aws

Developer Guide

Amazon Route 53

API Version 2013-04-01

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Route 53: Developer Guide

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

What is Amazon Route 53? 1
How domain registration works 3
How internet traffic is routed to your website or web application 4
Overview of how you configure Amazon Route 53 to route internet traffic for your domain 5
How Amazon Route 53 routes traffic for your domain 6
How Amazon Route 53 checks the health of your resources 7
Amazon Route 53 concepts 9
Domain registration concepts 10
Domain Name System (DNS) concepts 11
Control and data plane concepts 16
Health checking concepts 17
How to get started with Amazon Route 53 18
Accessing Amazon Route 53 19
AWS Identity and Access Management 19
Amazon Route 53 pricing and billing 20
Working with AWS SDKs 20
Getting started 22
Set up 22
Sign up for an AWS account 23
Download tools 23
Route DNS traffic to an Amazon S3 static website 24
Prerequisites 24
Step 1: Route DNS traffic for your domain to your website bucket 25
Step 2: Test your website 27
Route DNS traffic to a CloudFront distribution 27
Prerequisites 28
Step 1: Route DNS traffic for your domain to your CloudFront distribution 28
Integration with other services 31
Logging, monitoring, and tagging 31
Routing traffic to other AWS resources 32
DNS domain name format 34
Formatting domain names for domain name registration 34
Formatting domain names for hosted zones and records 34

Using an asterisk (*) in the names of hosted zones and records 35
Formatting internationalized domain names 36
Registering and managing domains 38
Registering new domains 39
Registering a new domain 39
Values that you specify when you register or transfer a domain 47
Values that Amazon Route 53 returns when you register a domain 53
Viewing the status of a domain registration 54
Updating domain settings 56
Updating contact information and ownership for a domain 56
Enabling or disabling privacy protection for contact information for a domain 65
Enabling or disabling automatic renewal for a domain 69
Locking a domain to prevent unauthorized transfer to another registrar 70
Extending the registration period for a domain 71
Updating name servers to use another registrar 73
Adding or changing name servers and glue records for a domain 73
Renewing registration for a domain 78
Restoring an expired or deleted domain 81
Replacing the hosted zone for a domain 82
Transferring domains 83
Choose your transfer type 84
Pre-transfer checklist 85
Transfer to Route 53 89
Common transfer issues 108
Transfer from Route 53 110
Transfer to a different AWS account 116
Transfer status 120
Transfer impact on domain expiration date 123
Resending authorization and confirmation emails 124
Updating your email address 126
Resending emails 126
Configuring DNSSEC for a domain 131
Overview of how DNSSEC protects your domain 132
Prerequisites and maximums for configuring DNSSEC for a domain 133
Adding public keys for a domain 134
Deleting public keys for a domain 135

Finding your registrar ... 136
Viewing information about domains ... 137
Deleting a domain name registration ... 139
Contacting AWS Support about domain registration issues ... 141
Contacting AWS Support when you can sign in to your AWS account ... 142
Contacting AWS Support when you can't sign in to your AWS account ... 143
Downloading a domain billing report ... 144
Domains that you can register with Amazon Route 53 ... 145
Index to supported top-level domains ... 146
Generic top-level domains ... 150
Geographic top-level domains ... 450
Configuring Amazon Route 53 as your DNS service ... 515
Making Route 53 the DNS service for an existing domain ... 516
Making Route 53 the DNS service for a domain that's in use ... 516
Making Route 53 the DNS service for an inactive domain ... 525
Configuring DNS routing for a new domain ... 529
Set up DNS for a domain you purchased from Route 53 ... 529
Set up DNS for a domain you purchased from another registrar ... 530
Routing traffic to your resources ... 531
Routing traffic for subdomains ... 532
Working with hosted zones ... 538
Working with public hosted zones ... 538
Working with private hosted zones ... 565
Hosted zone tiers ... 578
Hosted zone features ... 585
Migrating a hosted zone to a different AWS account ... 620
Working with records ... 631
Choosing a routing policy ... 632
Choosing between alias and non-alias records ... 656
Supported DNS record types ... 661
Creating records by using the Amazon Route 53 console ... 681
Resource record set permissions ... 684
Values you specify ... 685
Creating records by importing a zone file ... 780
Editing records ... 784
Deleting records ... 785

Listing records ... 785
Protecting records ... 788
Configuring DNSSEC signing ... 796
Enabling DNSSEC signing and establishing a chain of trust ... 797
Disabling DNSSEC signing ... 808
Working with customer managed keys ... 812
Working with key-signing keys (KSKs) ... 813
KMS key and ZSK management in Route 53 ... 815
DNSSEC proofs of nonexistence in Route 53 ... 817
Troubleshooting DNSSEC signing ... 817
Using AWS Cloud Map to create records and health checks ... 819
DNS constraints and behaviors ... 819
Maximum response size ... 819
Authoritative section processing ... 819
Additional section processing ... 819
Related topics ... 820
Traffic Flow ... 821
Traffic Flow advantages ... 821
Creating and managing traffic policies ... 822
Creating a traffic policy ... 823
Values that you specify when you create a traffic policy ... 824
Viewing a map that shows the effect of geoproximity settings ... 834
Creating additional versions of a traffic policy ... 836
Creating a traffic policy by using a JSON document ... 837
Viewing traffic policy versions and the associated policy records ... 838
Deleting traffic policy versions and traffic policies ... 840
Creating and managing policy records ... 841
Creating policy records ... 843
Values that you specify when you create or update a policy record ... 844
Updating policy records ... 845
Deleting policy records ... 845
Traffic Policy Document Format ... 846
Basic Syntax ... 847
Syntax for Endpoint Definitions ... 848
Syntax for Rule Definitions ... 851
Examples ... 862

What is VPC Resolver? ... 871
Resolving DNS queries between VPCs and your network ... 873
How DNS queries from your network are forwarded to VPC Resolver ... 877
How Resolver endpoints forward DNS queries from your VPCs to your network ... 879
Considerations when creating inbound and outbound endpoints ... 887
Route 53 VPC Resolver availability and scaling ... 891
Getting started with Route 53 VPC Resolver ... 893
Forwarding inbound DNS queries to your VPCs ... 895
Configuring inbound forwarding ... 895
Values that you specify when you create or edit inbound endpoints ... 896
Managing inbound endpoints ... 900
Forwarding outbound DNS queries to your network ... 903
Configuring outbound forwarding ... 904
Values that you specify when you create or edit outbound endpoints ... 905
Values that you specify when you create or edit rules ... 908
Managing outbound endpoints ... 911
Managing forwarding rules ... 913
Resolver delegation rules tutorial ... 921
Steps for outbound delegation ... 921
Delegation types ... 922
Enabling DNSSEC validation ... 927
Routing internet traffic to your AWS resources ... 929
Amazon API Gateway API ... 929
Prerequisites ... 930
Configuring Route 53 to route traffic to an API Gateway endpoint ... 931
Amazon CloudFront distribution ... 934
Prerequisites ... 935
Configuring Amazon Route 53 to route traffic to a CloudFront distribution ... 936
Amazon EC2 instance ... 939
Prerequisites ... 939
Configuring Amazon Route 53 to route traffic to an Amazon EC2 instance ... 940
App Runner service ... 941
Prerequisites ... 942
Configuring Amazon Route 53 to route traffic to an App Runner service ... 942
Global Accelerator ... 943
Prerequisites ... 944

Configuring Amazon Route 53 to route traffic to an accelerator ... 944
AWS Elastic Beanstalk environment ... 946
Deploying an application into an Elastic Beanstalk environment ... 946
Getting the domain name for your Elastic Beanstalk environment ... 946
Creating a Route 53 record ... 947
ELB load balancer ... 950
Prerequisites ... 951
Configuring Amazon Route 53 to route traffic to an ELB load balancer ... 951
Amazon S3 bucket ... 953
Prerequisites ... 954
Configuring Amazon Route 53 to route traffic to an S3 Bucket ... 955
Amazon Virtual Private Cloud interface endpoint ... 956
Prerequisites ... 957
Amazon VPC interface endpoint ... 957
Amazon WorkMail ... 959
Amazon OpenSearch Service ... 961
Prerequisites ... 962
Configuring Amazon Route 53 to route traffic to an Amazon OpenSearch Service domain endpoint ... 962
VPC Lattice ... 963
Prerequisites ... 963
Configuring Amazon Route 53 to route traffic to a VPC Lattice service domain endpoint ... 964
Other AWS resources ... 965
Creating health checks ... 966
Types of health checks ... 967
Private endpoint health checks ... 968
How private endpoint health checks work ... 969
Availability at launch ... 969
Setup considerations ... 970
Regional considerations ... 970
Health check sources ... 971
Creating a health check source ... 972
Viewing health check sources ... 973
Deleting a health check source ... 974
How Route 53 determines whether a health check is healthy ... 975
How Route 53 determines the status of health checks that monitor a public endpoint ... 975

How Route 53 determines the status of health checks that monitor other health checks .. 977
How Route 53 determines the status of health checks that monitor CloudWatch alarms ... 977
How Route 53 determines the status of a private endpoint health check 978
Creating, updating, and deleting health checks 978
Creating and updating health checks 980
Values that you specify when you create or update health checks 981
Values that Route 53 displays when you create a health check 993
Updating health checks when you change CloudWatch alarm settings 994
Disabling or enabling health checks 995
Inverting health checks 995
Deleting health checks 996
Updating or deleting health checks when DNS failover is configured 997
Configuring router and firewall rules for health checks 998
Configuring DNS failover 999
Task list for configuring DNS failover 1000
How health checks work in simple configurations 1001
How health checks work in complex configurations 1005
How Route 53 chooses records when health checking is configured 1012
Active-active and active-passive failover 1015
Configuring failover in a private hosted zone 1018
How Route 53 averts failover problems 1019
Naming and tagging health checks 1020
Tag restrictions 1021
Adding, editing, and deleting tags for health checks 1021
Using API versions before 2012-12-12 1022
Monitoring health check status and getting notifications 1023
Viewing health check status and the reason for health check failures 1023
Monitoring the latency between health checkers and your endpoint 1024
Monitoring health checks using CloudWatch 1026
View the status of your health check 1027
View health check alarms 1028
View health check metrics on the CloudWatch console 1029
Create an alarm with an SNS notification 1029
Resolver DNS Firewall 1030
How Resolver DNS Firewall works 1031
DNS Firewall components and settings 1031

How Resolver DNS Firewall filters DNS queries ... 1035
High-level steps for using DNS Firewall ... 1036
Using DNS Firewall rule groups in multiple Regions ... 1037
Region availability for DNS Firewall ... 1037
Getting started with Resolver DNS Firewall ... 1038
Resolver DNS Firewall walled garden example ... 1038
Resolver DNS Firewall block list example ... 1041
DNS Firewall rule groups and rules ... 1043
Rule group settings in DNS Firewall ... 1043
Rule settings in DNS Firewall ... 1044
Rule actions in DNS Firewall ... 1047
Managing rule groups and rules in DNS Firewall ... 1048
DNS Firewall Foundational Rules ... 1051
Managed Domain Lists ... 1052
Managing your own domain lists ... 1058
DNS Firewall Advanced Rules ... 1060
Advanced DNS Protections ... 1060
Advanced Managed Domain Lists ... 1062
Partner managed DNS threat protection with Palo Alto Networks ... 1065
Partner managed DNS threat protection with Infoblox ... 1071
Automated Allowlists ... 1077
Configuring query logging for DNS Firewall ... 1085
Sharing rule groups between accounts ... 1087
Enabling DNS Firewall protections for your VPC ... 1090
Managing associations between your VPC and firewall rule groups ... 1091
DNS Firewall VPC configuration ... 1092
Resolved IP validation ... 1093
Using resolved IP validation with and without DNS Firewall rules ... 1093
What are Amazon Route 53 Profiles? ... 1095
Profile prioritization ... 1096
Profile availability ... 1098
Using Profiles ... 1098
Create a Profile ... 1099
Associate DNS Firewall rule groups ... 1100
Associate private hosted zones ... 1102
Associate Resolver rules ... 1103

Associate VPC endpoints ... 1103
Associate query logging configurations ... 1104
Edit Profile configurations ... 1105
Associate VPCs ... 1107
Viewing and updating Profiles ... 1109
Deleting a Profile ... 1110
Viewing and updating resources associated to Profiles ... 1112
Disassociating a resource ... 1114
Viewing VPCs associated to a Profile ... 1115
Disassociating a VPC ... 1117
Updating association priority ... 1118
Working with shared Route 53 Profiles ... 1121
Granting permissions for sharing Route 53 Profiles ... 1122
Prerequisites for sharing Route 53 Profiles ... 1122
Sharing a Route 53 Profile ... 1122
Unsharing a shared Route 53 Profile ... 1123
Identifying a shared Route 53 Profile ... 1124
Responsibilities and permissions for shared Route 53 Profiles ... 1125
Billing and metering ... 1125
Instance quotas ... 1126
What is Amazon Route 53 on Outposts? ... 1127
Architecture overview ... 1127
Route 53 on Outposts features ... 1128
VPC Resolver behavior when AWS Outposts is disconnected from the VPC ... 1129
Getting started with VPC Resolver on AWS Outposts ... 1129
Second-generation AWS Outposts ... 1129
First-generation AWS Outposts ... 1129
Creating inbound endpoints (first-generation AWS Outposts only) ... 1131
Values that you specify when you create or edit inbound endpoints on an Outpost ... 1131
Creating outbound endpoints (first-generation AWS Outposts only) ... 1133
Values that you specify when you create or edit outbound endpoints in an AWS Outposts ... 1134
Creating forwarding rules for outbound endpoints ... 1136
Managing Resolver on Outpost ... 1136
Editing Resolver on Outpost (first-generation AWS Outposts only) ... 1137
Viewing Resolver on Outpost status ... 1137

Confirming that Resolver on Outpost is working ... 1138
Deleting Resolver on Outpost (first-generation AWS Outposts only) ... 1139
Managing inbound endpoints on Resolver on Outpost (first-generation AWS Outposts only) ... 1140
Viewing and editing inbound endpoints ... 1140
Viewing the status for inbound endpoints ... 1141
Deleting inbound endpoints ... 1142
Managing outbound endpoints on Resolver on Outpost (first-generation AWS Outposts only) ... 1143
Viewing and editing outbound endpoints ... 1143
Viewing the status for outbound endpoints ... 1144
Deleting outbound endpoints ... 1145
What is Route 53 Global Resolver? ... 1146
Why Route 53 Global Resolver? ... 1147
Concepts and terminology ... 1147
DNS resolver for clients at on-premises and remote locations ... 1148
DNS client authentication ... 1148
Split-traffic DNS resolution ... 1149
High availability and global presence ... 1150
DNS resolution and forwarding ... 1151
DNS filtering and domain lists ... 1151
Advanced DNS threat detection ... 1152
Monitoring and logging ... 1153
How it works ... 1154
What happens when clients make DNS queries ... 1154
Global anycast architecture ... 1154
Private anycast architecture with Cloud WAN ... 1155
DNS filtering and security ... 1155
Use cases ... 1156
Related services ... 1157
Accessing Global Resolver ... 1158
Supported AWS Regions ... 1159
Opt-in Region considerations ... 1160
Setting up account access ... 1161
Creating policies and roles ... 1162
Network considerations ... 1163

Tutorial: Create your first Route 53 Global Resolver 1163
Prerequisites 1163
Step 1: Create a global resolver 1164
Step 2: Create a DNS view and configure authentication 1164
Step 3: Configure DNS filtering rules (optional) 1165
Step 4: Test your configuration 1165
Step 5: Monitoring DNS activity 1166
Step 6: Clean up (optional) 1166
Next steps 1166
Tutorial: Create your first Route 53 Global Resolver with AWS Cloud WAN 1167
Prerequisites 1167
Step 1: Create a global resolver with public IP connectivity disabled 1168
Step 2: Connect a Cloud WAN core network 1168
Step 3: Create a DNS view and a segment-based access source 1169
Next steps 1169
Managing DNS views 1170
Managing DNS views 1170
Managing private hosted zone associations 1171
Configure settings for DNS views 1172
Managing access controls 1175
Access control methods 1175
Configuring access sources 1176
Managing access tokens 1178
Best practices 1183
Securing DNS 1184
Configure and manage DNS Firewall rules 1184
Managed Domain Lists 1188
Build domain lists 1192
DNS Firewall Advanced protections 1193
Manage security policies 1194
Configuring private hosted associations 1195
Manage private hosted zone associations 1195
Sharing DNS views between accounts 1196
Sharing a DNS view 1198
Using a shared DNS view 1200
Core network endpoints 1200

How it works ... 1201
Managing endpoints ... 1202
Monitoring DNS activity ... 1204
Gain DNS visibility ... 1205
Configure DNS monitoring ... 1212
Monitoring with CloudWatch ... 1213
Troubleshooting DNS issues ... 1221
Diagnose connectivity issues ... 1221
Resolve performance issues ... 1223
Troubleshoot configuration ... 1225
Troubleshooting internal resource access ... 1226
Quotas ... 1227
Soft quotas ... 1227
Creating AWS CloudFormation resources ... 1229
Route 53, VPC Resolver, and CloudFormation templates ... 1229
Best practices for Route 53 and CloudFormation ... 1230
Understanding eventual consistency ... 1230
DNS record ordering and logical IDs ... 1231
Handling rollback failures ... 1231
Learn more about CloudFormation ... 1232
Code examples ... 1233
Route 53 ... 1234
Basics ... 1234
Route 53 domain registration ... 1266
Basics ... 1267
Security ... 1349
Data protection ... 1349
Protection from dangling delegation records ... 1350
Identity and access management ... 1353
Authenticating with identities ... 1353
Access control ... 1355
Using Service-Linked Roles ... 1376
Service-linked roles for Private endpoint health checks ... 1380
AWS managed policies ... 1384
Using conditions ... 1400
Route 53 API permissions reference ... 1417

Logging and monitoring 1418
Compliance validation 1419
Resilience 1420
Infrastructure security 1421
Sending findings to Security Hub CSPM 1422
How findings work in Security Hub CSPM 1422
Types of findings that DNS Firewall sends 1423
Retrying when Security Hub CSPM is unavailable 1423
Updating existing findings in Security Hub CSPM 1423
Typical finding from DNS Firewall 1423
Enabling and configuring the integration 1426
Stopping the delivery of findings to Security Hub CSPM 1426
Monitoring 1427
Public DNS query logging 1427
Configuring logging for DNS queries 1428
Using Amazon CloudWatch to access DNS query logs 1430
Changing the retention period for logs and exporting logs to Amazon S3 1430
Stopping query logging 1431
Values that appear in DNS query logs 1431
Query log example 1433
Resolver query logging 1433
Resources that you can send Resolver query logs to 1435
Managing configurations 1437
Monitoring domain registrations 1445
Monitoring your resources with Amazon Route 53 health checks and Amazon CloudWatch . 1446
Metrics and dimensions for health checks 1446
Monitoring hosted zones using Amazon CloudWatch 1449
CloudWatch metrics for Route 53 public hosted zones 1449
CloudWatch dimension for Route 53 public hosted zone metrics 1451
Monitoring Route 53 VPC Resolver endpoints with Amazon CloudWatch 1451
Metrics and dimensions for VPC Resolver 1451
Monitoring Resolver DNS Firewall rule groups with Amazon CloudWatch 1459
Metrics and dimensions for DNS Firewall 1460
Using Amazon CloudWatch Contributor Insights with DNS Firewall data 1462
Analyzing DNS Firewall logs with Amazon CloudWatch Logs Insights 1463
Managing DNS Firewall events using EventBridge 1464

Resolver DNS Firewall events ... 1465
Sending DNS Firewall events ... 1465
Permissions ... 1468
Additional resources ... 1468
Events DNS Firewall detail reference ... 1468
Logging Amazon Route 53 API calls with AWS CloudTrail ... 1476
Route 53 information in CloudTrail ... 1476
Viewing Route 53 events in event history ... 1477
Understanding Route 53 log file entries ... 1477
Troubleshooting ... 1486
My domain is unavailable on the internet ... 1487
You registered a new domain, but you didn't choose the link in the confirmation email .. 1487
You transferred domain registration to Amazon Route 53, but you didn't transfer DNS service ... 1488
You transferred domain registration and specified the wrong name servers in the domain settings ... 1489
You transferred DNS service first, but you didn't wait long enough before transferring domain registration ... 1491
You deleted the hosted zone that Route 53 is using to route internet traffic for the domain ... 1491
Your domain has been suspended ... 1492
My domain is suspended (status is ClientHold) ... 1492
You registered a new domain, but you didn't choose the link in the confirmation email .. 1493
You disabled automatic renewal for the domain, and the domain expired ... 1494
You changed the email address for the registrant contact, but you didn't verify that the new email address is valid ... 1494
We suspended the domain for a violation of the AWS Acceptable Use Policy ... 1495
We suspended the domain because of a court order ... 1495
My domain operation failed ... 1495
Domain operation failed due to invalid contact information ... 1495
Transferring my domain to Amazon Route 53 failed ... 1496
You didn't choose the link in the authorization email ... 1496
The authorization code that you got from the current registrar is not valid ... 1497
"Parameters in request are not valid" error when trying to transfer a .es domain to Amazon Route 53 ... 1497

Is the internationalized domain name you're transferring to Amazon Route 53 listed in punycode? ... 1497
Common transfer error messages ... 1498
I changed DNS settings, but they haven't taken effect ... 1500
You transferred DNS service to Amazon Route 53 in the last 48 hours, so DNS is still using your previous DNS service ... 1501
You recently transferred DNS service to Amazon Route 53, but you didn't update the name servers with the domain registrar ... 1501
DNS resolvers still are using the old settings for the record ... 1502
You have more than one hosted zone with the same name, and you updated the one that isn't associated with the domain ... 1503
My browser displays a "Server not found" error ... 1505
You didn't create a record for the domain or subdomain name ... 1505
You created a record but specified the wrong value ... 1505
The resource that you're routing traffic to is unavailable ... 1505
I can't route traffic to an Amazon S3 bucket that's configured for website hosting ... 1506
I was billed twice for the same hosted zone ... 1506
I was charged multiple invoices for my domain ... 1506
My AWS account is closed or permanently closed, and my domain is registered with Route 53 ... 1507
IP address ranges ... 1509
IP address ranges of Route 53 name servers ... 1509
IP address ranges of Route 53 health checks ... 1509
Referencing prefix lists ... 1510
Internal IP address ranges of Route 53 health checks ... 1510
Tagging resources ... 1511
Tutorials ... 1512
Using Amazon Route 53 as the DNS service for subdomains without migrating the parent domain ... 1513
Creating a subdomain that uses Amazon Route 53 as the DNS service without migrating the parent domain ... 1514
Migrating DNS service for a subdomain to Amazon Route 53 without migrating the parent domain ... 1517
Transitioning to latency-based routing in Amazon Route 53 ... 1521
Adding another Region to your latency-based routing in Amazon Route 53 ... 1523

Using latency and weighted records in Amazon Route 53 to route traffic to multiple Amazon EC2 instances in a Region ... 1525
Managing over 100 weighted records in Amazon Route 53 ... 1526
Weighting fault-tolerant multi-record answers in Amazon Route 53 ... 1527
Best practices ... 1529
Best practices for Amazon Route 53 DNS ... 1530
Best practices for VPC Resolver ... 1532
Avoid loop configurations with Resolver endpoints ... 1534
Resolver endpoint scaling ... 1534
High availability for Resolver endpoints ... 1536
DNS zone walking ... 1536
Subnet compatibility for Resolver endpoints ... 1536
Best practices for Amazon Route 53 health checks ... 1537
Best practices for Amazon Route 53 domain registration ... 1538
Making API Requests ... 1540
Making API Requests for Hosted Zones, Resource Record Sets, and Health Checks ... 1540
REST Requests ... 1541
REST Responses ... 1542
Making API Requests for Domain Registration and for Route 53 Resolver ... 1544
RPC Requests ... 1544
RPC Responses ... 1546
Signing Amazon Route 53 API Requests ... 1547
Amazon Route 53 API actions by function ... 1547
Types of function ... 1547
Actions by function ... 1549
API throttling ... 1557
How throttling is applied ... 1557
Request rate token bucket sizes and refill rates ... 1558
Change throughput limiting ... 1559
Monitor API throttling ... 1560
Retries and exponential backoff ... 1560
Requesting a limit increase ... 1561
Best practices for API throttling ... 1561
Quotas ... 1562
Using Service Quotas to view and manage quotas ... 1562
Quotas on entities ... 1562

Quotas on domains ... 1563
Quotas on hosted zones ... 1563
Quotas on records ... 1564
Quotas on Route 53 VPC Resolver ... 1565
Quotas on health checks ... 1573
Quotas on query log configurations ... 1574
Quotas on traffic flow policies and policy records ... 1574
Quotas on reusable delegation sets ... 1575
Quotas on Route 53 Profiles ... 1575
Maximums on API requests ... 1576
Number of elements and characters in ChangeResourceRecordSets requests ... 1576
Number of elements in record set protection requests ... 1576
Frequency of Amazon Route 53 API requests ... 1577
Frequency of Route 53 VPC Resolver API requests ... 1578
Related information ... 1579
AWS resources ... 1579
Third-party tools and libraries ... 1580
Graphical user interfaces ... 1581
FAQs on supported protocols ... 1582
What are the AWS JSON and Smithy RPC v2 CBOR protocols, and how do they differ? ... 1582
How do I get started with the Smithy RPC v2 CBOR protocol? ... 1582
What are the risks of enabling the Smithy RPC v2 CBOR protocol? ... 1583
What if my solution does not support Smithy RPC v2 CBOR? ... 1583
What AWS Regions support the Smithy RPC v2 CBOR protocol? ... 1583
What latency improvements can I expect when upgrading to SDK versions that support Smithy RPC v2 CBOR? ... 1583
Will the AWS JSON protocol be deprecated? ... 1583
Where can I find more information about Smithy RPC v2 CBOR? ... 1584
Document history ... 1585
2026 releases ... 1585
2025 releases ... 1588
2024 releases ... 1590
2023 releases ... 1592
2022 releases ... 1593
2021 releases ... 1593
2020 releases ... 1594

2018 releases 1594
2017 releases 1596
2016 releases 1598
2015 releases 1601
2014 releases 1603
2013 releases 1607
2012 release 1608
2011 releases 1608
2010 release 1609

What is Amazon Route 53?

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. You can use Route 53 to perform three main functions in any combination: domain registration, DNS routing, and health checking.

If you choose to use Route 53 for all three functions, follow the order below:

1. Register domain names

Your website needs a name, such as example.com. With Route 53, you can register a name for your website or web application, known as a *domain name*.

- For an overview, see How domain registration works.
- For a procedure, see Registering a new domain.
- For a tutorial that takes you through registering a domain and creating a simple website in an Amazon S3 bucket, see Getting started with Amazon Route 53.

2. Route internet traffic to the resources for your domain

When a user opens a web browser and enters your domain name (example.com) or subdomain name (acme.example.com), Route 53 connects the browser with your website or web application.

- For an overview, see How internet traffic is routed to your website or web application.
- For procedures, see Configuring Amazon Route 53 as your DNS service.
- For a procedure on how to route email to Amazon WorkMail, see Routing traffic to Amazon WorkMail.

3. Check the health of your resources

Route 53 sends automated requests over the internet to a resource, such as a web server, to verify that it's reachable, available, and functional. You can receive notifications when a resource becomes unavailable. You can also route internet traffic away from unhealthy resources.

- For an overview, see How Amazon Route 53 checks the health of your resources.
- For procedures, see Creating Amazon Route 53 health checks.

Other Route 53 features

In addition to being a Domain Name System (DNS) web service, Route 53 offers the following features:

VPC Resolver

Get recursive DNS for your Amazon VPCs in AWS Regions, VPCs in AWS Outposts racks, or any other on-premises networks. Create conditional forwarding rules and Route 53 endpoints to resolve custom names in Route 53 private hosted zones or in your on-premises DNS servers.

For more information , see What is Route 53 VPC Resolver?.

Amazon Route 53 Resolver on Outposts

Connect VPC Resolver on Outpost racks with DNS servers in your on-premises data centers through Resolver endpoints. This resolves DNS queries between the Outposts racks and your other on-premises resources.

For more information , see What is Amazon Route 53 on Outposts?.

Resolver DNS Firewall

Protect your recursive DNS queries within the VPC Resolver. Create domain lists and build firewall rules that filter outbound DNS traffic against these rules.

For more information , see Using DNS Firewall to filter outbound DNS traffic.

Traffic Flow

Easy-to-use and cost-effective global traffic management: route end users to the best endpoint for your application based on geoproximity, latency, health, and other considerations.

For more information , see Using Traffic Flow to route DNS traffic.

Amazon Route 53 Profiles

With Route 53 Profiles, you can apply and manage DNS-related Route 53 configurations across many VPCs and in different AWS account.

For more information , see What are Amazon Route 53 Profiles?.

Topics

- How domain registration works

- How internet traffic is routed to your website or web application
- How Amazon Route 53 checks the health of your resources
- Amazon Route 53 concepts
- How to get started with Amazon Route 53
- Accessing Amazon Route 53
- AWS Identity and Access Management
- Amazon Route 53 pricing and billing
- Using Route 53 with an AWS SDK

How domain registration works

If you want to create a website or a web application, you start by registering the name of your website, known as a domain name. Your domain name is the name, such as example.com, that your users enter in a browser to display your website.

Here's an overview of how you register a domain name with Amazon Route 53:

1. You choose a domain name and confirm that it's available, meaning that no one else has registered the domain name that you want.

   If the domain name you want is already in use, you can try other names or change only the *top-level domain*, such as .com, to another top-level domain, such as .ninja or .hockey. For a list of the top-level domains that Route 53 supports, see Domains that you can register with Amazon Route 53.
2. You register the domain name with Route 53. When you register a domain, you provide names and contact information for the domain owner and other contacts.

   When you register a domain with Route 53, the service automatically makes itself the DNS service for the domain. It does the following:

   - Creates a hosted zone that has the same name as your domain.
   - Assigns a set of four name servers to the hosted zone. When someone uses a browser to access your website, such as www.example.com, these name servers tell the browser where to find your resources, such as a web server or an Amazon S3 bucket. (Amazon S3 is object storage for storing and retrieving any amount of data from anywhere on the web. A bucket is a container for objects that you store in S3.)

- Gets the name servers from the hosted zone and adds them to the domain.

For more information, see How internet traffic is routed to your website or web application.

3. At the end of the registration process, Route 53 sends your information to the registrar for the domain. The domain registrar is either Amazon Registrar, Inc. or our registrar associate, Gandi. To find out who the registrar is for your domain, see Finding your registrar.
4. The registrar sends your information to the *registry* for the domain. A registry is a company that sells domain registrations for one or more top-level domains, such as .com.
5. The registry stores the information about your domain in their own database and also stores some of the information in the public WHOIS database.

For more information about how to register a domain name, see Registering a new domain.

If you already registered a domain name with another registrar, you can transfer the domain registration to Route 53. This isn't required to use other Route 53 features. For more information, see Transferring registration for a domain to Amazon Route 53.

How internet traffic is routed to your website or web application

All computers on the internet, from your smart phone or laptop to the servers that serve content for massive retail websites, communicate with one another by using numbers. These numbers, known as *IP addresses*, are in one of the following formats:

- Internet Protocol version 4 (IPv4) format, such as 192.0.2.44
- Internet Protocol version 6 (IPv6) format, such as 2001:0db8:85a3:0000:0000:abcd:0001:2345

When you open a browser and go to a website, you don't have to remember and enter a long string of characters like that. Instead, you can enter a domain name like example.com and still get to the right place. A DNS service such as Amazon Route 53 makes the connection between domain names and IP addresses.

Topics

- Overview of how you configure Amazon Route 53 to route internet traffic for your domain
- How Amazon Route 53 routes traffic for your domain

Overview of how you configure Amazon Route 53 to route internet traffic for your domain

Here's an overview of how to use the Amazon Route 53 console to register a domain name and configure Route 53 to route internet traffic to your website or web application.

1. You register the domain name that you want your users to use to access your content. For an overview, see How domain registration works.
2. After you register your domain name, Route 53 automatically creates a public hosted zone that has the same name as the domain. For more information, see Working with public hosted zones.
3. To route traffic to your resources, you create *records*, also known as *resource record sets*, in your hosted zone. Each record includes information about how to route traffic for your domain, such as the following:

   Name

   The name of the record corresponds with the domain name (example.com) or subdomain name (www.example.com, retail.example.com) that you want Route 53 to route traffic for.

   The name of every record in a hosted zone must end with the name of the hosted zone. For example, if the name of the hosted zone is example.com, all record names must end in example.com. The Route 53 console does this for you automatically.

   Type

   The record type usually determines the type of resource that you want traffic to be routed to. For example, to route traffic to an email server, you specify MX for Type. To route traffic to a web server that has an IPv4 IP address, you specify A for Type.

   Value

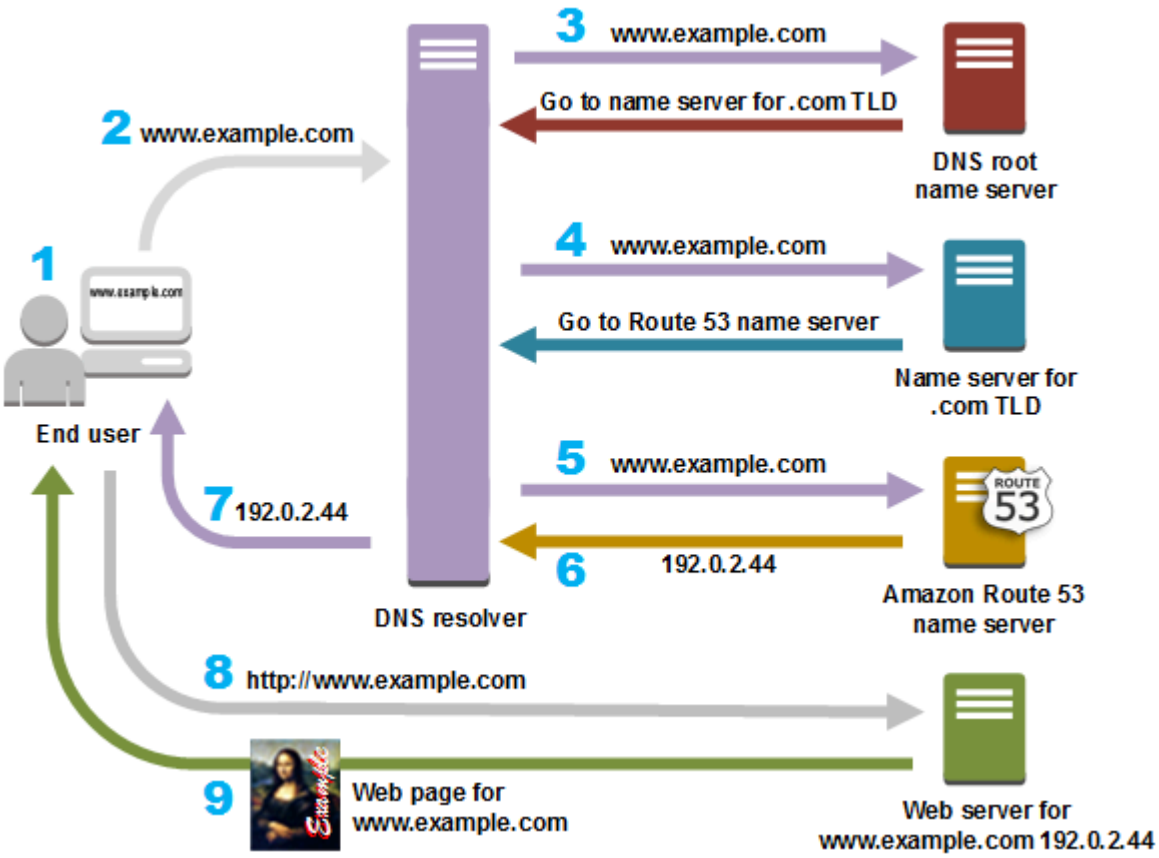
   Value is closely related to Type. If you specify MX for Type, you specify the names of one or more email servers for Value. If you specify A for Type, you specify an IP address in IPv4 format, such as 192.0.2.136.

For more information about records, see Working with records.

You can also create special Route 53 records, called alias records, that route traffic to Amazon S3 buckets, Amazon CloudFront distributions, and other AWS resources. For more information, see Choosing between alias and non-alias records and Routing internet traffic to your AWS resources.

How Amazon Route 53 routes traffic for your domain

After you configure Amazon Route 53 to route your internet traffic to your resources, such as web servers or Amazon S3 buckets, here's what happens in a few milliseconds when someone requests content for www.example.com:



1. A user opens a web browser, enters www.example.com in the address bar, and presses Enter.
2. The request for www.example.com is routed to a DNS resolver, which is typically managed by the user's internet service provider (ISP), such as a cable internet provider, a DSL broadband provider, or a corporate network.
3. The DNS resolver for the ISP forwards the request for www.example.com to a DNS root name server.
4. The DNS resolver forwards the request for www.example.com again, this time to one of the TLD name servers for .com domains. The name server for .com domains responds with the names of the four Route 53 name servers associated with the example.com domain.

The DNS resolver caches (stores) the four Route 53 name servers. The next time someone browses to example.com, the resolver skips steps 3 and 4 because it already has the name servers for example.com. The name servers are typically cached for two days.

5. The DNS resolver chooses a Route 53 name server and forwards the request for www.example.com to that name server.
6. The Route 53 name server looks in the example.com hosted zone for the www.example.com record, gets the associated value, such as the IP address for a web server, 192.0.2.44, and returns the IP address to the DNS resolver.
7. The DNS resolver now has the IP address that the user needs. The resolver returns that value to the web browser.

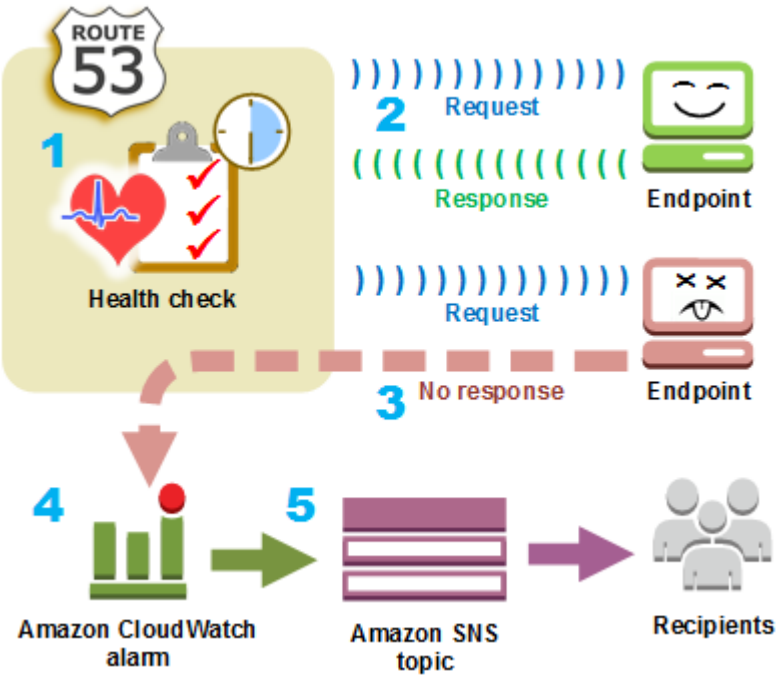
> **Note**
>
> The DNS resolver also caches the IP address for example.com for an amount of time that you specify so that it can respond more quickly the next time someone browses to example.com. For more information, see time to live (TTL).

8. The web browser sends a request for www.example.com to the IP address that it got from the DNS resolver. This is where your content is, for example, a web server running on an Amazon EC2 instance or an Amazon S3 bucket that's configured as a website endpoint.
9. The web server or other resource at 192.0.2.44 returns the web page for www.example.com to the web browser, and the web browser displays the page.

How Amazon Route 53 checks the health of your resources

Amazon Route 53 health checks monitor the health of your resources such as web servers and email servers. You can configure Amazon CloudWatch alarms for your health checks so that you receive notification when a resource becomes unavailable.

Here's an overview of how health checking works if you want to be notified when a resource becomes unavailable:



1. You create a health check and specify values that define how you want the health check to work, such as the following:
   - The IP address or domain name of the endpoint, such as a web server, that you want Route 53 to monitor. (You can also monitor the status of other health checks, or the state of a CloudWatch alarm.)
   - The protocol that you want Amazon Route 53 to use to perform the check: HTTP, HTTPS, or TCP.
   - How often you want Route 53 to send a request to the endpoint. This is the *request interval*.
   - How many consecutive times the endpoint must fail to respond before Route 53 considers it unhealthy. This is the *failure threshold*.
   - Optionally, how you want to be notified when Route 53 detects that the endpoint is unhealthy. When you configure notification, Route 53 sets a CloudWatch alarm. CloudWatch uses Amazon SNS to notify users that an endpoint is unhealthy.
2. Route 53 starts to send requests to the endpoint at the interval that you specified in the health check.

   If the endpoint responds to the requests, Route 53 considers the endpoint to be healthy and takes no action.
3. If the endpoint doesn't respond to a request, Route 53 starts to count the number of consecutive requests that the endpoint doesn't respond to:

- If the count reaches the value that you specified for the failure threshold, Route 53 considers the endpoint unhealthy.
- If the endpoint starts to respond again before the count reaches the failure threshold, Route 53 resets the count to 0, and CloudWatch doesn't contact you.

4. If Route 53 considers the endpoint unhealthy and if you configured notification for the health check, Route 53 notifies CloudWatch.

   If you didn't configure notification, you can still see the status of your Route 53 health checks in the Route 53 console. For more information, see Monitoring health check status and getting notifications.
5. If you configured notification for the health check, CloudWatch triggers an alarm and uses Amazon SNS to send notification to the specified recipients.

In addition to checking the health of a specified endpoint, you can configure a health check to check the health of one or more other health checks. This lets you know when a specified number of resources, such as two web servers out of five, are unavailable. You can also configure a health check to check the status of a CloudWatch alarm so that you can be notified based on a broad range of criteria, not just whether a resource is responding to requests.

If you have multiple resources that perform the same function, for example, web servers or database servers, you can configure DNS failover. Associate a health check with each record for that resource. If a health check determines that the underlying resource is unhealthy, Route 53 routes traffic away from the associated record.

For more information about using Route 53 to monitor the health of your resources, see Creating Amazon Route 53 health checks.

Amazon Route 53 concepts

Here's an overview of the concepts that are discussed throughout the *Amazon Route 53 Developer Guide*.

Topics

- Domain registration concepts
- Domain Name System (DNS) concepts
- Control and data plane concepts

- Health checking concepts

Domain registration concepts

Here's an overview of the concepts that are related to domain registration.

- domain name
- domain registrar
- domain registry
- domain reseller
- top-level domain (TLD)

domain name

The name, such as example.com, that a user types in the address bar of a web browser to access a website or a web application. To make your website or web application available on the internet, you start by registering a domain name. For more information, see How domain registration works.

domain registrar

A company that is accredited by ICANN (Internet Corporation for Assigned Names and Numbers) to process domain registrations for specific top-level domains (TLDs). To find the registrar of your domain, see Finding your registrar.

domain registry

A company that owns the right to sell domains that have a specific top-level domain. For example, VeriSign is the registry that owns the right to sell domains that have a .com TLD. A domain registry defines the rules for registering a domain, such as residency requirements for a geographic TLD. A domain registry also maintains the authoritative database for all of the domain names that have the same TLD. The registry's database contains information such as contact information and the name servers for each domain.

domain reseller

A company that sells domain names for registrars such as Amazon Registrar. Amazon Route 53 is a domain reseller for Amazon Registrar and for our registrar associate, Gandi.

top-level domain (TLD)

The last part of a domain name, such as .com, .org, or .ninja. There are two types of top-level domains:

Generic top-level domains

These TLDs typically give users an idea of what they'll find on the website. For example, domain names that have a TLD of *.bike* often are associated with websites for motorcycle or bicycle businesses or organizations. With a few exceptions, you can use any generic TLD you want, so a bicycle club could use the .hockey TLD for their domain name.

Geographic top-level domains

These TLDs are associated with geographic areas such as countries or cities. Some registries for geographic TLDs have residency requirements, while others, such as the section called ".io (British Indian Ocean Territory)", allow or even encourage use as a generic TLD.

For a list of the TLDs that you can use when you register a domain name with Route 53, see Domains that you can register with Amazon Route 53.

Domain Name System (DNS) concepts

Here's an overview of the concepts that are related to the Domain Name System (DNS).

- alias record
- authoritative name server
- CIDR block
- DNS query
- DNS resolver
- Domain Name System (DNS)
- hosted zone
- IP address
- name servers
- private DNS
- recursive name server
- record (DNS record)

- reusable delegation set
- routing policy
- subdomain
- time to live (TTL)

alias record

A type of record that you can create with Amazon Route 53 to route traffic to AWS resources such as Amazon CloudFront distributions and Amazon S3 buckets. For more information, see Choosing between alias and non-alias records.

authoritative name server

A name server that has definitive information about one part of the Domain Name System (DNS). It responds to requests from a DNS resolver by returning the applicable information. For example, an authoritative name server for the .com top-level domain (TLD) knows the names of the name servers for every registered .com domain. When a .com authoritative name server receives a request from a DNS resolver for example.com, it responds with the names of the name servers for the DNS service for the example.com domain.

Route 53 name servers are the authoritative name servers for every domain that uses Route 53 as the DNS service. The name servers know how you want to route traffic for your domain and subdomains based on the records that you created in the hosted zone for the domain. (Route 53 name servers store the hosted zones for the domains that use Route 53 as the DNS service.)

For example, if a Route 53 name server receives a request for www.example.com, it finds that record and returns the IP address, such as 192.0.2.33, that is specified in the record.

CIDR block

A CIDR block is an IP range used with IP-based routing. In Route 53, you can specify a CIDR block from /0 to /24 for IPv4 and /0 to /48 for IPv6. For example, a /24 IPv4 CIDR block includes 256 contiguous IP addresses. You can group CIDR blocks (or IP ranges) into CIDR locations, which are then grouped into reusable CIDR collections.

DNS query

A request that is submitted by a device, such as a computer or a smart phone, to the Domain Name System (DNS) for a resource associated with a domain name. The most common example of a DNS query is when a user opens a browser and types the domain name in the address bar.

The response to a DNS query is typically the IP address associated with a resource such as a web server. The device that initiated the request uses the IP address to communicate with the resource. For example, a browser can use the IP address to get a web page from a web server.

DNS resolver

A DNS server, often managed by an internet service provider (ISP), that acts as an intermediary between user requests and DNS name servers. When you open a browser and enter a domain name in the address bar, your query goes first to a DNS resolver. The resolver communicates with DNS name servers to get the IP address for the resource, such as a web server. A DNS resolver is also known as a recursive name server because it sends requests to a sequence of authoritative DNS name servers until it gets the response (typically an IP address) that it returns to the user's device, for example, a web browser on a laptop.

Domain Name System (DNS)

A worldwide network of servers that help computers, smart phones, tablets, and other IP-enabled devices to communicate with one another. The Domain Name System translates easily understood names such as example.com into the numbers, known as *IP addresses*, that allow computers to find each other on the internet.

See also IP address.

hosted zone

A container for records that include information about how to route traffic for a domain (such as example.com) and all of its subdomains (such as www.example.com, retail.example.com, and seattle.accounting.example.com). A hosted zone has the same name as the domain.

For example, the hosted zone for example.com might include a record that has information about routing traffic for www.example.com to a web server that has the IP address 192.0.2.243, and a record that has information about routing email for example.com to two email servers, mail1.example.com and mail2.example.com. Each email server also requires its own record.

See also record (DNS record).

IP address

A number that is assigned to a device on the internet—such as a laptop, a smart phone, or a web server—that allows the device to communicate with other devices on the internet. IP addresses are in one of the following formats:

- Internet Protocol version 4 (IPv4) format, such as 192.0.2.44

- Internet Protocol version 6 (IPv6) format, such as 2001:0db8:85a3:0000:0000:abcd:0001:2345

Route 53 supports both IPv4 and IPv6 addresses for the following purposes:

- You can create records that have a type of A, for IPv4 addresses, or a type of AAAA, for IPv6 addresses.
- You can create health checks that send requests either to IPv4 or to IPv6 addresses.
- If a DNS resolver is on an IPv6 network, it can use either IPv4 or IPv6 to submit requests to Route 53.

name servers

Servers in the Domain Name System (DNS) that help to translate domain names into the IP addresses that computers use to communicate with one another. Name servers are either recursive name servers (also known as DNS resolver) or authoritative name server.

For an overview of how DNS routes traffic to your resources, including the role of Route 53 in the process, see How Amazon Route 53 routes traffic for your domain.

private DNS

A local version of the Domain Name System (DNS) where you can route traffic for a domain and its subdomains to Amazon EC2 instances within one or more Amazon virtual private clouds (VPCs). For more information, see Working with private hosted zones.

record (DNS record)

An object in a hosted zone that you use to define how you want to route traffic for the domain or a subdomain. For example, you might create records for example.com and www.example.com that route traffic to a web server that has an IP address of 192.0.2.234.

For more information about records, including information about functionality that is provided by Route 53–specific records, see Configuring Amazon Route 53 as your DNS service.

recursive name server

See DNS resolver.

reusable delegation set

A set of four authoritative name servers that you can use with more than one hosted zone. By default, Route 53 assigns a random selection of name servers to each new hosted zone.

To make it easier to migrate DNS service to Route 53 for a large number of domains, you can create a reusable delegation set. Then associate the reusable delegation set with new hosted zones. (You can't change the name servers that are associated with an existing hosted zone.)

You create a reusable delegation set and associate it with a hosted zone programmatically; using the Route 53 console isn't supported. For more information, see CreateHostedZone and CreateReusableDelegationSet in the *Amazon Route 53 API Reference*. The same feature is also available in the AWS SDKs, the AWS Command Line Interface, and AWS Tools for Windows PowerShell.

routing policy

A setting for records that determines how Route 53 responds to DNS queries. Route 53 supports these routing policies:

- **Simple routing policy** – Use to route internet traffic to a single resource that performs a given function for your domain, for example, a web server that serves content for the example.com website.
- **Failover routing policy** – Use when you want to configure active-passive failover.
- **Geolocation routing policy** – Use when you want to route internet traffic to your resources based on the location of your users.
- **Geoproximity routing policy** – Use when you want to route traffic based on the location of your resources and, optionally, shift traffic from resources in one location to resources in another location.
- **Latency routing policy** – Use when you have resources in multiple locations and you want to route traffic to the resource that provides the best latency.
- **IP-based routing policy** – Use when you want to route traffic based on the location of your users, and have the IP addresses that the traffic originates from.
- **Multivalue answer routing policy** – Use when you want Route 53 to respond to DNS queries with up to eight healthy records selected at random.
- **Weighted routing policy** – Use to route traffic to multiple resources in proportions that you specify.
- **AZ-affinity routing policy** – Use when you want to route traffic to resources in the same Availability Zone (AZ), Zone Group, or AWS Region that the query comes from, with automatic fallback to a less specific location.

For more information, see Choosing a routing policy.

subdomain

A domain name that has one or more labels prepended to the registered domain name. For example, if you register the domain name example.com, then www.example.com is a subdomain. If you create the hosted zone accounting.example.com for the example.com domain, then seattle.accounting.example.com is a subdomain.

To route traffic for a subdomain, create a record that has the name that you want, such as www.example.com, and specify the applicable values, such as the IP address of a web server.

time to live (TTL)

The amount of time, in seconds, that you want a DNS resolver to cache (store) the values for a record before submitting another request to Route 53 to get the current values for that record. If the DNS resolver receives another request for the same domain before the TTL expires, the resolver returns the cached value.

A longer TTL reduces your Route 53 charges, which are based in part on the number of DNS queries that Route 53 responds to. A shorter TTL reduces the time that DNS resolvers route traffic to older resources after you change the values in a record, for example, by changing the IP address for the web server for www.example.com.

Control and data plane concepts

Here's an overview of the concepts that are related to how Amazon Route 53 divides its functionality into a control and a data plane. Like most AWS services, Route 53 includes a control plane and a data plane. The control plane enables you to perform management operations such as creating, updating, and deleting resources. The data plane provides the service's core functionality. Both are built to be reliable. The control planes are optimized for data consistency, whereas the data planes are optimized for availability. The data plane's resilient design allows it to maintain availability even during rare disruptive events, during which the control plane might become unavailable. For this reason, use data plane functions where availability is important.

For Route 53 public and private DNS and health checks, the control plane is located in the us-east-1 AWS Region and the data planes are globally distributed.

Amazon Route 53 is divided into control and data planes as follows:

- For Route 53 public and private DNS, the control plane consists of the Route 53 APIs, which allow you to manage DNS entries. This includes both the Route 53 and Traffic Flow APIs. The

Route 53 console is located in the us-east-1 AWS Region, but if AWS determines that there is an impairment in that Region, the Route 53 console is served by the us-west-2 AWS Region. The data plane is the authoritative DNS service. It runs across over 200 Points of Presence (PoP) locations, answering DNS queries based on your hosted zones and health check data.

- For Route 53 health checks, the control plane consists of the Route 53 APIs that you can use to create, update, and delete health checks. The Route 53 health checks console is located in the us-east-1 AWS Region, but if AWS determines that there is an impairment in that Region, the Route 53 health checks console is served by the us-west-2 AWS Region. The data plane is the globally distributed service that performs health checks, aggregates the results, and delivers them to the data planes of Route 53 public and private DNS and AWS Global Accelerator.
- For Route 53 VPC Resolver, the control plane consists of the VPC Resolver APIs that allow you to manage Amazon VPC settings, Resolver rules, query logging policies, and DNS Firewall policies. The data plane is the DNS resolver service. It answers DNS queries in your VPC, forwards queries to other resolvers through endpoints, and applies DNS Firewall policies to filter DNS queries. VPC Resolver is a Regional service. Its control and data planes run independently in each AWS Region.
- Route 53 domain registrations are managed only on the control plane in the us-east-1 AWS Region.

For more information about data planes, control planes, and how AWS builds services to meet high availability targets, see the Static stability using Availability Zones paper in the Amazon Builders' Library.

Health checking concepts

Here's an overview of the concepts that are related to Amazon Route 53 health checking.

- DNS failover
- endpoint
- health check

DNS failover

A method for routing traffic away from unhealthy resources and to healthy resources. When you have more than one resource performing the same function—for example, more than one web server or mail server—you can configure Route 53 health checks to check the health of your resources. Then configure records in your hosted zone to route traffic only to healthy resources.

For more information, see Configuring DNS failover.

endpoint

The resource, such as a web server or an email server, that you configure a health check to monitor the health of. You can specify an endpoint by IPv4 address (192.0.2.243), by IPv6 address (2001:0db8:85a3:0000:0000:abcd:0001:2345), or by domain name (example.com).

> **Note**
>
> You can also create health checks that monitor the status of other health checks or that monitor the alarm state of a CloudWatch alarm.

health check

A Route 53 component that you can use to do the following:

- Monitor whether a specified endpoint, such as a web server, is healthy
- Optionally, get notified when an endpoint becomes unhealthy
- Optionally, configure DNS failover, so you can reroute internet traffic from an unhealthy resource to a healthy resource

For more information about how to create and use health checks, see Creating Amazon Route 53 health checks.

How to get started with Amazon Route 53

For information about getting started with Amazon Route 53, see the following topics in this guide:

- Set up Amazon Route 53, which explains how to sign up for AWS, how to secure access to your AWS account, and how to set up programmatic access to Route 53
- Getting started with Amazon Route 53, which describes how to register a domain name, create an Amazon S3 bucket configured to host a static website, and route internet traffic to the website

Accessing Amazon Route 53

You can access Amazon Route 53 in the following ways:

- **AWS Management Console** – The procedures throughout this guide explain how to use the AWS Management Console to perform tasks.
- **AWS SDKs** – If you're using a programming language that AWS provides an SDK for, you can use an SDK to access Route 53. SDKs simplify authentication, integrate with your development environment, and provide access to Route 53 commands. For more information, see Tools for Amazon Web Services.
- **Route 53 API** – If you're using a programming language that an SDK isn't available for, see the Amazon Route 53 API Reference for information about API actions and about how to make API requests.
- **AWS Command Line Interface** – For more information, see Getting set up with the AWS Command Line Interface in the *AWS Command Line Interface User Guide*.
- **AWS Tools for Windows PowerShell** – For more information, see Setting up the AWS Tools for Windows PowerShell in the *AWS Tools for PowerShell User Guide*.

AWS Identity and Access Management

Amazon Route 53 integrates with AWS Identity and Access Management (IAM), a service that your organization can use to do the following:

- Create users and groups under your organization's AWS account
- Easily share your AWS account resources among the users in the account
- Assign unique security credentials to each user
- Granularly control user access to services and resources

For example, you can use IAM with Route 53 to control which users in your AWS account can create a new hosted zone or change records.

For general information about IAM, see the following:

- Identity and access management in Amazon Route 53
- Identity and Access Management (IAM)

- IAM User Guide

Amazon Route 53 pricing and billing

As with other AWS products, there are no contracts or minimum commitments for using Amazon Route 53. You pay only for the hosted zones that you configure and the number of DNS queries that Route 53 answers. For more information, see Amazon Route 53 Pricing.

For information about billing for AWS services, including how to view your bill and manage your account and payments, see the AWS Billing User Guide.

Using Route 53 with an AWS SDK

AWS software development kits (SDKs) are available for many popular programming languages. Each SDK provides an API, code examples, and documentation that make it easier for developers to build applications in their preferred language.

SDK documentation	Code examples
AWS SDK for C++	AWS SDK for C++ code examples
AWS CLI	AWS CLI code examples
AWS SDK for Go	AWS SDK for Go code examples
AWS SDK for Java	AWS SDK for Java code examples
AWS SDK for JavaScript	AWS SDK for JavaScript code examples
AWS SDK for Kotlin	AWS SDK for Kotlin code examples
AWS SDK for .NET	AWS SDK for .NET code examples
AWS SDK for PHP	AWS SDK for PHP code examples
AWS Tools for PowerShell	AWS Tools for PowerShell code examples
AWS SDK for Python v4	AWS SDK for Python v4 code examples
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) code examples

SDK documentation	Code examples
AWS SDK for Ruby	AWS SDK for Ruby code examples
AWS SDK for Rust	AWS SDK for Rust code examples
AWS SDK for SAP ABAP	AWS SDK for SAP ABAP code examples
AWS SDK for Swift	AWS SDK for Swift code examples

For examples specific to Route 53, see Code examples for Route 53 using AWS SDKs.

Example availability

Can't find what you need? Request a code example by using the **Provide feedback** link at the bottom of this page.

Getting started with Amazon Route 53

Get started by registering a domain with Amazon Route 53 and configuring Route 53 to respond to DNS queries for a static website. The first topic hosts a static website in an open Amazon S3 bucket. The second topic uses a Amazon CloudFront distribution to serve the website with SSL/TLS.

Estimated cost

- There's an annual fee to register a domain, which varies by top-level domain, such as .com. For more information, see Route 53 Pricing for Domain Registration. The pricing list doesn't include registry premium prices, which vary by domain name. To check whether a domain is premium and to see its price, search for it in the Route 53 console or use the `GetDomainPrices` API operation. This fee is not refundable.
- When you register a domain, a hosted zone with the same name is created. You use the hosted zone to specify where Route 53 routes traffic for your domain.
- As part of this setup, you create an Amazon S3 bucket and upload a sample web page. If you're a new AWS customer, you can get started with Amazon S3 for free. If you're an existing AWS customer, charges are based on how much data you store, the number of requests for your data, and the amount of data transferred. For more information, see Amazon S3 Pricing.
- CloudFront charges are based on the number of requests for your data, the number of edge locations you use, and the amount of data transferred. For more information, see CloudFront Pricing.

Topics

- Set up Amazon Route 53
- Use your domain for a static website in an Amazon S3 bucket
- Use an Amazon CloudFront distribution to serve a static website

Set up Amazon Route 53

Topics

- Sign up for an AWS account
- Download tools

Sign up for an AWS account

To get started with AWS, you need an AWS account. There are two ways to sign up for AWS. If you want complete control over account configuration and where your content is stored, or if you have regulated workloads such as HIPAA or FedRAMP, use Sign up for AWS (advanced). If you are a new user and want to use a login you already own to sign in to AWS and create an AWS environment with preconfigured defaults in select AWS Regions, use Sign up for AWS.

Download tools

The AWS Management Console includes a console for Amazon Route 53, but if you want to access the services programmatically, see the following:

- The API guide document the operations that the services support and provide links to the related SDK and CLI documentation:
  - Amazon Route 53 API Reference
- To call an API without having to handle low-level details like assembling raw HTTP requests, you can use an AWS SDK. The AWS SDKs provide functions and data types that encapsulate the functionality of AWS services. To download an AWS SDK and access installation instructions, see the applicable page:
  - Java
  - JavaScript
  - .NET
  - Node.js
  - PHP
  - Python
  - Ruby

  For a complete list of AWS SDKs, see Tools for Amazon Web Services.
- You can use the AWS Command Line Interface (AWS CLI) to control multiple AWS services from the command line. You can also automate your commands using scripts. For more information, see AWS Command Line Interface.
- AWS Tools for Windows PowerShell supports these AWS services. For more information, see AWS Tools for PowerShell Cmdlet Reference.

Use your domain for a static website in an Amazon S3 bucket

This topic shows you how to use Amazon Route 53 to route DNS traffic for your domain to an Amazon Simple Storage Service bucket that hosts a static website. You'll create alias records that point your domain to the S3 website endpoint.

This topic is part of a complete static website setup workflow. For general information about routing traffic to any S3 bucket, see Routing traffic to a website that is hosted in an Amazon S3 bucket.

After you complete these steps, you have the following resources configured:

- Amazon S3 buckets configured for website hosting and redirect (if using www subdomain)
- Website content uploaded to your bucket
- Public access configured for your website bucket

Visitors can access your static website using your custom domain name.

> **Note**
>
> You can also transfer an existing domain to Route 53, but the process is more complex and time-consuming than registering a new domain. For more information, see Transferring registration for a domain to Amazon Route 53.

Prerequisites

Before you begin, complete these steps:

- Complete the steps in Set up Amazon Route 53.
- Register a domain name using Amazon Route 53. For more information, see Registering a new domain.
- Configure an Amazon Simple Storage Service bucket for static website hosting. For complete instructions, see Tutorial: Configuring a static website using a custom domain registered with Route 53 in the *Amazon Simple Storage Service User Guide*.

Step 1: Route DNS traffic for your domain to your website bucket

Now that you have an Amazon Simple Storage Service bucket configured for static website hosting, use Amazon Route 53 to route DNS traffic for your domain to the bucket. This enables visitors to access your website using your custom domain name.

To route traffic to your website

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.

   > **Note**
   >
   > When you registered your domain, Amazon Route 53 automatically created a hosted zone with the same name. A hosted zone contains information about how you want Route 53 to route traffic for the domain.

3. In the list of hosted zones, choose the name of your domain.
4. Choose **Create record**.
5. Specify the following values:

   Record name

   Leave blank to create a record for your root domain.

   Record type

   Choose **A - Routes traffic to an IPv4 address and some AWS resources**.

   Alias

   Turn on **Alias**.

   Route traffic to

   Choose **Alias to S3 website endpoint**.

   Choose the Region where you created your S3 bucket.

   Choose your S3 bucket. The bucket name should match the name of your domain. In the list, the bucket name appears with the Amazon S3 website endpoint for the Region

where the bucket was created, for example, `s3-website-us-west-1.amazonaws.com (example.com)`.

If your bucket does not appear in the list, enter the Amazon S3 website endpoint for the Region where the bucket was created, for example, **`s3-website-us-west-2.amazonaws.com`**. For a complete list of Amazon S3 website endpoints, see Amazon S3 Website endpoints.

Evaluate target health

Accept the default value of **No**.

6. Choose **Create records**.

(Optional) To add an alias record for your subdomain (`www.example.com`)

If you created a bucket for your subdomain, add an alias record for it also.

1. Choose **Create record**.
2. Specify the following values:

Record name

Enter **`www`**.

Record type

Choose **A - Routes traffic to an IPv4 address and some AWS resources**.

Alias

Turn on **Alias**.

Route traffic to

Choose **Alias to S3 website endpoint**.

Choose the Region where you created your S3 bucket.

Choose your S3 bucket for the subdomain, for example, `s3-website-us-west-2.amazonaws.com (www.example.com)`.

Evaluate target health

Accept the default value of **No**.

3. Choose **Create records**.

Step 2: Test your website

To verify that the website is working correctly, open a web browser and browse to the following URLs:

- http://*your-domain-name*, for example, `example.com` – Displays the index document in the *your-domain-name* bucket
- http://www.*your-domain-name* for example, `www.example.com` – Redirects your request to the *your-domain-name* bucket

In some cases, you might need to clear the cache to see the expected behavior.

For more advanced information about routing your internet traffic, see Configuring Amazon Route 53 as your DNS service. For information about routing your internet traffic to AWS resources, see Routing internet traffic to your AWS resources.

> **Note**
>
> Amazon S3 does not support HTTPS access to the website. If you want to use HTTPS, you can use Amazon CloudFront to serve a static website hosted on Amazon S3. For more information, see Use an Amazon CloudFront distribution to serve a static website.

Use an Amazon CloudFront distribution to serve a static website

This topic shows you how to route DNS traffic for your domain to Amazon CloudFront distributions that serve a static website. You create alias records that point your domain and subdomain to your CloudFront distributions.

This topic is part of a static website setup workflow. For general information about routing traffic to a CloudFront distribution, see Routing traffic to an Amazon CloudFront distribution by using your domain name.

After you complete these steps, you have the following resources configured:

- An SSL/TLS certificate for your domain in AWS Certificate Manager
- Amazon S3 buckets configured for website hosting and redirect
- CloudFront distributions for both your root domain and subdomain

Visitors can then reach your website at your custom domain name with HTTPS security from CloudFront.

Prerequisites

Before you begin, complete these steps:

- Complete the steps in Set up Amazon Route 53.
- Register a domain name using Amazon Route 53. For more information, see Registering a new domain.
- Create a secure static website with Amazon CloudFront and Amazon Simple Storage Service. For instructions, see Get started with a secure static website in the *Amazon CloudFront Developer Guide*.

Step 1: Route DNS traffic for your domain to your CloudFront distribution

Now that you have Amazon CloudFront distributions for your website, route DNS traffic for your domain to those distributions. This lets visitors reach your website at your custom domain name.

For more information about routing traffic to CloudFront distributions, see Routing traffic to an Amazon CloudFront distribution by using your domain name.

To route traffic to your website

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.

> **Note**
>
> When you registered your domain, Amazon Route 53 created a hosted zone with the same name. This hosted zone stores information about how Route 53 routes traffic for the domain.

3. In the list of hosted zones, choose the name of your domain.
4. Choose **Create record**.
5. Specify the following values:

   Record name

   For your subdomain record, enter **`www`**.

   Record type

   Choose **A - Routes traffic to an IPv4 address and some AWS resources**.

   Alias

   Turn on **Alias**.

   Route traffic to

   Choose **Alias to CloudFront distribution**.

   Choose the us-east-1 Region.

   Choose your CloudFront distribution. The distribution name matches the value in the **Domain name** column in the CloudFront console (for example, `d111111abcdef8.cloudfront.net`).

   Evaluate target health

   Accept the default value of **No**.
6. Choose **Create records**.

To add an alias record for your root domain (`example.com`)

Add an alias record for your root domain too. Point it to the CloudFront distribution that redirects traffic to `www.example.com`.

1. Choose **Create record**.
2. Specify the following values:

   Record name

   Leave blank to create a record for your root domain.

   Record type

   Choose **A - Routes traffic to an IPv4 address and some AWS resources**.

   Alias

   Turn on **Alias**.

   Route traffic to

   Choose **Alias to CloudFront distribution**.

   Choose the us-east-1 Region.

   Choose your root domain CloudFront distribution.

   Evaluate target health

   Accept the default value of **No**.
3. Choose **Create records**.

Step 2: Test your website

To check that your website works, open a web browser and go to the following URLs:

- https://www.*your-domain-name*, for example, `www.example.com` – Displays the index document in the *www.your-domain-name* bucket
- https://*your-domain-name* for example, `example.com` – Redirects your request to the *www.your-domain-name* bucket

In some cases, you might need to clear the cache to see the expected behavior.

For more information about routing internet traffic, see Configuring Amazon Route 53 as your DNS service. To route traffic to AWS resources, see Routing internet traffic to your AWS resources.

Integration with other services

You can integrate Amazon Route 53 with other AWS services to log requests sent to the Route 53 API, monitor the status of your resources, and assign tags to your resources. You can also use Route 53 to route internet traffic to your AWS resources.

Topics

- Logging, monitoring, and tagging
- Routing traffic to other AWS resources

Logging, monitoring, and tagging

AWS CloudTrail

Amazon Route 53 is integrated with AWS CloudTrail, a service that captures information about every request sent to the Route 53 API by your AWS account. You can use the CloudTrail log files to see which requests were made to Route 53, the source IP address, who made the request, and when.

For more information, see Logging Amazon Route 53 API calls with AWS CloudTrail.

Amazon CloudWatch

You can use Amazon CloudWatch to monitor the status—healthy or unhealthy—of your Route 53 health checks. Health checks monitor the health and performance of your web applications, web servers, and other resources. At regular intervals that you specify, Route 53 sends requests over the internet to your resource to verify that it's reachable and functional.

For more information, see Monitoring health checks using CloudWatch.

Tag Editor

A tag is a label that you assign to an AWS resource, including Route 53 domains, hosted zones, and health checks. Each tag consists of a key and a value, both of which you define. For example, you might assign a tag to a domain registration that has the key "Customer" and the value "Example Corp." Tags serve many purposes. One common use is to categorize and track your AWS costs.

For more information, see Tagging Amazon Route 53 resources.

Routing traffic to other AWS resources

You can use Amazon Route 53 to route traffic to a variety of AWS resources.

Amazon API Gateway

Amazon API Gateway lets you create, publish, maintain, monitor, and secure APIs at any scale. You can create APIs that access AWS or other web services, and data stored in the AWS Cloud.

You can use Route 53 to route traffic to an API Gateway API. For more information, see Routing traffic to an Amazon API Gateway API by using your domain name.

Amazon CloudFront

To speed up delivery of your web content, you can use Amazon CloudFront, the AWS content delivery network (CDN). CloudFront can deliver your entire website—including dynamic, static, streaming, and interactive content—by using a global network of edge locations. CloudFront routes requests for your content to the edge location that gives your users the lowest latency. You can use Route 53 to route traffic for your domain to your CloudFront distribution. For more information, see Routing traffic to an Amazon CloudFront distribution by using your domain name.

Amazon EC2

Amazon EC2 provides scalable computing capacity in the AWS Cloud. You can launch an EC2 instance using a preconfigured template (an Amazon Machine Image, or AMI). The AMI includes the operating system (Linux or Microsoft Windows) and software such as a web server or database.

If you host a website or run a web application on an EC2 instance, you can route traffic for your domain to your server by using Route 53. For more information, see Routing traffic to an Amazon EC2 instance.

AWS Elastic Beanstalk

If you use AWS Elastic Beanstalk to deploy and manage applications in the AWS Cloud, you can use Route 53 to route DNS traffic for your domain to an Elastic Beanstalk environment. For more information, see Routing traffic to an AWS Elastic Beanstalk environment.

Elastic Load Balancing

If you host a website on multiple Amazon EC2 instances, you can distribute traffic across the instances by using an Elastic Load Balancing (ELB) load balancer. ELB scales the load balancer

as traffic to your website changes over time. The load balancer also monitors the health of its registered instances and routes traffic only to healthy instances.

You can use Route 53 to route traffic for your domain to your Classic, Application, or Network Load Balancer. For more information, see Routing traffic to an ELB load balancer.

Amazon Lightsail

Amazon Lightsail provides compute, storage, and networking to deploy and manage websites, web apps, and databases in the cloud for a low monthly price.

If you use Lightsail, you can use Route 53 to route traffic to your instance. For more information, see Using Route 53 to point a domain to an Amazon Lightsail instance.

Amazon S3

Amazon Simple Storage Service (Amazon S3) provides secure, durable, highly scalable cloud storage. You can configure an S3 bucket to host a static website that can include web pages and client-side scripts. (S3 doesn't support server-side scripting.) You can use Route 53 to route traffic to an Amazon S3 bucket. For more information, see the following topics:

- For information about routing traffic to a bucket, see Routing traffic to a website that is hosted in an Amazon S3 bucket.
- For a more detailed explanation of how to host a static website in an S3 bucket, see Getting started with Amazon Route 53.

Amazon Virtual Private Cloud (Amazon VPC)

An interface endpoint lets you connect to services powered by AWS PrivateLink. These services include some AWS services, services hosted by other AWS customers and partners in their own VPCs (called *endpoint services*), and supported AWS Marketplace partner services.

You can use Route 53 to route traffic to an interface endpoint. For more information, see Routing traffic to an Amazon Virtual Private Cloud interface endpoint by using your domain name.

Amazon WorkMail

If you use Amazon WorkMail for business email and Route 53 as your DNS service, you can use Route 53 to route traffic to your Amazon WorkMail email domain. For more information, see Routing traffic to Amazon WorkMail.

For more information see Routing internet traffic to your AWS resources.

DNS domain name format

Domain names (including the names of domains, hosted zones, and records) consist of a series of labels separated by dots. Each label can be up to 63 bytes long. The total length of a domain name cannot exceed 255 bytes, including the dots. Amazon Route 53 supports any valid domain name.

Naming requirements depend on whether you're registering a domain name or you're specifying the name of a hosted zone or a record. See the applicable topic.

Formatting domain names for domain name registration

For domain name registration, a domain name can contain only the characters a-z, 0-9, and - (hyphen). You can't specify a hyphen at the beginning or end of a label.

For information about how to register an internationalized domain name (IDN), see Formatting internationalized domain names.

Formatting domain names for hosted zones and records

For hosted zones and records, the domain name can include any of the following printable ASCII characters (excluding spaces):

- a-z
- 0-9
- - (hyphen)
- ! " # $ % & ' ( ) * + , - / : ; < = > ? @ [\ ] ^ _ ` { | } ~ .

Amazon Route 53 stores alphabetic characters as lowercase letters (a-z), regardless of how you specify them: as uppercase letters, lowercase letters, or the corresponding letters in escape codes.

If your domain name contains any of the following characters, you must specify the characters by using escape codes in the format *three-digit octal code*:

- Characters 000 to 040 octal (0 to 32 decimal, 0x00 to 0x20 hexadecimal)
- Characters 177 to 377 octal (127 to 255 decimal, 0x7F to 0xFF hexadecimal)

- . (period), character 056 octal (46 decimal, 0x2E hexadecimal), when used as a character in a domain name. When using . as a delimiter between labels, you do not need to use an escape code.

If the domain name includes any characters other than a to z, 0 to 9, - (hyphen), or _ (underscore), Route 53 API actions return the characters as escape codes. This is true whether you specify the characters as characters or as escape codes when you create the entity. The Route 53 console displays the characters as characters, not as escape codes.

For a list of ASCII characters the corresponding octal codes, do an internet search on "ascii table".

To specify an internationalized domain name (IDN), convert the name to Punycode. For more information, see Formatting internationalized domain names.

Using an asterisk (*) in the names of hosted zones and records

You can create hosted zones and records that include * in the name.

Hosted zones

- You can't include an * in the leftmost label in a domain name. For example, *.example.com is not allowed.
- If you include * in other positions, DNS treats it as an * character (ASCII 42), not as a wildcard.

Records

DNS treats the * character either as a wildcard or as the * character (ASCII 42), depending on where it appears in the name. Note the following restrictions on using * as a wildcard in the name of a record:

- The * must replace the leftmost label in a domain name, for example, *.example.com or *.acme.example.com. If you include * in any other position, such as prod.*.example.com, DNS treats it as an * character (ASCII 42), not as a wildcard.
- The * must replace the entire label. For example, you can't specify *prod.example.com or prod*.example.com.
- Specific domain names take precedence. For example, if you create records for *.example.com and acme.example.com, Route 53 always responds to DNS queries for acme.example.com with the values in the acme.example.com record.

- The * applies to DNS queries for the subdomain level that includes the asterisk, and all the subdomains of that subdomain. For example, if you create a record named *.example.com, Route 53 uses the values in that record to respond to DNS queries for zenith.example.com, acme.zenith.example.com, and pinnacle.acme.zenith.example.com (if there are no records of any type for that hosted zone).

  If you create a record named *.example.com and there's no example.com record, Route 53 responds to DNS queries for example.com with `NXDOMAIN` (non-existent domain).

  > **Note**
  >
  > If you're using Route 53 with AWS Certificate Manager, note that ACM wildcard certificates only protect one subdomain level. For example, an ACM certificate for *.example.com protects login.example.com and test.example.com, but not test.login.example.com. For more information, see AWS Certificate Manager certificate characteristics and limitations.

- You can configure Route 53 to return the same response to DNS queries both for all subdomains at the same level and for the domain name. For example, you can configure Route 53 to respond to DNS queries such as acme.example.com and zenith.example.com using the example.com record. Perform the following steps:

  1. Create a record for the domain, such as example.com.
  2. Create an alias record for the subdomain, such as *.example.com. Specify the record that you created in step 1 as the target for the alias record.

- You can't use the * as a wildcard for records that have a type of NS.

Formatting internationalized domain names

When you register a new domain name or create hosted zones and records, you can specify letters other than a-z (for example, the ç in French), characters in other alphabets (for example, Cyrillic or Arabic), and characters in Chinese, Japanese, or Korean. Amazon Route 53 stores these internationalized domain names (IDNs) in Punycode, which represents Unicode characters as ASCII strings.

If you're registering a domain name, note the following:

- You can use characters other than a-z, 0-9, and - (hyphen) only if the top-level domain (TLD) supports IDNs and supports the language that you want to use. To determine which languages a TLD supports, see Domains that you can register with Amazon Route 53.
- You can specify a name in an unsupported language if the name contains only the letters a-z. For example, if a TLD doesn't support French but the name that you want to use includes only the characters a-z without diacritical marks, you can still use that name. In this example, a name that includes a "c" is allowed; a name that contains a "ç" is not.
- If a TLD doesn't support IDNs or doesn't support the language that you want to use for your domain name, you also can't specify the name in Punycode even though the Punycode includes only a-z, 0-9, and -.

The following example shows the Punycode representation of the internationalized domain name 中国.asia:

```
xn--fiqs8s.asia
```

When you enter an IDN in the address bar of a modern browser, the browser converts it to Punycode before submitting a DNS query or making an HTTP request.

How you enter an IDN depends on what you're creating (domain names, hosted zones, or records), and how you're creating it (API, SDK, or Route 53 console):

- If you're using the Route 53 API or one of the AWS SDKs, you can programmatically convert a Unicode value to Punycode. For example, if you're using Java, you can convert a Unicode value to Punycode by using the **toASCII** method of the java.net.IDN library.
- If you're using the Route 53 console to register a domain name, you can paste the name, including Unicode characters, into the name field, and the console converts the value to Punycode before saving it.
- If you're using the Route 53 console to create hosted zones or records, you need to convert the domain name to Punycode before you enter the name in the applicable **Name** field. For information about online converters, perform an internet search on "punycode converter".

If you're registering a domain name, note that not all top-level domains (TLDs) support IDNs. For a list of TLDs supported by Route 53, see Domains that you can register with Amazon Route 53. TLDs that don't support IDNs are noted.

Registering and managing domains using Amazon Route 53

When you want to get a new domain name, such as the **example.com** part of the URL http://example.com, you can register it with Amazon Route 53. You can also transfer the registration for existing domains from other registrars to Route 53 or transfer the registration for domains that you register with Route 53 to another registrar.

The procedures in this chapter explain how to register and transfer domains using the Route 53 console, and how to edit domain settings and view domain status. If you're only registering and managing a few domains, using the console is the easiest way.

If you need to register and manage a lot of domains, you might prefer to make changes programmatically. For more information, see Set up Amazon Route 53.

> **Note**
>
> If you are using a language for which an AWS SDK exists, use the SDK rather than trying to work your way through the APIs. The SDKs make authentication simpler, integrate easily with your development environment, and provide easy access to Route 53 commands.

Domain name registration services are provided under our Domain Name Registration Agreement.

Topics

- Registering new domains
- Updating domain settings
- Renewing registration for a domain
- Restoring an expired or deleted domain
- Replacing the hosted zone for a domain that is registered with Route 53
- Transferring domains
- Resending authorization and confirmation emails
- Configuring DNSSEC for a domain
- Finding your registrar and other information about your domain

- Deleting a domain name registration
- Contacting AWS Support about domain registration issues
- Downloading a domain billing report
- Domains that you can register with Amazon Route 53

Registering new domains

This section covers the following topics related to registering new domains with Amazon Route 53:

1. Registering a new domain:
   - Learn the step-by-step procedure for registering a new domain using the Route 53 console.
   - Understand the considerations and prerequisites for domain registration, such as contacting AWS Support for issues, pricing, supported top-level domains (TLDs), and automatic hosted zone creation.
2. Values that you specify when you register or transfer a domain:
   - Discover the values you need to provide when registering or transferring a domain, including contact information, privacy protection settings, and automatic renewal options.
   - Understand the implications of changing certain values, such as the domain owner or registrant email address.
3. Values that Amazon Route 53 returns when you register a domain:
   - Learn about the values that Route 53 returns after successful domain registration, including registration date, expiration date, domain status codes, transfer lock status, and name servers.
4. Viewing the status of a domain registration:
   - Find out how to view the current status of your domain registration, including the ICANN status codes and any actions required from your end, such as verifying the registrant email address.

Registering a new domain

Register a new domain or update name servers for an existing domain

You can use Amazon Route 53 with domains you register with Route 53, and with domains you have registered with other DNS providers. Depending on your DNS provider, choose one of the following procedures to register and use a new domain with Route 53:

- For registering a new domain, see To register a new domain using Route 53.
- For an existing domain, see Making Amazon Route 53 the DNS service for an existing domain.
- For moving a domain to another registrar, see update name servers when you want to use another DNS service.

Considerations for domain registration

Before you start, note the following:

Contacting AWS Support

If you encounter issues while registering a domain, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

Domain registration pricing

For information about the cost to register domains, see Amazon Route 53 Pricing for Domain Registration.

Supported domains

For a list of supported TLDs, see Domains that you can register with Amazon Route 53.

You can't change a domain name after you register it

If you accidentally register the wrong domain name, you can't change it. Instead, you need to register another domain name and specify the correct name. You also can't get a refund for a domain name that you registered accidentally.

AWS credits

You can't use AWS credits to pay the fee for registering a new domain with Route 53.

Charges for hosted zones

When you register a domain with Route 53, Route 53 automatically creates a hosted zone for the domain and charges a small monthly fee for the hosted zone in addition to the annual charge for the domain registration. This hosted zone is where you store information about how to route traffic for your domain, for example, to an Amazon EC2 instance or a CloudFront distribution. If you don't want to use your domain right now, you can delete the hosted zone; if you delete it within 12 hours of registering the domain, there won't be any charge for the hosted zone on your AWS bill. Route 53 also charges a small fee for the DNS queries that it receives for your domain. For more information, see Amazon Route 53 Pricing.

Replacing the hosted zone for a domain

If you create a new hosted zone for a domain, you must also update the name servers for the domain to use the same name servers as the new hosted zone. For details see, Replacing the hosted zone for a domain that is registered with Route 53

Do not use a Route 53 domain for the root user email on the same account

> **Important**
>
> Do not use a domain registered in Route 53 as the email domain for the root user of the same AWS account. If the account is closed or suspended, the domain registration is suspended within 5 days. Domain suspension stops email delivery, which prevents root user password recovery. Without email access, you cannot recover the account or reinstate the domain. After 30 days, the domain is permanently deleted with no remediation path.
>
> To avoid this circular dependency, use one of the following architectures:
>
> - Register domains used for root user email on a separate, dedicated AWS account (such as a management account).
> - Use an email address hosted outside AWS for the root user.

For more information, see Best practices for Amazon Route 53 domain registration.

To register a new domain using Route 53

To register a new domain using Route 53

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Domains** and then **Registered domains**.
3. On the **Registered domains** page, choose **Register domains**.

   a. In the **Search for domain** section, enter the domain name that you want to register, and choose **Search** to find out whether the domain name is available.

   If the domain name that you want to register contains characters other than a-z, A-Z, 0-9, and - (hyphen), note the following:

- You can enter the name using the applicable characters. You don't need to convert the name to Punycode.
- A list of languages appears. Choose the language of the specified name. For example, if you enter příklad ("example" in Czech), choose Czech (CES) or Czech (CZE).

> **Note**
>
> For languages that have more than one code, you might need to try both of them. Even though CES and CZE are synonymous, some TLD registries support only one or the other.

For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.

If the domain you entered is available, it will be displayed, if not, similar domains will be displayed as suggestions.

You can choose up to five domains to register. The domains you select appear in the **Selected domains** list.

b. To register more domains, repeat steps 3a through 3b.

4. Choose **Proceed to checkout**.
5. On the **Pricing** page, choose the number of years that you want to register the domain for and whether you want Route 53 to automatically renew your domain registration before the expiration date.

> **Note**
>
> Multi-year registration is not supported for premium domains.

> **Note**
>
> Domain name registrations and renewals are not refundable. If you enable automatic domain renewal and you decide that you don't want the domain name after Route 53 renews the registration, you can't get a refund for the cost of the renewal.

Choose **Next**.

6. On the **Contact information** page, enter contact information for the domain registrant, admin, tech, and billing contacts. The values that you enter here are applied to all of the domains that you're registering. For more information, see Values that you specify when you register or transfer a domain.

> **Important**
>
> The contact you list as the registrant during domain registration will have certain rights as the Registered Name Holder of the domain name, under the ICANN Transfer Policy. Most domains will be deleted upon closure of your AWS account (for more information, see My AWS account is closed or permanently closed, and my domain is registered with Route 53), however if a domain remains in a closed account, the contact you listed as the registrant might have the ability to request a transfer of the domain name to an external registrar. Therefore, it is important that the registrant contact you list is either yourself or another person you trust to act responsibly.

Note the following considerations:

First Name and Last Name

For **First Name** and **Last Name**, we recommend that you specify the name on your official ID. For some changes to domain settings, some domain registries require that you provide proof of identity. The name on your ID must match the name of the registrant contact for the domain.

Different contacts

By default, Route 53 uses the same information for all three contacts. If you want to enter different information for one or more contacts, change the value of **Same as registrant contact** toggle switch to off position.

> **Note**
>
> For .it domains, the registrant and administrative contacts must be the same.

> **Note**
>
> For .jp domains, the technical and administrative contacts must be the same.

Multiple domains

If you're registering more than one domain, Route 53 uses the same contact information for all of the domains.

Additional required information

For some top-level domains (TLDs), additional information is required. For these TLDs, enter the applicable values after the **Postal/Zip Code** field.

Privacy protection

Choose whether you want to hide your contact information from WHOIS queries.

> **Note**
>
> You must specify the same privacy setting for the administrative, registrant, technical, and billing contacts.

For more information, see the following topics:

- Enabling or disabling privacy protection for contact information for a domain
- Domains that you can register with Amazon Route 53

> **Note**
>
> To enable privacy protection for .uk, .co.uk, .me.uk, and .org.uk domains, you must open a support case and request privacy protection.

Choose **Next**.

7. On the **Review** page, review the information that you entered, and optionally correct it, read the terms of service, and select the check box to confirm that you've read the terms of service.

   Choose **Submit**.

8. In the navigation pane, choose **Domains** and then **Requests**.

   On this page you can view the status of domain and also if you need to respond to registrant contact verification email. You can also choose to resend the verification email.

   If you specified an email address for the registrant contact that has never been used to register a domain with Route 53, some TLD registries require you to verify that the address is valid.

   Route 53 sends a verification email from one of the following email addresses:

   - **noreply@registrar.amazon** – for TLDs registered by Amazon Registrar.
   - **noreply@domainnameverification.net** or **noreply@emailverification.info** – for TLDs registered by our registrar associate, Gandi. To determine who the registrar is for your TLD, see Finding your registrar.

   > **Important**
   >
   > The registrant contact must follow the instructions in the email to verify that the email was received, or Route 53 must suspend the domain as required by ICANN. When a domain is suspended, it's not accessible on the internet.

   > **Note**
   >
   > The verification emails might contain links to www.registrar.amazon. These links are safe to use.

For some TLDs, the verification email might not include the domain name in the subject or body of the email.

a. When you receive the verification email, choose the link in the email that verifies that the email address is valid. If you don't receive the email immediately, check your junk email folder.

b. Return to the **Requests** page. If the status doesn't automatically update to say **email-address is verified**, refresh the browser.

9. When domain registration is complete, your next step depends on whether you want to use Route 53 or another DNS service as the DNS service for the domain:

- **Route 53** – In the hosted zone that Route 53 created when you registered the domain, create records to tell Route 53 how you want to route traffic for the domain and subdomains.

  For example, when someone enters your domain name in a browser and that query is forwarded to Route 53, do you want Route 53 to respond to the query with the IP address of a web server in your data center or with the name of an ELB load balancer?

  For more information, see Working with records.

  Important

  If you create records in a hosted zone other than the one that Route 53 creates automatically, you must update the name servers for the domain to use the name servers for the new hosted zone.

- **Another DNS service** – Configure your new domain to route DNS queries to the other DNS service. Perform the procedure Updating name servers to use another registrar.

Values that you specify when you register or transfer a domain

> **Note**
>
> Route 53 has updated the domains console. During the transition period, you can continue to use the old console, or use the new console. Most of the information returned by Route 53 is the same for both consoles. The differences are noted in the following list.

When you register a domain or transfer domain registration to Amazon Route 53, you specify the values that are described in this topic.

> **Note**
>
> If you're registering more than one domain, Route 53 uses the values that you specify for all of the domains that are in your shopping cart.

You can also change values for a domain that is currently registered with Route 53. Note the following:

- If you change contact information for the domain, Route 53 sends an email notification to the registrant contact about the change. This email comes from noreply@registrar.amazon. For most changes, the registrant contact is not required to respond.
- For changes to contact information that also constitutes a change in ownership, Route 53 sends the registrant contact an additional email. ICANN requires that the registrant contact confirms receiving the email. For more information, see **First Name, Last Name** and **Organization** later in this section.

For more information about changing settings for an existing domain, see Updating domain settings.

Values that you specify

- My Registrant, Administrative, and Technical contacts are all the same
- Contact Type
- First Name, Last Name

- Organization
- Email
- Phone
- Address 1
- Address 2
- Country
- State
- City
- Postal/Zip Code
- Fields for selected top-level domains
- Privacy Protection
- Auto-renew

Same as the registrant contact

Specifies whether you want to use the same contact information for the registrant of the domain, the administrative contact, and the technical contact.

Contact Type

Category for this contact. Note the following:

- If you choose an option other than **Person**, you must enter an organization name.
- For some TLDs, the privacy protection available depends on the value that you choose for **Contact Type**. For the privacy protection settings for your TLD, see Domains that you can register with Amazon Route 53.
- For .es domains, the value of **Contact Type** must be **Person** for all three contacts.

First Name, Last Name

The first and last names of the contact.

> **Important**
>
> For **First Name** and **Last Name**, we recommend that you specify the name on your official ID. For some changes to domain settings, you must provide proof of identity, and the name on your ID must match the name of the registrant contact for the domain.

If you're transferring a domain to Route 53 and the following are true, then you're changing the owner of the domain:

- The contact type is **Person**.
- You're changing the **First Name** and/or **Last Name** fields for the registrant contact from the current settings.

In that case, ICANN requires that Route 53 email the registrant contact to get approval. Route 53 sends email from one of the following email addresses:

TLDs	Email address that approval email comes from
TLDs registered by Amazon Registrar	noreply@registrar.amazon
.fr	nic@nic.fr (The email is sent both to the current registrant contact and the new registrant contact.)
All others	noreply@domainnameverification.net or noreply@emailverification.info

To determine who the registrar is for your TLD, see Domains that you can register with Amazon Route 53.

Important

The registrant contact must follow the instructions in the email to confirm that the email was received, or Route 53 must suspend the domain as required by ICANN. When a domain is suspended, it's not accessible on the internet.

If you change the email address of the registrant contact, this email is sent to the former email address and the new email address for the registrant contact.

Some TLD registrars charge a fee for changing the domain owner. When you change one of these values, the Route 53 console displays a message that tells you whether there is a fee.

Organization

The organization that is associated with the contact, if any. For the registrant and administrative contacts, this is typically the organization that is registering the domain. For the technical contact, this might be the organization that manages the domain.

When the contact type is any value except **Person** and you change the **Organization** field for the registrant contact, you change the owner of the domain. ICANN requires that Route 53 email the registrant contact to get approval. Route 53 sends email from one of the following email addresses:

TLDs	Email address that approval email comes from
TLDs registered by Amazon Registrar	noreply@registrar.amazon
.fr	nic@nic.fr (The email is sent both to the current registrant contact and the new registrant contact.)
All others	noreply@domainnameverification.net or noreply@emailverification.info

To determine who the registrar is for your TLD, see Domains that you can register with Amazon Route 53.

If you change the email address of the registrant contact, this email is sent to the former email address and the new email address for the registrant contact.

Some TLD registrars charge a fee for changing the domain owner. When you change the value of **Organization**, the Route 53 console displays a message that tells you whether there is a fee.

Email

The email address for the contact.

If you change the email address for the registrant contact, Route 53 sends a notification email to the former email address and the new email address. This email comes from noreply@registrar.amazon.

Phone

The phone number for the contact:

- If you're entering a phone number for locations in the United States or Canada, enter **1** in the first field and the 10-digit area code and phone number in the second field.
- If you're entering a phone number for any other location, enter the country code in the first field, and enter the rest of the phone number in the second field. For a list of phone country codes, see the Wikipedia article List of country calling codes.

Address 1

The street address for the contact.

Address 2

Additional address information for the contact, for example, apartment number or mail stop.

Country

The country for the contact.

State

The state or province for the contact, if any.

City

The city for the contact.

Postal/Zip Code

The postal or zip code for the contact.

Fields for selected top-level domains

The following top-level domains (TLDs) require that you specify additional values:

- .com.au and .net.au
- .ca
- .es
- .fi
- .fr
- .it
- .ru
- .se

- .sg
- .co.uk, .me.uk, .org.uk, and .uk

In addition, many TLDs require a VAT identification number.

For information about valid values, see ExtraParam in the *Amazon Route 53 API Reference*.

Privacy Protection

Whether you want to conceal your contact information from WHOIS queries. If you select **Turn on privacy protection** (new console) or **Hide contact information** (old console), WHOIS ("who is") queries will return contact information for the registrar or the value "Protected by policy."

> **Note**
>
> You must specify the same privacy setting for the administrative, registrant, technical, and billing contacts.

If you select **Don't hide contact information**, you'll get more email spam at the email address that you specified.

Anyone can send a WHOIS query for a domain and get back all of the contact information for that domain. The WHOIS command is available in many operating systems, and it's also available as a web application on many websites.

> **Important**
>
> Although there are legitimate users for the contact information associated with your domain, the most common users are spammers, who target domain contacts with unwanted email and bogus offers. In general, we recommend that you choose **Hide contact information** for **Privacy Protection**.

To enable or disable privacy protection for some domains, you must open a support case and request privacy protection.

For more information about privacy protection, see the following topics:

- Enabling or disabling privacy protection for contact information for a domain
- Domains that you can register with Amazon Route 53

Auto Renew (Only available when editing domain settings)

Whether you want Route 53 to automatically renew the domain before it expires. The registration fee is charged to your AWS account. On the old console this setting is only available when editing domain settings. For more information, see Renewing registration for a domain.

> **Important**
>
> If you disable automatic renewal, registration for the domain will not be renewed when the expiration date passes, and you might lose control of the domain name.

The period during which you can renew a domain name varies by top-level domain (TLD). For an overview about renewing domains, see Renewing registration for a domain. For information about extending domain registration for a specified number of years, see Extending the registration period for a domain.

Values that Amazon Route 53 returns when you register a domain

When you register your domain with Amazon Route 53, Route 53 returns the following values in addition to the values that you specified.

Registered on

The date on which the domain was originally registered with Route 53.

Expires on

The date and time on which the current registration period expires, in Greenwich Mean Time (GMT).

The registration period is typically one year, although the registries for some top-level domains (TLDs) have longer registration periods. For the registration and renewal period for your TLD, see Domains that you can register with Amazon Route 53.

For most TLDs, you can extend the registration period by up to ten years. For more information, see Extending the registration period for a domain.

Domain name status code

The current status of the domain.

ICANN, the organization that maintains a central database of domain names, has developed a set of domain name status codes (also known as EPP status codes) that tell you the status of a variety of operations on a domain name. For example, registering a domain name, transferring a domain name to another registrar, renewing the registration for a domain name, and so on. All registrars use this same set of status codes.

For a current list of domain name status codes and an explanation of what each code means, go to the ICANN website and search for **epp status codes**. (Search on the ICANN website; web searches sometimes return an old version of the document.)

Transfer lock

Whether the domain is locked to reduce the possibility of someone transferring your domain to another registrar without your permission. If the domain is locked, the value of **Transfer Lock** is **On**. If the domain is not locked, the value is **Off**.

Auto renew

Whether Route 53 will automatically renew the registration for this domain shortly before the expiration date.

Authorization code

The code that is required if you want to transfer registration of this domain to another registrar. An authorization code is only generated when you request it. For information about transferring a domain to another registrar, see Transferring a domain from Amazon Route 53 to another registrar.

Name servers

The Route 53 servers that respond to DNS queries for this domain. We recommend that you don't delete Route 53 name servers.

For information about adding, changing, or deleting name servers, see Adding or changing name servers and glue records for a domain.

Viewing the status of a domain registration

ICANN, the organization that maintains a central database of domain names, has developed a set of domain name status codes (also known as EPP status codes) that tell you the status of a variety of operations, for example, registering a domain name, transferring a domain name to another

registrar, renewing the registration for a domain name, and so on. All registrars use this same set of status codes.

To view the status code for your domains, perform the following procedure.

To view the ICANN status code of a domain

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, expand **Domains** and choose **Registered Domains**.
3. Select the linked name of your domain.
4. If you need to take an action, such as resending the verification email to the registrant contact, a banner on top of the page will indicate the action you need to take.
5. For the current status of your domain, see the value of the **Domain status code** field.

   For a current list of domain name status codes and an explanation of what each code means, go to the ICANN website and search for **epp status codes**. (Search on the ICANN website; web searches sometimes return an old version of the document.)

You can also view the status of the registration on the **Requests** page.

To view the registration status

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, expand **Domains** and choose **Requests**.
3. On the **Requests** page, you can view the registration status, and additionally the status of any other actions, such as deleting domains, ,locking domain transfers, adding or deleting DNSSEC keys, you have taken on domains,

   Any action you might have to take to complete a process, such as verifying your email, is also listed.

   - To respond to an action request, select the radio-button next to the domain name and then select the action from the **Action** drop-down.

Updating domain settings

This section provides information on the following topics related to managing domain settings in Route 53:

1. Updating contact information and ownership for a domain:
   - Learn how to update contact information for a domain, including administrative, technical, registrant, and billing contacts.
   - Understand the process for changing the owner of a domain when the registry requires a Change of Domain Ownership form.
2. Enabling or disabling privacy protection for contact information for a domain:
   - Discover how to enable or disable privacy protection for contact information, which hides or reveals your personal details from WHOIS queries.
3. Enabling or disabling automatic renewal for a domain:
   - Find out how to enable or disable automatic renewal for a domain, which determines whether Route 53 automatically renews the registration before expiration.
4. Locking a domain to prevent unauthorized transfer to another registrar:
   - Learn how to lock a domain to prevent unauthorized transfer to another registrar, and how to disable the lock when needed.
5. Extending the registration period for a domain:
   - Understand the process for extending the registration period for a domain, typically up to ten years in one-year increments.
6. Updating name servers to use another registrar and Adding or changing name servers and glue records for a domain:
   - Discover how to update name servers to use another DNS service or configure white-label (vanity) name servers.
   - Learn about considerations and best practices when changing name servers and glue records.

Updating contact information and ownership for a domain

For the administrative and technical contacts for a domain, you can change all contact information without having to authorize the changes. For more information, see Updating contact information for a domain.

For the registrant contact, you can change most values without having to authorize the changes. However, for some TLDs, changing the owner of a domain requires authorization. For more information, see the applicable topic.

Topics

- What triggers a domain ownership change?
- TLDs that require special processing to change the owner
- Updating contact information for a domain
- Changing the owner of a domain when the registry requires a Change of Domain Ownership form

What triggers a domain ownership change?

> **Important**
>
> The contact you list as the registrant will have certain rights as the Registered Name Holder of the domain name, under the ICANN Transfer Policy. Most domains will be deleted upon closure of your AWS account (for more information, see My AWS account is closed or permanently closed, and my domain is registered with Route 53), however if a domain remains in a closed account, the contact you listed as the registrant might have the ability to request a transfer of the domain name to an external registrar. Therefore, it is important that the registrant contact you list is either yourself or another person you trust to act responsibly.

The following changes to registrant contact information trigger the domain ownership change process:

For Person contact type

Changing the **First Name** or **Last Name** fields triggers an ownership change.

For non-Person contact types (Company, Association, Public Body)

Changing the **Organization** field triggers an ownership change.

Additional triggers (all contact types)

Depending on your domain's TLD requirements, other contact field changes might also trigger the ownership change process, including:

- Changes to the registrant email address
- Changes to other identifying information fields

The specific fields that trigger ownership changes vary by TLD and are determined by the domain registry's policies.

> **Warning**
>
> **Important:** When an ownership change is triggered, you may receive an authorization email. If you don't respond to this email within 3-15 days (depending on your TLD), we may cancel the change request.
> If the change triggers email verification and you don't confirm within 3-15 days (depending on your TLD), we may suspend your domain, which can interrupt your website and email services.
> Always make sure you can access the registrant contact email address before making any contact changes.

Note the following about changing the owner of a domain:

- For some TLDs, there's a fee to change the owner of a domain. To determine whether there's a fee for the TLD for your domain, see the "Change Ownership Price" column in Amazon Route 53 Pricing for Domain Registration.

  > **Note**
  >
  > You can't use AWS credits to pay the fee, if any, to change the owner of a domain.

- For some TLDs, when you change the owner of a domain, Route 53 sends an authorization email to the email address for the registrant contact. The registrant contact must follow the instructions in the email to authorize the change.
- For some TLDs, you need to fill out a Change of Domain Ownership Form and provide proof of identity so that an Amazon Route 53 support engineer can update the values for you. If the TLD

for your domain requires a Change of Domain Ownership form, the console displays a message that links to a form for opening a support case. For more information, see Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

TLDs that require special processing to change the owner

When you change the owner of a domain, the registries for some TLDs require special processing. If you're changing the owner for any of the following domains, perform the applicable procedure. If you're changing the owner for any other domain, you can change the owner yourself, either programmatically or using the Route 53 console. See Updating contact information for a domain.

The following TLDs require special processing to change the owner of the domain:

.be, .cl, .com.br, .es, .fi, .ru, .se, .sh

.be

You must get a transfer code from the registry for .be domains, and then open a case with AWS Support.

- To get the transfer code, see https://www.dnsbelgium.be/en/manage-your-domain-name/change-holder#transfer, and follow the prompts.
- To open a case, see Contacting AWS Support about domain registration issues.

.cl

You must complete and submit a form to AWS Support. See Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

.com.ar

You must complete and submit a form to AWS Support. See Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

.com.br

You must complete and submit a form to AWS Support. See Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

.es

You must complete and submit a form to AWS Support. See Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

.fi

Initiate the owner change on the Route 53 console. After you have initiated the change, you will receive a **Holder transfer key** from *fi-domain-tech@traficom.fi* email address. After you receive the key, open a support case with AWS Support, and share the key code with us. See Contacting AWS Support about domain registration issues.

.qa

You must complete and submit a form to AWS Support. See Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

.ru

You must complete and submit a form to AWS Support. See Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

.se

You must complete and submit a form to AWS Support. See Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

.sh

You must complete and submit a form to AWS Support. See Changing the owner of a domain when the registry requires a Change of Domain Ownership form.

Updating contact information for a domain

To update contact information for a domain, perform the following procedure.

> **Warning**
>
> **Important:** If your contact changes trigger the owner change process, you may receive an authorization email. If you don't respond to this email within 3-15 days (depending on your TLD), we may cancel the change request.
> If the change triggers email verification and you don't confirm within 3-15 days (depending on your TLD), we may suspend your domain, which can interrupt your website and email services.
> Ensure you can access the registrant contact email address before making any contact changes.

Before updating domain contacts

Before making any changes to your domain's registrant contact information, complete this checklist:

1. ☐ **Verify email access** - Confirm you can receive email at the current registrant email address
2. ☐ **Understand owner change triggers** - Review the section called "What triggers a domain ownership change?" to understand what changes might trigger the owner change process
3. ☐ **Plan for authorization** - If owner change is triggered, you have 3-15 days to respond to the authorization email or we may cancel the change request

To update contact information for a domain

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered domains**.
3. Choose the name of the domain that you want to update contact information for.
4. In the **Contact information** tab, choose **Edit**.
5. If you don't have access to the email address for the registrant contact, perform the following steps. If you have access to the email address for the registrant contact, skip to step 6.

    a. Change *only* the email address for the registrant contact. Don't change any other values for any of the contacts for the domain. If you also want to change other values, you change them later in the process.

    Choose **Save changes**.

    To verify the new email address, Route 53 sends a verification email to the new address (if required for the TLD). You must choose the link in the email to verify that the new email address is valid. If verification is required, and you don't verify the new email address, Route 53 suspends the domain as required by ICANN.

    If you need to resend the verification email, navigate to the **Registered domains** page, choose the radio button next to the domain name you updated, and choose the name of the domain that you're updating. On the **Verify your email to avoid domain suspension** alert, choose **Send email again**.

b. If you want to change other values for the registrant, admin, tech, or billing contacts for the domain, return to step 1 and repeat the procedure.

6. Update the applicable values. You can also choose **Copy registrant contact** to automatically fill in the same information you entered for the registrant contact. For more information, see Values that you specify when you register or transfer a domain.

   Depending on the TLD for your domain and the values that you're changing, the console might display the following message:

   "To change the registrant name or organization, open a case."

   If you see that message, skip the rest of this procedure and see Changing the owner of a domain when the registry requires a Change of Domain Ownership form for more information.

7. Choose **Save**.

8. If you changed the domain owner, as described in What triggers a domain ownership change?, Route 53 sends email to the registrant contact for the domain. The email asks for authorization for the change of owner.

   If Route 53 doesn't receive authorization for the change within 3 to 15 days, depending on the top-level domain, Route 53 must cancel the request as required by ICANN.

   The email comes from one of the following email addresses.

TLDs	Email address that authorization email comes from
.fr	nic@nic.fr
.com.au .net.au	noreply@emailverification.info
All others	One of the following email addresses: • noreply@registrar.amazon • noreply@domainnameverification.net

TLDs	Email address that authorization email comes from
	• noreply@emailverification.info

9. If you encounter issues while updating contact information, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

For information about the API you can use to update the contact information, see UpdateDomainContact.

Changing the owner of a domain when the registry requires a Change of Domain Ownership form

If the registry for your domain requires you to complete a Change of Domain Ownership and submit the form to AWS Support, perform the following procedure. To determine whether you need to perform this procedure, see the following topics:

- To determine whether the value you're changing is considered a change of owner, see What triggers a domain ownership change?.
- To determine whether a Change of Domain Ownership form is required for your domain, see TLDs that require special processing to change the owner.

To change the owner of a domain when the registry requires a Change of Domain Ownership form

1. See the introduction to this topic to determine whether the registry for your domain requires special processing to change the owner of the domain. If so, and if a Change of Domain Ownership form is required, continue with this procedure.

   If no Change of Domain Ownership form is required, perform the procedure in the applicable topic instead.
2. Download the Change of Domain Ownership Form. The file is compressed into a .zip file.
3. Fill out the form.

4. For the registrant contact for the former owner of the domain *and* for the new owner, get a copy of a signed proof of identity (identity card, driver's license, passport, or other legal proof of identity).

   In addition, if a legal entity is listed as the registrant organization, gather the following information for the former owner of the domain *and* for the new owner:

   - Proof that the organization that the domain is registered to exists.
   - Proof that the representatives for the former owner and the new owner are authorized to act on the organization's behalf. This document must be a certified legal document that contains both the name of the organization and the names of the representatives as signing officers (for example, CEO, President, or Executive Director).

5. Scan the Change of Domain Ownership form and the required proof. Save the scanned documents in a common format, such as a .pdf file or a .png file.

6. Using the AWS account that the domain is currently registered to, sign in to the AWS Support Center.

   > **Important**
   >
   > You must sign in either by using the root account or by using a user that has been granted IAM permissions in one or more of the following ways:
   >
   > - The user is assigned the **AdministratorAccess** managed policy.
   > - The user is assigned the **AmazonRoute53DomainsFullAccess** managed policy.
   > - The user is assigned the **AmazonRoute53FullAccess** managed policy.
   >
   > If you don't sign in either by using the root account or by using a user that has the required permissions, Route 53 can't update the domain owner. This requirement prevents unauthorized users from changing the owner of a domain.

7. Specify the following values:

   Regarding

   Accept the default value of **Account and billing**.

Service

Accept the default value of **Domains**.

Category

Accept the default value of **Change of Ownership**.

Severity

Accept the default value of **General question**.

Choose **Next step: Additional information**

Subject

Specify Change the owner of a domain.

Description

Provide the following information:

- Domain that you want to change the owner for
- 12-digit account ID of the AWS account that the domain is registered to

Add attachment

Upload the documents that you scanned in step 5.

Contact method

Specify a contact method and enter the applicable values.

8. Choose **Submit**.

An AWS Support engineer reviews the information that you provided and updates the settings. The engineer will either contact you when the update is finished or contact you for more information.

Enabling or disabling privacy protection for contact information for a domain

When you register a domain with Amazon Route 53 or transfer a domain to Route 53, Route 53 enables privacy protection by default for all the contacts for the domain. This typically hides most of your contact information from WHOIS ("Who is") queries and reduces the amount of spam

that you receive. When you enable privacy protection, your contact information is replaced with contact information for the registrar or with the phrase "REDACTED FOR PRIVACY", or "On behalf of <domain name> owner."

If you choose to disable privacy protection, you must disable it for all contacts for a domain. If you do disable privacy protection, anyone can send a WHOIS query for the domain and, for most top-level domains (TLDs), might be able to get all the contact information that you provided when you registered or transferred the domain, including name, address, phone number, and email address. The WHOIS command is widely available; it's included in many operating systems, and it's also available as a web application on many websites.

If you are transferring a domain to another registrar, and privacy protection is enabled for the domain contacts, the email to verify the transfer will be delivered from identity-protect.org addresses for TLDs registered with Amazon Registrar. To determine who the registrar is for your TLD, see Finding your registrar.

The information that you can hide from WHOIS queries depends on two main factors:

The registry for the top level domain

Most TLD registries hide all contact information automatically, some allow you to choose to hide all contact information, some allow you to hide only some information, and some do not allow you to hide any information. **Some TLD registries intentionally suppress privacy protection to comply with local regulations or registry policies.**

When privacy protection on a domain is enabled, your contact information is replaced either with contact information for the privacy protection service, or with the phrase "REDACTED FOR PRIVACY." The privacy protection service applies spam prevention features (address rotation and SPF/DKIM/spam analysis) and will, in most cases, automatically forward emails passing these filters. However, it is not advisable to send critical emails to privacy protected email addresses because the spam mechanism might prevent them from being forwarded.

Additionally, the choice of which privacy protection mechanism is used for a domain is not configurable and is auto-selected by the system. The contact detail for our privacy protection service can't be manually updated.

> **Note**
>
> To enable or disable privacy protection for some domains, you must open a support case and request privacy protection. For more information, see the applicable section in Domains that you can register with Amazon Route 53:
>
> - .co.uk (United Kingdom)
> - .me.uk (United Kingdom)
> - .org.uk (United Kingdom)
> - .link

The registrar

When you register a domain with Route 53 or transfer a domain to Route 53, the registrar for the domain is either Amazon Registrar or our registrar associate, Gandi. Amazon Registrar and Gandi hide different information by default:

- **Amazon Registrar** – By default, all of your contact information is hidden. However, regulations for the TLD registry take precedence.
- **Gandi** – By default, all of your contact information is hidden except organization name, if any. However, regulations for the TLD registry take precedence.

  For geographic TLDs that don't allow privacy protection, your personal information will be marked as "redacted" on the Whois Directory Search page on the Gandi website. However, your personal information might be available at the domain registry or on third-party WHOIS websites.

To find out what information is hidden for the TLD for your domain, see Domains that you can register with Amazon Route 53.

When you want to enable or disable privacy protection for a domain that you registered using Route 53, perform the following procedure.

> **Important**
>
> **Privacy protection availability:** Privacy protection is not available for all TLDs. Common TLDs that don't support privacy protection include .de, .us, .com.au, .es, and several others.

If you don't see a "Privacy protection" section when editing your domain's contact information, privacy protection is not supported for your domain's TLD.
For a complete list of TLDs that don't support privacy protection, see TLDs that don't support privacy protection. For detailed information about privacy protection support for any TLD, see Domains that you can register with Amazon Route 53.

To enable or disable privacy protection for contact information for a domain

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered domains**.
3. Choose the name of the domain that you want to enable or disable privacy protection for.
4. In the **Contact information** section, choose **Edit**.
5. In the **Privacy protection** section, choose whether to hide contact information. You must specify the same privacy setting for all four contacts: admin, registrant, technical, and billing.

   Note

   If the **Privacy protection** section is not displayed, privacy protection is not supported for your domain's TLD. Your contact information will be publicly visible in WHOIS queries for domains that don't support privacy protection.

6. Choose **Save changes**.
7. If you encounter issues while enabling or disabling privacy protection, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

TLDs that don't support privacy protection

The following TLDs do not support privacy protection. If your domain uses one of these TLDs, your contact information will be publicly visible in WHOIS queries:

Generic TLDs

.audio, .ceo, .diet, .flowers, .guitars, .hosting, .juegos, .property, .sexy

Geographic TLDs

.cl (Chile), .com.ar (Argentina), .com.au (Australia), .com.br (Brazil), .com.sg (Singapore), .de (Germany), .es (Spain), .in (India), .net.au (Australia), .qa (Qatar), .ru (Russia), .sg (Singapore), .us (United States), .vg (British Virgin Islands)

> **Note**
>
> This list includes the most common TLDs that don't support privacy protection. For complete and up-to-date information about privacy protection support for a specific TLD, see the individual TLD page in Domains that you can register with Amazon Route 53.

Troubleshooting privacy protection issues

Privacy protection option is not available in the console

This indicates that your domain's TLD does not support privacy protection. Your contact information will be publicly visible in WHOIS queries. This is normal behavior for certain TLDs due to registry policies or local regulations.

To verify whether your TLD supports privacy protection, check the individual TLD page in Domains that you can register with Amazon Route 53 or see TLDs that don't support privacy protection.

Contact information is still visible after enabling privacy protection

Some registries maintain their own WHOIS databases and might continue to show contact information even when privacy protection is enabled with Route 53. This is controlled by the TLD registry, not by Route 53.

Additionally, some TLD registries intentionally maintain limited privacy protection or redaction services for their domains.

Enabling or disabling automatic renewal for a domain

When you want to change whether Amazon Route 53 automatically renews registration for a domain shortly before the expiration date, or you want to see the current setting for automatic renewal, perform the following procedure.

Note that you can't use AWS credits to pay the fee for renewing registration for a domain.

> **Note**
>
> Make sure you turn off automatic renewal if you intend to cancel your AWS account. Otherwise, you will continue to receive renewal notices from AWS. Your domain will not, however, be renewed, unless you re-activate your account.

To enable or disable automatic renewal for a domain

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered domains**.
3. Choose the name of the domain that you want to update.
4. On the **Details** section, in the **Actions** dropdown, choose **Turn on auto-renew**

   In the **Turn on auto-renew for <domain name>?** agree to pay the yearly rate, and choose **Turn on**.

   > **Note**
   >
   > The price listed is for the current registration period, and can change. For more information, see Amazon Route 53 Pricing for Domain Registration. The pricing list doesn't include registry premium prices, which vary by domain name.

5. To turn off auto-renew, select **Turn off auto-renew** in the **Actions** dropdown.
6. If you encounter issues while enabling or disabling automatic renewal, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

Locking a domain to prevent unauthorized transfer to another registrar

The domain registries for all generic TLDs and many geographic TLDs support locking a domain to prevent someone from transferring the domain to another registrar without your permission. To determine whether the registry for your domain supports locking the domain, see Domains that you can register with Amazon Route 53. If locking is supported and you want to lock your domain,

perform the following procedure. You can also use the procedure to disable the lock if you want to transfer a domain to another registrar.

To lock a domain to prevent unauthorized transfer to another registrar

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered Domains**.
3. Choose the name of the domain that you want to update.
4. On the **Details** section, in the **Actions** dropdown, choose **Turn on transfer lock** or **Turn off transfer lock**, depending on whether you want to turn the transfer lock on or off.

   You can navigate to the **Requests** page to see the progress of your request.
5. If you encounter issues while locking a domain, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

In WHOIS search, this status shows up as: `clientTransferProhibited`. Some TLDs might have these statuses in addition:

- `clientUpdateProhibited`
- `clientDeleteProhibited`

Extending the registration period for a domain

When you register a domain with Amazon Route 53 or you transfer domain registration to Route 53, Route 53 configures the domain to renew automatically. The automatic renewal period is typically one year, although the registries for some top-level domains (TLDs) have longer renewal periods.

Note the following:

Maximum renewal period

All generic TLDs and many country-code TLDs let you extend domain registration for longer periods, typically up to ten years in one-year increments. To determine whether you can extend the registration period for your domain, see Domains that you can register with Amazon Route 53. If longer registration periods are allowed, perform the following procedure.

Restrictions on when you can renew or extend a domain registration

Some TLD registries have restrictions on when you can renew or extend a domain registration, for example, the last two months before the domain expires. Even if the registry allows extending the registration period for a domain, they might not allow it at the current number of days before the domain expires.

> **Note**
>
> For example, if the maximum allowable renewal period is 5 years for the TLDs that you have your domain with, you can add yearly renewals at any time until you reach the 5-year limit. More specifically, if you have a domain that currently has 2.5 years of validity, you can only renew it for up to 2 more years.

AWS credits

You can't use AWS credits to pay the fee for extending the registration period for a domain.

To extend the registration period for your domain

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered Domains**.
3. Choose the name of the domain for which you want to extend the registration period.
4. In the **Details** section, in the **Actions** dropdown choose **Renew domain registration**.
5. In the **Renew domain registrations** dialog box, in the **Renewal period** dropdown, choose the number of years that you want to extend the registration for.

   The list shows all the current options based on the current expiration date and the maximum registration period allowed by the registry for this domain. The expiration date with that number of years applied is listed under the duration.
6. Choose **Renew domain registration**.

   When Route 53 receives confirmation from the registry that they've updated your expiration date, Route 53 sends you an email to confirm that the expiration date has changed.

7. If you encounter issues while extending the registration period for a domain, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

Updating name servers to use another registrar

If you want to move DNS management to another registrar, you need to update the name servers to point to

To update the name servers for your domain when you want to use another DNS service

1. Use the process that is provided by your DNS service to get the name servers for the domain.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Registered domains**.
4. Choose the name of the domain that you want to configure to use another DNS service.
5. In the **Details** section, under the **Actions** drop-down, choose **Edit name servers**.
6. Delete the existing name servers, and then add the names of the name servers to the name servers that you got from your DNS service in step 1.
7. Choose **Save changes**.
8. (Optional) Delete the hosted zone that Route 53 created automatically when you registered your domain. This prevents you from being charged for a hosted zone that you aren't using.

    a. In the navigation pane, choose **Hosted Zones**.

    b. Select the radio button for the hosted zone that has the same name as your domain.

    c. Choose **Delete Hosted Zone**.

    d. Choose **Confirm** to confirm that you want to delete the hosted zone.

Adding or changing name servers and glue records for a domain

When you register a domain with Route 53, Route 53 automatically creates a hosted zone for the domain, assigns four name servers to the hosted zone, and then updates the domain registration to use those name servers. You typically don't need to change those settings unless you want to use another DNS service or you want to use white-label name servers.

The maximum number of name servers per domain in Route 53 is 6.

> **Warning**
>
> If you change name servers to the wrong values, specify the wrong IP addresses in glue records, or delete one or more name servers without specifying new ones, your website or application might become unavailable on the internet for up to two days.

Topics

- Considerations for changing name servers and glue records
- Adding or changing name servers or glue records

Considerations for changing name servers and glue records

Consider the following issues before you change your configuration.

Topics

- You want to make Route 53 the DNS service for your domain
- You want to use another DNS service
- You want to use white-label name servers
- You're changing name servers for a .it domain

You want to make Route 53 the DNS service for your domain

If you're currently using another DNS service and you want to make Route 53 the DNS service for your domain, see Making Amazon Route 53 the DNS service for an existing domain for detailed instructions on how to migrate DNS service to Route 53.

> **Important**
>
> If you don't rigorously follow the migration process, your domain can become unavailable on the internet for up to two days.

You want to use another DNS service

If you want to use a DNS service other than Route 53 for your domain, use the following procedure to change the name servers for the domain registration to the name servers that are provided by the other DNS service.

> **Note**
>
> If you change name servers and Route 53 returns the following error message, the registry for the TLD doesn't recognize the name servers that you specified as valid name servers:
> `"We're sorry to report that the operation failed after we forwarded your request to our registrar associate. This is because: One or more of the specified name servers are not known to the domain registry."`
> TLD registries commonly support name servers provided by public DNS services but don't support private DNS servers, such as DNS servers that you configured on Amazon EC2 instances, unless the registry has IP addresses for those name servers. Route 53 doesn't support using name servers that aren't recognized by the TLD registry. If you encounter this error, you must change to name servers for Route 53 or another public DNS service.

You want to use white-label name servers

If you want the names of your name servers to be subdomains of your domain name, you can create white-label name servers. (White-label name servers are also known as vanity name servers or private name servers.) For example, you might create name servers ns1.example.com through ns4.example.com for the domain example.com. To use white-label name servers, use the following procedure to specify the IP addresses of your name servers instead of the names. These IP addresses are known as glue records.

For more information about configuring white-label name servers, see Configuring white-label name servers.

You're changing name servers for a .it domain

IName servers for your IT domain must pass a DNS check. We suggest that you check the name servers at https://dns-check.nic.it/ before you submit the change request. The registry

continues to respond to DNS queries using the name servers from before you made the change. If the previous name servers are no longer available, your domain becomes unavailable on the internet.

> **Important**
>
> Whenever you change name servers for a domain, confirm that DNS is responding to queries with the new name servers before you cancel the old DNS service or you delete the Route 53 hosted zone that used the old name servers.

For information about getting help from AWS to correct the names of your name servers with the registry for .it domains, see Contacting AWS Support about domain registration issues.

Adding or changing name servers or glue records

To add or change name servers or glue records, perform the following procedure.

> **Note**
>
> By default, DNS resolvers typically cache the names of name servers for two days. As a result, your changes can take two days to take effect. For more information, see How Amazon Route 53 routes traffic for your domain.

To add or change name servers or glue records for a domain

1. Review Considerations for changing name servers and glue records and address the applicable issues, if any.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Registered domains**.
4. Choose the name of the domain for which you want to edit settings.
5. In the **Details** section, in the **Actions** dropdown choose **Edit name servers**.
6. In the **Edit name servers** dialog box, you can do the following:

   - Change the DNS service for the domain by doing one of the following:

- Replace the name servers for another DNS service with the name servers for a Route 53 hosted zone
- Replace the name servers for a Route 53 hosted zone with the name servers for another DNS service
- Replace the name servers for one Route 53 hosted zone with the name servers for a different Route 53 hosted zone

For information about changing the DNS service for a domain, see Making Amazon Route 53 the DNS service for an existing domain. For information about getting the name servers for the Route 53 hosted zone that you want to use for DNS service for the domain, see Getting the name servers for a public hosted zone.

- Add one or more name servers.
- Replace the name of an existing name server.
- If you specify white-label name servers, add or change the IP addresses in glue records. You can enter addresses in IPv4 or IPv6 format. If a name server has multiple IP addresses, enter each address on a separate line.

  A white-label name server includes your domain name, such as example.com, in the name of the name server, such as ns1.example.com. When you specify a white-label name server, Route 53 prompts you to specify one or more IP addresses for the name server. The IP address is known as a glue record. For more information, see Configuring white-label name servers.
- Delete a name server. Choose the x icon on the right side of the field for that name server.

> **Warning**
>
> If you change name servers to the wrong values, specify the wrong IP addresses in glue records, or delete one or more name servers without specifying new ones, your website or application might become unavailable on the internet for up to two days.

7. Choose **Update**.
8. If you encounter issues while adding or changing name servers or glue records, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

Renewing registration for a domain

When you register a domain with Amazon Route 53 or you transfer domain registration to Route 53, Route 53 configures the domain to renew automatically. The automatic renewal period is typically one year, although the registries for some top-level domains (TLDs) have longer renewal periods. For the registration and renewal period for your TLD, see Domains that you can register with Amazon Route 53.

Note

You can't use AWS credits to pay the fee for renewing registration for a domain.

For most top-level domains (TLDs), you can change the expiration date for a domain. For more information, see Extending the registration period for a domain.

Important

If you turn off automatic renewal, be aware of the following effects on your domain:

- Some TLD registries delete domains even before the expiration date if you don't renew early enough. We strongly recommend that you leave automatic renewal enabled if you want to keep a domain name.
- We also strongly recommend that you don't plan to re-register a domain after it has expired. Some registrars allow others to register domains immediately after the domains expire, so you might not be able to re-register before the domain is taken by someone else.
- Some registries charge a large premium to restore expired domains.
- On or near the expiration date, the domain becomes unavailable on the internet.

To determine whether automatic renewal is enabled for your domain, see Enabling or disabling automatic renewal for a domain.

If automatic renewal is enabled, here's what happens:

45 days before expiration

Route 53 sends an email to the registrant contact that tells you that automatic renewal is currently enabled and gives instructions about how to disable it. Keep your registrant contact email address current so you don't miss this email.

35 or 30 days before expiration

For all domains except .com.ar and .com.br domains, Route 53 renews domain registration 35 days before the expiration date to allow time to resolve any issues with your renewal before the domain name expires.

The registries for .com.ar and .com.br domains require that Route 53 renew the domains no more than 30 days before expiration. You'll get a renewal email from Gandi, our registrar associate, 30 days before expiration, which is the same day that Route 53 renews your domain if you have automatic renewal enabled.

> **Note**
>
> When Route 53 renews your domain, Route 53 sends you an email to let you know that it renewed it. If the renewal failed, Route 53 sends you an email to explain why it failed.

If automatic renewal is disabled, here's what happens as the expiration date for a domain name approaches:

45 days before expiration

Route 53 sends an email to the registrant contact for the domain that tells you that automatic renewal is currently disabled and gives instructions about how to enable it. Keep your registrant contact email address current so you don't miss this email.

30 days and 7 days before expiration

If automatic renewal is disabled for the domain, ICANN, the governing body for domain registration, requires the registrar to send you an email. The email comes from one of the following email addresses:

- **noreply@registrar.amazon** – For domains for which the registrar is Amazon Registrar.
- **noreply@domainnameverification.net** or **noreply@emailverification.info** – For domains for which the registrar is our registrar associate, Gandi.

To determine who the registrar is for your TLD, see Domains that you can register with Amazon Route 53.

If you enable automatic renewal less than 30 days before expiration, and the renewal period has not passed, Route 53 renews the domain within 24 hours.

> **Important**
>
> Some TLD registries have restrictions on when you can renew a domain. For details specific to your domain, see Domains that you can register with Amazon Route 53. In addition, processing a renewal can take up to a day. If you delay too long before enabling automatic renewal, the domain might expire before renewal can be processed, and you might lose the domain. If the expiration date is approaching, we recommend that you manually extend the expiration date for the domain. For more information, see Extending the registration period for a domain.

For more information about renewal periods, see the Deadlines for renewing and restoring domains section for your TLD in Domains that you can register with Amazon Route 53.

After the expiration date

The registrar holds most domains for a brief time after expiration, so you might be able to renew an expired domain after the expiration date, but if you want to keep your domain, we strongly recommend that you keep automatic renewal enabled.

If your domain was already deleted or it's past the end of its late-renewal period, you might need to restore it instead of renewing it. For more information, see Restoring an expired or deleted domain.

If your domain expires but late renewal is allowed for the domain, you can renew the domain for its renewal price. To determine whether a domain is still within the late-renewal period, perform the procedure in the Extending the registration period for a domain section. If the domain is still listed, it's within the late-renewal period.

For more information about renewal periods, see the Deadlines for renewing and restoring domains section for your TLD in Domains that you can register with Amazon Route 53.

Restoring an expired or deleted domain

If you don't renew a domain before the end of the late-renewal period or if you accidentally delete the domain, for some registries for top-level domains (TLDs), you can restore the domain before it becomes available for others to register.

When a domain is deleted or it passes the end of the late-renewal period, it no longer appears in the Amazon Route 53 console.

Note

You can't use AWS credits to pay the fee for restoring a domain.

Important

Restoring a domain typically costs more than registering or renewing one. For the current price, see the Restoration Price column in Amazon Route 53 Pricing for Domain Registration. If the domain is a premium domain, its restoration price isn't listed here and varies by domain name; search for the domain in the Route 53 console or use the `GetDomainPrices` API operation to see its price.

Important

For generic TLD domains deleted within the first 5 days after registration, the standard 30-day redemption period doesn't apply. Anyone can register these domains immediately. To check the redemption period for a TLD, see Domains that you can register with Amazon Route 53.

To restore domain registration when a domain is deleted or the late-renewal period has expired

1. Determine whether the TLD registry for the domain supports restoring domains and, if so, the period during which restoration is allowed.

   a. See Domains that you can register with Amazon Route 53.

b. Find the TLD for your domain, and review the values in the Deadlines for renewing and restoring domains section.

2. Using the AWS account that the domain was registered to, call the RestoreDomain API action for the domain that you want to restore.
3. If the restoration succeeds, the domain reappears in the console and in the response to GetDomainDetail. The new expiration date might be different from the domain's previous expiration date. If the domain can no longer be restored, the request fails with no charge.

Replacing the hosted zone for a domain that is registered with Route 53

If you delete the hosted zone for a domain, you need to create another hosted zone when you're ready to make the domain available on the internet. Perform the following procedure.

To replace the hosted zone for a domain

1. Create a public hosted zone. For more information, see Creating a public hosted zone.
2. Create records in the hosted zone. Records define how you want to route traffic for the domain (example.com) and subdomains (acme.example.com, zenith.example.com). For more information, see Working with records.
3. Update the domain configuration to use the name servers for the new hosted zone. For more information, see Adding or changing name servers and glue records for a domain.

> **Important**
>
> When you create a hosted zone, Route 53 assigns a set of four name servers to the hosted zone. If you delete a hosted zone and then create one, Route 53 assigns another set of four name servers. Typically, none of the name servers for the new hosted zone match any of the name servers for the previous hosted zone. If you don't update the domain configuration to use the name servers for the new hosted zone, the domain will remain unavailable on the internet.

4. If you encounter issues while replacing the hosted zone for a domain, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

Transferring domains

You can transfer domain registration from another registrar to Amazon Route 53, from one AWS account to another, or from Route 53 to another registrar. There is no cost for transferring domains from one AWS account to another.

The topics in this section cover the following topics related to transferring domains:

1. Choose your transfer type
   - Understand the difference between transferring domain registration to Route 53 versus using Route 53 for DNS hosting only.
   - Learn which option is right for your needs based on your goals for domain management and DNS hosting.
2. Pre-transfer checklist
   - Complete essential preparation steps before transferring your domain to Route 53 to avoid common transfer failures.
   - Verify domain eligibility, obtain authorization codes, and prepare your DNS settings for a smooth transfer process.
3. Transferring registration for a domain to Amazon Route 53
   - Learn the step-by-step procedure for transferring a domain from another registrar to Route 53, including prerequisites, authorization codes, and updating DNS settings.
   - Understand how transferring a domain affects the expiration date and the considerations for different top-level domains (TLDs).
4. Common transfer issues
   - Prevent common transfer issues by understanding domain requirements, authorization processes, and timing considerations.
   - Learn how to resolve transfer problems and what to do if your transfer is delayed or rejected.
5. Transferring a domain from Amazon Route 53 to another registrar
   - Understand the process of transferring a domain from Route 53 to another registrar, including obtaining the authorization code, updating DNS settings, and responding to confirmation emails.
   - Be aware of the considerations when transferring DNS service to another provider and the potential impact on Route 53-specific features like alias records and routing policies.
6. Transferring a domain to a different AWS account

- Find out how to transfer a domain from one AWS account to another, including the roles and permissions required for initiating and accepting the transfer.
- Learn about the optional step of migrating the hosted zone to the new account after the domain transfer.

7. Transfer status
   - Discover how to view the status of a domain transfer request and the meaning of different status codes during the transfer process.
8. How transferring a domain to Amazon Route 53 affects the expiration date for your domain registration
   - Find out how transferring a domain to Route 53 might affect the expiration date for the domain.

By following the information provided in the topics listed above, you can effectively transfer domains to and from Route 53, manage the transfer process, and ensure a smooth transition while maintaining proper DNS configuration and routing.

Choose your Route 53 transfer type

When you want to use Amazon Route 53 with your domain, you have two options. Understanding the difference will help you choose the right approach for your needs:

Transfer domain registration to Route 53

With this option, Route 53 becomes your domain registrar. This means you'll manage your domain renewals, contact information, and DNS settings all through Route 53.

Choose this option if you want to:

- Consolidate domain management and DNS hosting in one place
- Take advantage of Route 53 domain management features
- Simplify billing by having domain renewals on your AWS bill

What to expect:

- **Time to complete:** 5-7 days
- **Prerequisites:** Authorization code from current registrar, unlock domain, verify registrant email

- **Who you pay for renewals:** Route 53

Next step: Complete the Pre-transfer checklist for domain transfers to prepare for the transfer.

Use Route 53 for DNS hosting only

With this option, you keep your current domain registrar but use Route 53 as your DNS service.

Choose this option if you want to:

- Keep your existing registrar relationship
- Use Route 53 advanced DNS features without changing registrars
- Get started quickly with minimal prerequisites

What to expect:

- **Time to complete:** Up to 2 days (depends on DNS propagation)
- **Prerequisites:** None
- **Who you pay for renewals:** Your current registrar

Next step: Go to Configuring Amazon Route 53 as your DNS service to set up DNS hosting.

> **Important**
>
> You don't have to transfer your domain registration to use Route 53 as your DNS service. You can keep your existing registrar and use Route 53 only for DNS hosting.

Pre-transfer checklist for domain transfers

Before you begin transferring your domain registration to Amazon Route 53, complete the following checks to avoid common transfer failures and ensure a smooth process.

Decide what you want to transfer

Choose the option that matches your needs:

Transfer domain registration to Route 53

Your registrar becomes Amazon Route 53. You pay Route 53 for domain renewals and manage domain settings through the Route 53 console.

Next step: Continue with this checklist, then follow Transferring registration for a domain to Amazon Route 53.

Use Route 53 for DNS hosting only

Keep your current registrar but use Route 53 name servers for DNS resolution. You continue paying your current registrar for renewals.

Next step: Skip this checklist and follow Configuring Amazon Route 53 as your DNS service.

Prerequisites checklist

Complete all of the following before starting your domain transfer.

Pre-transfer checklist (Complete all items)

1. ☐ **Verify registrant email access**

   Confirm that you can receive email at the registrant email address, as it will receive the domain transfer authorization email.

   > **Important**
   >
   > If you cannot access the current registrant email address, update the email address for the registrant contact with your current registrar and wait 60 days before transferring.

2. ☐ **Unlock domain**

   Access your current registrar domain settings and disable the "Domain Lock" or "Transfer Lock" protection, if enabled.

   > **Note**
   >
   > After disabling the lock, allow a few minutes for the change to take effect before starting your transfer.

3. ☐ **Obtain authorization code**

   Request an authorization code (auth code) from your current registrar and maintain its confidentiality. You'll enter this code in the Route 53 console later in the transfer process.

4. ☐ **Verify domain age and eligibility**

   Confirm your domain meets ICANN transfer requirements:

   - The domain must be at least 60 days old since initial registration.
   - If the domain registration expired and was restored, it must have been restored at least 60 days ago.

   > **Tip**
   >
   > ICANN rules might temporarily block transfers if registrant contact details were updated recently (for example, changing your name or email). Some registrars let you opt out of this hold — check with your current registrar if you're unsure.

5. ☐ **Prepare payment method**

   Ensure your AWS account has a valid payment method. When you transfer a domain to Route 53, the transfer fee that we apply to your AWS account varies by top-level domain.

   For current pricing, see Route 53 Pricing.

   > **Note**
   >
   > You cannot use AWS credits to pay for domain transfers. Route 53 charges the transfer fee before starting the process and provides an immediate refund if the transfer fails.

6. ☐ **Disable DNSSEC (if enabled)**

   If DNSSEC is currently enabled with your current registrar, temporarily disable it before starting the transfer. This ensures the transfer can complete without DNS validation issues.

   Why this matters:

When you transfer a domain, the DNSSEC keys (KSK/ZSK) and DS records stored at your current registrar are not automatically transferred to Route 53. If you re-enable DNSSEC without reconfiguring these keys in Route 53, your domain might fail DNS validation.

Safe approach (recommended):

1. Disable DNSSEC at your current registrar before initiating the transfer.
2. Complete the domain transfer and create your Route 53 hosted zone (if not already created).
3. Verify that all DNS records are resolving correctly without DNSSEC enabled.
4. Configure DNSSEC in Route 53 using your hosted zone's new KSK/ZSK values.
5. Re-enable DNSSEC by publishing the new DS record with your TLD registrar (now Route 53).

> **Tip**
>
> If you are transferring your domain along with creating a new hosted zone in Route 53, disable DNSSEC before transferring, then wait until the hosted zone setup is verified and DNS resolution works correctly. After you confirm this, you can safely enable DNSSEC again from the Route 53 console. Always verify your DNS configuration in Route 53 before re-enabling DNSSEC. For more information about DNSSEC with Route 53, see Configuring DNSSEC for a domain.

Optional: Transfer DNS service first

Consider transferring your DNS service to Route 53 before transferring domain registration. This approach provides these benefits:

- Reduces the risk of website or email downtime during the transfer
- Allows you to test Route 53 DNS functionality before committing to the full transfer
- Provides a fallback option if the domain transfer encounters issues

To transfer DNS service first, see Configuring Amazon Route 53 as your DNS service.

Next steps

After completing this checklist:

1. Follow the step-by-step transfer process in Transferring registration for a domain to Amazon Route 53.
2. Monitor your email for authorization messages during the transfer process.

Transferring registration for a domain to Amazon Route 53

> **Important**
>
> During the transfer of any country code top-level domains (ccTLDs) to Route 53, except for .cc and .tv, updates to the owner contact are ignored and the owner contact data from the registry is used. You can update the owner contact after the transfer is complete. For more information, see Updating contact information and ownership for a domain.

To transfer the registration for a domain to Amazon Route 53, follow the procedures in this topic.

Transfer process overview

Domain transfer involves these key steps:

1. **Complete pre-transfer checklist**

   Verify prerequisites to avoid transfer failures. See Pre-transfer checklist for domain transfers.
2. **Request transfer in Route 53 console**

   Enter domain name and authorization code. See Step 5: Request the transfer.
3. **Authorize via email** (*Customer action required)*

   Click authorization link sent to registrant email within 5 days. See Step 6: Click the link in the confirmation and authorization emails.
4. **Wait for completion**

   Transfer completes automatically unless rejected by current registrar. Monitor status in console.

Most common failure points

The following are solutions for common issues:

- **Email not received**: Verify registrant email is accessible and check spam folder
- **Domain locked**: Unlock at current registrar before starting transfer
- **Invalid auth code**: Request new authorization code from current registrar
- **60-day rule violation**: 60-day ICANN rule (eligibility window). Transfers are blocked within 60 days of initial registration or a registrant contact change. Some registrars allow an opt-out before the change—ask your current registrar if applicable.

> **Important**
>
> If you skip a step, your domain might become unavailable on the internet.

Note the following:

Contacting AWS Support

If you encounter issues while transferring a domain, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

Expiration date

For information about how transferring your domain affects the current expiration date, see How transferring a domain to Amazon Route 53 affects the expiration date for your domain registration.

Transfer fee

When you transfer a domain to Route 53, the transfer fee that we apply to your AWS account depends on the top-level domain, such as .com or .org. For more information, see Route 53 Pricing.

You can't use AWS credits to pay the fee, if any, for transferring a domain to Route 53.

> **Note**
>
> Route 53 charges the fee for transferring your domain before we start the transfer process. If a transfer fails for some reason, we immediately credit your account for the cost of the transfer.

Domain quotas

The default maximum number of domains per AWS account is 20. You can request a higher quota. For more information, see Quotas on domains.

Name servers limit

The maximum number of name servers per domain in Route 53 is 6.

Topics

- Transfer requirements for top-level domains
- Step 1: Confirm that Amazon Route 53 supports the top-level domain
- Step 2 (Optional): Transfer your DNS service to Amazon Route 53 or another DNS service provider
- Step 3: Change settings with the current registrar
- Step 4: Get the names of your name servers
- Step 5: Request the transfer
- Step 6: Click the link in the confirmation and authorization emails
- Step 7: Update the domain configuration

Transfer requirements for top-level domains

Most domain registrars enforce requirements on transferring a domain to another registrar. The primary purpose of these requirements is to prevent the owners of fraudulent domains from repeatedly transferring the domains to different registrars. Requirements vary, but the following requirements are typical:

- You must have registered or transferred the domain to the current registrar at least 60 days ago.

- If the registration for a domain name expired and had to be restored, it must have been restored at least 60 days ago.
- The domain cannot have any of the following domain name status codes:
  - clientTransferProhibited
  - pendingDelete
  - pendingTransfer
  - redemptionPeriod
  - serverTransferProhibited
- The registries for some top-level domains don't allow transfers until changes are complete, such as changes to the domain owner.

For a current list of domain name status codes and an explanation of what each code means, go to the website for ICANN, and search for EPP status codes. (Search on the ICANN website; web searches sometimes return an old version of the document.)

> **Note**
>
> ICANN is the organization that establishes policies governing registration and transfer of domain names.

You can also search for your domain name in website for Whois to see status codes and other information for your domain.

Step 1: Confirm that Amazon Route 53 supports the top-level domain

See Domains that you can register with Amazon Route 53. If the top-level domain for the domain that you want to transfer is on the list, you can transfer the domain to Amazon Route 53.

If a TLD is not on the list, you can't currently transfer the domain registration to Route 53. We occasionally add more TLDs to the list, so check back to see if we've added support for your domain.

Step 2 (Optional): Transfer your DNS service to Amazon Route 53 or another DNS service provider

Why transfer DNS first?

Some registrars provide free DNS service that might be disabled as soon as they receive a request from Route 53 to transfer the domain's registration. If you'd like Route 53 to provide DNS service for your domain, see Making Amazon Route 53 the DNS service for an existing domain.

Step 3: Change settings with the current registrar

Using the method provided by your current registrar, do each of the following for each domain that you want to transfer.

- Confirm that the email for the registrant contact for your domain is up to date
- Unlock the domain so it can be transferred
- Confirm that the domain status allows you to transfer the domain
- Disable DNSSEC for the domain
- Get an authorization code
- Renew your domain registration before you transfer the domain (selected geographic TLDs)

Confirm that the email for the registrant contact for your domain is up to date

We'll send email to that email address to request authorization for the transfer. You need to choose a link in the email to authorize the transfer. If you don't choose the link, we must cancel the transfer.

> **Important**
>
> The contact you list as the registrant will have certain rights as the Registered Name Holder of the domain name, under the ICANN Transfer Policy. Most domains will be deleted upon closure of your AWS account (for more information, see My AWS account is closed or permanently closed, and my domain is registered with Route 53), however if a domain remains in a closed account, the contact you listed as the registrant might have the ability to request a transfer of the domain name to an external registrar. Therefore, it is important that the registrant contact you list is either yourself or another person you trust to act responsibly.

Unlock the domain so it can be transferred

ICANN, the governing body for domain registrations requires that you unlock your domain before you transfer it.

Confirm that the domain status allows you to transfer the domain

For more information, see Transfer requirements for top-level domains.

Disable DNSSEC for the domain

If you use DNSSEC with a domain and you transfer the domain registration to Route 53, you must remove the DS records for your domain at your current registrar before transferring. After removing the DS records, wait at least 24 hours for the change to propagate before proceeding with the transfer. Then, after you transfer the domain registration, you can set up DNSSEC for the domain in Route 53. Route 53 supports DNSSEC for domain registration and for DNSSEC signing. For more information, see Configuring DNSSEC signing in Amazon Route 53.

> **Important**
>
> If you transfer a domain registration to Route 53 while DNSSEC is configured, the DNSSEC public keys are transferred, too. If you transfer DNS service to a provider that doesn't support DNSSEC, DNS resolution fails intermittently until you delete the DNSSEC keys from the domain. For more information, see Deleting public keys for a domain.

Get an authorization code

An authorization code from the current registrar authorizes us to request that registration for the domain be transferred to Route 53. You'll enter this code in the Route 53 console later in the process.

Some top-level domains have additional requirements:

.co.za domains

You don't need to get an authorization code to transfer a .co.za domain to Route 53.

.uk, .co.uk, .me.uk, and .org.uk domains

If you're transferring a .uk, .co.uk, .me.uk, or .org.uk domain to Route 53, you don't need to get an authorization code. Instead, use the method provided by your current domain registrar to update the value of the IPS tag for the domain to **GANDI**, all uppercase. (An IPS tag is required by Nominet, the registry for .uk domain names.) If your registrar doesn't provide a way to change the value of the IPS tag, contact Nominet.

Note the following about changing the IPS tag:

You must request the transfer within five days

If you don't request the transfer within five days after you change the IPS tag, the tag changes back to the previous value. You must change the value of the IPS tag again, or the transfer request will fail.

Viewing the IPS tag in WHOIS queries

The change to the IPS tag doesn't appear in WHOIS queries until after the transfer to Route 53 has completed.

Email from Gandi

You might receive an email from our registrar associate, Gandi, about the transfer process. If you receive an email from Gandi (transfer-auth@gandi.net) about transferring your domain, ignore the instructions in the email because they aren't relevant to a .uk, .co.uk, .me.uk, or .org.uk domains. Follow the instructions in this topic instead.

Renew your domain registration before you transfer the domain (selected geographic TLDs)

For most TLDs, when you transfer a domain, the registration is automatically extended by one year. However, for some geographic TLDs, registration is not extended when you transfer the domain. If you're transferring a domain to Route 53 that has one of these TLDs, we recommend that you renew the domain registration before you transfer the domain, especially if the expiration date is approaching.

> ⚠ **Important**
>
> If you don't renew the domain before you transfer it, the registration could expire before the transfer is complete. If this happens, the domain becomes unavailable on the internet, and the domain name could become available for others to purchase.

Registration is not automatically extended when you transfer the following domains to another registrar:

- .ch (Switzerland)
- .cl (Chile)
- .co.uk (United Kingdom)
- .co.za (South Africa)
- .com.au (Australia)

- .cz (Czech Republic)
- .es (Spain)
- .fi (Finland)
- .im (Isle of Man)
- .jp (Japan)
- .me.uk (United Kingdom)
- .net.au (Australia)
- .org.uk (United Kingdom)
- .se (Sweden)
- .uk (United Kingdom)

Step 4: Get the names of your name servers

If you're using Amazon Route 53 as your DNS service or you're continuing to use the existing DNS service, we'll get the names of the name servers for you automatically later in the process. Skip to Step 5: Request the transfer.

If you want to change the DNS service to a provider other than Route 53 at the same time that you're transferring the domain to Route 53, use the procedure provided by the DNS service provider to get the names of the name servers for each domain that you want to transfer.

> **Important**
>
> If the registrar for your domain is also the DNS service provider for the domain, transfer your DNS service to Route 53 or another DNS service provider *before* you continue with the process to transfer the domain registration.
>
> If you transfer DNS service at the same time that you transfer domain registration, your website, email, and the web applications associated with the domain might become unavailable. For more information, see Step 2 (Optional): Transfer your DNS service to Amazon Route 53 or another DNS service provider.

Step 5: Request the transfer

To transfer domain registration from the current registrar to Amazon Route 53, use the Route 53 console to request the transfer. Route 53 handles the communication with the current registrar for the domain.

You can use the console to transfer up to five domains.

The procedure that you use depends on whether you want to transfer a single domain or up to five domains:

- To transfer domain registration of a single domain to Route 53
- To transfer domain registration to Route 53 for up to five domains

Use the **Transfer domain to your account** process to transfer a single domain to your account.

To transfer domain registration of a single domain to Route 53

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered domains**.
3. On the **Registered Domains** page, choose **Single domain** from the **Transfer in** dropdown.
4. On the **Transfer domain to your account** page, in the **Check domain transferability** section, enter the name of the domain for which you want to transfer registration to Route 53, and choose **Check**.
5. If the domain registration is available for transfer, verify that you have completed the transfer requirements for top-level domains, and choose **Next**.

   If the domain registration is not available for transfer, the Route 53 console lists the reasons. Contact your registrar for information about how to resolve the issues that prevent you from transferring the registration.
6. On the **DNS service** page, review the information about name servers, and choose **Next**.
7. If prompted, enter the authorization code or IPS tag that you got from your current registrar in Step 3: Change settings with the current registrar.

> **Note**
>
> You don't need to enter an authorization code to transfer a .co.za, .uk, .co.uk, .me.uk, or .org.uk domain to Route 53.

Choose **Next**.

8. On the **Domain pricing options** page, choose the number of years that you want to register the domain you are transferring for and whether you want us to automatically renew your domain registration before the expiration date.

> **Note**
>
> Domain name registrations and renewals are not refundable. If you enable automatic domain renewal and you decide that you don't want the domain name after we renew the registration, you can't get a refund for the cost of the renewal.

Choose **Next**.

9. On the **Contact information** page, enter contact information for the domain registrant, admin, technical, and billing contacts. The values that you enter here are applied to all of the domains that you're registering. For more information, see Values that you specify when you register or transfer a domain.

Note the following considerations:

First Name and Last Name

For **First Name** and **Last Name**, we recommend that you specify the name on your official ID. For some changes to domain settings, some domain registries require that you provide proof of identity. The name on your ID must match the name of the registrant contact for the domain.

Different contacts

By default, we use the same information for all three contacts. If you want to enter different information for one or more contacts, change the value of **Same as registrant contact** toggle switch to off position.

> **Note**
>
> For .it domains, the registrant and admin contacts must be the same.

Additional required information

For some top-level domains (TLDs), we're required to collect additional information. For these TLDs, enter the applicable values after the **Postal/Zip Code** field.

Privacy protection

Choose whether you want to hide your contact information from WHOIS queries.

> **Note**
>
> You must specify the same privacy setting for the admin, registrant, and technical contacts.

For more information, see the following topics:

- Enabling or disabling privacy protection for contact information for a domain
- Domains that you can register with Amazon Route 53

> **Note**
>
> To enable privacy protection for .uk, .co.uk, .me.uk, and .org.uk domains, you must open a support case and request privacy protection.

Choose **Next**.

10. On the **Review** page, review the information that you entered, and optionally correct it. Read the terms of service, and select the check box to confirm that you've read the terms of service.

    Choose **Submit request**.

11. In the navigation pane, choose **Domains** and then **Requests**.

On this page, you can view the status of domain and also if you need to respond to registrant contact verification email. You can also choose to resend the verification email.

If you specified an email address for the registrant contact that has never been used to register a domain with Route 53, some TLD registries require you to verify that the address is valid.

We send a verification email from one of the following email addresses:

- **noreply@registrar.amazon** – for TLDs registered by Amazon Registrar.
- **noreply@domainnameverification.net** or **noreply@emailverification.info** – for TLDs registered by our registrar associate, Gandi. To determine who the registrar is for your TLD, see Finding your registrar.

> **Important**
>
> The registrant contact must follow the instructions in the email to verify that the email was received, or we must suspend the domain as required by ICANN. When a domain is suspended, it's not accessible on the internet.

a. When you receive the verification email, choose the link in the email that verifies that the email address is valid. If you don't receive the email immediately, check your junk email folder.

b. Return to the **Requests** page. If the status doesn't automatically update to say **email-address is verified**, choose **Refresh status**.

12. When domain transfer is complete, your next step depends on whether you want to use Route 53 or another DNS service as the DNS service for the domain:

- **Route 53** – In the hosted zone that Route 53 created when you registered the domain, create records to tell Route 53 how you want to route traffic for the domain and subdomains.

  For example, when someone enters your domain name in a browser and that query is forwarded to Route 53, do you want Route 53 to respond to the query with the IP address of a web server in your data center or with the name of an Elastic Load Balancing load balancer?

For more information, see Working with records.

> **Important**
>
> If you create records in a hosted zone other than the one that Route 53 creates automatically, you must update the name servers for the domain to use the name servers for the new hosted zone.

- **Another DNS service** – Configure your new domain to route DNS queries to the other DNS service. Perform the procedure Updating name servers to use another registrar.

Use the following procedure to transfer up to five domains to your account.

To transfer domain registration to Route 53 for up to five domains

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered domains**.
3. On the **Registered domains** page, choose **Multiple domains** from the **Transfer in** dropdown.
4. On the **Transfer multiple domains to your account** page, enter up to five domains you want to transfer, and their authorization code, if required, per line, and choose **Check**.
5. If the domain registration is available for transfer, it is listed in the **Domain availability** list as available. Select the check-box next to each domain for which you want to transfer the registration, verify that you have completed the transfer requirements for top-level domains, and choose **Next**.

   If the domain registration is not available for transfer, the Route 53 console lists the reasons. Contact your registrar for information about how to resolve the issues that prevent you from transferring the registration.
6. On the **DNS service** page, review the information about name servers, and choose **Next**.

> **Note**
>
> Domain name registrations and renewals are nonrefundable. If you enable automatic domain renewal and you decide that you don't want the domain name after we renew the registration, you can't get a refund for the cost of the renewal.

7. On the **Contact information** page, enter contact information for the domain registrant, admin, and tech contacts. The values that you enter here are applied to all the domains that you're transferring.

   > **Important**
   >
   > We recommend that you specify the following values for the registrant contact (the domain owner):
   >
   > - **First and last name**: We recommend that you specify the name that appears on your official ID. For some changes to domain settings, some domain registries require that you provide proof of identity. The name on your ID must match the name of the registrant contact for the domain.
   > - **Contact details**: During the domain transfer, we recommend that you specify the same values as are specified with the current registrar. When you change contact details for the registrant contact, you change the domain owner, and some TLD registries don't allow you to change the domain owner during a domain transfer. If you change contact details for the registrant contact, the transfer might fail. You can change contact details for the registrant contact after you transfer the domain.

   By default, we use the same information for all three contacts. If you want to enter different information for one or more contacts, set the value of **Same as the registrant contact** to off position.

   > **Note**
   >
   > For .it domains, the registrant and admin contacts must be the same.

   For more information, see Values that you specify when you register or transfer a domain.
8. For some TLDs, we're required to collect additional information. For these TLDs, enter the applicable values after the **Postal/Zip Code** field.
9. If the value of **Contact Type** is **Person**, choose whether you want to hide your contact information from WHOIS queries. For more information, see Enabling or disabling privacy protection for contact information for a domain.
10. Choose **Submit**.

11. Review the information you entered, read the terms of service, and select the check box to confirm that you've read the terms of service.
12. Choose **Submit request**.

    We confirm that the domains are eligible for transfer, and we send an email to the registrant contacts for the domain to request authorization to transfer the domain.
13. In the navigation pane, choose **Domains** and then **Requests**.

    On this page you can view the status of domain and also if you need to respond to registrant contact verification email. You can also choose to resend the verification email.

    If you specified an email address for the registrant contact that has never been used to register a domain with Route 53, some TLD registries require you to verify that the address is valid.

    We send a verification email from one of the following email addresses:

    - **noreply@registrar.amazon** – for TLDs registered by Amazon Registrar.
    - **noreply@domainnameverification.net** or **noreply@emailverification.info** – for TLDs registered by our registrar associate, Gandi. To determine who the registrar is for your TLD, see Finding your registrar.

    > **Important**
    >
    > The registrant contact must follow the instructions in the email to verify that the email was received, or we must suspend the domain as required by ICANN. When a domain is suspended, it's not accessible on the internet.

    a. When you receive the verification email, choose the link in the email that verifies that the email address is valid. If you don't receive the email immediately, check your junk email folder.

    b. Return to the **Requests** page. If the status doesn't automatically update to say **email-address is verified**, choose **Refresh status**.
14. When domain transfer is complete, your next step depends on whether you want to use Route 53 or another DNS service as the DNS service for the domain:

- **Route 53** – In the hosted zone that Route 53 created when you registered the domain, create records to tell Route 53 how you want to route traffic for the domain and subdomains.

  For example, when someone enters your domain name in a browser and that query is forwarded to Route 53, do you want Route 53 to respond to the query with the IP address of a web server in your data center or with the name of an ELB load balancer?

  For more information, see Working with records.

  > **Important**
  >
  > If you create records in a hosted zone other than the one that Route 53 creates automatically, you must update the name servers for the domain to use the name servers for the new hosted zone.

- **Another DNS service** – Configure your new domain to route DNS queries to the other DNS service. Perform the procedure Updating name servers to use another registrar.

Step 6: Click the link in the confirmation and authorization emails

Soon after you request the transfer, we might send one or more emails to the registrant contact for the domain:

Email to confirm that the registrant contact is reachable

If you've never registered a domain with Route 53 or transferred a domain to Route 53, we send you an email that asks you to confirm that the email address is valid. We retain this information so we don't have to send this confirmation email again.

Email to get authorization to transfer the domain

For some TLDs, you need to respond to an email to authorize transfer of the domain.

Generic TLDs such as .com, .net, and .org

Authorization isn't required for domains that have a generic TLD, such as .com, .net, or .org.

Geographic TLDs such as .co.uk and .jp

For domains that have a geographic TLD, we're required to get your authorization to transfer the domain. If you transfer 10 domains, we have to send you 10 emails, and you have to choose the authorization link in each one.

The emails all go to the registrant contact for the domain:

- If you're the registrant contact for the domain, follow the instructions in the email to authorize the transfer.
- If someone else is the registrant contact, ask that person to follow the instructions in the email to authorize the transfer.

> **Important**
>
> If you're transferring a domain that has a geographic TLD, we wait up to five days for the registrant contact to authorize the transfer. If the registrant contact doesn't respond within five days, we cancel the transfer operation and send an email to the registrant contact about the cancellation.

Topics

- Authorization email for a new owner or email address
- Email addresses that authorization emails come from
- Approval from the current registrar
- What happens next

Authorization email for a new owner or email address

If you changed the following values, we send you a separate email that asks for your authorization:

Domain owner

If you change the owner of the domain, as described in What triggers a domain ownership change?, we send email to the registrant contact for the domain.

Email address for the registrant contact (only for some TLDs)

For some TLDs, if you change the email address for the registrant contact, we send an email to the old and the new email address for the registrant contact. Someone at both email addresses must follow the instructions in the email to authorize the change.

For changes to the domain owner or the email address for the registrant contact, if we don't receive authorization for the change within 3-15 days, depending on the top-level domain, we may cancel the change request.

Email addresses that authorization emails come from

All email comes from one of the following email addresses.

TLDs	Email address that authorization email comes from
.com.au and .net.au	One of the following email addresses: • no-reply@ispapi.net (contains a link to http://transfers.ispapi.net) • icann-transfers@key-systems.net
.fr	nic@nic.fr, if you're changing the registrant contact for a .fr domain name at the same time that you're transferring the domain. (The email is sent both to the current registrant contact and the new registrant contact.)
All others	One of the following email addresses: • noreply@registrar.amazon • noreply@domainnameverification.net • noreply@emailverification.info •

TLDs	Email address that authorization email comes from
	icann-transfers@key-systems.net

To determine who the registrar is for your TLD, see Domains that you can register with Amazon Route 53.

Approval from the current registrar

If the registrant contact authorizes the transfer, we start to work with your current registrar to transfer your domain. This step might take up to ten days, depending on the TLD for your domain:

- Generic top-level domains – take up to seven days
- Geographic top-level domains (also known as country code top-level domains) – take up to ten days

If your current registrar doesn't reply to our transfer request, which is common among registrars, the transfer happens automatically. If your current registrar rejects the transfer request, we send an email notification to the current registrant contact. The registrant needs to contact the current registrar and resolve the issues with the transfer.

What happens next

When your domain transfer has been approved, we send another email to the registrant contact. For more information about the process, see Viewing the status of a domain transfer.

We charge your AWS account for the domain transfer as soon as the transfer is complete. For a list of charges by TLD, see Amazon Route 53 Pricing for Domain Registration.

> **Note**
>
> This is a one-time charge, so the charge doesn't appear in your CloudWatch billing metrics. For more information about CloudWatch metrics, see Using Amazon CloudWatch metrics in the *Amazon CloudWatch User Guide*.

Step 7: Update the domain configuration

After the transfer is complete, you can optionally change the following settings:

Transfer lock

To transfer the domain to Route 53, you had to disable the transfer lock. If you want to re-enable the lock to prevent unauthorized transfers, see Locking a domain to prevent unauthorized transfer to another registrar.

Automatic renewal

We configure the transferred domain to automatically renew as the expiration date approaches. For information about how to change this setting, see Enabling or disabling automatic renewal for a domain.

Extended registration period

By default, Route 53 renews the domain annually. If you want to register the domain for a longer period, see Extending the registration period for a domain.

DNSSEC

For information about configuring DNSSEC for the domain, see Configuring DNSSEC for a domain.

Preventing common Route 53 transfer issues

If you encounter issues while your domain transfer is in progress, here are the most common problems and how to resolve them:

> **Note**
>
> If your transfer has already failed, see Transferring my domain to Amazon Route 53 failed for post-failure troubleshooting.

You didn't receive the authorization email

If you don't receive the authorization email from Route 53, here are the most common causes and solutions:

- **Check your spam folder** - Authorization emails are sometimes filtered by email providers.
- **Verify your email address** - Confirm that the registrant email address in your domain's contact information is current and accessible.
- **Update your contact information** - If the email address is outdated, update it with your current registrar before starting the transfer.
- **Resend the authorization email** - In the Route 53 console, click "Resend authorization email" for the domain in transfer status.

You're getting an "invalid authorization code" error

If you receive an "invalid authorization code" error, try these solutions:

- **Double-check the code** - Authorization codes are case-sensitive. Copy and paste the code directly from your registrar's interface to avoid typing errors.
- **Check if the code has expired** - Authorization codes can expire between 5-30 days depending on your current registrar. Request a new code if yours has expired.
- **Remove extra spaces** - Make sure there are no leading or trailing spaces when you enter the code.
- **Get a new authorization code** - If the code continues to fail, request a new authorization code from your current registrar.

Your domain is locked at the current registrar

Domain transfers can fail when your current registrar has a transfer lock enabled (also known as "Domain Lock" or "Client Transfer Prohibited"). This is a common security setting that prevents unauthorized transfers.

To resolve this issue:

- **Unlock your domain** - Log in to your current registrar's control panel and disable the domain transfer lock.
- **Verify the lock is removed** - You can use any public WHOIS lookup tool to confirm that the domain status shows "OK" instead of "clientTransferProhibited."
- **Get help from your registrar** - If you don't see an option to unlock your domain, contact your current registrar's support team. They can help you remove the lock.

Your domain isn't eligible for transfer

If your domain transfer is rejected due to eligibility issues, check the following:

- **Verify domain age** - Your domain must be at least 60 days old since registration or last transfer.
- **Check TLD support** - Confirm that Route 53 supports your domain's top-level domain (TLD). See Domains that you can register with Amazon Route 53 for supported TLDs.
- **Review domain status** - Domains with certain status codes (such as "pendingTransfer" or "redemptionPeriod") cannot be transferred.
- **Check expiration date** - Domains that are expired or within 15 days of expiration might not be transferable.
- **Resolve billing issues** - Make sure all outstanding fees with your current registrar are paid.

Your transfer is taking longer than expected

If your transfer seems to be delayed, here's what you should know:

- **Normal timeframe** - Most transfers complete within 5-7 days, but can take up to 10 days.
- **Monitor transfer status** - Check your transfer progress in the Route 53 console. For more information, see Viewing the status of a domain transfer.
- **Respond to emails quickly** - Both registrars might send confirmation emails that require your response to proceed.
- **Contact your current registrar** - If the transfer is delayed, contact your current registrar to make sure they're not blocking the transfer.

When to contact AWS Support

Contact AWS Support if you've tried the solutions above and still have issues, or if you encounter:

- Transfer failures after multiple attempts with correct information
- Billing or payment issues during the transfer process
- Technical errors in the Route 53 console that prevent you from starting a transfer
- Transfers that remain in "pending" status for more than 10 days

When you contact support, include your domain name, transfer request ID (if available), and details about the specific error or issue you're experiencing.

Transferring a domain from Amazon Route 53 to another registrar

When you transfer a domain from Amazon Route 53 to another registrar, you get some information from Route 53 and provide it to the new registrar. The new registrar will do the rest.

Important

If you're currently using Route 53 as your DNS service provider and you also want to transfer DNS service to another provider, be aware that the following Route 53 features don't have direct parallels with features provided by other DNS service providers. You'll need to work with the new DNS service provider to determine how to achieve comparable functionality:

- Alias records. For more information, see Choosing between alias and non-alias records.
- Routing policies other than the simple routing policy. For more information, see Choosing a routing policy.
- Health checks that are associated with records. For more information, see Configuring DNS failover.

Most domain registrars enforce requirements on transferring a domain to another registrar. The primary purpose of these requirements is to prevent the owners of fraudulent domains from repeatedly transferring the domains to different registrars. Requirements vary, but the following requirements are typical:

- You must have registered the domain with the current registrar or transferred registration for the domain to the current registrar at least 14 days ago.
- The domain cannot have any of the following domain name status codes:
  - pendingDelete
  - pendingTransfer
  - redemptionPeriod
  - clientTransferProhibited
  - serverTransferProhibited

For a current list of domain name status codes and an explanation of what each code means, go to the ICANN website and search for **epp status codes**. (Search on the ICANN website; web searches sometimes return an old version of the document.)

> **Note**
>
> If you want to transfer your domain to another domain registrar but the AWS account that the domain is registered with is closed, suspended, or terminated, you can contact AWS Support for help. Domains cannot be transferred within the first 14 days of registration. For more information, see Contacting AWS Support about domain registration issues.

> **Note**
>
> If the new registrar requires a REG-ID code, you can contact AWS Support for help. For more information, see Contacting AWS Support about domain registration issues.

To transfer a domain from Route 53 to another registrar

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered domains**.
3. Choose the name of the domain that you want to transfer to another registrar.
4. On the *domain name* page, check the value of **Domain name status code**. If it is one of the following values, you can't currently transfer the domain:

   - pendingDelete
   - pendingTransfer
   - redemptionPeriod
   - clientTransferProhibited
   - serverTransferProhibited

   For a current list of domain name status codes and an explanation of what each code means, go to the ICANN website and search for **epp status codes**. (Search on the ICANN website; web searches sometimes return an old version of the document.)

   If the value of **Domain name status code** is **serverTransferProhibited**, you can contact AWS Support for free to learn what you must do so you can transfer the domain. For more information, see Contacting AWS Support about domain registration issues.

5. If the value of **Transfer lock** is **On**, choose **Turn off transfer lock** from the **Actions** dropdown.
6. *All domains except .be, .co.za, .ru, .uk, .co.uk, .me.uk, and .org.uk domains* – On the *domain name* page, choose **Transfer to another registrar** from the **Transfer out** dropdown.

   In the **Transfer to another registrar** dialog box, choose **Copy** to copy the authorization code for the domain transfer. You'll provide this value to your registrar later in this procedure.

   > **Note**
   >
   > For .eu domains, you can also generate the auth code by using the "My.eu" panel at the registry: https://my.eurid.eu/.

   .be, .co.za, .es, .ru, .uk, .co.uk, .me.uk, and .org.uk domains – Do the following:

   .be domains

   Get the authorization code from the registry for .be domains at DNS Belgium website.

   .co.za domains

   You don't need to get an authorization code to transfer a .co.za domain to another registrar.

   .ru domains

   Get the authorization code from the registry for .ru domains at https://www.nic.ru/en/auth/recovery/:

   a. Choose the option to recover credentials by domain name.
   b. Enter your domain name, and choose **Continue**.
   c. Follow the on-screen prompts to get access to the RU-CENTER admin page.
   d. In the **Manage your account** section, choose **Domain transfer**.
   e. Confirm the transfer with REGRU-RU.

   .uk, .co.uk, .me.uk, and .org.uk domains

   Change the IPS tag to the value for the new registrar:

   a. Go to the Find a Registrar page on the Nominet website, and find the IPS tag for the new registrar. (Nominet is the registry for .uk, .co.uk, .me.uk, and .org.uk domains.)

b. On the **Registered Domains >** *domain name* page, select the **Transfer out**, drop-down, then **Update IPS Tag**, and specify the value that you got in step 6a.

c. Choose **Update**.

> **Note**
>
> You can also update the IPS tag on the Nominet console. For instructions, see Switch the registrar.

7. If you're not currently using Route 53 as the DNS service provider for your domain, skip to step 10.

   If you are currently using Route 53 as the DNS service provider for the domain, perform the following steps:

   a. Choose **Hosted Zones.**

   b. Choose the name of the hosted zone for your domain. The domain and the hosted zone have the same name.

   c. *If you want to continue using Route 53 as the DNS service provider for the domain:* Get the names of the four name servers that Route 53 assigned to your hosted zone. For more information, see Getting the name servers for a public hosted zone.

      If you do not want to continue using Route 53 as the DNS service provider for the domain: Make note of the settings for all of your records except the NS and SOA records. For Route 53–specific features such as alias records, you'll need to work with your new DNS service provider to determine how to achieve comparable functionality.

8. If you're transferring DNS service to another provider, use the methods that are provided by the new DNS service to perform the following tasks:

   - Create a hosted zone
   - Create records that reproduce the functionality of your Route 53 records
   - Get the name servers that the new DNS service assigned to your hosted zone

9. Use the process that is provided by the new registrar to request a transfer of the domain.

   All domains except .co.za, .uk, .co.uk, .me.uk, and .org.uk domains – You'll be prompted to enter the authorization code that you got from the Route 53 console in step 6 of this procedure.

10. If you still want to use Route 53 as your DNS service provider, use the process that is provided by the new registrar to specify the names of the Route 53 name servers that you got in step 7. If you want to use another DNS service provider, specify the names of the name servers that the new provider gave you when you created a new hosted zone in step 8.
11. Act on the confirmation email:

    All domains except .jp domains

    Route 53 sends a confirmation email to the registrant contact for the domain:

    - Depending on the TLD, the confirmation email includes a link from one of the following domains. Use this link to approve or reject the transfer.
      - registrar.amazon
      - domainnameverification.net
      - emailverification.info
      - key-systems.net
    - If you don't take action, the transfer will proceed automatically on the specified date.

    .jp domains

    Route 53 sends a confirmation email to the registrant contact for the domain from address *noreply@domainnameverification.net* or *noreply@emailverification.info* with a link to confirm the transfer:

    - Use this link to approve or reject the transfer. You will need to provide the domain authorization code that you obtained in step 7.
    - If you don't take action, the transfer will be canceled on the specified date.

    In addition you might receive an email from WIXI.jp. You can ignore this email.
12. If the registrar that you're transferring the domain to reports that the transfer failed, contact that registrar for more information. When you transfer a domain to another registrar, all status updates go to the new registrar, so Route 53 has no information about why a transfer failed.

    If the new registrar reports that the transfer failed because the authorization code that you got from Route 53 isn't valid, open a case with AWS Support. (You don't need a support contract, and there's no fee.) For more information, see Contacting AWS Support about domain registration issues.

> **Note**
>
> Authorization codes generated by Gandi are valid for about 5 days. If your transfer attempt occurs after this period, it might fail due to an expired code.

13. If you transferred DNS service to another DNS service provider, you can delete the records in the hosted zone and delete the hosted zone after DNS resolvers stop responding to DNS queries with the names of Route 53 name servers. This typically takes two days, the amount of time that DNS resolvers commonly cache the names of name servers for a domain.

    > **Important**
    >
    > If you delete the hosted zone while DNS resolvers are still responding to DNS queries with the names of Route 53 name servers, your domain will become unavailable on the internet.

    After you delete the hosted zone, Route 53 will stop billing you the monthly charge for a hosted zone. For more information, see the following documentation:

    - Deleting records
    - Deleting a public hosted zone
    - Route 53 Pricing

Transferring a domain to a different AWS account

If you registered a domain using one AWS account and you want to transfer the domain to another AWS account, you can easily transfer it by using the new console, or by using the AWS CLI or other programmatic methods.

Topics

- Step 1: Transfer a domain to a different AWS account
- Step 2 (Optional): Migrate a hosted zone to a different AWS account

Step 1: Transfer a domain to a different AWS account

Domains cannot be transferred within the first 14 days of registration.

When you initiate the domain transfer, you must sign in either by using the root account or by using a user that has been granted IAM permissions in one or more of the following ways:

- The user is assigned the **AdministratorAccess** managed policy.
- The user is assigned the **AmazonRoute53DomainsFullAccess** managed policy.
- The user is assigned the **AmazonRoute53FullAccess** managed policy.
- The user is assigned the **PowerUserAccess** managed policy.
- The user has permission to perform all the following actions: `TransferDomains`, `DisableDomainTransferLock`, and `RetrieveDomainAuthCode`.

If you don't sign in either by using the root account or by using a user that has the required permissions, Route 53 can't perform the transfer. This requirement prevents unauthorized users from transferring domains to other AWS accounts.

The transfer process has two steps. First the originating account owner starts the transfer: in the initiate a transfer to another AWS account procedure, and then the destination account owner accepts the transfer in the accept a transfer from another AWS account procedure.

To transfer a domain to a different AWS account

1. Sign in to AWS by using the AWS account that the domain is currently registered to.
2. Open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Registered domains**.
4. Choose the name of the domain that you want to transfer to another AWS account.
5. Above the **Details** section, in the **Transfer out** dropdown, choose **Transfer to another AWS account**.
6. On the **Transfer to another AWS account** dialog, enter the destination account ID. You can get this ID from the destination AWS account owner.
7. Choose **Confirm**.
8. On the **Generate password** dialog, copy the password, and forward it to the receiving AWS account owner.

On the **Requests** page, the **Status** for the domain displays **In progress**, and the **Type** displays **Internal transfer out**.

To accept a domain transfer from a different AWS account

1. Sign into AWS by using the AWS account that is receiving the domain.
2. Open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Requests**.
4. On the **Requests** page, select the radio button next to the domain name you are transferring from another AWS account. If the domain is ready to be transferred the **Status** is **Action required** and the **Type** is **Internal transfer of domain in**.

   You have three days to accept the request. If the transfer isn't accepted in three days, the transfer request is cancelled.
5. In the **Action** dropdown, choose **Accept**.

   You can also choose **Reject** to cancel the transfer process.
6. If you accepted, on the **Transfer domain to your account** page, in the **Password** section, enter the password you received from the originating account owner.

   Accept the terms and conditions, and choose **Next**.
7. Navigate to the **Requests** page to monitor the transfer status and other steps to complete.
8. After the transfer completes, you can update the contact information. For more information, see Updating contact information and ownership for a domain.

Transfer the domain programmatically

You can also transfer the domain programmatically by using the AWS CLI, one of the AWS SDKs, or the Route 53 API. For more information, see the following documentation:

- For an overview of the transfer process and documentation about the API actions that you use to transfer a domain using the Route 53 domain registration API, see TransferDomainToAnotherAwsAccount in the *Amazon Route 53 API Reference*.
- For documentation about other options for transferring domains programmatically, see "SDKs & Toolkits" in the Guides and API References section of the "AWS documentation" page.

- The receiving account has three days to accept the transfer from the originating account, by using the transfer-domain-to-another-aws-account API. If the transfer isn't accepted in three days, the transfer request is cancelled.

> **Important**
>
> When you transfer a domain to a different AWS account, the hosted zone for the domain isn't transferred. If you also want to transfer the hosted zone, wait until the domain has been transferred, and then see Step 2 (Optional): Migrate a hosted zone to a different AWS account.

Step 2 (Optional): Migrate a hosted zone to a different AWS account

If you're using Route 53 as the DNS service for the domain, Route 53 doesn't transfer the hosted zone when you transfer a domain to a different AWS account. If domain registration is associated with one account and the corresponding hosted zone is associated with another account, neither domain registration nor DNS functionality is affected. The only effect is that you'll need to sign into the Route 53 console using one account to see the domain, and sign in using the other account to see the hosted zone.

If you own the account that you're transferring the domain from and the account that you're transferring the domain to, you can optionally migrate the hosted zone for the domain to a different account, but it's not required. Route 53 continues to use the records in the existing hosted zone to route traffic for the domain.

> **Important**
>
> If you don't own both the account that you're transferring the domain from and the account that you're transferring the domain to, you must either migrate the existing hosted zone to the AWS account that you're transferring the domain to, or create a new hosted zone in an AWS account that you own. If you don't own the account that created the hosted zone that routes traffic for the domain, you can't control how traffic is routed.

To migrate the existing hosted zone to the new account, see Migrating a hosted zone to a different AWS account.

To create a new hosted zone, see Making Amazon Route 53 the DNS service for an existing domain. This topic is typically used when you're transferring domains from another registrar to Route 53, but the process is the same when you're transferring domains from one AWS account to another.

Important

White-label name servers: If your domain uses white-label name servers (custom name server hostnames like ns1.example.com), additional steps are required:

- **When transferring domain registration:** Update glue records with your new registrar after transfer completion
- **When migrating hosted zones:** Update glue records with your domain registrar after zone migration, as IP addresses change even though hostnames remain the same

For detailed instructions, see the section called "Step 7: Create glue records and change the registrar's name servers".

Viewing the status of a domain transfer

After you initiate the transfer of a domain from another domain registrar to Amazon Route 53, you can track the status on the **Requests** page (new console) or the **Pending requests** (old console) page of the Route 53 console. The **Status** column includes a brief description of the current step. The following list includes the text in the console and a more detailed description of each step.

Note

When you submit a transfer request, the initial status is **Domain transfer request submitted**, which indicates that we've received your request.

Determining whether the domain meets transfer requirements (step 1 of 14)

We're confirming that your domain's status is eligible for transfer. You must unlock your domain, and the domain can't have any of the following status codes when you submit the transfer request:

- clientTransferProhibited

- pendingDelete
- pendingTransfer
- redemptionPeriod

Geographic TLDs only – verifying WHOIS information (step 2 of 14)

If you're transferring a domain that has a geographic TLD, we sent a WHOIS query for your domain to determine whether you've disabled the privacy protection for the domain. If privacy protection is still enabled with your current registrar, we won't be able to access the information we need to transfer the domain.

> **Note**
>
> Authorization isn't required for domains that have a generic TLD, such as .com, .net, or .org.

Geographic TLDs only – Sent email to registrant contact to get transfer authorization (step 3 of 14)

If you're transferring a domain that has a geographic TLD, we've sent an email to the registrant contact for the domain. The purpose of the email is confirm that the transfer was requested by an authorized contact of the domain.

> **Note**
>
> Authorization isn't required for domains that have a generic TLD, such as .com, .net, or .org.

Verifying transfer with current registrar (step 4 of 14)

We've sent a request to the current registrar for the domain to initiate the transfer.

Geographic TLDs only – Awaiting authorization from registrant contact (step 5 of 14)

We sent email to the registrant contact for the domain (see step 3 of 14), and we're waiting for the registrant contact to choose a link in the email to authorize the transfer. If you're transferring a domain that has a geographic TLD and you didn't receive the email for some reason, see Resending authorization and confirmation emails.

Contacted current registrar to request transfer (step 6 of 14)

We're working with the current registrar for the domain to finalize the transfer.

Waiting for the current registrar to complete the transfer (step 7 of 14)

Your current registrar is confirming that your domain meets the requirements for being transferred. This step might take up to ten days, depending on the TLD for your domain:

- Generic top-level domains – take up to seven days
- Geographic top-level domains (also known as country code top-level domains) – take up to ten days

> **Note**
>
> If you have approved the confirmation email sent from Route 53 when transferring a .JP domain, but it has stopped for several days in STEP 7, contact AWS Support Center for assistance.

For most registrars, the process is entirely automated and can't be accelerated. Some registrars send you an email that asks you to approve the transfer; if your registrar sends this confirmation email, the transfer process might be much faster than seven to ten days.

For information about reasons that a registrar might reject the transfer, see Transfer requirements for top-level domains.

Confirming with the registrant contact that the contact initiated the transfer (step 8 of 14)

Some TLD registries send the registrant contact another email to confirm that the domain transfer was requested by an authorized user.

Synchronizing name servers with the registry (step 9 of 14)

This step occurs only if the name servers that you provided as part of the transfer request are different from the name servers that are listed with the current registrar. We'll try to update your name servers to the new name servers that you provided.

Synchronizing settings with the registry (step 10 of 14)

We're verifying that the transfer has completed successfully, and we're synchronizing your domain-related data with our registrar associate.

Sending updated contact information to the registry (step 11 of 14)

If you changed the ownership of the domain when you requested the transfer, we're trying to make this change. However, most registries don't allow a transfer of ownership as part of the domain transfer process.

Finalizing the transfer to Route 53 (step 12 of 14)

We're confirming that the transfer process was successful.

Finalizing transfer (step 13 of 14)

We're setting up your domain in Route 53.

Transfer Complete (step 14 of 14)

Your transfer has been successfully completed.

How transferring a domain to Amazon Route 53 affects the expiration date for your domain registration

When you transfer a domain between registrars, some TLD registries let you keep the same expiration date for your domain, some registries add a year to the expiration date, and some registries change the expiration date to one year after the transfer date.

> **Note**
>
> For most TLDs, you can extend the registration period for a domain by up to ten years after you transfer it to Amazon Route 53. For more information, see Extending the registration period for a domain.

Generic TLDs

When you transfer a domain that has a generic TLD (for example, .com) to Route 53, the new expiration date for the domain is the expiration date with your previous registrar plus one year.

Geographic TLDs

When you transfer a domain that has a geographic TLD (for example, .co.uk) to Route 53, the new expiration date for the domain depends on the TLD. Find your TLD in the following table to determine how transferring your domain affects the expiration date.

Continent	Geographic TLDs and the effect of transferring a domain on the expiration date
Africa	**.co.za** – The expiration date remains the same.
Americas	**.cl, .com.ar, .com.br** – The expiration date remains the same. **.ca, .co, .mx, .us** – One year is added to the old expiration date.
Asia/Oceania	**.com.au, .com.sg, .jp, .net.au, .sg** – The expiration date remains the same. **.co.nz, .in, .net.nz, .org.nz** – One year is added to the old expiration date.
Europe	**.ch, .co.uk, .es, .fi, .me.uk, .nl, .org.uk, .se** – The expiration date remains the same. **.berlin, .eu, .io, .me, .ruhr, .wien** – One year is added to the old expiration date. **.be, .de, .fr, .it** – The new expiration date is one year after the date of transfer.

Resending authorization and confirmation emails

For several operations related to domain registration, ICANN requires that we get authorization from the registrant contact for the domain or confirmation that the email address for the registrant contact is valid. To get authorization or confirmation, we send an email that contains a link. You have between 3 and 15 days to choose the link, depending on the operation and the top-level domain. After that time, the link stops working.

If you don't choose the link in the email in the allotted amount of time, ICANN generally requires that we suspend the domain or cancel the operation, depending on what you were trying to do:

Register a domain

We suspend the domain, so that it's not accessible on the internet. To resend the confirmation email, see To resend the confirmation email for a domain registration.

Geographic TLDs only – Transfer a domain to Amazon Route 53

If you're transferring a domain that has a geographic TLD, we cancel the transfer. To resend the authorization email, see To resend the authorization email for a domain transfer.

> **Note**
>
> Authorization isn't required for domains that have a generic TLD, such as .com, .net, or .org.

Change the name or email address of the registrant contact for the domain (the owner)

We cancel the change. To resend the authorization email, see To resend the authorization email to update the registrant contact or delete a domain.

Delete a domain

We cancel the deletion request. To resend the authorization email, see To resend the authorization email to update the registrant contact or delete a domain.

Geographic TLDs only – Transfer a domain from Route 53 to another registrar

If you're transferring a domain that has a geographic TLD, the new registrar cancels the transfer.

> **Note**
>
> Authorization isn't required for domains that have a generic TLD, such as .com, .net, or .org.

Topics

- Updating your email address
- Resending emails

Updating your email address

We always send confirmation and authorization emails to the email address for the registrant contact for a domain. For some TLDs, we're required to send email to the old and new email addresses for the registrant contact in the following cases:

- You're changing the email address for a domain that is already registered with Amazon Route 53
- You're changing the email address for a domain that you're transferring to Route 53

Resending emails

Use the applicable procedure to resend confirmation or authorization emails.

- To resend the confirmation email for a domain registration
- To resend the authorization email for a domain transfer
- To resend the authorization email to update the registrant contact or delete a domain

To resend the confirmation email for a domain registration

1. Check the email address for the registrant contact and, if necessary, update it. For more information, see Updating contact information and ownership for a domain.
2. Check the spam folder in your email application for an email from one of the following email addresses.

   If too much time has passed, the link won't work any longer, but you'll know where to look for the confirmation email when we send you another one.

TLDs	Email address that the approval or confirmation email comes fro m
.fr	nic@nic.fr
All others	One of the following email addresses: • noreply@registrar.amazon

TLDs	Email address that the approval or confirmation email comes fro m
	• noreply@domainnameverification.net • noreply@emailverification.info

> **Note**
>
> The emails might contain links to www.registrar.amazon. These links are safe to use. For some TLDs, the verification email might not include the domain name in the subject or body of the email.

3. Use the Amazon Route 53 console to resend the confirmation email:

   a. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   b. In the navigation pane, choose **Registered domains**.

   c. Choose the name of the domain that you want to resend the email for.

   d. In the warning box with the heading "Your domain might be suspended," choose **Send email again**.

   > **Note**
   >
   > If there's no warning box, you already confirmed that the email address for the registrant contact is valid.

4. If you encounter issues while resending the confirmation email, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

To resend the authorization email for a domain transfer

This method doesn't work for .jp domain transfer out requests.

1. Use the method provided by the current domain registrar to confirm that privacy protection for the domain is disabled. If not, disable it.

   We send the authorization email to the email address that the current registrar saved in the WHOIS database. When privacy protection is enabled, that email address typically is obfuscated. The current registrar might not forward to your actual email address the email that Amazon Route 53 sends to the email address in the WHOIS database.

   > **Note**
   >
   > If the current registrar for the domain won't let you turn off privacy protection, we can still transfer the domain if you specified a valid authorization code in Step 5: Request the transfer.

2. Check the email address for the registrant contact and, if necessary, update it. Use the method provided by the current registrar for the domain.
3. Check the spam folder in your email application for an email from one of the following email addresses.

   If too much time has passed, the link won't work any longer, but you'll know where to look for the authorization email when we send you another one.

TLDs	Email address that the approval or confirmation email comes from
.com.au and .net.au	no-reply@ispapi.net The email contains a link to https://approve.domainadmin.com.
.fr	nic@nic.fr
All others	One of the following email addresses: • noreply@registrar.amazon • noreply@domainnameverification.net

TLDs	Email address that the approval or confirmation email comes fro m
	• noreply@emailverification.info

> **Note**
>
> The emails might contain links to www.registrar.amazon. These links are safe to use. For some TLDs, the verification email might not include the domain name in the subject or body of the email.

4. If the transfer is no longer in process (if we already canceled it because too much time has passed), request the transfer again, and we'll send you another authorization email.

> **Note**
>
> For the first 15 days after you request a transfer, you can determine the status of the transfer by checking the **Notifications** table on the **Dashboard** page in the Route 53 console. After 15 days, use the AWS CLI to get the status. For more information, see route53domains in the *AWS CLI Command Reference*.

If the transfer is still in progress, perform the following steps to resend the authorization email.

a. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

b. In the **Notifications** table, find the domain that you want to transfer.

c. In the **Status** column for that domain, choose **Resend email**.

5. If you encounter issues while resending the authorization email for a domain transfer, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

To resend the authorization email to update the registrant contact or delete a domain

1. Check the email address for the registrant contact and, if necessary, update it. For more information, see Updating contact information and ownership for a domain.
2. Check the spam folder in your email application for an email from one of the following email addresses.

   If too much time has passed, the link won't work any longer, but you'll know where to look for the authorization email when we send you another one.

TLDs	Email address that the authorization email comes from
.fr	nic@nic.fr
All others	One of the following email addresses: • noreply@registrar.amazon • noreply@domainnameverification.net • noreply@emailverification.info

> **Note**
>
> The emails might contain links to www.registrar.amazon. These links are safe to use. For some TLDs, the verification email might not include the domain name in the subject or body of the email.

3. Cancel the change or deletion. You have two options:

   - You can wait for the 3 to 15 day waiting period to pass, after which we automatically cancel the requested operation.
   - Alternatively, you can contact AWS Support and ask them to cancel the operation.

> **Important**
>
> **Transfer types and cancellation methods:** The cancellation method depends on the type of transfer:
>
> - **External registrar to Route 53:** Follow the procedures in this topic (wait for auto-cancellation or contact AWS Support)
> - **Between AWS accounts:** Use the `CancelDomainTransferToAnotherAwsAccount` API or contact AWS Support
>
> Don't contact your current external registrar to cancel a transfer to Route 53, as they can't cancel transfers that are already in progress with AWS.

4. After the change or deletion is canceled, you can change the contact information or delete the domain again, and we'll send you another authorization email.
5. If you encounter issues while resending the authorization email, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

Configuring DNSSEC for a domain

Attackers sometimes hijack traffic to internet endpoints such as web servers by intercepting DNS queries and returning their own IP addresses to DNS resolvers in place of the actual IP addresses for those endpoints. Users are then routed to the IP addresses provided by the attackers in the spoofed response, such as fake websites.

You can protect your domain from this type of attack, known as DNS spoofing or a man-in-the-middle attack, by configuring Domain Name System Security Extensions (DNSSEC), a protocol for securing DNS traffic.

> **Important**
>
> Amazon Route 53 supports DNSSEC signing and DNSSEC for domain registration. If you want to configure DNSSEC signing for a domain that's registered with Route 53, see Configuring DNSSEC signing in Amazon Route 53.

Topics

- Overview of how DNSSEC protects your domain
- Prerequisites and maximums for configuring DNSSEC for a domain
- Adding public keys for a domain
- Deleting public keys for a domain

Overview of how DNSSEC protects your domain

When you configure DNSSEC for your domain, a DNS resolver establishes a chain of trust for responses from intermediate resolvers. The chain of trust begins with the TLD registry for the domain (your domain's parent zone) and ends with the authoritative name servers at your DNS service provider. Not all DNS resolvers support DNSSEC. Only resolvers that support DNSSEC perform signature or authenticity checks.

Here's how you configure DNSSEC for domains registered with Amazon Route 53 to protect your internet hosts from DNS spoofing:

1. Use the method provided by your DNS service provider to *sign* the records in your hosted zone with the *private key* in an asymmetric key pair.

   > **Important**
   >
   > Route 53 supports DNSSEC signing and DNSSEC for domain registration. To learn more, see Configuring DNSSEC signing in Amazon Route 53.

2. Provide the *public key* from the key pair to your domain registrar, and specify the algorithm that was used to generate the key pair. The domain registrar forwards the public key and the algorithm to the top-level domain (TLD) registry.

   For information about how to perform this step for domains that you registered with Route 53, see Adding public keys for a domain.

After you configure DNSSEC, here's how it protects your domain from DNS spoofing:

1. Submit a DNS query, for example, by browsing to a website or by sending an email message.
2. The request is routed to a DNS resolver. Resolvers return the correct value to clients based on the request, such as the IP address for a web server or an email server.

3. If the IP address is cached on the DNS resolver, the resolver returns the IP address to the client. The client then uses the IP address to access the host.

   If the IP address isn't cached on the DNS resolver, the resolver sends a request to the parent zone for your domain, at the TLD registry. The TLD registry returns two values:

   - The Delegation Signer (DS) record, which is a public key that corresponds with the private key that was used to sign the record.
   - The IP addresses of the authoritative name servers for your domain.
4. The DNS resolver sends the request to another DNS resolver. If that resolver doesn't have the IP address, it repeats the process until a resolver sends the request to a name server at your DNS service provider. The name server returns two values:
   - The record for the domain, such as example.com. Typically this contains the IP address of a host.
   - The signature for the record, which you created when you configured DNSSEC.
5. The DNS resolver uses the public key that you provided to the domain registrar and the registrar forwarded to the TLD registry to do two things:
   - Establish a chain of trust.
   - Verify that the signed response from the DNS service provider is legitimate and hasn't been replaced with a bad response from an attacker.
6. If the response is authentic, the resolver returns the value to the client that submitted the request.

   If the response can't be verified, the resolver returns an error to the user.

   If the TLD registry for the domain doesn't have the public key for the domain, the resolver responds to the DNS query by using the response that it got from the DNS service provider.

Prerequisites and maximums for configuring DNSSEC for a domain

To configure DNSSEC for a domain, your domain and DNS service provider must meet the following prerequisites:

- The registry for the TLD must support DNSSEC. To determine whether the registry for your TLD supports DNSSEC, see Domains that you can register with Amazon Route 53.
- The DNS service provider for the domain must support DNSSEC.

> **Important**
>
> Route 53 supports DNSSEC signing and DNSSEC for domain registration. To learn more, see Configuring DNSSEC signing in Amazon Route 53.

- You must configure DNSSEC with the DNS service provider for your domain before you add public keys for the domain to Route 53.
- The number of public keys that you can add to a domain depends on the TLD for the domain:
  - **.com and .net domains** – up to thirteen keys
  - **All other domains** – up to four keys

Adding public keys for a domain

When you're rotating keys or you're enabling DNSSEC for a domain, perform the following procedure after you configure DNSSEC with the DNS service provider for the domain.

To add public keys for a domain

1. If you haven't already configured DNSSEC with your DNS service provider, use the method provided by your service provider to configure DNSSEC.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Registered domains**.
4. Choose the name of the domain that you want to add keys for.
5. In the **DNSSEC keys** tab, choose **Add key**.
6. Specify the following values:

   Key type

   Choose whether you want to upload a key-signing key (KSK) or a zone-signing key (ZSK).

   Algorithm

   Choose the algorithm that you used to sign the records for the hosted zone.

Public key

Specify the public key from the asymmetric key pair that you used to configure DNSSEC with your DNS service provider.

Note the following:

- Specify the public key, not the digest.
- You must specify the key in base64 format.

7. Choose **Add**.

> **Note**
>
> You can only add one public key at a time. If you need to add more keys, wait until you receive a confirmation email from Route 53.

8. When Route 53 receives a response from the registry, we send an email to the registrant contact for the domain. The email either confirms that the public key has been added to the domain at the registry or explains why the key couldn't be added.

Deleting public keys for a domain

When you're rotating keys or you're disabling DNSSEC for the domain, delete public keys using the following procedure before you disable DNSSEC with your DNS service provider. Note the following:

- If you're rotating public keys, we recommend that you wait for up to three days after you add the new public keys to delete the old public keys.
- If you're disabling DNSSEC, delete public keys for the domain first. We recommend that you wait for up to three days before you disable DNSSEC with the DNS service for the domain.

> **Important**
>
> If DNSSEC is enabled for the domain and you disable DNSSEC with the DNS service, DNS resolvers that support DNSSEC will return a `SERVFAIL` error to clients, and the clients won't be able to access the endpoints that are associated with the domain.

To delete public keys for a domain

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered domains**.
3. Choose the name of the domain that you want to delete keys from.
4. In the **DNSSEC keys** tab, choose the radio button next to the key you want to delete, and then choose **Delete key**.
5. In the **Delete DNSSEC key** dialog box, enter *delete* in the text box to confirm you want to delete the key, and then choose **Delete**.

   > **Note**
   >
   > You can only delete one public key at a time. If you need to delete more keys, wait until you receive a confirmation email from Amazon Route 53.

6. When Route 53 receives a response from the registry, we send an email to the registrant contact for the domain. The email either confirms that the public key has been deleted from the domain at the registry or explains why the key couldn't be deleted.

Finding your registrar and other information about your domain

To view domain information by using the GetDomainDetail API, you can use any of the SDKs or AWS CLI. For more information, see get-domain-detail.

To view information about domains with `get-domain-detail` CLI

- Use the following CLI:

```
aws route53domains get-domain-detail \
    --region us-east-1 \
    --domain-name example.com
```

> **Note**
>
> This command only runs in us-east-1 AWS Region.

All the information about your domain will be listed in the output, including the registrar, registration date, and privacy setting.

Viewing information about domains that are registered with Route 53

> **Note**
>
> Effective January 28, 2025, some domain registries might transition from WHOIS to the Registration Data Access Protocol (RDAP) for domain registration data queries. If you access domain registration information directly from registries via WHOIS, you might need to update your applications to use RDAP endpoints if the relevant registry discontinues WHOIS support.
> Route 53 domains service functionality remains unchanged. This notice applies only to direct registry WHOIS queries performed outside of Route 53 services.

You can view information about domains that were registered using Route 53. This information includes details such as when the domain was originally registered and contact information for the domain owner and for the technical, administrative, and billing contacts.

WHOIS

WHOIS is a free, publicly available directory containing information about domains sponsored by domain registrars and registries. It is provided as both a service that accepts queries on port 43, and as a website, each accessible via both IPv4 and IPv6. WHOIS is a distributed hierarchical lookup. For more information, see About WHOIS.

A WHOIS request to different levels of the hierarchy can provide different information:

- A request to the root WHOIS (whois.iana.org) provides information about the registry.
- A request to registry WHOIS provides information about the registrar, and some public information about the domain.

- A request to registrar WHOIS provides all public information about the domain.

Because there are multiple levels of WHOIS, including WHOIS lookups operated by the TLD registry and the domain registrar, turning your privacy protection off on the Route 53 console might only turn it off on the registrar-provided WHOIS. Some registries intentionally maintain privacy protection or redaction services for their WHOIS lookup services regardless of whether you have turned it off with Route 53. To get full information about your domain, we recommend that you use the registrar-provided WHOIS.

Note the following:

Emailing domain contacts when privacy protection is enabled

If privacy protection is enabled for the domain, contact information for the registrant, technical, and administrative contacts is replaced with contact information for the Amazon Registrar privacy service. For example, if the example.com domain is registered with Amazon Registrar and if privacy protection is enabled, the value of **Registrant Email** in the response to a WHOIS query would be similar to owner1234@example.com.identity-protect.org.

To contact one or more domain contacts when privacy protection is enabled, send an email to the corresponding email addresses. Route 53 automatically forwards your email to the applicable contact.

Reporting abuse

To report any illegal activity or violation of the Acceptable Use Policy, including inappropriate content, phishing, malware, or spam, send an email to trustandsafety@support.aws.com.

To view information about domains that are registered with Route 53

1. In a web browser, go to one of the following websites:
   - **Amazon Registrar:** https://registrar.amazon/lookup
   - **Gandi WHOIS:** https://whois.gandi.net (use this if your domain registrar is Gandi; see the section called “Finding your registrar”)
2. Enter the name of the domain that you want to view information about, and choose **Search**.

Deleting a domain name registration

For most top-level domains (TLDs), you can delete the registration if you no longer want it. If the registry allows you to delete the registration, perform the procedure in this topic.

Note the following:

The registration fee is not refundable

If you delete a domain name registration before the registration is scheduled to expire, AWS does not refund the registration fee.

TLDs that allow you to delete a domain registration

To determine whether you can delete the registration for your domain, see Domains that you can register with Amazon Route 53. If the section for your TLD doesn't include a "Deletion of domain registration" subsection, you can delete the domain. Before you delete the domain, make sure you have disabled the domain lock. For more information about disabling the domain lock, see DisableDomainTransferLock.

What if you can't delete a domain registration?

If the registry for your domain doesn't allow you to delete a domain name registration, you must wait for the domain to expire. To make sure that the domain isn't automatically renewed, disable automatic renewal for the domain. When the **Expires on** date passes, Route 53 automatically deletes the registration for the domain. For information about how to change the automatic renewal setting, see Enabling or disabling automatic renewal for a domain.

Delay before a domain is deleted and available to register again

Almost all registries prevent anyone from immediately registering a domain that has just expired. The typical delay is one to three months, depending on the TLD. For more information, see the "Deadlines for renewing and restoring domains" section for your TLD in Domains that you can register with Amazon Route 53.

> **Important**
>
> Don't delete a domain and expect to reregister it if you just want to transfer the domain between AWS accounts or transfer the domain to another registrar. See the applicable documentation instead:
>
> - Transferring a domain to a different AWS account

- Transferring a domain from Amazon Route 53 to another registrar

To delete a domain name registration

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered domains**.
3. Choose the name of your domain.

   If you want to delete a .co.uk, .me.uk, .org.uk, or .uk domain, see To delete .co.uk, .me.uk, .org.uk, and .uk domain name registrations.
4. If the registry for your TLD allows deleting a domain name registration, choose **Delete domain**.

   Some domains might require that Route 53 send an email to the registrant for the domain to verify that the registrant wants to delete the domain. If you receive an email, it will from one of the following email addresses:

   - noreply@registrar.amazon – for TLDs registered by Amazon Registrar.
   - noreply@domainnameverification.net or noreply@emailverification.info – for TLDs registered by our registrar associate, Gandi.

   To determine who the registrar is for your TLD, see Domains that you can register with Amazon Route 53.
5. If you receive the verification email, choose the link in the email, and either approve or reject the request to delete the domain.

   Important

   The registrant contact must immediately follow the instructions in the email, or Route 53 must cancel the deletion request as soon as after one day, as required by ICANN.

   You'll receive another email when your domain has been deleted. To determine the current status of your request, see Viewing the status of a domain registration.

6. Delete the records in the hosted zone for the deleted domain, and then delete the hosted zone. After you delete the hosted zone, Route 53 stops billing you the monthly charge for a hosted zone. For more information, see the following documentation:

   - Deleting records
   - Deleting a public hosted zone
   - Route 53 Pricing

7. If you encounter issues while deleting a domain name registration, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

To delete .co.uk, .me.uk, .org.uk, and .uk domain name registrations

If you want to delete a .co.uk, .me.uk, .org.uk, or .uk domain, you create an account with Nominet, the registry for .uk domains. For more information, see "Cancelling your domain name" on the Nominet website, https://www.nominet.uk/domain-support/.

> **Important**
>
> If you delete (cancel) a .uk domain name, it will be deleted within a few days and becomes available for anyone to register. If you just want to transfer the domain, do not delete it.

Here's an overview of the process:

1. On the Nominet website, follow the instructions for logging in for the first time. See https://secure.nominet.org.uk/auth/login.html. Nominet sends you an email with instructions for creating a password.
2. Follow the instructions in the email that you receive from Nominet.
3. Log in to the Nominet website, and follow the instructions for canceling (deleting) a domain name.

Contacting AWS Support about domain registration issues

AWS provides a Basic support plan, free of charge, for all AWS customers. The plan includes assistance for the following issues related to domain registration:

- Transferring domains to or from Amazon Route 53
- Transferring domains between AWS accounts
- Increasing quotas on Route 53 entities, such as the number of domains that you can register (See Quotas.)
- Changing the owner of a domain
- Changing contact information for the owner of a domain
- Resending confirmation and authorization emails
- Renewing domains
- Restoring expired domains
- Getting information about Route 53 billing
- Providing proof of identity for .uk domains
- Deleting domains or disabling automatic renewal after you close your AWS account

To contact AWS Support about these and other issues related to domain registration, perform the applicable procedure.

Topics

- Contacting AWS Support when you can sign in to your AWS account
- Contacting AWS Support when you can't sign in to your AWS account

Contacting AWS Support when you can sign in to your AWS account

To contact AWS Support when you're able to sign in to your AWS account, perform the following procedure:

1. Using the AWS account that the domain is currently registered to, sign in to the AWS Support.

   > **Important**
   >
   > You must sign in by using the root account that the domain is currently registered to. This requirement prevents unauthorized users from hijacking your account.

2. Choose **Account and Billing**
3. Specify the following values:

Regarding

Accept the default value of **Account and Billing Support**.

Service

Accept the default value of **Domains**.

Category

Accept the default value of **Registration Issue**.

Severity

Choose the applicable severity.

Subject

Enter a brief summary of the issue.

Description

Describe the issue that you're having in more detail, and attach any relevant documents or screenshots.

Contact method

Choose the contact method, **Web**. AWS contacts you using the email address that's associated with your AWS account.

4. Choose **Submit**.

Contacting AWS Support when you can't sign in to your AWS account

To contact AWS Support when you can't sign in to your AWS account, perform the following procedure:

1. Go to the I'm an AWS customer and I'm looking for billing or account support page.
2. Fill out the form.
3. Choose **Submit**.

Downloading a domain billing report

If you manage multiple domains and you want to view charges by domain for a specified time period, you can download a domain billing report. This report includes all charges that apply to domain registration, including the following:

- Registering a domain
- Renewing registration for a domain
- Transferring a domain to Amazon Route 53
- Changing the owner of a domain (for some TLDs, this operation is free)

Sometimes your billing report can show billing periods into the future. This happens because the domain auto renewal process starts the month before the domain expires. Therefore, for example, in your August report, you might see a billing period that starts the September after and runs until the September of the following year.

When you run the report using the console, you can choose the following options:

- **Last 12 months**: The report includes charges from one year before you ran the report until the current day. For example, if you run the report on June 3, it includes charges from June 3 the previous year until the current day.
- **Individual months in the last year**: The report includes charges for the specified month.

If you run the report programmatically, you can get charges for any date range, starting with July 31, 2014. That's the date that Route 53 started to support domain registration. For example, see view-billing in the *AWS CLI Command Reference*.

The billing report is in CSV format, and the contents are described by the ViewBilling API.

To download a domain billing report

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Registered Domains**.
3. Choose **Domain billing report**.
4. Choose the date range for the report, and then choose **Download domain report**.

5. Follow the prompts to open the report or to save it.
6. If you encounter issues while downloading a domain billing report, you can contact AWS Support for free. For more information, see Contacting AWS Support about domain registration issues.

Domains that you can register with Amazon Route 53

> **Important**
>
> The top-level domain restrictions below apply to domain registration only. You can use the Route 53 DNS service with any top-level domain you choose and with any domain registrar. The information on this page pertains only to the domains you can register with Route 53. For more information about Route 53 as a DNS service, see How internet traffic is routed to your website or web application.

The following lists of generic and geographic top-level domains show the top-level domains (TLDs) that you can use to register domains with Amazon Route 53.

Registering domains with Route 53

The TLDs that you can register with Route 53 are included in the following lists. If the TLD isn't included, you can't register the domain with Route 53.

Transferring domains to Route 53

You can transfer a domain to Route 53 if the TLD is included on the following lists. If the TLD isn't included, you can't transfer the domain to Route 53.

For most TLDs, you need to get an authorization code from the current registrar to transfer a domain. To determine whether you need an authorization code, see the "Authorization code required for transfers" section for your TLD.

Pricing for domain registration and transfers

For information about the cost to register domains or transfer them to Route 53, see Amazon Route 53 Pricing for Domain Registration. Certain domain names are designated as premium, often because they are shorter, more descriptive, or easier to remember. The registry sets prices for each premium domain individually. Prices for operations such as registration, transfer,

renewal, and restoration might differ from the standard TLD prices and can change over time. The pricing list doesn't include registry premium prices, which vary by domain name. To check whether a domain is premium and to see its price, search for it in the Route 53 console or use the `GetDomainPrices` API operation.

Using Route 53 as your DNS service

You can use Route 53 as the DNS service for any domain, even if the TLD for the domain isn't included on the following lists. For more information about Route 53 as a DNS service, see How internet traffic is routed to your website or web application. For information about how to transfer DNS service for your domain to Route 53, see Making Amazon Route 53 the DNS service for an existing domain.

Internationalized domain names

Not all TLDs support internationalized domain names (IDNs), meaning domain names that include characters other than ASCII characters a-z, 0-9, and - (hyphen). The listing for each TLD indicates whether that TLD supports IDNs. For more information about internationalized domain names, see DNS domain name format.

Registering geographic domains with TLDs

The rules for registration of geographic TLDs vary by country. Some countries are unrestricted, meaning that anyone in the world can register, while others have certain restrictions, such as residency. The listing for each geographic TLD indicates any restrictions.

Index to supported top-level domains

Topics

- Generic top-level domains
- Geographic top-level domains

Generic top-level domains

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | WXYZ

A

.ac, .academy, .accountants, .actor, .adult, .agency, .airforce, .apartments, .app, .art, .associates, .auction, .audio

B

.band, .bar, .bargains, .beer,.bet, .bid, .bike, .bingo, .bio, .biz, .black, .blue, .boo, .bot, .boutique, .build, .builders, .business, .buzz

C

.cab, .cafe, .camera, .camp, .capital, .cards, .care, .careers, .cash, .casino, .catering, .cc, .center, .ceo, .chat, .cheap, .christmas, .church, .city, .claims, .cleaning, .click, .clinic, .clothing, .cloud, .club, .coach, .codes, .coffee, .college, .com, .community, .company, .computer, .condos, .construction, .consulting, .contact, .contractors, .cool, .coupons, .credit, .creditcard, .cruises

D

.dad, .dance, .dating, .day, .deal, .deals, .degree, .delivery, .democrat, .dental, .design, .dev, .diamonds, .diet, .digital, .direct, .directory, .discount, .diy, .dog, .domains

E

.earth, .education, .email, .energy, .engineering, .enterprises, .equipment, .esq, .estate, .events, .exchange, .expert, .exposed, .express

F

.fail, .fan, .farm, .finance, .financial, .fish, .fit, .fitness, .flights, .florist, .flowers, .fm, .foo, .food, .football, .forsale, .forum, .foundation, .free, .fun, .fund, .furniture, .futbol, .fyi

G

.gallery, .games, .gift, .gifts, .gives, .glass, .global, .gmbh, .gold, .golf, .graphics, .gratis, .green, .gripe, .group, .guide, .guitars, .guru

H

.haus, .health, .healthcare, .help, .hiv, .hockey, .holdings, .holiday, .host, .hosting, .hot, .house, .how

I

.im, .immo, .immobilien, .industries, .info, .ink, .institute, .insure, .international, .investments, .io, .irish

J

.jewelry, .juegos

K

.kaufen, .kim, .kitchen, .kiwi

L

.land, .law, .lease, .legal, .lgbt, .life, .lifestyle, .lighting, .limited, .limo, .link, .live, .living, .llc, .loan, .loans, .lol, .love, .ltd

M

.maison, .management, .marketing, .mba, .media, .memorial, .menu, .mobi, .moda, .moi, .money, .mortgage, .mov, .movie

N

.name, .net, .network, .news, .nexus, .ninja, .now

O

.one, .onl, .online, .org

P

.page, .partners, .parts, .phd, .photo, .photography, .photos, .pics, .pictures, .pink, .pizza, .place, .plumbing, .plus, .poker, .porn, .press, .pro, .productions, .prof, .properties, .property, .pub. .pw (Palau)

Q

.qpon

R

.realty, .recipes, .red, .reise, .reisen, .rentals, .repair, .report, .republican, .rest, .restaurant, .reviews, .rip, .rocks, .rsvp, .run

S

.sale, .sarl, .school, .schule, .services, .sex, .sexy, .shiksha, .shoes, .shop, .shopping, .show, .singles, .site, .ski, .soccer, .social, .solar, .solutions, .software, .soy, .space, .spot, .store, .stream, .studio, .style, .sucks, .supplies, .supply, .support, .surgery, .systems

T

.tattoo, .tax, .taxi, .team, .tech,.technology, .tennis, .theater, .tienda, .tips, .tires, .today, .tools, .tours, .town, .toys, .trade, .training, .tv

U

.university, .uno

V

.vacations, .vegas, .ventures, .vg, .viajes, .video, .villas, .vision, .vote, .voyage

WXYZ

.watch, .website, .wedding, .wiki, .win, .wine, .work, .works, .world, .wtf, .xxx, .xyz, .zip, .zone

Geographic top-level domains

Africa

.ac (Ascension Island), .co.za (South Africa), .sh (Saint Helena)

Americas

.ai (Anguilla), .ca (Canada), .cl (Chile), .co (Colombia), .com.ar (Argentina), .com.br (Brazil), .com.mx (Mexico), .mx (Mexico), .us (United States), .vc (Saint Vincent and the Grenadines), .vg (British Virgin Islands)

Asia/Oceania

.au (Australia), .cc (Cocos (Keeling) Islands), .co.nz (New Zealand), .com.au (Australia), .com.sg (Republic of Singapore), .fm (Federated States of Micronesia), .in (India), .jp (Japan), .io (British Indian Ocean Territory), .my (Malaysia), .net.au (Australia), .net.nz (New Zealand), .nz (New Zealand), .org.nz (New Zealand), .pw (Palau), .qa (Qatar), .ru (Russian Federation), .sg (Republic of Singapore)

Europe

.be (Belgium), .berlin (city of Berlin in Germany), .ch (Switzerland), .co.uk (United Kingdom), .cz (Czech Republic), .de (Germany), .es (Spain), .eu (European Union), .fi (Finland), .fr (France), .gg (Guernsey), .im (Isle of Man), .it (Italy), .me (Montenegro), .me.uk (United Kingdom), .nl (the Netherlands), .org.uk (United Kingdom), .ruhr (Ruhr region, western part of Germany), .se (Sweden), .uk (United Kingdom), .wien (city of Vienna in Austria)

Generic top-level domains

Generic top-level domains (gTLDs) are global extensions that are used and recognized around the world, such as .com, .net, and .org. They also include specialty domains such as .bike, .condos, and .marketing.

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | WXYZ

A

.ac, .academy, .accountants, .actor, .adult, .agency, .airforce, .apartments, .app, .art, .associates, .auction, .audio

B

.band, .bar, .bargains, .beer, .bet, .bid, .bike, .bingo, .bio, .biz, .black, .blue, .boo, .bot, .boutique, .build, .builders, .business, .buzz

C

.cab, .cafe, .camera, .camp, .capital, .cards, .care, .careers, .cash, .casino, .catering, .cc, .center, .ceo, .chat, .cheap, .church, .christmas, .city, .claims, .cleaning, .click, .clinic, .clothing, .cloud, .club, .coach, .codes, .coffee, .college, .com, .community, .company, .computer, .condos, .construction, .consulting, .contact, .contractors, .cool, .coupons, .credit, .creditcard, .cruises

D

.dad, .dance, .dating, .day, .deal, .deals, .degree, .delivery, .democrat, .dental, .design, .dev, .diamonds, .diet, .digital, .direct, .directory, .discount, .diy, .dog, .domains

E

.earth, .education, .email, .energy, .engineering, .enterprises, .equipment, .esq, .estate, .events, .exchange, .expert, .exposed, .express

F

.fail, .fan, .farm, .finance, .financial, .fish, .fit, .fitness, .flights, .florist, .flowers, .fm, .foo, .food, .football, .forsale, .forum, .foundation, .free, .fun, .fund, .furniture, .futbol, .fyi

G

.gallery, .games, .gift, .gifts, .gives, .glass, .global, .gmbh, .gold, .golf, .graphics, .gratis, .green, .gripe, .group, .guide, .guitars, .guru

H

.haus, .health, .healthcare, .help, .hiv, .hockey, .holdings, .holiday, .host, .hosting, .hot, .house, .how

I

.im, .immo, .immobilien, .industries, .info, .ink, .institute, .insure, .international, .investments, .io, .irish

J

.jewelry, .juegos

K

.kaufen, .kim, .kitchen, .kiwi

L

.land, .law, .lease, .legal, .lgbt, .life, .lifestyle, .lighting, .limited, .limo, .link, .live, .living, .llc, .loan, .loans, .lol, .love, .ltd

M

.maison, .management, .marketing, .mba, .media, .memorial, .menu, .mobi, .moda, .moi, .money, .mortgage, .mov, .movie

N

.name, .net, .network, .news, .nexus, .ninja, .now

O

.one, .onl, .online, .org

P

.page, .partners, .parts, .phd, .photo, .photography, .photos, .pics, .pictures, .pink, .pizza, .place, .plumbing, .plus, .poker, .porn, .press, .pro, .productions, .prof, .properties, .property, .pub

Q

.qpon

R

.realty, .recipes, .red, .reise, .reisen, .rentals, .repair, .report, .republican, .rest, .restaurant, .reviews, .rip, .rocks, .rsvp, .run

S

.sale, .sarl, .school, .schule, .services, .sex, .sexy, .shiksha, .shoes, .shop, .shopping, .show, .singles, .site, .ski, .soccer, .social, .solar, .solutions, .software, .soy, .space, .spot, .store, .stream, .studio, .style, .sucks, .supplies, .supply, .support, .surgery, .systems

T

.tattoo, .tax, .taxi, .team,.tech, .technology, .tennis, .theater, .tienda, .tips, .tires, .today, .tools, .tours, .town, .toys, .trade, .training, .tv

U

.university, .uno

V

.vacations, .vegas, .ventures, .vg, .viajes, .video, .villas, .vision, .vote, .voyage

WXYZ

.watch, .website, .wedding, .wiki, .win, .wine, .work, .works, .world, .wtf, .xxx, .xyz, .zip, .zone

.ac

See .ac (Ascension Island).

Return to index

.academy

Used by educational institutions such as schools and universities. Also used by recruiters, advisors, advertisers, students, teachers, and administrators who are affiliated with educational institutions.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.accountants

Used by businesses, groups, and individuals affiliated with the accounting profession.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.actor

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.adult

Used for websites that host adults-only content.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.agency

Used by any businesses or groups that identify as agencies.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.airforce

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.apartments

Used by real estate agents, landlords, and renters.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.app

Used as a secure domain for developers, entrepreneurs, and businesses to showcase their apps.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.art

Used for art-related websites, portfolios, galleries, and creative projects.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.associates

Used by businesses and firms that include the term "associates" in their titles. Also used by any groups or agencies that want to indicate the professional nature of their organizations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.auction

Used for events related to auctions and auction-based buying and selling.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Spanish, and Latin.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.audio

> **Important**
>
> You can no longer use Route 53 to register new .audio domains or transfer .audio domains to Route 53. We'll continue to support .audio domains that are already registered with Route 53.

Used by the audiovisual industry and anyone interested in broadcasting, sound equipment, audio production, and audio streaming.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Not supported. You can no longer transfer .audio domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.band

Used for sharing information about musical bands and band events. Also used by musicians to connect with their fan base and sell band-related merchandise.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Spanish, and Latin.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.bar

Used by bars, pubs, and nightlife-related businesses and enthusiasts.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.bargains

Used for information about sales and promotions.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.beer

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.bet

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.bid

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.bike

Used by businesses or groups that cater to cyclists, such as bike stores, motorcycle dealerships, and repair shops.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.bingo

Used for online gaming websites or for sharing information about the game of bingo.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.bio

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.biz

Used for business or commercial use.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Simplified Chinese, Traditional Chinese, Danish, Finnish, German, Hungarian, Japanese, Korean, Latvian, Lithuanian, Norwegian, Polish, Portuguese, Spanish, and Swedish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.black

Used by those who like the color black or those who want to associate the color black with their business or brand.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.blue

Used by those who like the color blue or those who want to associate the color blue with their business or brand.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.boo

Used as a fun and memorable secure domain, whether for love, laughs, or a surprise.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.bot

Used for chatbot and AI-related products and services. This domain extension helps convey the nature of automated systems, conversational interfaces, and emerging technologies in artificial intelligence.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.boutique

Used for information about boutiques and small specialty shops.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.build

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.builders

Used by companies and individuals affiliated with the construction industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.business

Used by any kind of business. Can be used as an alternative to the .biz extension.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.buzz

Used for information about the latest news and events.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cab

Used by companies and individuals affiliated with the taxicab industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cafe

Used by cafe businesses and those who have an interest in cafe culture.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.camera

Used by photography enthusiasts and anyone who wants to share photos.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.camp

Used by parks and recreation departments, summer camps, writers' workshops, fitness camps, and camping enthusiasts.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.capital

Used as a general category that describes any kind of capital, such as financial capital or the capital of a city.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cards

Used by businesses that specialize in cards such as ecards, printed greeting cards, business cards, and playing cards. Also ideal for gamers who want to discuss the rules and strategies of card games.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.care

Used by businesses or agencies in the care-giving field. Also used by charitable organizations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.careers

Used for information about job recruitment.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cash

Used by any organization, group, or individual engaged in money-related activities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.casino

Used by the gambling industry or by gamers who want to share information about gambling and casino games.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.catering

Used by catering businesses or those who share information about food-related events.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cc

See .cc (Cocos (Keeling) Islands).

Return to index

.center

Used as a generic extension for everything from research organizations to community centers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.ceo

Used for information about CEOs and their equals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for German.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.chat

Used by any kind of online chat website.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cheap

Used by e-commerce websites to promote and sell inexpensive products.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.christmas

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 43 days after expiration
- Domain is deleted from Route 53: 44 days after expiration
- Restoration with the registry is possible: Between 44 days and 86 days after expiration
- Domain is deleted from the registry: 86 days after expiration

.church

Used by churches of any size or denomination to connect with their congregations and to publish information about church-related events and activities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.city

Used to provide information about specific cities, such as points of interest, top local spots to visit, or neighborhood activities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.claims

Used by companies that handle insurance claims or provide legal services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cleaning

Used by businesses or individuals that provide cleaning services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.click

Used by businesses that want to associate the action of clicking with their websites, for example, clicking products on a website to purchase them.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.clinic

Used by the health care industry and by medical professionals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.clothing

Used by those in the fashion industry, including retailers, department stores, designers, tailors, and outlets.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cloud

Used as a general extension, but ideal for companies that provide cloud computing technologies and services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.club

Used by any type of club or organization.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Spanish and Japanese.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.coach

Used by anyone with an interest in coaching, such as sports professionals, lifestyle coaches, or corporate trainers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.codes

Used as a generic extension for all kinds of code, such as codes of conduct, building codes, or programming code.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.coffee

Used by those in the coffee industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.college

Used by educational institutions such as schools and universities. Also used by recruiters, advisors, advertisers, students, teachers, and administrators who are affiliated with educational institutions.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic, simplified and traditional Chinese, Cyrillic, Greek, Hebrew, Japanese, and Thai.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.com

Used for commercial websites. It is the most popular extension on the internet.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.community

Used by any type of community, club, organization, or special interest group.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.company

Used as a generic extension for companies of all kinds.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.computer

Used as a generic extension for information about computers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.condos

Used by individuals and businesses associated with condominiums.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.construction

Used by those in the construction industry, such as builders and contractors.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.consulting

Used by consultants and others who are affiliated with the consulting industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic, Chinese, French, Cyrillic, Devanagari, German, Greek, Hebrew, Japanese, Korean, Latin, Spanish, Tamil, and Thai.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.contact

Used by churches of any size or denomination to connect with their congregations and to publish information about church-related events and activities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.contractors

Used by contractors, such as contractors in the construction industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cool

Used by organizations and groups who want to associate their brand with the latest trends.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.coupons

Used by retailers and manufacturers that provide online coupons and coupon codes.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.credit

Used by the credit industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.creditcard

Used by companies or banks that issue credit cards.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.cruises

Used by the voyage industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.dad

Used for fatherhood-related content, blogs, and marketing to dads.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.dance

Used by dancers, dance instructors, and dance schools.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.dating

Used for dating websites.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.day

Used to bring attention to any important day, from birthdays to weddings to special occasions.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.deal

Used for websites offering deals, discounts, and special offers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.deals

Used to provide information about online bargains and sales.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.degree

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.delivery

Used by companies that deliver any kind of merchandise or service.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.democrat

Used for information about the Democratic Party. Also used by officials running for elected office, elected officials, political enthusiasts, consultants, and advisors.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.dental

Used by dental professionals and dental suppliers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.design

Used by churches of any size or denomination to connect with their congregations and to publish information about church-related events and activities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.dev

Used as a secure domain for developers and technology.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.diamonds

Used by diamond enthusiasts and those in the diamond industry, including sellers, resellers, and merchandisers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.diet

> **Important**
>
> You can no longer use Route 53 to register new .diet domains or transfer .diet domains to Route 53. We'll continue to support .diet domains that are already registered with Route 53.

Used by health and fitness professionals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Not supported. You can no longer transfer .diet domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.digital

Used for anything and everything digital, but ideal for technology businesses.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.direct

Used as a general extension, but ideal for those who sell products directly to customers through an e-commerce website.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.directory

Used by the media sector.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.discount

Used for discount websites and businesses that slash prices.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.diy

Used for do-it-yourself projects, crafts, and home improvement content.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.dog

Used by dog lovers and those who provide canine services and products.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.domains

Used for information about domain names.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.earth

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.education

Used for information about education.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.email

Used for information about promoting email.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.energy

Used as a general extension, but ideal for those in the energy or energy conservation fields.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.engineering

Used by engineering firms and professionals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.enterprises

Used for information about enterprises and businesses.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.equipment

Used for information about equipment, equipment retailers or manufacturers, and rental shops.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.esq

Used by lawyers and distinguished people to showcase credentials and experience.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.estate

Used for information about housing and the housing sector.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.events

Used for information about events of all kinds.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.exchange

Used for any type of exchange: the stock exchange, the exchange of goods, or even the simple exchange of information.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.expert

Used by those who have specialized knowledge in a variety of fields.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.exposed

Used as a generic extension for a variety of subjects, including photography, tabloids, and investigative journalism.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.express

Used as a general extension, but ideal for those who want to emphasize the speedy delivery of good or services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fail

Used by anyone who has made mistakes, but ideal for publishing humorous "fail" blunders and bloopers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration

- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fan

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.farm

Used by those in the farming industry, such as farmers and agricultural engineers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.finance

Used by the financial sector.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.financial

Used by the financial sector.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fish

Used as a general extension, but ideal for websites related to fish and fishing.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fit

Used for fitness, health, and wellness-related websites and businesses.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fitness

Used to promote fitness and fitness services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.flights

Used by travel agents, airlines, and anyone affiliated with the travel industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.florist

Used by florists.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.flowers

> **Important**
>
> You can no longer use Route 53 to register new .flowers domains or transfer .flowers domains to Route 53. We'll continue to support .flowers domains that are already registered with Route 53.

Used for anything related to flowers, such as online flower sales or information about flower growing and breeding.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Not supported. You can no longer transfer .flowers domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fm

See .fm (Federated States of Micronesia).

Return to index

.foo

Used as a secure domain for developers. Use .foo for everything and anything.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.food

Used for food-related businesses, recipes, and culinary content.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.football

Used by anyone involved in the sport of football.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.forsale

Used for selling goods and services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.forum

Used for online forums, discussion boards, and community platforms.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.foundation

Used by non-profit organizations, charities, and other kinds of foundations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.free

Used for websites offering free products, services, or content.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration

- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fun

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fund

Used as a general extension for anything related to funding.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.furniture

Used by furniture makers and sellers and anyone affiliated with the furniture industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.futbol

Used for information about soccer (futbol).

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.fyi

Used as a general extension, but ideal for sharing information of all kinds. "FYI" is an acronym for "for your information."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.gallery

Used by owners of art galleries.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.games

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.gift

Used by businesses or organizations that sell gifts or provide gift-related services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.gifts

Used by businesses or organizations that sell gifts or provide gift-related services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.gives

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.glass

Used by those in the glass industry, such as glass cutters and window installers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.global

Used by businesses or groups with an international market or vision.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic, Belarusian, Bosnian, Bulgarian, Chinese (Simplified) Chinese (Traditional), Danish, German, Hindi, Hungarian, Icelandic, Korean, Latvian, Lithuanian, Macedonian, Montenegrin, Polish, Russian, Serbian, Spanish, Swedish, and Ukrainian.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.gmbh

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.gold

Used as a general extension, but ideal for companies that purchase or sell gold or gold-related products.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.golf

Used for websites devoted to the game of golf.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.graphics

Used by those in the graphics industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.gratis

Used for websites that offer free products, such as promotional items, downloads, or coupons. "Gratis" is a Spanish word that means "free."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.green

Used for websites devoted to conservation, ecology, the environment, and the green lifestyle.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.gripe

Used for sharing complaints and criticism.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.group

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.guide

Used as a general extension, but ideal for websites that focus on travel destinations, services, and products.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.guitars

> **Important**
>
> You can no longer use Route 53 to register new .guitars domains or transfer .guitars domains to Route 53. We'll continue to support .guitars domains that are already registered with Route 53.

Used by guitar enthusiasts.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Not supported. You can no longer transfer .guitars domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.guru

Used by those who want to share their knowledge about a variety of subjects.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.haus

Used by real estate and construction industries. "Haus" is a German word that means "house."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.health

Used for health, medical, and wellness-related websites and organizations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.healthcare

Used by the healthcare sector.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.help

Used as a general extension, but ideal for websites that provide online help and information.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.hiv

Used for websites devoted to the fight against HIV.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.hockey

Used for websites devoted to the game of hockey.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.holdings

Used by financial advisors, stockbrokers, and those who work with investments.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.holiday

Used by those in the travel industry and individuals and businesses involved in party planning and special occasions.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.host

Used by companies that provide web hosting platforms and services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic, Simplified Chinese, Traditional Chinese, Greek, Hebrew, Korean, and Thai.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.hosting

Important

You can no longer use Route 53 to register new .hosting domains or transfer .hosting domains to Route 53. We'll continue to support .hosting domains that are already registered with Route 53.

Used for hosting websites or by those in the hosting industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Not supported. You can no longer transfer .hosting domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.hot

Used for websites featuring trending topics, popular content, or time-sensitive offers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.house

Used by real estate agents and buyers and sellers of houses.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.how

Used by thinkers, tinkerers, and knowledge-seekers to explore everything from "how do" to "how to."

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.im

See .im (Isle of Man).

Return to index

.immo

Used by the real estate sector.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.immobilien

Used for information about real estate.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.industries

Used by any business or commercial enterprise that wants to identify as an industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.info

Used for the dissemination of information.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.ink

Used by tattoo enthusiasts or any industry related to ink, such as printing and publishing industries.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic and Latin.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.institute

Used by any organization or group, especially research and educational organizations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.insure

Used by insurance companies and insurance brokers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.international

Used by businesses that have international chains, individuals who travel internationally, or charity organizations with an international influence.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.investments

Used as a general extension, but ideal for promoting investment opportunities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.io

See .io (British Indian Ocean Territory).

Return to index

.irish

Used for promoting Irish culture and organizations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic, Simplified Chinese, Traditional Chinese, French, German, Greek, Hebrew, Japanese, Korean, Spanish, Tamil, and Thai.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.jewelry

Used by jewelry sellers and buyers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.juegos

> **Important**
>
> You can no longer use Route 53 to register new .juegos domains or transfer .juegos domains to Route 53. We'll continue to support .juegos domains that are already registered with Route 53.

Used for gaming websites of all kinds. "Juegos" is a Spanish word that means "games."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Not supported. You can no longer transfer .juegos domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration

- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.kaufen

Used for information about e-commerce.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.kim

Used by people whose name or surname is Kim.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.kitchen

Used by kitchen retailers, cooks, food bloggers, and anyone in the food industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.kiwi

Used by companies and individuals who want to support New Zealand kiwi culture. It is also used as a platform for charitable aid in the reconstruction of Christchurch, damaged by earthquakes in 2010 and 2011.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Maori.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.land

Used by farmers, real estate agents, commercial developers, and anyone with an interest in property.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.law

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.lease

Used by realtors, landlords, and renters.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.legal

Used by members of the legal profession.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.lgbt

Used by the community of lesbian, gay, bisexual, and transgender people.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.life

Used as a general extension, and suitable for a wide range of businesses, groups, and individuals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.lifestyle

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.lighting

Used by photographers, designers, architects, engineers, and others with an interest in lighting.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.limited

Used as a general extension, and suitable for a wide range of businesses, groups, and individuals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.limo

Used by chauffeurs, limousine companies, and car rental agencies.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.link

Used for information about the creation of online shortcut links.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Uniregistry is the registry for .LINK domains. Due to Uniregistry policy, the registry-level WHOIS shows "REDACTED FOR PRIVACY". Removing our privacy protection feature will only affect the information displayed at the registrar-level Amazon Registrar lookup.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.live

Used as a general extension, and suitable for a wide range of businesses, groups, and individuals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.living

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.llc

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.loan

Used by lenders, borrowers, and credit professionals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Danish, German, Norwegian, and Swedish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.loans

Used by lenders, borrowers, and credit professionals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.lol

Used for humor and comedy websites. "LOL" is an acronym for "laugh out loud."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic, French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.love

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.ltd

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.maison

Used by the real estate sector.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.management

Used for information about the business world and company management.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.marketing

Used by the marketing sector for a variety of purposes.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.mba

Used for websites that provide information about the master's degree in business administration (MBA).

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.media

Used by the media and entertainment sectors.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.memorial

Used by commemorative organizations dedicated to honoring events and people.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.menu

Used by restaurants, cafes, and food service businesses to showcase their menus.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.mobi

Used by companies and individuals who want to have their websites accessible on mobile phones.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.moda

Used for information about fashion.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.moi

French for "me." Used to signal French-language content to Francophone end users. Suitable for personal blogs, CVs, portfolios, and hobby or interest sites in French.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.money

Used for websites that focus on money and money-related activities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.mortgage

Used by the mortgage industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.mov

Used for moving pictures and other things that move.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.movie

Used for websites that provide information about movies and movie-making. Suitable for both professionals and fans.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chines, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.name

Used by anyone who wants to create a personalized web presence.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Verisign, the registry for .name TLDs, allows you to register both second-level domains (*name*.name) and third-level domains (*firstname.lastname*.name). Route 53 supports only second-level domains, both for registering domains and for transferring existing domains to Route 53.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.net

Used for all types of websites. The .net extension is an abbreviation of network.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.network

Used by those in the network industry or those who want to build connections through networking.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.news

Used for distributing any newsworthy information such as current events or information related to journalism and communication.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.nexus

Used as a secure hub that brings it all together for whatever you're doing online.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.ninja

Used by individuals and businesses who want to associate themselves with the abilities of a ninja.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.now

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.one

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.onl

The .onl extension is an abbreviation for "online," and it is also the short term in Spanish for non-profit organization.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.online

The .onl extension is an abbreviation for "online," and it is also the short term in Spanish for non-profit organization.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.org

Used by all kinds of organizations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.page

Used to build a simple, secure online presence.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.partners

Used by law firms, investors, and a variety of companies. Also used for social websites that build relationships.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.parts

Used as a general extension, but ideal for parts manufacturers, sellers, and buyers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.phd

Used by anyone with a doctorate to showcase their publications and accomplishments.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.photo

Used by photographers and anyone interested in photos.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.photography

Used by photographers and anyone interested in photos.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.photos

Used by photographers and anyone interested in photos.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.pics

Used by photographers and anyone interested in photos.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.pictures

Used by anyone interested in photography, art, and media.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.pink

Used by those who like the color pink or those who want to associate the color pink with their business or brand.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.pizza

Used by pizza restaurants and pizza lovers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.place

Used as a general extension, but ideal for the home and travel sectors.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.plumbing

Used by those in the plumbing industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.plus

Used as a general extension, but ideal for plus-size clothing, add-on software, or any product that offers "extra" features or dimensions.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.poker

Used by poker players and gaming websites.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.porn

Used for adults-only websites.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.press

Used for adults-only websites.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.pro

Used by licensed and credentialed professionals and professional organizations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.productions

Used by studios and production houses that make commercials, radio ads, and music videos.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.prof

Used by professors to showcase teaching credentials, research, and publications.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.properties

Used for information about any type of property, including real estate or intellectual property. Also used by those who have houses, buildings, or land to sell, lease, or rent.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.property

Used for information about any type of property, including real estate or intellectual property. Also used by those who have houses, buildings, or land to sell, lease, or rent.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Not supported. You can no longer transfer .property domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.pub

Used by those in the publication, advertising, or brewing business.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.qpon

Used for coupons and promo codes.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.realty

Used by real estate agents, brokers, and property-related businesses.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.recipes

Used by those with recipes to share.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.red

Used by those who like the color red or those who want to associate the color red with their business or brand.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.reise

Used for websites related to travels or journeys. "Reise" is a German word that means "rise," "arise," or "set out on a journey."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.reisen

Used for websites related to travels or journeys. "Reisen" is a German word that means "to travel."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.rentals

Used for all types of rentals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.repair

Used by repair services or by those who want to teach others how to repair all kinds of items.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.report

Used as a general extension, but ideal for information about business reports, community publications, book reports, or news reporting.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.republican

Used for information about the Republican Party. Also used by officials running for elected office, elected officials, political enthusiasts, consultants, and advisors.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.rest

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.restaurant

Used by the restaurant industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.reviews

Used by those who want give their opinions and read the comments of others.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.rip

Used for websites dedicated to death and memorials. "RIP" is an acronym for "rest in peace."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.rocks

Used as a general extension, but ideal for anyone who "rocks": musicians, geologists, jewelers, climbers, and more.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.rsvp

Used for events and reservations, from weddings to fundraisers to business bookings.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.run

Used as a general extension, but ideal for the fitness and sports industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.sale

Used by e-commerce websites.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.sarl

Used by limited liability companies typically located in France. "SARL" is an acronym for Société à Responsabilité Limité.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.school

Used for information about education, educational institutions, and school-related activities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.schule

Used for information about German-based education, educational institutions, and school-related activities. "Schule" is a German word that means "school."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.services

Used for websites that focus on services of any kind.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.sex

Used for adults-only content.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.sexy

Used for sexual content. Also used for describing the most popular and exciting brands, products, information, and websites.

Return to index

> **Important**
>
> You can no longer use Route 53 to register new .sexy domains or transfer .sexy domains to Route 53. We'll continue to support .sexy domains that are already registered with Route 53.

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Not supported. You can no longer transfer .sexy domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.shiksha

Used by educational institutions. "Shiksha" is an Indian term for school.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.shoes

Used by shoe retailers, designers, manufacturers, or fashion bloggers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.shop

Used for online and brick-and-mortar retail businesses, as well as service providers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.shopping

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.show

Used as a general extension, but ideal for the entertainment industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.singles

Used by dating services, resorts, and other businesses that cater to those who want to make a connection.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.site

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.ski

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.soccer

Used for websites dedicated to the game of soccer.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.social

Used for information about social media, forums, and online conversations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.solar

Used for information about the solar system or solar energy.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.solutions

Used by consultants, do-it-yourself services, and advisors of all kinds.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.software

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.soy

Used for the digital needs of the US Hispanic market, for brands, publishers, and consumers.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.space

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.spot

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.store

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.stream

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.studio

Used as a general extension, but ideal for those in the real estate, art, or entertainment industries.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.style

Used as a general extension, but ideal for websites dedicated to the latest trends, especially trends in fashion, design, architecture, and art.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.sucks

Used as a general extension, but ideal for those who want to share negative experiences or warn others about scams, frauds, or faulty products.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.supplies

Used by businesses that sell goods online.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.supply

Used by businesses that sell goods online.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.support

Used by businesses, groups, or charities that offer any kind of support, including customer, product, or system support or emotional, financial, or spiritual support.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.surgery

Used for information about surgery, medicine, and healthcare.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.systems

Used primarily by the technology industry and those who offer technology services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tattoo

Used by tattoo enthusiasts and the tattoo industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Cyrillic (primarily Russian), French, German, Italian, Portuguese, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tax

Used for information about taxes, tax preparation, and tax law.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.taxi

Used by cab, chauffeur, and shuttle companies.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.team

Used by any business or organization that wants to identify as a team.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tech

Used by technology enthusiasts and those dedicated to technology in companies, services, and manufacturers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.technology

Used by technology enthusiasts and those dedicated to technology in companies, services, and manufacturers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tennis

Used for information related to the game of tennis.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.theater

Used for websites dedicated to theaters, plays, and musicals.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tienda

Used by retail businesses that want to connect with Spanish-speaking consumers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tips

Used by those who want to share their knowledge and advice on virtually any topic.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tires

Used by manufacturers, distributors, or buyers of tires.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.today

Used for information about current events, news, weather, entertainment, and more.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tools

Used for information about any kind of tool.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tours

Used as a general extension, but ideal for travel companies.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.town

Used to promote a city's locale, culture, and community.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.toys

Used by the toy industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.trade

Used as a general extension, but ideal for commerce websites or trading services.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Danish, German, Norwegian, and Swedish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.training

Used by trainers, coaches, and educators.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration

- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.tv

Used for information about television and media.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

None.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.university

Used by universities and other educational organizations.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.uno

Used for information about the Hispanic, Portuguese, and Italian communities.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.vacations

Used by the travel and tourism industry.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.vegas

Used to promote the city of Las Vegas and the Las Vegas lifestyle.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.ventures

Used by entrepreneurs, startups, venture capitalists, investment banks, and financiers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.vg

See .vg (British Virgin Islands).

Return to index

.viajes

Used by travel agencies, tour operators, travel blogs, tour companies, rental services, travel bloggers, and travel retailers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.video

Used by media and video industries.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German, Latin, and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.villas

Used by real estate agents and property owners who have villas to sell, rent, or lease.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.vision

Used as a general extension, but ideal for vision specialists such as optometrists and ophthalmologists.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.vote

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.voyage

Used by travel agencies, tour operators, travel blogs, tour companies, rental services, travel bloggers, and travel retailers.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.watch

Used for information about streaming websites, web TVs, video, or watches.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.website

Used for information about website development, promotion, improvements, and experiences.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic, Simplified Chinese, Traditional Chinese, Greek, Hebrew, Japanese, Korean, and Thai.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.wedding

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

None.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Chinese, French, German and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.wiki

Used for information about online documentation.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic and Latin.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.win

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 75 days after expiration

.wine

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

Supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.work

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.works

Used by businesses, organizations, and individuals for information about work, job, and employment services. This extension can be used as an alternative to the .com, .net, or .org extensions.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.world

Used by anyone who wants to provide information about global subjects.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.wtf

Used by anyone who wants to identify with the popular (but profane) acronym "WTF."

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.XXX

Used for websites that host adults-only content.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.xyz

Used as a general extension for any purpose.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.zip

Used for tying things together or moving really fast.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

HTTPS is required for all websites on this TLD. For the domain to work properly in browsers, you must configure HTTPS for your website. For more information about configuring HTTPS with AWS, including resources to obtain an SSL certificate, see What Is AWS Certificate Manager?.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.zone

Used for information about any kind of zone, including time zones, climate zones, and sports zones.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for French and Spanish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

Geographic top-level domains

The following domain extensions are grouped by geography and include official country-specific extensions, known as *country code top-level domains* (ccTLDs). Examples include .be (Belgium), .in (India), and .mx (Mexico). The rules for registration of ccTLDs vary by country. Some countries are unrestricted, meaning that anyone in the world can register, while others have certain restrictions, such as residency. The listing for each ccTLD indicates any restrictions.

Important

During the transfer of any ccTLDs to Route 53, except for .cc and .tv, updates to the owner contact are ignored and the owner contact data from the registry is used. You can update the owner contact after the transfer is complete. For more information, see Updating contact information and ownership for a domain.

Return to index

Africa

.ac (Ascension Island), .co.za (South Africa), .sh (Saint Helena)

Americas

.ai (Anguilla), .ca (Canada), .cl (Chile), .co (Colombia), .com.ar (Argentina), .com.br (Brazil), .com.mx (Mexico), .mx (Mexico), .us (United States), .vc (Saint Vincent and the Grenadines), .vg (British Virgin Islands)

Asia/Oceania

.au (Australia), .cc (Cocos (Keeling) Islands), .co.nz (New Zealand), .com.au (Australia), .com.sg (Republic of Singapore), .fm (Federated States of Micronesia), .in (India), .jp (Japan), .io (British Indian Ocean Territory), .my (Malaysia), .net.au (Australia), .net.nz (New Zealand), .nz (New Zealand), .org.nz (New Zealand), .pw (Palau), .qa (Qatar), .ru (Russian Federation), .sg (Republic of Singapore)

Europe

.be (Belgium), .berlin (city of Berlin in Germany), .ch (Switzerland), .co.uk (United Kingdom), .cz (Czech Republic), .de (Germany), .es (Spain), .eu (European Union), .fi (Finland), .fr (France), .gg (Guernsey), .im (Isle of Man), .it (Italy), .me (Montenegro), .me.uk (United Kingdom), .nl

(the Netherlands), .org.uk (United Kingdom), .ruhr (Ruhr region, western part of Germany), .se (Sweden), .uk (United Kingdom), .wien (city of Vienna in Austria)

Africa

You can use the following top-level domains (TLDs) for Africa to register domains with Amazon Route 53.

, ,

Return to index

.ac (Ascension Island)

Return to index

Also used as a generic TLD that is popular for those in academia.

Lease period for registration and renewal

One year.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Determined by the registry.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 80 days after expiration

.co.za (South Africa)

Return to index

Lease period for registration and renewal

One year.

Restrictions

Only second-level domains are available for the .za extension. Route 53 supports the second-level domain .co.za.

Open to the public, with some restrictions:

- Registration is open to identifiable legal entities (individuals and legal persons).
- The domain name must pass a zone check during the registration process.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

No

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until one day before the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: 1 day before expiration
- Restoration with the registry is possible: Between 1 day and 9 days after expiration
- Domain is deleted from the registry: 9 days after expiration

.sh (Saint Helena)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration

- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 80 days after expiration

Americas

You can use the following top-level domains (TLDs) for the Americas to register domains with Amazon Route 53.

, , , , , , , , , , ,

Return to index

.ai (Anguilla)

Return to index

Lease period for registration and renewal

Registration: Two to ten years. Renewal: Two to nine years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date

- Late renewal with Route 53 is possible: Until 45 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.ca (Canada)

Return to index

Variants, with (à) or without (a) an accent mark, of a domain name are automatically reserved for the registrant and become part of an administrative bundle. In order to activate a domain in a bundle, the registrant must make a registration request for the domain. All domains within a bundle must be registered by the same registrant and with the same registrar. The registrant will also need to submit transfer request for all the domains in a bundle to complete the transfer.

Confirmation email from the TLD registry

When you register a .ca domain, you will receive an email with a link to the acceptation procedure of the registrant agreement. You must complete the procedure within seven days or your domain will not be registered.

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- Registration is open to individuals or organizations connected to Canada, as described by the Canadian Presence Requirements for Registrants.
- Registrant contact: You must provide the full and exact legal name of the owner of the domain.
- Admin and tech contacts: You must specify **Person** as the contact type and provide contact information for individuals living in Canada.
- You must select one of the following legal types during the registration process:
  - ABO: Aboriginal Peoples (individuals or groups) indigenous to Canada
  - ASS: Canadian unincorporated association
  - CCO: Canadian corporation, or Canadian province or territory

- CCT: Canadian citizen
- EDU: Canadian educational institution
- GOV: Government or government entity in Canada
- HOP: Canadian hospital
- INB: Indian Band recognized by the Indian Act of Canada
- LAM: Canadian library, archive, or museum
- LGR: Legal Representative of a Canadian Citizen or Permanent Resident
- MAJ: Her/His Majesty the Queen/King
- OMK: Official mark registered in Canada
- PLT: Canadian political party
- PRT: Partnership registered in Canada
- RES: Permanent resident of Canada
- TDM: Trade-mark registered in Canada (by a non-Canadian owner type)
- TRD: Canadian trade union
- TRS: Trust established in Canada

Privacy protection

- **Person** – For all contacts, contact name, address, phone number, fax number, and email address are hidden, because CIRA automatically applies its privacy protection to a person. The privacy protection option will be applied at the registrar Whois only.
- **Company, association, or public body** – Not supported at the registry level.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: After the restoration period ends.

Deletion of domain registration

The registry for .ca domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

.cl (Chile)

> **Important**
>
> You can no longer use Route 53 to register new .cl domains or transfer .cl domains to Route 53. We'll continue to support .cl domains that are already registered with Route 53.

Return to index

Renewal period

Two years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Authorization code required for transfers

Not supported. You can no longer transfer .cl domains to Route 53.

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Contact AWS Support.
- Late renewal with Route 53 is possible: Contact AWS Support.
- Domain is deleted from Route 53: After the late-renewal period ends.
- Restoration with the registry is possible: Yes, for a limited time after the domain is deleted.
- Domain is deleted from the registry: After the restoration period ends.

.co (Colombia)

Return to index

Lease period for registration and renewal

One to five years.

Restrictions

Open to the public, with no restrictions.

Privacy protection (applies to: person)

All information is hidden.

If the contact type is not a person, company name and country is displayed by WHOIS.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: Between 30 days and 45 days after expiration
- Domain is deleted from the registry: 50 days after expiration

.com.ar (Argentina)

> **Important**
>
> You can no longer use Route 53 to register new .com.ar domains or transfer .com.ar domains to Route 53. We'll continue to support .com.ar domains that are already registered with Route 53.

Return to index

Renewal period

One year.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. To prevent unauthorized transfers, restrict access to the registrant email address and to the Route 53 APIs that could allow ownership change, for example, UpdateDomainContact. For more information, see Actions, resources, and condition keys for Route 53 Domains in the *Service Authorization Reference* and Example permissions for a domain record owner.

Authorization code required for transfers

Not supported. You can no longer transfer .com.ar domains to Route 53.

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Contact AWS Support.
- Late renewal with Route 53 is possible: Contact AWS Support.
- Domain is deleted from Route 53: After the late-renewal period ends.
- Restoration with the registry is possible: Yes, for a limited time after the domain is deleted.
- Domain is deleted from the registry: After the restoration period ends.

.com.br (Brazil)

> **Important**
>
> You can no longer use Route 53 to register new .com.br domains or transfer .com.br domains to Route 53. We'll continue to support .com.br domains that are already registered with Route 53.

Return to index

Renewal period

One year.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Authorization code required for transfers

Not supported. You can no longer transfer .com.br domains to Route 53.

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Between 30 days before expiration and the expiration date

- Late renewal with Route 53 is possible: Until 119 days after expiration
- Domain is deleted from Route 53: 119 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 119 days after expiration

.com.mx (Mexico)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Determined by the registry.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration

- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.mx (Mexico)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Determined by the registry.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.us (United States)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

The registry for .us domains doesn't allow domain names that contain any of the seven words identified in the "Appendix to Opinion of the Court" of Federal Communications Commission v. Pacifica Foundation No. 77-528.

Open to the public, with one restriction:

- The .us extension is for websites or activities that are located in the United States of America.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: Between 30 days and 60 days after expiration

- Domain is deleted from the registry: 65 days after expiration

.vc (Saint Vincent and the Grenadines)

Also used as a generic TLD, often by those involved in venture capital financing, varsity colleges, and so on.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration

- Domain is deleted from the registry: 80 days after expiration

.vg (British Virgin Islands)

Also used as a generic TLD, often by organizations involved in video gaming.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until 44 days after the expiration date
- Late renewal with Route 53 is possible: Yes
- Domain is deleted from Route 53: 45 days after expiration
- Domain is deleted from the registry: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 74 days after expiration
- Domain is made publicly available again: 80 days after expiration

Asia/Oceania

You can use the following top-level domains (TLDs) for Asia and Oceania to register domains with Amazon Route 53.

, , , , , , , , , , , , , , , , ,

Return to index

.au (Australia)

Return to index

Confirmation email from the TLD registry

Our registrar associate, Gandi, resells .au domains through DomainDirectors. When you transfer a domain name to Route 53, DomainDirectors sends an email to the registrant contact for the domain to verify contact information or to authorize transfer requests.

Lease period for registration and renewal

One to five years.

Restrictions

Open to the public, with some restrictions:

- The .au domains are open to legal persons, trading, partnerships, or sole traders registered in Australia; to foreign companies licensed to trade in Australia; and to owners or applicants of an Australian-registered trademark. Individuals cannot register .au domains. The registrant contact must be a company.
- During the registration process, you must indicate the following:
  - Your registration type: ABN (Australian Business Number), ACN (Australian Company Number), or TM (Trademark) if the domain name corresponds to your trademark.
  - Your ID number, which can be an Australian Business Name (ABN), Australian Company Number (ACN), or TM (trademark).
  - Your registrant name, which is the name of the legal entity registering the domains. This value must exactly match the name on the provided ABN (Australian Business Number), ACN (Australian Company Number), or TM (Trademark).
  - Your eligibility type, which refers to the type of legal entity registering the domain name.
  - The reason why you are allowed to register this domain name (Policy reason):

- The domain name matches the acronym or abbreviation of the registrant's company or trading name, organization or association name, or trademark. (Policy reason 1)
- The domain name meets the allocation rules but is not an acronym or abbreviation of the r egistrant's company or trading name, organization or association name, or trademark. (Policy reason 2). To view the allocation rules for .au, see SCHEDULE A from the .au registry.

- Incorrect or mismatched contact information, including name, ABN, or Trademark (TM) number will result in registration, trade, and renewals failures. An ownership change might be required to correct information for existing domains.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes. In addition to the Route 53 console, you can also obtain the transfer code from the .au registry.

DNSSEC

Supported for domain registration. When you set the key, you must choose DNS security algorithm 2 (DH). For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Between 60 days before expiration and the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 29 days after expiration
- Restoration with the registry is possible: No

- Domain is deleted from the registry: 30 days after expiration

Deletion of domain registration

The registry for .au domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

Changing ownership

Change the owner by using the Route 53 console. See Updating contact information for a domain. Then complete the following process to complete the ownership change:

1. Both the old and new registrants must choose the link they receive in an email from *transfers@1api.net* to their listed email addresses. If this isn't completed within 14 days, you have to start the process again.

2. After the responses are confirmed, the owner change in the registry will be processed in a short time without any further confirmation.

.cc (Cocos (Keeling) Islands)

Return to index

Also used as a generic TLD, often by organizations with "cc" in their names, such as consulting companies, cloud computing companies, or cycling clubs. The extension is a popular alternative to ".com."

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection (applies to all contact types: person, company, association, and public body)

- **Hidden** – address, phone number, fax number, and email address
- **Not hidden** – contact name and organization name

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 30 days and 60 days after expiration
- Domain is deleted from the registry: 65 days after expiration

.co.nz (New Zealand)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- Individuals must be at least 18.
- Organizations must be registered.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you

also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 44 days after expiration
- Restoration with the registry is possible: Between 44 days and 134 days after expiration
- Domain is deleted from the registry: 134 days after expiration

.com.au (Australia)

Return to index

Confirmation email from the TLD registry

Our registrar associate, Gandi, resells .com.au domains through DomainDirectors. When you transfer a domain name to Route 53, DomainDirectors sends an email to the registrant contact for the domain to verify contact information or to authorize transfer requests.

Lease period for registration and renewal

One to five years.

Restrictions

Open to the public, with some restrictions:

- The .com.au and .net.au domains are open to partnerships or sole traders registered in Australia; to foreign companies licensed to trade in Australia; and to owners or applicants of an Australian-registered trademark. Individuals cannot register .com.au/.net.au domains. The registrant contact must be a company.
- Your domain name must be identical to your name (as registered with the relevant Australian authorities) or to your trademark (or to the abbreviation or acronym for your trademark).
- The domain name should indicate your activity. For example, it should indicate a product that you sell or a service that you provide.
- During the registration process, you must provide the following information:
  - Your registration type: ABN (Australian Business Number), ACN (Australian Company Number), or TM (Trademark) if the domain name corresponds to your trademark.
  - Your ID number, which can be an Australian Business Number (ABN), an Australian Company Number (ACN), or your trademark number (TM) if the domain name corresponds to your trademark.
  - Your registrant name, which is the name of the legal entity registering the domain. This value must exactly match the name on the provided.
  - Your eligibility type, which refers to the type of the legal entity registering this domain name.
  - The reason why you are allowed to register this domain name (Policy reason):
    - The domain name matches the acronym or abbreviation of the registrant's company or trading name, organization or association name, or trademark. (Policy reason 1)
    - The domain name meets the allocation rules but is not an acronym or abbreviation of the registrant's company or trading name, organization or association name, or trademark. (Policy reason 2). To view the allocation rules for .com.au, see SCHEDULE C from the .com.au registry.
- Incorrect or mismatched contact information, including name, ABN, or Trademark (TM) number will result in registration, trade, and renewals failures. An ownership change might be required to correct information for existing domains.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you

also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. When you set the key, you must choose DNS security algorithm 2 (DH). For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Between 60 days before expiration and the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 29 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 30 days after expiration

Deletion of domain registration

The registry for .com.au domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

Changing ownership

Change the owner, either programmatically or by using the Route 53 console. See Updating contact information for a domain. Then complete the following process to complete the ownership change:

1. Both the old and new registrants must choose the link they receive in an email from *transfers@1api.net* to their listed email addresses. If this isn't completed within 14 days, you have to start the process again.

2. After the responses are confirmed, the owner change in the registry will be processed in a short time without any further confirmation.

.com.sg (Republic of Singapore)

> **Important**
>
> You can no longer use Route 53 to register new .com.sg domains or transfer .com.sg domains to Route 53. We'll continue to support .com.sg domains that are already registered with Route 53.

Return to index

Renewal period

One or two years.

Deletion of domain registration

The registry for .com.sg domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Authorization code required for transfers

Not supported. You can no longer transfer .com.sg domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: Between 30 days and 60 days after expiration
- Domain is deleted from the registry: 60 days after expiration

.fm (Federated States of Micronesia)

Also used as a generic TLD, often by organizations involved in online media and broadcasting.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection (applies to all contact types: person, company, association, and public body)

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 44 days after expiration
- Restoration with the registry is possible: Between 44 days and 79 days after expiration
- Domain is deleted from the registry: 84 days after expiration

.in (India)

Return to index

Important

If your domain has a serverHold status, you need to reach out to the .IN registry (NIXI - National Internet Exchange of India) directly to resolve it. NIXI might require verification of your contact information to lift the status, in which case you will need to provide a photo ID and address proof.
Contact NIXI at support@nixi.in.

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: Between 30 days and 60 days after expiration
- Domain is deleted from the registry: 65 days after expiration

.jp (Japan)

Return to index

Lease period for registration and renewal

One year.

Restrictions

Open to the public, with one restriction:

- Only individuals or companies in Japan can register a .jp domain name.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported for Japanese.

Authorization code required for transfers

Yes.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until 16 days before the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: 6 days before expiration
- Restoration with the registry is possible: Yes, for a limited time after the domain is deleted.
- Domain is deleted from the registry: After the restoration period ends.

> **Note**
>
> Registering non-general-purpose JP domains such as .co.jp and .or.jp is currently not possible.

.io (British Indian Ocean Territory)

Also used as a generic TLD, often by computer-related organizations such as online services, browser-based games, and startup companies.

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

All information is hidden except state/province and country.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

The registry for .io domains also uses the authorization code as a single-use password for some operations, such as enabling or disabling privacy protection. If you want to perform more than one operation that requires a password, you must generate another authorization code for each operation.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 90 days after expiration

.my (Malaysia)

> **Important**
>
> Since August 2024, .my domains cannot be registered by residents of Israel. This restriction applies to both new registrations and modifications, including changes of address to an Israeli location.

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.net.au (Australia)

Return to index

Confirmation email from the TLD registry

Our registrar associate, Gandi, resells .net.au domains through DomainDirectors. When you transfer a domain name to Route 53, DomainDirectors sends an email to the registrant contact for the domain to verify contact information or to authorize transfer requests.

Lease period for registration and renewal

One to five years.

Restrictions

Open to the public, with some restrictions:

- The .com.au and .net.au domains are open to legal persons, trading, partnerships, or sole traders registered in Australia; to foreign companies licensed to trade in Australia; and to owners or applicants of an Australian-registered trademark.
- Your domain name must be identical to your name, as registered with the relevant Australian authorities or to your trademark (or to the abbreviation or acronym).
- The domain name should indicate your activity. For example, it should indicate a product that you sell or a service that you provide.
- During the registration process, you must indicate the following:
  - Your registration type: ABN (Australian Business Number), ACN (Australian Company Number), or TM (Trademark) if the domain name corresponds to your trademark.
  - Your ID number, which can be an Australian Business Number (ABN), an Australian Company Number (ACN), or your trademark number (TM) if the domain name corresponds to your trademark.

- Your registrant name, which is the name of the legal entity registering the domain. This value must exactly match the name on the provided.
- Your eligibility type, which refers to the type of the legal entity registering this domain name.
- The reason why you are allowed to register this domain name (Policy reason):
  - The domain name matches the acronym or abbreviation of the registrant's company or trading name, organization or association name, or trademark. (Policy reason 1)
  - The domain name meets the allocation rules but is not an acronym or abbreviation of the registrant's company or trading name, organization or association name, or trademark. (Policy reason 2). To view the allocation rules for .net.au, see SCHEDULE E from the net.au registry.
- Incorrect or mismatched contact information, including name, ABN, or Trademark (TM) number will result in registration, trade, and renewals failures. An ownership change might be required to correct information for existing domains.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. When you set the key, you must choose DNS security algorithm 2 (DH). For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Between 60 days before expiration and the expiration date

- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 29 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 30 days after expiration

Deletion of domain registration

The registry for .net.au domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

Changing ownership

Change the owner, either programmatically or by using the Route 53 console. See Updating contact information for a domain. Then complete the following process to complete the ownership change:

1. Both the old and new registrants must choose the link they receive in an email from *transfers@1api.net* to their listed email addresses. If this isn't completed within 14 days, you have to start the process again.

2. After the responses are confirmed, the owner change in the registry will be processed in a short time without any further confirmation.

.net.nz (New Zealand)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- Individuals must be at least 18.
- Organizations must be registered.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 44 days after expiration
- Restoration with the registry is possible: Between 44 days and 134 days after expiration
- Domain is deleted from the registry: 134 days after expiration

.nz (New Zealand)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- Individuals must be at least 18.
- Organizations must be registered.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 44 days after expiration
- Restoration with the registry is possible: Between 44 days and 134 days after expiration
- Domain is deleted from the registry: 134 days after expiration

.org.nz (New Zealand)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- Individuals must be at least 18.
- Organizations must be registered.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 44 days after expiration
- Restoration with the registry is possible: Between 44 days and 134 days after expiration
- Domain is deleted from the registry: 134 days after expiration

.pw (Palau)

Return to index

The .pw was originally reserved for the residents of Palau, an island country in the Micronesia sub region of Oceania in the western Pacific, however, now it is commonly used to represent 'Professional Web' and is available to everyone.

Lease period for registration and renewal

One to ten years.

Privacy protection (applies to all contact types: person, company, association, and public body)

All information is hidden except organization name.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 75 days after expiration

.qa (Qatar)

> **Important**
>
> You can no longer use Route 53 to register new .qa domains or transfer .qa domains to Route 53. We'll continue to support .qa domains that are already registered with Route 53.

Return to index

Renewal period

One to five years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Authorization code required for transfers

Not supported. You can no longer transfer .qa domains to Route 53.

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 31 days after expiration

.ru (Russian Federation)

Important

You can no longer use Route 53 to register new .ru domains, transfer .ru domains to Route 53, or renew existing .ru domains with Route 53. If you have existing .ru domains with Route 53, transfer them to a registrar that supports .ru immediately to avoid any service disruption.

Return to index

Lease period for registration and renewal

One year.

> **Note**
>
> The registry for .ru domains updates the expiration date for a domain on the day that the domain expires. WHOIS queries will show the old expiration date for the domain until that date regardless of when you renew the domain with Route 53.

Restrictions

Open to the public, with some restrictions:

- Individuals might need to provide a passport number or government-issued ID number.
- Foreign companies might need to provide a company ID or company registration.

Privacy protection

Determined by the registry.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Not supported.

Authorization code required for transfers

Not supported. You can no longer transfer .ru domains to Route 53.

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

Not supported. You can no longer renew or restore .ru domains with Route 53.

Deletion of domain registration

The registry for .ru domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

.sg (Republic of Singapore)

> **Important**
>
> You can no longer use Route 53 to register new .sg domains or transfer .sg domains to Route 53. We'll continue to support .sg domains that are already registered with Route 53.

Return to index

Renewal period

One or two years.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Authorization code required for transfers

Not supported. You can no longer transfer .sg domains to Route 53.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: Between 30 days and 60 days after expiration

- Domain is deleted from the registry: 60 days after expiration

Deletion of domain registration

The registry for .sg domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

Europe

You can use the following top-level domains (TLDs) for Europe to register domains with Amazon Route 53.

, ,

Return to index

.be (Belgium)

Return to index

Lease period for registration and renewal

One year.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes. You can obtain the transfer code from DNS Belgium website.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: On the expiration date
- Restoration with the registry is possible: Until 40 days after expiration
- Domain is deleted from the registry: 40 days after expiration

.berlin (city of Berlin in Germany)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- The owner, administrative, or technical contact must provide an address in Berlin, and the administrative contact must be an individual.
- You must activate and use your .berlin domain within 12 months following its registration (applies to a website, redirection, or email address).
- If you publish a website under your .berlin domain, or if your .berlin domain redirects to another website, the content of the website must be related to Berlin.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Latin and Cyrillic.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 80 days after expiration

.ch (Switzerland)

Return to index

Lease period for registration and renewal

One year.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 9 days after expiration
- Domain is deleted from Route 53: 9 days after expiration
- Restoration with the registry is possible: Between 9 days and 49 days after expiration
- Domain is deleted from the registry: 49 days after expiration

.co.uk (United Kingdom)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported

Internationalized domain names

Not supported.

Authorization code required for transfers

If you're transferring a .co.uk domain to Route 53, you don't need to get an authorization code. Instead, use the method provided by your current domain registrar to update the value of

the IPS tag for the domain to **GANDI**, all uppercase. (An IPS tag is required by Nominet, the registry for .uk domain names.) If your registrar will not change the value of the IPS tag, contact Nominet.

> **Note**
>
> When you register a .co.uk domain, Route 53 automatically sets the IPS tag for the domain to **GANDI**.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Between 180 days before and 30 days after the expiration date
- Late renewal with Route 53 is possible: Between 30 days and 90 days after expiration
- Domain is deleted from Route 53: 90 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 92 days after expiration

Deletion of domain registration

The registry for .co.uk domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

.cz (Czech Republic)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported, but email address and phone number are hidden for all contacts.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

If your current registrar doesn't provide an authorization code, go to `https://www.nic.cz/whois/send-password/` to request it to be sent to the registrant email address by the CZ domain registry.

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 58 days after expiration
- Domain is deleted from Route 53: 59 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 60 days after expiration

.de (Germany)

Return to index

Lease period for registration and renewal

One year.

Restrictions

Open to the public, with some restrictions:

- You must reside in Germany or have an administrative contact (physical person) who resides in Germany and has an address other than a P.O. box.
- During registration, the DNS (A, MX, and CNAME) of the domain name must be correctly configured so that it can pass the registry's zone check. Three servers of two different C classes are required.
- If you're using a DNS service other than Route 53, the name servers for the domain must pass a check to make sure that they're correctly configured. To determine whether the name servers for your domain will pass the check, see https://www.denic.de/en/service/tools/nast/.
- You must provide contact information that meets registry validation requirements. Invalid or incorrectly formatted contact details can cause domain renewals to fail.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: On the expiration date

- Restoration with the registry is possible: Only if you deleted the domain. Expired .de domains can't be restored. See the following note.
- Domain is deleted from the registry: After the restoration period ends.

> **Note**
>
> How you recover a .de domain depends on its status with DENIC, the registry for .de.
>
> - If you deleted the domain and it's within DENIC's redemption grace period, you can recover the domain by restoring it.
> - If the domain expires, it leaves Route 53 and enters DENIC's TRANSIT status. DENIC mails the domain holder a letter with an authorization code. If you don't receive it, you can request a new code from DENIC. Within the period stated in the letter, you can recover the domain by transferring it back to Route 53 using that code.

.es (Spain)

Return to index

Domain purchase or transfer

> **Important**
>
> You currently can purchase new .es domain names or transfer .es domains to Route 53 . The contact type for the registrant contact has no restriction. Admin / Tech / Billing contact type must be a **person**.

Lease period for registration and renewal

One to five years.

Restrictions

Open to the public, for those who have an interest in or connection with Spain.

As of 2016, the .ES domain registrants must provide a registrant contact email. If you haven't provided this information, you need to do so at the current registrar before you transfer your domain to Route 53.

You will need the following information:

- ESNIC identifier similar to *AAAA0-ESNIC-F0*.
- If you don't know your ESNIC identifier, you can get it from the current registrar. You can find your registrar at: https://www.dominios.es/en.

Depending on whether you remember your password at the registrar, or not, you can follow one of the following procedures to update your registrant email:

- If you remember you password, sign in at https://www.nic.es/sgnd/login.action by using you ESNIC identifies and password.

  After you have signed in, you can edit the registrant email contact by choosing the **Edit** tab on the registry page.
- If you forgot your password, browse to https://www.nic.es/sgnd/peticion/editCorreo.action?request_locale=en.

  Fill out the form with your ESNIC identifier, your new and valid registrant email contact. Then, validate the form by choosing **Processing without eID/Certificate**, and upload the requested identity document.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode operation. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

The .es registry manages the authorization code with a time-to-live, so it might expire. If your authorization code doesn't work, contact AWS Support.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until 6 days before the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: 6 days before expiration
- Restoration with the registry is possible: Between 6 days before and 4 days after expiration
- Domain is deleted from the registry: 4 days after expiration

.eu (European Union)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with one restriction:

- You must provide a valid postal address from one of the 30 states of the European Economic Area (EEA) or if you're a citizen of one of the 27 member-states of the European Union (EU), you must specify your EU country of citizenship.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

You can also generate the auth code by using the "My.eu" panel at the registry: https://my.eurid.eu/.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: On the expiration date
- Restoration with the registry is possible: Until 40 days after expiration
- Domain is deleted from the registry: 40 days after expiration

WHOIS Search

For information about existing .eu domains, see https://whois.eurid.eu/en/.

.fi (Finland)

Return to index

Lease period for registration and renewal

One to five years.

Restrictions

Open to the public, with some restrictions:

- The .fi extension is available to individuals who have a domicile in Finland and have a Finnish identity number, and legal persons or private entrepreneurs registered in Finland.
- If the registrant contact address is in Finland, then Finnish identity number is required for an individual registrant and Finnish company number is required for a company registrant, and you must provide the following information during registration:
  - Whether or not the contact is based on a physical or moral person in Finland.

- The identifier of the register where the name is recorded, if based on a moral person's name.
- The number of the record in the register where the name is recorded, if based on a moral person's name.
- The identification number for a moral person in Finland.
- The identification number for a physical person in Finland.
- If the registrant is Non-Finnish Company, then you must provide Business Number as VAT_Number.

- If the registrant address isn't located in Finland, then no Finnish identity or company number is required.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: No

- Domain is deleted from the registry: No

Deletion of domain registration

For information about deleting domains, see Deleting a domain name registration.

.fr (France)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- Individuals must be at least 18 and must provide their date-of-birth.
- Organizations must be located in the European Economic Area or in Switzerland.
- Organizations should fill out all company identification fields (VAT number, SIREN, WALDEC, DUNS, and so on), as this will facilitate any verification that AFNIC might perform at a later date.
- The same eligibility conditions apply to the administrative contact.
- Names and terms are subject to an AFNIC prior review (Naming Charter Article 2.4) and to the following additional conditions:
  - Domain names previously reserved or prohibited are open to applicants that justify a legitimate right and act in good faith.
  - Names beginning with ville, mairie, agglo, cc, cg, and cr are subject to AFNIC naming conventions.

Privacy protection

Determined by the registry.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 27 days after expiration
- Domain is deleted from Route 53: 28 days after expiration
- Restoration with the registry is possible: Between 28 days and 58 days after expiration
- Domain is deleted from the registry: 58 days after expiration

.gg (Guernsey)

Return to index

Lease period for registration and renewal

One year.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: Between 30 days and 35 days after expiration
- Domain is deleted from the registry: 35 days after expiration

.im (Isle of Man)

Also used as a generic TLD, often by instant messaging services or for individuals who want to develop an "I am" personal brand.

Return to index

Lease period for registration and renewal

One or two years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: On the expiration date
- Restoration with the registry is possible: Until 30 days after expiration
- Domain is deleted from the registry: 30 days after expiration

.it (Italy)

Return to index

Lease period for registration and renewal

One year.

Restrictions

Open to the public, with some restrictions:

- Individuals or organizations must have a registered address in the European Union.
- If your country of origin is Italy, you must enter a fiscal code. If your country of origin is within the European Union, you must enter an identity document number (ID number).
- Name servers for your domain must pass a DNS check. We suggest that you check the name servers at https://dns-check.nic.it/ before you submit the change request. If your domain name does not comply with the technical requirements (for example, it isn't associated with an operational name server), and you do not correct it within 30 days, your domain name will be deleted by the registry. We don't issue refunds for domains that are deleted because they don't meet technical requirements.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs,

and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported.

Authorization code required for transfers

Yes

DNSSEC

Not supported.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 13 days after expiration
- Domain is deleted from the registry: 49 days after expiration
- Restoration with the registry is possible: Between 14 days and 44 days after expiration
- Domain is deleted from the registry: After the restoration period ends.

.me (Montenegro)

Return to index

Lease period for registration and renewal

One to ten years.

Privacy protection

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Arabic, Belarusian, Bosnian, Bulgarian, Simplified Chinese, Traditional Chinese, Croatian, Danish, French, German, Hindi, Hungarian, Icelandic, Italian, Korean, Latvian, Lithuanian, Mongolian, Montenegrin, Polish, Portuguese, Russian, Serbian, Spanish, Swedish, Turkish, and Ukrainian.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 29 days after expiration
- Domain is deleted from Route 53: 30 days after expiration
- Restoration with the registry is possible: Between 30 days and 60 days after expiration
- Domain is deleted from the registry: 65 days after expiration

.me.uk (United Kingdom)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported

Internationalized domain names

Not supported.

Authorization code required for transfers

If you're transferring a .me.uk domain to Route 53, you don't need to get an authorization code. Instead, use the method provided by your current domain registrar to update the value of the IPS tag for the domain to **GANDI**, all uppercase. (An IPS tag is required by Nominet, the

registry for .uk domain names.) If your registrar will not change the value of the IPS tag, contact Nominet.

> **Note**
>
> When you register a .me.uk domain, Route 53 automatically sets the IPS tag for the domain to **GANDI**.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Between 180 days before and 30 days after the expiration date
- Late renewal with Route 53 is possible: Between 30 days and 90 days after expiration
- Domain is deleted from Route 53: 90 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 92 days after expiration

Deletion of domain registration

The registry for .me.uk domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

.nl (the Netherlands)

Return to index

Lease period for registration and renewal

One year.

Restrictions

Open to the public, with some restrictions:

- The owner or the administrative contact must provide a valid address in the Netherlands. A local presence is required.

- If you do not have a valid address in the Netherlands, the Registry SIDN will provide you with a domicile address, as per the Domicile Address Procedure.
- The domain name must be 3-63 characters, excluding .nl.

Privacy protection

Determined by the registry.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Not supported.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until 1 day before the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: 1 day before expiration
- Restoration with the registry is possible: Between 1 day before and 39 days after expiration
- Domain is deleted from the registry: 39 days after expiration

.org.uk (United Kingdom)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported

Internationalized domain names

Not supported.

Authorization code required for transfers

If you're transferring a .org.uk domain to Route 53, you don't need to get an authorization code. Instead, use the method provided by your current domain registrar to update the value of the IPS tag for the domain to **GANDI**, all uppercase. (An IPS tag is required by Nominet, the registry for .uk domain names.) If your registrar will not change the value of the IPS tag, contact Nominet.

> **Note**
>
> When you register a .org.uk domain, Route 53 automatically sets the IPS tag for the domain to **GANDI**.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Between 180 days before and 30 days after the expiration date
- Late renewal with Route 53 is possible: Between 30 days and 90 days after expiration
- Domain is deleted from Route 53: 90 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 92 days after expiration

Deletion of domain registration

The registry for .org.uk domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

.ruhr (Ruhr region, western part of Germany)

Return to index

The .ruhr extension is for the Ruhr region (western part of Germany).

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with one restriction:

- The administrative contact must be an individual who has an address in Germany.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported (ä, ö, ü, ß).

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration

- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: After the restoration period ends.

.se (Sweden)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- If you are located in Sweden, you must provide a valid Swedish ID number. The ID number format is `YYMMDD-NNNN`.
- If you are located outside of Sweden, you must enter a valid ID number such as a tax ID number.

Additional verification requirements

As of October 1, 2025, .se domain registrations require additional identity verification to comply with the registry's new NIS 2.0 requirements. When you register a .se domain, we'll contact you to request verification documents. Your domain registration will remain pending until you provide the required documents and verification is complete. To submit your documents, contact AWS Support.

The .se registry also conducts periodic audits of existing domains and might request additional verification for recently registered domains or when you make certain changes to your domain (such as updating the domain owner or contact information). If your domain is placed in `serverHold` status, this means the registry requires verification before your domain can be fully activated.

Documents you'll need to provide:

- **For individuals:** A government-issued ID document (such as an identity card or passport) and proof of address less than 3 months old.
- **For companies or organizations:** Registration certificate of the company or organization, and proof of address if different from the registration certificate.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Not supported. We recommend that you prevent unauthorized transfers by restricting access to the RetrieveDomainAuthCode API action. (When you restrict access to this Route 53 API, you also restrict who can generate an authorization code using the Route 53 console, AWS SDKs, and other programmatic methods.) For more information, see Identity and access management in Amazon Route 53.

Internationalized domain names

Supported for Latin, Swedish, and Yiddish.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until 1 day before the expiration date
- Late renewal with Route 53 is possible: No
- Domain is deleted from Route 53: 1 day before expiration
- Restoration with the registry is possible: Between 1 day before and 59 days after expiration
- Domain is deleted from the registry: 64 days after expiration

.uk (United Kingdom)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with no restrictions.

Privacy protection

All information is hidden.

Domain locking to prevent unauthorized transfers

Supported

Internationalized domain names

Not supported.

Authorization code required for transfers

If you're transferring a uk domain to Route 53, you don't need to get an authorization code. Instead, use the method provided by your current domain registrar to update the value of the IPS tag for the domain to **GANDI**, all uppercase. (An IPS tag is required by Nominet, the registry for .uk domain names.) If your registrar will not change the value of the IPS tag, contact Nominet.

> **Note**
>
> When you register a .uk domain, Route 53 automatically sets the IPS tag for the domain to **GANDI**.

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Between 180 days before and 30 days after the expiration date
- Late renewal with Route 53 is possible: Between 30 days and 90 days after expiration
- Domain is deleted from Route 53: 90 days after expiration
- Restoration with the registry is possible: No
- Domain is deleted from the registry: 92 days after expiration

Deletion of domain registration

The registry for .uk domains doesn't allow you to delete domain registrations. Instead, you must disable automatic renewal and wait for the domain to expire. For more information, see Deleting a domain name registration.

.wien (city of Vienna in Austria)

Return to index

Lease period for registration and renewal

One to ten years.

Restrictions

Open to the public, with some restrictions:

- You must show an economic, cultural, tourist, historical, social, or other affinity with the city of Vienna in Austria.
- The .wien domain names must be used in connection with the above conditions, throughout the term of registration.

Privacy protection

Not supported.

Domain locking to prevent unauthorized transfers

Supported.

Internationalized domain names

Supported for Latin.

Authorization code required for transfers

Yes

DNSSEC

Supported for domain registration. For more information, see Configuring DNSSEC for a domain.

Deadlines for renewing and restoring domains

- Renewal is possible: Until the expiration date
- Late renewal with Route 53 is possible: Until 44 days after expiration
- Domain is deleted from Route 53: 45 days after expiration
- Restoration with the registry is possible: Between 45 days and 75 days after expiration
- Domain is deleted from the registry: 80 days after expiration

Configuring Amazon Route 53 as your DNS service

You can use Amazon Route 53 as the DNS service for your domain, such as example.com. Route 53 routes internet traffic to your website by translating domain names like www.example.com into IP addresses like 192.0.2.1. Computers use these IP addresses to connect to each other. When someone types your domain name in a browser or sends you an email, a DNS query goes to Route 53. It responds with the right value, such as the IP address for your web server.

> **DNS hosting vs. domain registration**
>
> This chapter covers using Route 53 for *DNS hosting only*. Your domain stays registered with your current registrar. You keep paying them for renewals. Route 53 only handles your DNS settings and DNS queries.
>
> If you want to transfer your domain to Route 53 as well (making Route 53 both your registrar and DNS service), see Pre-transfer checklist for domain transfers and Transferring registration for a domain to Amazon Route 53.

This chapter explains how to set up Route 53 to route internet traffic to the right places. It also covers how to move DNS service to Route 53 from another provider, and how to use Route 53 as the DNS service for a new domain.

Topics

- Making Amazon Route 53 the DNS service for an existing domain
- Configuring DNS routing for a new domain
- Routing traffic to your resources
- Working with hosted zones
- Working with records
- Configuring DNSSEC signing in Amazon Route 53
- Using AWS Cloud Map to create records and health checks
- DNS constraints and behaviors
- Related topics

Making Amazon Route 53 the DNS service for an existing domain

If you're transferring one or more domain registrations to Route 53, and your current registrar doesn't provide paid DNS service, you need to migrate DNS service before you transfer the domain. Otherwise, the registrar will stop providing DNS service when you transfer your domains. The associated websites and web applications will become unavailable on the internet. (You can also migrate DNS service to another DNS service provider. You don't have to use Route 53 as the DNS service provider for domains registered with Route 53.)

The process depends on whether you're currently using the domain:

- If the domain is currently getting traffic—for example, if your users are using the domain name to browse to a website or access a web application—see Making Route 53 the DNS service for a domain that's in use.
- If the domain isn't getting any traffic (or is getting very little traffic), see Making Route 53 the DNS service for an inactive domain.

For both options, your domain should remain available during the entire migration process. However, in the unlikely event that there are issues, the first option lets you roll back quickly. With the second option, your domain could be unavailable for a couple of days.

If you want to connect with an expert at AWS, visit Sales support.

Making Route 53 the DNS service for a domain that's in use

You can migrate DNS service to Amazon Route 53 for a domain that is already receiving traffic. For example, users might already browse to a website or access a web application on the domain. Follow the procedures in this topic to migrate without disrupting service.

Topics

- Step 1: Get your current DNS configuration from the current DNS service provider (optional but recommended)
- Step 2: Create a hosted zone
- Step 3: Create records
- Step 4: Lower TTL settings

- Step 5: (If you have DNSSEC configured) Remove the DS record from the parent zone
- Step 6: Wait for the old TTL to expire
- Step 7: Update the NS records to use Route 53 name servers
- Step 8: Monitor traffic for the domain
- Step 9: Change the TTL for the NS record back to a higher value
- Step 10: Transfer domain registration to Amazon Route 53
- Step 11: Re-enable DNSSEC signing (if required)

Step 1: Get your current DNS configuration from the current DNS service provider (optional but recommended)

When you migrate DNS service from another provider to Route 53, you reproduce your current DNS configuration in Route 53. In Route 53, you create a hosted zone that has the same name as your domain, and you create records in the hosted zone. Each record indicates how you want to route traffic for a domain name or subdomain name. For example, when someone enters your domain name in a web browser, do you want traffic to go to a web server in your data center, to an Amazon EC2 instance, to a CloudFront distribution, or to some other location?

The process depends on the complexity of your current DNS configuration:

- **If your current DNS configuration is simple** – If you're routing internet traffic for just a few subdomains to a small number of resources, such as web servers or Amazon S3 buckets, then you can manually create a few records in the Route 53 console.
- **If your current DNS configuration is more complex, and you just want to reproduce your current configuration** – You can simplify the migration if you can get a zone file from the current DNS service provider and import it into Route 53. (Not all DNS service providers offer zone files.) When you import a zone file, Route 53 reproduces the existing configuration by creating the corresponding records in your hosted zone.

  Try asking customer support with your current DNS service provider how to get a *zone file* or a *records list*. For information about the required format of the zone file, see Creating records by importing a zone file.
- **If your current DNS configuration is more complex, and you're interested in Route 53 routing features** – Review the following documentation to see whether you want to use Route 53 features not available from other DNS service providers. If so, you can create records manually, or you can import a zone file and then create or update records later:

- Choosing between alias and non-alias records explains the advantages of Route 53 alias records, which route traffic to some AWS resources, such as CloudFront distributions and Amazon S3 buckets, for no charge.
- Choosing a routing policy explains the Route 53 routing options, for example, routing based on user location, latency between users and resources, resource health, and weights that you specify.

> **Note**
>
> You can also import a zone file and later change your configuration to take advantage of alias records and complex routing policies.

If you can't get a zone file or if you want to manually create records in Route 53, the records that you're likely to migrate include the following:

- **A (Address) records** – associate a domain name or subdomain name with the IPv4 address (for example, 192.0.2.3) of the corresponding resource
- **AAAA (Address) records** – associate a domain name or subdomain name with the IPv6 address (for example, 2001:0db8:85a3:0000:0000:abcd:0001:2345) of the corresponding resource
- **Mail server (MX) records** – route traffic to mail servers
- **CNAME records** – reroute traffic for one domain name (example.net) to another domain name (example.com)
- **Records for other supported DNS record types** – For a list of supported record types, see Supported DNS record types.

Step 2: Create a hosted zone

To tell Amazon Route 53 how you want to route traffic for your domain, you create a hosted zone that has the same name as your domain, and then you create records in the hosted zone.

> **Important**
>
> You can create a hosted zone only for a domain that you have permission to administer. Typically, this means that you own the domain, but you might also be developing an application for the domain registrant.

When you create a hosted zone, Route 53 automatically creates a name server (NS) record and a start of authority (SOA) record for the zone. The NS record identifies the four name servers that Route 53 associated with your hosted zone. To make Route 53 the DNS service for your domain, update the registration for the domain to use these four name servers.

> **Important**
>
> Don't create additional name server (NS) or start of authority (SOA) records, and don't delete the existing NS and SOA records.

To create a hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. If you're new to Route 53, choose **Get started** under **DNS management**, and then choose **Create hosted zones**.

   If you're already using Route 53, choose **Hosted zones** in the navigation pane, and then choose **Create hosted zones**.
3. In the **Create hosted zone** pane, enter a domain name and, optionally, a comment. For more information about a setting, choose to open the help panel on the right side.

   For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.
4. For **Type**, accept the default value of **Public hosted zone**.
5. Choose **Create hosted zone**.

Step 3: Create records

After you create a hosted zone, you create records in the hosted zone that define where you want to route traffic for a domain (example.com) or subdomain (www.example.com). For example, if you want to route traffic for example.com and www.example.com to a web server on an Amazon EC2 instance, you create two records, one named example.com and the other named www.example.com. In each record, you specify the IP address for your EC2 instance.

You can create records in a variety of ways:

Import a zone file

This is the easiest method if you got a zone file from your current DNS service in Step 1: Get your current DNS configuration from the current DNS service provider (optional but recommended). Amazon Route 53 can't predict when to create alias records or to use special routing types such as weighted or failover. As a result, if you import a zone file, Route 53 creates standard DNS records using the simple routing policy.

For more information, see Creating records by importing a zone file.

Create records individually in the console

If you didn't get a zone file and you just want to create a few records with a routing policy of Simple to get started, you can create the records in the Route 53 console. You can create both alias and non-alias records.

For more information, see the following topics:

- Choosing a routing policy
- Choosing between alias and non-alias records
- Creating records by using the Amazon Route 53 console

Create records programmatically

You can create records by using one of the AWS SDKs, the AWS CLI, or AWS Tools for Windows PowerShell. For more information, see AWS Documentation.

If you're using a programming language that AWS doesn't provide an SDK for, you can also use the Route 53 API. For more information, see the Amazon Route 53 API Reference.

Step 4: Lower TTL settings

The TTL (time to live) setting for a record specifies how long you want DNS resolvers to cache the record and use the cached information. When the TTL expires, a resolver sends another query to the DNS service provider for the latest information.

The typical TTL setting for the NS record is 172800 seconds, or two days. The NS record lists the name servers that DNS can use to get information about routing traffic for your domain. Lowering the TTL for the NS record, both with your current DNS service provider and with Amazon Route 53, reduces downtime for your domain if you discover a problem while migrating DNS to Route 53. If you don't lower the TTL, your domain could be unavailable on the internet for up to two days if something goes wrong.

> **Note**
>
> Some full resolvers might cache the TTL of the NS record of the parent authoritative server, therefore the TTL of NS records registered on the parent authoritative DNS server must also be reduced.

We recommend that you change the TTL on the following NS records:

- On the NS record in the hosted zone for the current DNS service provider. (Your current provider might use different terminology.)
- On the NS record in the hosted zone that you created in Step 2: Create a hosted zone.

To lower the TTL setting on the NS record with the current DNS service provider

- Use the method provided by the current DNS service provider for the domain to change the TTL for the NS record in the hosted zone for your domain.

To lower the TTL setting on the NS record in a Route 53 hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. Choose **Hosted Zones** in the navigation pane.
3. Choose the name of the hosted zone.
4. Choose the NS record, and choose **Edit**.
5. Change the value of **TTL (Seconds)**. We recommend that you specify a value between 60 seconds and 900 seconds (15 minutes).
6. Choose **Save changes**.

Step 5: (If you have DNSSEC configured) Remove the DS record from the parent zone

If you've configured DNSSEC for your domain, remove the Delegation Signer (DS) record from the parent zone before you migrate your domain to Route 53.

If the parent zone is hosted through Route 53 or another registrar, contact them to remove the DS record.

You can't have DNSSEC signing enabled across two providers at the same time. You must remove any DS or DNSKEYs to turn off DNSSEC. This temporarily signals DNS resolvers to skip DNSSEC validation. In step 11, you can re-enable DNSSEC validation after the transition to Route 53 is done.

For more information, see Deleting public keys for a domain.

Step 6: Wait for the old TTL to expire

If your domain is in use—for example, if users are browsing to a website or accessing a web application—then DNS resolvers have cached the names of the name servers provided by your current DNS service provider. A DNS resolver that cached that information a few minutes ago will save it for almost two more days.

To make sure that migrating DNS service to Route 53 happens all at one time, wait for two days after you lowered the TTL. After the two-day TTL expires and resolvers request the name servers for your domain, the resolvers will get the current name servers and the new TTL that you specified in Step 4: Lower TTL settings.

Step 7: Update the NS records to use Route 53 name servers

To begin using Amazon Route 53 as the DNS service for a domain, use the method provided by the registrar, or the parent zone, to replace the current name servers in the NS record with Route 53 name servers.

> **Note**
>
> When you update the NS record with the current DNS service provider to use Route 53 name servers, you're updating the DNS configuration for the domain. (This is similar to updating the NS record in the Route 53 hosted zone for a domain, except that you're updating the setting with the DNS service that you're migrating away from.)

To update the NS record at the registrar, or the parent zone, to use Route 53 name servers

1. In the Route 53 console, get the name servers for your hosted zone:

   a. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

b. In the navigation pane, choose **Hosted zones**.

c. On the **Hosted zones** page, choose the name for the applicable hosted zone.

d. Make note of the four names listed for **Name servers** in the **Hosted zone details** section.

2. Use the method that is provided by the current DNS service for the domain to update the NS record for the hosted zone. If the domain is registered with Route 53, see Adding or changing name servers and glue records for a domain.The process depends on whether the current DNS service lets you delete name servers:

   If you can delete name servers

   - Make note of the names of the current name servers in the NS record for the hosted zone. If you need to revert to the current DNS configuration, these are the servers that you'll specify.
   - Delete the current name servers from the NS record.
   - Update the NS record with the names of all four of the Route 53 name servers that you got in step 1 of this procedure.

   > **Note**
   >
   > When you're finished, the only name servers in the NS record will be the four Route 53 name servers.

   If you cannot delete name servers

   - Choose the option to use custom name servers.
   - Add all four Route 53 name servers that you got in step 1 of this procedure.

Step 8: Monitor traffic for the domain

Monitor traffic for the domain, including website or application traffic, and email:

- **If the traffic slows or stops** – Use the method provided by the previous DNS service to change the name servers for the domain back to the previous name servers. These are the name servers that you made note of in step 7 of To update the NS record at the registrar, or the parent zone, to use Route 53 name servers. Then determine what went wrong.
- **If the traffic is unaffected** – Continue to Step 9: Change the TTL for the NS record back to a higher value.

Step 9: Change the TTL for the NS record back to a higher value

In the Amazon Route 53 hosted zone for the domain, change the TTL for the NS record to a more typical value, for example, 172800 seconds (two days). This improves latency for your users because they don't have to wait as often for DNS resolvers to send a query for your domain's name servers.

To change the TTL for the NS record in the Route 53 hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. Choose **Hosted Zones** in the navigation pane.
3. Choose the name of the hosted zone.
4. In the list of records for the hosted zone, choose the NS record.
5. Choose **Edit**.
6. Change **TTL (Seconds)** to the number of seconds that you want DNS resolvers to cache the names of the name servers for your domain. We recommend a value of 172800 seconds.
7. Choose **Save changes**.

Step 10: Transfer domain registration to Amazon Route 53

Now that you've transferred DNS service for a domain to Amazon Route 53, you can optionally transfer registration for the domain to Route 53. For more information, see Transferring registration for a domain to Amazon Route 53.

Step 11: Re-enable DNSSEC signing (if required)

Now that you've transferred DNS service for a domain to Amazon Route 53, you can re-enable DNSSEC signing.

Enabling DNSSEC signing has two steps:

- Step 1: Enable DNSSEC signing for Route 53, and request that Route 53 create a key signing key (KSK) based on a customer managed key in AWS Key Management Service (AWS KMS).
- Step 2: Create a chain of trust for the hosted zone by adding a Delegation Signer (DS) record to the parent zone. This lets DNS responses be verified with trusted cryptographic signatures.

  For instructions, see Enabling DNSSEC signing and establishing a chain of trust.

Making Route 53 the DNS service for an inactive domain

If you want to migrate DNS service to Amazon Route 53 for a domain that isn't getting any traffic (or is getting very little traffic), perform the procedures in this section.

Topics

- Step 1: Get your current DNS configuration from the current DNS service provider (inactive domains)
- Step 2: Create a hosted zone (inactive domains)
- Step 3: Create records (inactive domains)
- Step 4: Update the domain registration to use Amazon Route 53 name servers (inactive domains)

Step 1: Get your current DNS configuration from the current DNS service provider (inactive domains)

When you migrate DNS service from another provider to Route 53, you reproduce your current DNS configuration in Route 53. In Route 53, you create a hosted zone that has the same name as your domain, and you create records in the hosted zone. Each record indicates how you want to route traffic for a domain name or subdomain name. For example, when someone enters your domain name in a web browser, do you want traffic to go to a web server in your data center, to an Amazon EC2 instance, to a CloudFront distribution, or to some other location?

The process depends on the complexity of your current DNS configuration:

- **If your current DNS configuration is simple** – If you're routing internet traffic for just a few subdomains to a small number of resources, such as web servers or Amazon S3 buckets, then you can manually create a few records in the Route 53 console.
- **If your current DNS configuration is more complex, and you just want to reproduce your current configuration** – You can simplify the migration if you can get a zone file from the current DNS service provider and import it into Route 53. (Not all DNS service providers offer zone files.) When you import a zone file, Route 53 reproduces the existing configuration by creating the corresponding records in your hosted zone.

  Try asking customer support with your current DNS service provider how to get a *zone file* or a *records list*. For information about the required format of the zone file, see Creating records by importing a zone file.

- **If your current DNS configuration is more complex, and you're interested in Route 53 routing features** – Review the following documentation to see whether you want to use Route 53 features not available from other DNS service providers. If so, you can create records manually, or you can import a zone file and then create or update records later:
  - Choosing between alias and non-alias records explains the advantages of Route 53 alias records, which route traffic to some AWS resources, such as CloudFront distributions and Amazon S3 buckets, for no charge.
  - Choosing a routing policy explains the Route 53 routing options, for example, routing based on user location, latency between users and resources, resource health, and weights that you specify.

> **Note**
>
> You can also import a zone file and later change your configuration to take advantage of alias records and complex routing policies.

If you can't get a zone file or if you want to manually create records in Route 53, the records that you're likely to migrate include the following:

- **A (Address) records** – associate a domain name or subdomain name with the IPv4 address (for example, 192.0.2.3) of the corresponding resource
- **AAAA (Address) records** – associate a domain name or subdomain name with the IPv6 address (for example, 2001:0db8:85a3:0000:0000:abcd:0001:2345) of the corresponding resource
- **Mail server (MX) records** – route traffic to mail servers
- **CNAME records** – reroute traffic for one domain name (example.net) to another domain name (example.com)
- **Records for other supported DNS record types** – For a list of supported record types, see Supported DNS record types.

Step 2: Create a hosted zone (inactive domains)

To tell Amazon Route 53 how you want to route traffic for your domain, you create a hosted zone that has the same name as your domain, and then you create records in the hosted zone.

> **Important**
>
> You can create a hosted zone only for a domain that you have permission to administer. Typically, this means that you own the domain, but you might also be developing an application for the domain registrant.

When you create a hosted zone, Route 53 automatically creates a name server (NS) record and a start of authority (SOA) record for the zone. The NS record identifies the four name servers that Route 53 associated with your hosted zone. To make Route 53 the DNS service for your domain, update the registration for the domain to use these four name servers.

> **Important**
>
> Don't create additional name server (NS) or start of authority (SOA) records, and don't delete the existing NS and SOA records.

To create a hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. If you're new to Route 53, choose **Get started**.

   If you're already using Route 53, choose **Hosted zones** in the navigation pane.
3. Choose **Create hosted zone**.
4. In the **Create hosted zone** pane, enter a domain name and, optionally, a comment. For more information about a setting, pause the mouse pointer over its label to see a tool tip.

   For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.
5. For **Record type**, accept the default value of **Public hosted zone**.
6. Choose **Create hosted zone**.

Step 3: Create records (inactive domains)

After you create a hosted zone, you create records in the hosted zone that define where you want to route traffic for a domain (example.com) or subdomain (www.example.com). For example, if you want to route traffic for example.com and www.example.com to a web server on an Amazon EC2 instance, you create two records, one named example.com and the other named www.example.com. In each record, you specify the IP address for your EC2 instance.

You can create records in a variety of ways:

Import a zone file

This is the easiest method if you got a zone file from your current DNS service in Step 1: Get your current DNS configuration from the current DNS service provider (inactive domains). Amazon Route 53 can't predict when to create alias records or to use special routing types such as weighted or failover. As a result, if you import a zone file, Route 53 creates standard DNS records using the simple routing policy.

For more information, see Creating records by importing a zone file.

Create records individually in the console

If you didn't get a zone file and you just want to create a few records with a routing policy of Simple to get started, you can create the records in the Route 53 console. You can create both alias and non-alias records.

For more information, see the following topics:

- Choosing a routing policy
- Choosing between alias and non-alias records
- Creating records by using the Amazon Route 53 console

Create records programmatically

You can create records by using one of the AWS SDKs, the AWS CLI, or AWS Tools for Windows PowerShell. For more information, see AWS Documentation.

If you're using a programming language that AWS doesn't provide an SDK for, you can also use the Route 53 API. For more information, see the Amazon Route 53 API Reference.

Step 4: Update the domain registration to use Amazon Route 53 name servers (inactive domains)

When you've finished creating records for the domain, you can change the DNS service for your domain to Amazon Route 53. Perform the following procedure to update settings with the domain registrar.

To update the name servers for the domain

1. In the Route 53 console, get the name servers for your Route 53 hosted zone:
   a. Open the Route 53 console at https://console.aws.amazon.com/route53/.
   b. In the navigation pane, choose **Hosted zones**.
   c. On the **Hosted zones** page, choose the radio button (not the name) for the hosted zone, then choose **View details**.
   d. On the details page for the hosted zone, choose **Hosted zone details**.
   e. Make note of the four servers listed for **Name servers**.
2. Use the method provided by the registrar for the domain to change the name servers for the domain to use the four Route 53 name servers that you got in step 2 of this procedure.

   If the domain is registered with Route 53, see Adding or changing name servers and glue records for a domain.

Configuring DNS routing for a new domain

To route internet traffic to your website or web application, you configure DNS routing for your domain. The steps depend on whether you purchased your domain from Route 53 or from another registrar.

Set up DNS for a domain you purchased from Route 53

When you register a domain with Route 53, Route 53 makes itself the DNS service for the domain. It creates a hosted zone with the same name as the domain, assigns four name servers to the zone, and updates the domain to use those name servers.

Set up DNS for a domain you purchased from another registrar

When you buy a domain from another registrar, for example, because the top-level domain (TLD) isn't offered by Route 53, you have two options:

- **Use Route 53 for DNS hosting only:** Keep your current registrar but let Route 53 handle DNS. Follow the steps below to create a hosted zone and update your registrar's name servers.
- **Transfer the domain to Route 53:** Make Route 53 your registrar and DNS service. See Pre-transfer checklist for domain transfers for what to do first, and Transferring registration for a domain to Amazon Route 53 for the transfer steps.

For more information about TLDs supported by Route 53, see Domains that you can register with Amazon Route 53.

Create a hosted zone and delegate DNS

Follow these steps to create a public hosted zone and then add the name servers to your registrar:

To create a hosted zone for a non-Route 53 domain

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and then choose **Create hosted zone**.
3. For the **Name**, enter the name of the domain you want to create a hosted zone for, such as `example.com`, add a description if you want, choose **Public hosted zone**, and then choose **Create hosted zone**.
4. After you create the hosted zone, note the four name server (NS) records. Each starts with "ns-".

   At your domain registrar, enter these name servers to point the domain to your Route 53 hosted zone.

Route DNS traffic

To tell Route 53 how to route internet traffic for the domain, you create records in the hosted zone. For example, if you want to route requests for example.com to a web server on an Amazon EC2 instance, you create a record in the example.com hosted zone and specify the Elastic IP address for that instance. For more information, see the following topics:

- For information about how to create records in your hosted zone, see Working with records.
- For information about how to route traffic to selected AWS resources, see Routing internet traffic to your AWS resources.
- For information about how DNS works, see How internet traffic is routed to your website or web application.
- To check DNS repose, see Checking DNS responses from Route 53.

Routing traffic to your resources

When users request your website or web application, for example, by entering the name of your domain in a web browser, Amazon Route 53 helps route users to your resources, such as an Amazon S3 bucket or a web server in your data center. To configure Route 53 to route traffic to your resources, do the following:

1. Create a hosted zone. You can create either a public hosted zone or a private hosted zone:

   Public hosted zone

   Create a public hosted zone if you want to route internet traffic to your resources, for example, so your customers can view the company website that you're hosting on EC2 instances. For more information, see Working with public hosted zones.

   Private hosted zone

   Create a private hosted zone if you want to route traffic within an Amazon VPC. For more information, see Working with private hosted zones.

2. Create records in the hosted zone. Records define where you want to route traffic for each domain name or subdomain name. For example, to route traffic for www.example.com to a web server in your data center, you typically create a www.example.com record in the example.com hosted zone.

   For more information, see the following topics:

   - Working with records
   - Routing traffic for subdomains
   - Routing internet traffic to your AWS resources

Routing traffic for subdomains

When you want to route traffic to your resources for a subdomain, such as acme.example.com or zenith.example.com, you have two options:

Create records in the hosted zone for the domain

Typically, to route traffic for a subdomain, you create a record in the hosted zone that has the same name as the domain. For example, to route internet traffic for acme.example.com to a web server in your data center, you create a record named acme.example.com in the example.com hosted zone. For more information, see the topic Working with records and its subtopics.

Create a hosted zone for the subdomain, and create records in the new hosted zone

You can also create a hosted zone for the subdomain. Using a separate hosted zone to route internet traffic for a subdomain is sometimes known as "delegating responsibility for a subdomain to a hosted zone" or "delegating a subdomain to other name servers." Here's an overview of how it works:

1. You create a hosted zone that has the same name as the subdomain that you want to route traffic for, such as acme.example.com.
2. You create records in the new hosted zone that define how you want to route traffic for the subdomain (acme.example.com) and its subdomains, such as backend.acme.example.com.
3. You get the name servers that Route 53 assigned to the new hosted zone when you created it.
4. You create a new NS record in the hosted zone for the domain (example.com). The NS record name must be the subdomain name (acme.example.com), and you specify the four name servers that you got in step 3 as the record values.

When you use a separate hosted zone to route traffic for a subdomain, you can use IAM permissions to restrict access to the hosted zone for the subdomain. If you have multiple subdomains managed by different groups, creating a hosted zone for each subdomain can reduce the number of people who need access to records in the hosted zone for the domain.

To implement this delegation process, you need the following IAM permissions. For more information about Route 53 IAM policies, see Identity and access management in Amazon Route 53:

Parent account (owns example.com)

The user or role needs permissions to modify records in the parent domain's hosted zone:

Child account (creates acme.example.com)

The user or role needs permissions to create and manage the subdomain hosted zone:

Using a separate hosted zone for a subdomain also allows you to use different DNS services for the domain and the subdomain. For more information, see Using Amazon Route 53 as the DNS service for subdomains without migrating the parent domain.

There's a small performance impact to this configuration for the first DNS query from each DNS resolver. The resolver must get information from the hosted zone for the root domain and then from the hosted zone for the subdomain. After the first DNS query for a subdomain, the resolver caches the information and doesn't need to get it again until the TTL expires and another client requests the subdomain from that resolver. For more information, see TTL (seconds) in the section Values that you specify when you create or edit Amazon Route 53 records.

Topics

- Creating another hosted zone to route traffic for a subdomain
- Routing traffic for additional levels of subdomains

Creating another hosted zone to route traffic for a subdomain

One way to route traffic for a subdomain is to create a hosted zone for the subdomain, and then create records for the subdomain in the new hosted zone. (The more common option is to create records for the subdomain in the hosted zone for the domain.)

> **Note**
>
> This procedure shows how to delegate a subdomain to Route 53. You can also delegate a subdomain to other DNS services by creating NS records that point to those services' name servers instead.

Here's an overview of the process:

1. Create a hosted zone for the subdomain. For more information, see Creating a new hosted zone for a subdomain.
2. Add records to the hosted zone for the subdomain. If the hosted zone for the domain contains any records that belong in the hosted zone for the subdomain, duplicate those records in the hosted zone for the subdomain. For more information, see Creating records in the hosted zone for the subdomain
3. Create an NS record for the subdomain in the hosted zone for the domain, which delegates responsibility for the subdomain to the name servers in the new hosted zone. If the hosted zone for the domain contains any records that belong in the hosted zone for the subdomain, delete the records from the hosted zone for the domain. (You created copies in the hosted zone for the subdomain in step 2.) For more information, see Updating the hosted zone for the domain.

Creating a new hosted zone for a subdomain

To create a hosted zone for a subdomain using the Route 53 console, perform the following procedure.

To create a hosted zone for a subdomain (console)

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. If you're new to Route 53, choose **Get started**.

   If you're already using Route 53, choose **Hosted zones** in the navigation pane.
3. Choose **Create hosted zone**.
4. In the right pane, enter the name of the subdomain, such as acme.example.com. You can also optionally enter a comment.

   For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.
5. For **Type**, accept the default value of **Public hosted zone**.
6. At the bottom of the right pane, choose **Create hosted zone**.

Creating records in the hosted zone for the subdomain

To define how you want Route 53 to route traffic for the subdomain (acme.example.com) and its subdomains (backend.acme.example.com), create records in the hosted zone for the subdomain.

Note the following about creating records in the hosted zone for the subdomain:

- Don't create additional name server (NS) or start of authority (SOA) records in the hosted zone for the subdomain, and don't delete the existing NS and SOA records.
- Create all records for the subdomain in the hosted zone for the subdomain. For example, if you have hosted zones for example.com and for acme.example.com domain, create all records for the acme.example.com subdomain in the acme.example.com hosted zone. This includes records such as backend.acme.example.com and beta.backend.acme.example.com.
- If the hosted zone for the domain (example.com) already contains records that belong in the hosted zone for the subdomain (acme.example.com), duplicate those records in the hosted zone for the subdomain. In the last step of the process, you delete the duplicate records from the hosted zone for the domain later.

> **Important**
>
> If you have some records for the subdomain in both the hosted zone for the domain and the hosted zone for the subdomain, DNS behavior will be inconsistent. Behavior depends on which name servers a DNS resolver has cached: the name servers for the domain hosted zone (example.com) or the name servers for the subdomain hosted zone (acme.example.com). In some cases, Route 53 will return NXDOMAIN (non-existent domain) when the record exists but not in the hosted zone that DNS resolvers query.

For more information, see Working with records.

Updating the hosted zone for the domain

When you create a hosted zone, Route 53 automatically assigns four name servers to the zone. The NS record for a hosted zone identifies the name servers that respond to DNS queries for the domain or subdomain. To start using the records in the hosted zone for the subdomain to route internet traffic, create a new NS record in the hosted zone for the domain (example.com). Give it the name of the subdomain (acme.example.com). For the value of the NS record, specify the names of the name servers from the hosted zone for the subdomain.

Here's what happens when Route 53 receives a DNS query from a DNS resolver for the subdomain acme.example.com or one of its subdomains:

1. Route 53 looks in the hosted zone for the domain (example.com) and finds the NS record for the subdomain (acme.example.com).
2. Route 53 gets the name servers from the acme.example.com NS record in the hosted zone for the domain, example.com, and returns those name servers to the DNS resolver.
3. The resolver resubmits the query for acme.example.com to the name servers for the acme.example.com hosted zone.
4. Route 53 responds to the query using a record in the acme.example.com hosted zone.

To configure Route 53 to route traffic for the subdomain using the hosted zone for the subdomain, and to delete any duplicate records from the hosted zone for the domain, perform the following procedure:

To configure Route 53 to use the hosted zone for the subdomain (console)

1. In the Route 53 console, get the name servers for the hosted zone for the subdomain:
   a. In the navigation pane, choose **Hosted zones**.
   b. On the **Hosted zones** page, choose the name for the hosted zone for the subdomain.
   c. In the right pane, copy the names of the four servers listed for **Name servers** in the **Hosted zones details** section.
2. Choose the name of the hosted zone for the domain (example.com), not for the subdomain.
3. Choose **Create record**.
4. Choose **Simple routing** and choose **Next**.
5. Choose **Define simple record**.
6. Specify the following values:

   Name

   Enter the name of the subdomain (for example, `acme.example.com`). This must match exactly the name of the subdomain hosted zone you created.

   Value/Route traffic to

   Choose **IP address or another value depending on the record type**, and paste the names of the name servers that you copied in step 1.

Record type

Choose **NS – Name servers for a hosted zone**.

TTL (Seconds)

Change to a more common value for an NS record, such as 172800 seconds.

7. Choose **Define simple record**, and choose **Create records**.
8. If the hosted zone for the domain contains any records that you recreated in the hosted zone for the subdomain, delete those records from the hosted zone for the domain. For more information, see Deleting records.

   When you're finished, all records for the subdomain should be in the hosted zone for the subdomain.

Routing traffic for additional levels of subdomains

You route traffic to a subdomain of a subdomain, such as backend.acme.example.com, the same way that you route traffic to a subdomain, such as acme.example.com. You create records in the hosted zone for the domain, or you create a hosted zone for the lower-level subdomain and create records in that new hosted zone.

If you create a separate hosted zone for the lower-level subdomain, create the NS record for the lower-level subdomain in the hosted zone for the subdomain that is one level closer to the domain name. This makes sure that traffic is correctly routed to your resources. For example, suppose you want to route traffic for the following subdomains:

- subdomain1.example.com
- subdomain2.subdomain1.example.com

To use another hosted zone to route traffic for subdomain2.subdomain1.example.com, you do the following:

1. Create a hosted zone named subdomain2.subdomain1.example.com.
2. Create records in the subdomain2.subdomain1.example.com hosted zone. For more information, see Creating records in the hosted zone for the subdomain.
3. Copy the names of the name servers for the subdomain2.subdomain1.example.com hosted zone.

4. In the subdomain1.example.com hosted zone, create an NS record named subdomain2.subdomain1.example.com, and paste in the names of the name servers for the subdomain2.subdomain1.example.com hosted zone.

   In addition, delete any duplicate records from the subdomain1.example.com. For more information, see Updating the hosted zone for the domain.

   After you create this NS record, Route 53 starts to use the subdomain2.subdomain1.example.com hosted zone to route traffic for the subdomain2.subdomain1.example.com subdomain.

Working with hosted zones

A hosted zone is a container for records. Records hold information about how to route traffic for a domain, such as example.com, and its subdomains (acme.example.com, zenith.example.com). A hosted zone has the same name as its domain. There are two types:

- *Public hosted zones* contain records that specify how you want to route traffic on the internet. For more information, see Working with public hosted zones.
- *Private hosted zones* contain records that specify how you want to route traffic in an Amazon VPC. For more information, see Working with private hosted zones.

Every public and private hosted zone also has a tier, which determines the features that are available for the zone. For more information, see Hosted zone tiers and Hosted zone features.

Working with public hosted zones

A public hosted zone is a container that holds information about how you want to route traffic on the internet for a domain, such as example.com, and its subdomains (acme.example.com, zenith.example.com). You get a public hosted zone in one of two ways:

- When you register a domain with Route 53, Route 53 creates a hosted zone for you automatically.
- When you transfer DNS service for an existing domain to Route 53, you start by creating a hosted zone for the domain. For more information, see Making Amazon Route 53 the DNS service for an existing domain.

In both cases, you then create records in the hosted zone to route traffic for the domain and subdomains. For example, you might create a record to route traffic for www.example.com to a CloudFront distribution or to a web server in your data center. For more information about records, see Working with records.

This topic explains how to use the Amazon Route 53 console to create, list, and delete public hosted zones.

> **Note**
>
> You can also use a Route 53 *private* hosted zone to route traffic within one or more VPCs that you create with the Amazon VPC service. For more information, see Working with private hosted zones.

Topics

- Considerations when working with public hosted zones
- Creating a public hosted zone
- Getting the name servers for a public hosted zone
- Listing public hosted zones
- Viewing DNS query metrics for a public hosted zone
- Deleting a public hosted zone
- Checking DNS responses from Route 53
- Configuring white-label name servers
- NS and SOA records that Amazon Route 53 creates for a public hosted zone

Considerations when working with public hosted zones

Note the following considerations when working with public hosted zones:

NS and SOA records

When you create a hosted zone, Amazon Route 53 creates a name server (NS) record and a start of authority (SOA) record for the zone. The NS record lists the four name servers that you give to your registrar or your DNS service so that DNS queries reach Route 53 name servers. For more information about NS and SOA records, see NS and SOA records that Amazon Route 53 creates for a public hosted zone.

Multiple hosted zones that have the same name

You can create more than one hosted zone with the same name and add different records to each. Route 53 assigns four name servers to every hosted zone, and the name servers differ for each one. When you update your registrar's name server records, be careful to use the Route 53 name servers for the correct hosted zone—the one that has the records you want Route 53 to use when responding to queries for your domain. Route 53 never returns values for records in other hosted zones that have the same name.

Reusable delegation sets

By default, Route 53 assigns a unique set of four name servers (called a delegation set) to each hosted zone that you create. If you want to create many hosted zones, you can create a reusable delegation set with the API. (Reusable delegation sets aren't available in the Route 53 console.) You can then create hosted zones with the API and assign the same four name servers to each one.

Reusable delegation sets make it simpler to migrate DNS service to Route 53. You can tell your registrar to use the same four name servers for all the domains that you want Route 53 to serve. For more information, see CreateReusableDelegationSet in the *Amazon Route 53 API Reference*.

Creating a public hosted zone

A public hosted zone is a container that holds information about how you want to route traffic on the internet for a domain, such as example.com, and its subdomains (acme.example.com, zenith.example.com). After you create a hosted zone, you create records to route traffic for the domain and subdomains.

Restrictions

Note the following restrictions for creating hosted zones with Route 53.

- You can create a hosted zone only for a domain that you have permission to administer. Typically, this means that you own the domain, but you might also be developing an application for the domain registrant.
- You can create hosted zones for domains and subdomains only. Route 53 doesn't support hosting top-level domains (TLD) such as `.com`.

To create a public hosted zone using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. If you're new to Route 53, choose **Get started** under **DNS management**.

   If you're already using Route 53, choose **Hosted zones** in the navigation pane.
3. Choose **Create hosted zone**.
4. In the **Create Hosted Zone** pane, enter the name of the domain that you want to route traffic for. You can also optionally enter a comment.

   For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.
5. For **Type**, accept the default value of **Public Hosted Zone**.
6. Choose **Create**.
7. Create records that specify how you want to route traffic for the domain and subdomains. For more information, see Working with records.
8. To use records in the new hosted zone to route traffic for your domain, see the applicable topic:

   - If you're making Route 53 the DNS service for a domain that is registered with another domain registrar, see Making Amazon Route 53 the DNS service for an existing domain.
   - If the domain is registered with Route 53, see Adding or changing name servers and glue records for a domain.

Getting the name servers for a public hosted zone

You get the name servers for a public hosted zone if you want to change the DNS service for your domain registration. For information about how to change your DNS service, see Making Amazon Route 53 the DNS service for an existing domain.

> **Note**
>
> Some registrars only let you specify name servers using IP addresses; they don't accept fully qualified domain names. If your registrar needs IP addresses, you can get the IP addresses for your name servers using the dig utility (for Mac, Unix, or Linux) or the

nslookup utility (for Windows). Route 53 rarely changes the IP addresses of name servers; if Route 53 needs to change them, it will notify you in advance.

To get the name servers for a hosted zone using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, click **Hosted zones**.
3. On the **Hosted zones** page, choose the radio button (not the name) for the hosted zone, then choose **View details**.
4. On the details page for the hosted zone, choose **Hosted zone details**.
5. Make note of the four servers listed for **Name servers**.

Listing public hosted zones

You can use the Amazon Route 53 console to list all of the hosted zones that you created with the current AWS account. For information about how to list hosted zones using the Route 53 API, see ListHostedZones in the *Amazon Route 53 API Reference*.

To list the public hosted zones associated with an AWS account using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**. The page displays a list of the hosted zones that are associated with the AWS account that you are currently signed in with.
3. To filter hosted zones, use the search bar located at the top of the table.

   Search behavior depends on whether the hosted zone contains up to 2,000 records or more than 2,000 records:

   Up to 2,000 hosted zones

   - To display the records that have specific values, choose the search bar, choose a property in the dropdown list, and enter a value. You can also enter a value directly in the search bar and press Enter. For example, to display the hosted zones that have a name beginning with **abc**, enter that value in the search bar and press Enter.

- To display only the hosted zones that have the same hosted zone type, select the type in the dropdown list, and enter the type.

More than 2,000 hosted zones

- You can search for properties based on exact domain name, all properties, and type.
- Search using the exact domain name for faster search results.

Viewing DNS query metrics for a public hosted zone

You can view the total number of DNS queries that Route 53 responds to for a public hosted zone or a set of public hosted zones. The metrics show up in CloudWatch, so you can view a graph, choose the time period, and customize the metrics in other ways. You can also create alarms and set up notifications when the number of DNS queries goes above or below a level that you choose.

> **Note**
>
> Route 53 automatically sends the number of DNS queries to CloudWatch for all public hosted zones, so you don't need to configure anything before you can view query metrics. There's no charge for DNS query metrics.

Which DNS queries are counted?

Metrics include only the queries that DNS resolvers forward to Route 53. If a DNS resolver has already cached the response to a query (such as the IP address for a load balancer for example.com), it will keep returning the cached response until the TTL for that record expires.

Depending on how many DNS queries are sent for a domain name (example.com) or subdomain name (www.example.com), which resolvers your users use, and the TTL for the record, metrics might show only one query out of every few thousand queries sent to DNS resolvers. For more information about how DNS works, see How Amazon Route 53 routes traffic for your domain.

When do query metrics for a hosted zone start to appear in CloudWatch?

After you create a hosted zone, there's a delay of up to several hours before the hosted zone can appear in CloudWatch. In addition, you must submit a DNS query for a record in the hosted zone so there's data to display.

Metrics are available only in US East (N. Virginia)

To get metrics on the console, you must choose US East (N. Virginia) for the Region. To get metrics using the AWS CLI, you must either leave the AWS Region unspecified, or specify `us-east-1` as the Region. Route 53 metrics aren't available if you choose any other Region.

CloudWatch metric and dimension for DNS queries

For information about the CloudWatch metric and dimension for DNS queries, see Monitoring hosted zones using Amazon CloudWatch. For information about CloudWatch metrics, see Using Amazon CloudWatch metrics in the *Amazon CloudWatch User Guide*.

Getting more detailed data about DNS queries

To get more detailed information about each DNS query that Route 53 responds to, including the following values, you can configure query logging:

- Domain or subdomain that was requested
- Date and time of the request
- DNS record type (such as A or AAAA)
- Route 53 edge location that responded to the DNS query
- DNS response code, such as `NoError` or `ServFail`

For more information, see Public DNS query logging.

How to get DNS query metrics

Shortly after you create a hosted zone, Amazon Route 53 starts to send metrics and dimensions once a minute to CloudWatch. You can use the following procedures to view the metrics on the CloudWatch console or view them by using the AWS Command Line Interface (AWS CLI).

Topics

- Viewing DNS query metrics for a public hosted zone in the CloudWatch console
- Getting DNS query metrics using the AWS CLI

Viewing DNS query metrics for a public hosted zone in the CloudWatch console

To view DNS query metrics for public hosted zones in the CloudWatch console, perform the following procedure.

To view DNS query metrics for a public hosted zone on the CloudWatch console

1. Sign in to the AWS Management Console and open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. In the navigation pane, choose **Metrics**.
3. On the AWS Region list in the upper right corner of the console, choose **US East (N. Virginia)**. Route 53 metrics aren't available if you choose any other AWS Region.
4. On the **All metrics** tab, choose **Route 53**.
5. Choose **Hosted Zone Metrics**.
6. Select the check box for one or more hosted zones that have the metric name **DNSQueries**.
7. On the **Graphed metrics** tab, change the applicable values to view the metrics in the format that you want.

   For **Statistic**, choose **Sum** or **SampleCount**; these statistics both display the same value.

Getting DNS query metrics using the AWS CLI

To get DNS query metrics using the AWS CLI, you use the get-metric-data command. Note the following:

- You specify most values for the command in a separate JSON file. For more information, see get-metric-data.
- The command returns one value for each interval that you specify for `Period` in the JSON file. `Period` is in seconds, so if you specify a five-minute time period and specify `60` for `Period`, you get five values. If you specify a five-minute time period and specify `300` for `Period`, you get one value.
- In the JSON file, you can specify any value for `Id`.
- Either leave the AWS Region unspecified, or specify `us-east-1` as the Region. Route 53 metrics aren't available if you choose any other Region. For more information, see Configuring the AWS CLI in the *AWS Command Line Interface User Guide*.

Here's the AWS CLI command that you use to get DNS query metrics for the five-minute time period between 4:01 and 4:07 on May 1, 2019. The `metric-data-queries` parameter references the sample JSON file that follows the command.

```
aws cloudwatch get-metric-data --metric-data-queries file://./metric.json --start-time
 2019-05-01T04:01:00Z --end-time 2019-05-01T04:07:00Z
```

Here's the sample JSON file:

```
[
    {
        "Id": "my_dns_queries_id",
        "MetricStat": {
            "Metric": {
                "Namespace": "AWS/Route53",
                "MetricName": "DNSQueries",
                "Dimensions": [
                    {
                        "Name": "HostedZoneId",
                        "Value": "Z1D633PJN98FT9"
                    }
                ]
            },
            "Period": 60,
            "Stat": "Sum"
        },
        "ReturnData": true
    }
]
```

Here's the output from this command. Note the following:

- The start time and end time in the command cover a seven-minute time period, `2019-05-01T04:01:00Z` to `2019-05-01T04:07:00Z`.
- There are only six return values. There's no value for `2019-05-01T04:05:00Z` because there were no DNS queries during that minute.
- The value of `Period` specified in the JSON file is `60` (seconds), so the values are reported in one-minute intervals.

```
{
    "MetricDataResults": [
        {
            "Id": "my_dns_queries_id",
            "StatusCode": "Complete",
```

```
            "Label": "DNSQueries",
            "Values": [
                101.0,
                115.0,
                103.0,
                127.0,
                111.0,
                120.0
            ],
            "Timestamps": [
                "2019-05-01T04:07:00Z",
                "2019-05-01T04:06:00Z",
                "2019-05-01T04:04:00Z",
                "2019-05-01T04:03:00Z",
                "2019-05-01T04:02:00Z",
                "2019-05-01T04:01:00Z"
            ]
        }
    ]
}
```

Deleting a public hosted zone

This section explains how to delete a public hosted zone using the Amazon Route 53 console.

You can delete a hosted zone only if it has no records other than the default SOA and NS records. If your hosted zone has other records, delete them first. This prevents you from removing a hosted zone that still has records in use.

Topics

- Preventing traffic from being routed to your domain
- Deleting public hosted zones that were created by another service
- Using the Route 53 console to delete a public hosted zone

Preventing traffic from being routed to your domain

If you want to keep your domain registration but stop routing internet traffic to your website or web application, we suggest that you delete *records* in the hosted zone instead of deleting the hosted zone.

Important

If you delete a hosted zone, you can't undo it. You must create a new hosted zone and update the name servers for your domain, which can take up to 48 hours to take effect. Also, if you delete a hosted zone, someone could hijack the domain and route traffic to their own resources using your domain name.

If you gave a subdomain its own hosted zone and you want to delete that child hosted zone, you must also update the parent hosted zone by deleting the NS record with the same name as the child hosted zone. For example, if you want to delete the hosted zone acme.example.com, you must also delete the NS record acme.example.com in the example.com hosted zone. We suggest that you delete the NS record first, and wait for the TTL on that NS record to expire before you delete the child hosted zone. This ensures that no one can hijack the child hosted zone while DNS resolvers still have the child hosted zone's name servers cached.

If you want to avoid the monthly charge for the hosted zone, you can move DNS service for the domain to a free DNS service. When you move DNS service, you must update the name servers for the domain. If the domain is registered with Route 53, see Adding or changing name servers and glue records for a domain for information about how to replace Route 53 name servers with name servers for the new DNS service. If the domain is registered with another registrar, use the method provided by that registrar to update name servers for the domain. For more information, search the internet for "free DNS service."

Deleting public hosted zones that were created by another service

If another service created a hosted zone, you can't delete it using the Route 53 console. Instead, use the right process for that service:

- **AWS Cloud Map** – To delete a hosted zone that AWS Cloud Map created when you created a public DNS namespace, delete the namespace. AWS Cloud Map deletes the hosted zone automatically. For more information, see Deleting namespaces in the *AWS Cloud Map Developer Guide*.
- **Amazon Elastic Container Service (Amazon ECS) Service Discovery** – To delete a public hosted zone that Amazon ECS created when you created a service using service discovery, delete the Amazon ECS services that are using the namespace, and delete the namespace. For more information, see Deleting a service in the *Amazon Elastic Container Service Developer Guide*.

Using the Route 53 console to delete a public hosted zone

To use the Route 53 console to delete a public hosted zone, perform the following procedure.

To delete a public hosted zone using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and choose the highlighted link for the hosted zone you want to delete.
3. Confirm that the hosted zone that you want to delete contains only an NS and an SOA record. If it contains additional records, delete them. You will also need to disable DNSSEC signing:

   > **Note**
   >
   > If you attempt to delete the hosted zone without completing these requirements, Route 53 will return an error:
   >
   > - If DNSSEC signing is enabled: The specified hosted zone contains DNSSEC Key Signing Keys and so cannot be deleted
   > - If other resource record sets exist (other than the default SOA and NS records): The specified hosted zone contains non-required resource record sets and so cannot be deleted
   > - If record set protections exist: Hosted Zone *`hosted-zone-id`* has resource record set protections. All resource record set protections need to be deleted (DeleteResourceRecordSetProtections) before deleting hosted zone.

   a. Delete any record set protections on the **Protections** tab. You can't delete a protected record or a hosted zone that has protections. For more information about record set protections, see Protecting records from changes.

   b. On the hosted zone detail page, in the **Records** list, if the list of records includes any records for which the value of the **Type** column is something other than NS or SOA, choose the row, and choose **Delete**.

      To select multiple, consecutive records, choose the first row, press and hold the **Shift** key, and choose the last row. To select multiple, non-consecutive records, choose the first row, press and hold the **Ctrl** key, and choose the remaining rows.

> **Note**
>
> If you created any NS records for subdomains in the hosted zone, delete those records, too.

4. Go back to the the **Hosted zones** page, and choose the row for the hosted zone that you want to delete.
5. Choose **Delete**.
6. Type the confirmation key and choose **Delete**.
7. If you want to make the domain unavailable on the internet, we recommend that you transfer DNS service to a free DNS service and then delete the Route 53 hosted zone. This prevents future DNS queries from possibly being misrouted.

   If the domain is registered with Route 53, see Adding or changing name servers and glue records for a domain for information about how to replace Route 53 name servers with name servers for the new DNS service. If the domain is registered with another registrar, use the method provided by the registrar to change name servers for the domain.

   > **Note**
   >
   > If you're deleting a hosted zone for a subdomain (acme.example.com), you don't need to change name servers for the domain (example.com).

Checking DNS responses from Route 53

If you created an Amazon Route 53 hosted zone for your domain, you can use the DNS checking tool in the console to see how Route 53 responds to DNS queries when you set up your domain to use Route 53 as your DNS service. For geolocation, geoproximity, and latency records, you can also test queries from a given DNS resolver or client IP address to see what response Route 53 would return.

Important

The tool doesn't submit queries to the Domain Name System, it only responds based on the settings in the records in the hosted zone. The tool returns the same information regardless of whether the hosted zone is currently being used to route traffic for the domain.

The DNS checking tool works only for public hosted zones. The tool doesn't support AZ-affinity records. For information about how to test AZ-affinity records, see Considerations for AZ-affinity routing.

Note

The DNS checking tool returns information similar to what you would expect from the answer section of the `dig` command. Therefore, if you query for the name servers of a subdomain that point to the parent name servers, those will not be returned.

Topics

- Using the checking tool to see how Amazon Route 53 responds to DNS queries
- Using the checking tool to simulate queries from specific IP addresses (geolocation and latency records only)

Using the checking tool to see how Amazon Route 53 responds to DNS queries

You can use the tool to see what response Amazon Route 53 returns in response to a DNS query for a record.

To use the checking tool to see how Route 53 responds to DNS queries

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted Zones**.
3. On the **Hosted Zones** page, choose the name of a hosted zone. The console displays the list of records for that hosted zone.
4. To go directly to the **Check response from Route 53** page, choose **Test record**.

5. Specify the following values:

   - The name of the record, excluding the name of the hosted zone. For example, to check **www.example.com**, enter **www**. To check **example.com**, leave the **Record name** field blank.
   - The type of the record that you want to check, such as **A** or **CNAME**.

6. Choose **Get Response**.

7. The **Response returned by Route 53** section includes the following values:

   DNS response code

   A code that indicates whether the query was valid or not. The most common response code is **NOERROR**, meaning that the query was valid. If the response is not valid, Route 53 returns a response code that explains why not. For a list of possible response codes, see DNS RCODES on the IANA website.

   Protocol

   The protocol that Amazon Route 53 used to respond to the query, either **UDP** or **TCP**.

   Response returned by Route 53

   The value that Route 53 would return to a web application. The value is one of the following:

   - For non-alias records, the response contains the value or values in the record.
   - For multiple records that have the same name and type, which includes weighted, latency, geolocation, and failover, the response contains the value from the appropriate record, based on the request.
   - For alias records that refer to AWS resources other than another record, the response contains an IP address or a domain name for the AWS resource, depending on the type of resource.
   - For alias records that refer to other records, the response contains the value or values from the referenced record.

Using the checking tool to simulate queries from specific IP addresses (geolocation and latency records only)

If you have created latency or geolocation records, you can use the checking tool to simulate queries from the IP address for a DNS resolver and a client.

To use the checking tool to simulate queries from specified IP addresses

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted Zones**.
3. On the **Hosted Zones** page, choose the name of a hosted zone. The console displays the list of records for that hosted zone.
4. To go directly to the **Check response from Route 53** page, choose **Test record set**.

   To go to the **Check response from Route 53** page for a specific record, choose the check box for that record and choose **Test record set**.
5. If you chose **Test record set** without first choosing a record, specify the following values:

   - The name of the record, excluding the name of the hosted zone. For example, to check **www.example.com**, enter **www**. To check **example.com**, leave the **Record name** field blank.
   - The type of the record that you want to check, such as **A** or **CNAME**.
6. Specify the applicable values:

   Resolver IP address

   Specify an IPv4 or IPv6 address to simulate the location of the DNS resolver that a client uses to make requests. This is useful for testing latency and geolocation records. If you omit this value, the tool uses the IP address of a DNS resolver in the AWS US East (N. Virginia) Region (us-east-1).

   EDNS0 client subnet IP

   If the resolver supports EDNS0, enter the client subnet IP for an IP address in the applicable geographic location, for example, **192.0.2.0** or **2001:db8:85a3::8a2e:370:7334**.

   Subnet mask

   If you specify an IP address for **EDNS0 client subnet IP**, you can optionally specify the number of bits of the IP address that you want the checking tool to include in the DNS query. For example, if you specify **192.0.2.44** for **EDNS0 client subnet IP** and **24** for **Subnet mask**, the checking tool will simulate a query from **192.0.2.0/24**. The default value is 24 bits for IPv4 addresses and 64 bits for IPv6 addresses.
7. Choose **Get Response**.
8. The **Response returned by Route 53** section includes the following values:

DNS query sent to Route 53

The query, in BIND format, that the checking tool sent to Route 53. This is the same format that a web application would use to send a query. The three values are typically the name of the record, **IN** (for internet), and the type of the record.

DNS response code

A code that indicates whether the query was valid or not. The most common response code is **NOERROR**, meaning that the query was valid. If the response is not valid, Route 53 returns a response code that explains why not. For a list of possible response codes, see DNS RCODES on the IANA website.

Protocol

The protocol that Amazon Route 53 used to respond to the query, either **UDP** or **TCP**.

Response returned by Route 53

The value that Route 53 would return to a web application. The value is one of the following:

- For non-alias records, the response contains the value or values in the record.
- For multiple records that have the same name and type, which includes weighted, latency, geolocation, and failover, the response contains the value from the appropriate record, based on the request.
- For alias records that refer to AWS resources other than another record, the response contains an IP address or a domain name for the AWS resource, depending on the type of resource.
- For alias records that refer to other records, the response contains the value or values from the referenced record.

Configuring white-label name servers

Each Amazon Route 53 hosted zone is linked to four name servers, known as a delegation set. By default, the name servers have names like ns-2048.awsdns-64.com. If you want your name servers to match the domain name of your hosted zone, for example, ns1.example.com, you can set up white-label name servers, also known as vanity name servers or private name servers.

The following steps explain how to configure one set of four white-label name servers that you can reuse for multiple domains. For example, suppose you own the domains example.com, example.org, and example.net. With these steps, you can configure white-label name servers for example.com and reuse them for example.org and example.net.

Topics

- Step 1: Create a Route 53 reusable delegation set
- Step 2: Create or recreate Amazon Route 53 hosted zones, and change the TTL for NS and SOA records
- Step 3: Recreate records for your hosted zones
- Step 4: Get IP addresses
- Step 5: Create records for white-label name servers
- Step 6: Update NS and SOA records
- Step 7: Create glue records and change the registrar's name servers
- Step 8: Monitor traffic for the website or application
- Step 9: Change TTLs back to their original values
- Step 10: (Optional) contact recursive DNS services

Step 1: Create a Route 53 reusable delegation set

White-label name servers are associated with a Route 53 reusable delegation set. You can use white-label name servers for a hosted zone only if the hosted zone and the reusable delegation set were created by the same AWS account.

To create a reusable delegation set, you can use the Route 53 API, the AWS CLI, or one of the AWS SDKs. For more information, see the following documentation:

- **Route 53 API** – See CreateReusableDelegationSet in the *Amazon Route 53 API Reference*
- **AWS CLI** – See create-reusable-delegation-set in the *AWS CLI Command Reference*
- **AWS SDKs** See the applicable SDK documentation on the AWS Documentation page

Step 2: Create or recreate Amazon Route 53 hosted zones, and change the TTL for NS and SOA records

Create or recreate Amazon Route 53 hosted zones:

- **If you aren't currently using Route 53 as the DNS service for the domains for which you want to use white-label name servers** – Create the hosted zones and specify the reusable delegation set that you created in the previous step with each hosted zone. For more information, see CreateHostedZone in the *Amazon Route 53 API Reference*.
- **If you are using Route 53 as the DNS service for the domains for which you want to use white-label name servers** – You must recreate the hosted zones and link them to the reusable delegation set that you created in the step before.

> **Important**
>
> You cannot change the name servers for an existing hosted zone. You can link a reusable delegation set to a hosted zone only when you create it.

When you create the hosted zones and before you try to access the resources for the corresponding domains, change the following TTL values for each hosted zone:

- Change the TTL for the NS record for the hosted zone to 60 seconds or less.
- Change the minimum TTL for the SOA record for the hosted zone to 60 seconds or less. This is the last value in the SOA record.

If you give your registrar the wrong IP addresses for your white-label name servers, your website becomes unavailable and stays that way for the TTL after you fix the problem. By setting a low TTL, you reduce how long your website is down.

For more information about creating hosted zones and specifying a reusable delegation set for the name servers for the hosted zones, see CreateHostedZone in the *Amazon Route 53 API Reference*.

Step 3: Recreate records for your hosted zones

Create records in the hosted zones that you created in Step 2:

- **If you're migrating DNS service for your domains to Amazon Route 53** – You might be able to create records by importing information about your existing records. For more information, see Creating records by importing a zone file.
- **If you're replacing existing hosted zones so that you can use white-label name servers** – In the new hosted zones, recreate the records that appear in your current hosted zones. Route 53 doesn't provide a method of exporting records from a hosted zone, but some third-party vendors

do. You can then use the Route 53 import feature to import non-alias records for which the routing policy is simple. There is no way to export and re-import alias records or records for which the routing policy is anything other than simple.

For information about creating records by using the Route 53 API, see CreateHostedZone in the *Amazon Route 53 API Reference*. For information about creating records by using the Route 53 console, see Working with records.

Step 4: Get IP addresses

Get the IPv4 and IPv6 addresses of the name servers in the reusable delegation set, and fill in the following table.

Name of a name server in your reusable delegation set (example: Ns-2048.awsdns-64.com)	IPv4 and IPv6 addresses	Name that you want to assign to the white-label name server (example: ns1.example.com)
	IPv4: IPv6:	
	IPv4: IPv6:	
	IPv4: IPv6:	
	IPv4: IPv6:	

For example, suppose the four name servers for your reusable delegation set are:

- ns-2048.awsdns-64.com
- ns-2049.awsdns-65.net
- ns-2050.awsdns-66.org

- ns-2051.awsdns-67.co.uk

Here are the Linux and Windows commands that you'd run to get the IP addresses for the first of your four name servers:

dig commands for Linux

```
% dig A ns-2048.awsdns-64.com +short
192.0.2.117
```

```
% dig AAAA ns-2048.awsdns-64.com +short
2001:db8:85a3::8a2e:370:7334
```

nslookup command for Windows

```
c:\> nslookup ns-2048.awsdns-64.com
Non-authoritative answer:
Name:    ns-2048.awsdns-64.com
Addresses:  2001:db8:85a3::8a2e:370:7334
          192.0.2.117
```

Step 5: Create records for white-label name servers

In the hosted zone that has the same name (such as example.com) as the domain name of the white-label name servers (such as ns1.example.com), create eight records:

- One A record for each white-label name server
- One AAAA record for each white-label name server

> **Important**
>
> If you're using the same white-label name servers for two or more hosted zones, do not perform this step for the other hosted zones.

For each record, specify the following values. Refer to the table that you filled in for the previous step:

Routing policy

Specify **Simple routing**.

Record name

The name that you want to assign to one of your white-label name servers, for example, ns1.example.com. For the prefix (ns1 in this example), you can use any value that is valid in a domain name.

Value/Route traffic to

The IPv4 or IPv6 address of one of the Route 53 name servers in your reusable delegation set.

> **Important**
>
> If you specify the wrong IP addresses when you created records for your white-label name servers, your website or web application will become unavailable on the internet when you perform subsequent steps. Even if you correct the IP addresses immediately, your website or web application will remain unavailable for the duration of the TTL.

Record type

Specify **A** when you're creating records for the IPv4 addresses.

Specify **AAAA** when you're creating records for the IPv6 addresses.

TTL (seconds)

This value is the amount of time that DNS resolvers cache the information in this record before forwarding another DNS query to Route 53. We recommend that you specify an initial value of 60 seconds or less, so that you can recover quickly if you accidentally specify incorrect values in these records.

Step 6: Update NS and SOA records

Update SOA and NS records in the hosted zones that you want to use white-label name servers for. Perform Step 6 through Step 8 for one hosted zone and the corresponding domain at a time, then repeat for another hosted zone and domain.

> **Important**
>
> Start with the Amazon Route 53 hosted zone that has the same domain name (such as example.com) as the white-label name servers (such as ns1.example.com).

1. Update the SOA record by replacing the name of the Route 53 name server with the name of one of your white-label name servers

   Example

   Replace the name of the Route 53 name server:

   ```
   ns-2048.awsdns-64.net. hostmaster.example.com. 1 7200 900 1209600 60
   ```

   with the name of one of your white-label name servers:

   ```
   ns1.example.com. hostmaster.example.com. 1 7200 900 1209600 60
   ```

   > **Note**
   >
   > You changed the last value, the time to live (TTL), in Step 2: Create or recreate Amazon Route 53 hosted zones, and change the TTL for NS and SOA records.

   For information about updating records by using the Route 53 console, see Editing records.
2. In the NS record, make note of the names of the current name servers for the domain, so you can revert to these name servers if necessary.
3. Update the NS record. Replace the name of the Route 53 name servers with the names of your four white-label name servers, for example, `ns1.example.com`, `ns2.example.com`, `ns3.example.com`, and `ns4.example.com`.

Step 7: Create glue records and change the registrar's name servers

Use the method provided by the registrar to create glue records and change the registrar's name servers:

1. Add glue records:

- **If you're updating the domain that has the same domain name as the white-label name servers** – Create four glue records for which the names and IP addresses match the values that you got in step 4. Include both the IPv4 and the IPv6 address for a white-label name server in the corresponding glue record, for example:

  ns1.example.com – IP addresses = 192.0.2.117 and 2001:db8:85a3::8a2e:370:7334

  Registrars use a variety of terminology for glue records. You might also see this referred to as registering new name servers or something similar.
- **If you're updating another domain** – If Route 53 is your DNS service, you must first complete the step in the previous bullet and create the glue records that match the domain name. Then skip to step 2 in this procedure.

2. Change the name servers for the domain to the names of your white-label name servers.

If you're using Amazon Route 53 as your DNS service, see Adding or changing name servers and glue records for a domain.

Step 8: Monitor traffic for the website or application

Monitor the traffic for the website or application for which you created glue records and changed name servers in Step 7:

- **If the traffic stops** – Use the method provided by the registrar to change the name servers for the domain back to the previous Route 53 name servers. These are the name servers that you made note of in step 6b. Then determine what went wrong.
- **If the traffic is unaffected** – Repeat Step 6 through Step 8 for the rest of the hosted zones for which you want to use the same white-label name servers.

Step 9: Change TTLs back to their original values

For all of the hosted zones that are now using white-label name servers, change the following values:

- Change the TTL for the NS record for the hosted zone to a more typical value for NS records, for example, 172800 seconds (two days).
- Change the minimum TTL for the SOA record for the hosted zone to a more typical value for SOA records, for example, 900 seconds. This is the last value in the SOA record.

Step 10: (Optional) contact recursive DNS services

Optional If you're using Amazon Route 53 geolocation routing, contact the recursive DNS services that support the edns-client-subnet extension of EDNS0, and give them the names of your white-label name servers. This ensures that these DNS services will continue to route DNS queries to the optimal Route 53 location based on the approximate geographical location that the query came from.

NS and SOA records that Amazon Route 53 creates for a public hosted zone

For each public hosted zone that you create, Amazon Route 53 automatically creates a name server (NS) record and a start of authority (SOA) record. You rarely need to change these records.

Topics

- The name server (NS) record
- The start of authority (SOA) record

The name server (NS) record

Amazon Route 53 automatically creates a name server (NS) record that has the same name as your hosted zone. It lists the four name servers that are the authoritative name servers for your hosted zone. Except in rare circumstances, we recommend that you don't add, change, or delete name servers in this record.

The following examples show the format for the names of Route 53 name servers (these are examples only; don't use them when you're updating your registrar's name server records):

- *ns-2048.awsdns-64.com*
- *ns-2049.awsdns-65.net*
- *ns-2050.awsdns-66.org*
- *ns-2051.awsdns-67.co.uk*

To get the list of name servers for your hosted zone:

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, click **Hosted zones**.

3. On the **Hosted zones** page, choose the radio button (not the name) for the hosted zone, then choose **View details**.
4. On the details page for the hosted zone, choose **Hosted zone details**.
5. Make note of the four servers listed for **Name servers**.

For information about migrating DNS service from another DNS service provider to Route 53, see Making Amazon Route 53 the DNS service for an existing domain.

The start of authority (SOA) record

The start of authority (SOA) record holds base DNS information about the domain, for example:

```
ns-2048.awsdns-64.net. hostmaster.example.com. 1 7200 900 1209600 86400
```

A SOA record includes the following elements:

- The Route 53 name server that created the SOA record, for example, `ns-2048.awsdns-64.net`.
- The email address of the administrator. The `@` symbol is replaced by a period, for example, `hostmaster.example.com`. The default value is an amazon.com email address that is not monitored.
- A serial number that you can add to whenever you update a record in the hosted zone. Route 53 doesn't change the number on its own. (The serial number is used by DNS services that support secondary DNS.) In the example, this value is `1`.
- A refresh time in seconds that secondary DNS servers wait before querying the primary DNS server's SOA record to check for changes. In the example, this value is `7200`.
- The retry interval in seconds that a secondary server waits before retrying a failed zone transfer. Normally, the retry time is less than the refresh time. In the example, this value is `900` (15 minutes).
- The time in seconds that a secondary server will keep trying to finish a zone transfer. If this time passes without a successful transfer, the secondary server stops answering queries because it views its data as too old to be reliable. In the example, this value is `1209600` (two weeks).
- The minimum time to live (TTL). This value helps set how long recursive resolvers should cache the following responses from Route 53:

NXDOMAIN

There is no record of any type with the name that is specified in the DNS query, such as example.com. There also are no records that are children of the name that is specified in the DNS query, such as zenith.example.com.

NODATA

There is at least one record with the name that is specified in the DNS query, but none of those records have the type (such as A) that is specified in the DNS query.

When a DNS resolver caches an NXDOMAIN or NODATA response, this is referred to as *negative caching*.

The duration of negative caching is the lesser of the following values:

- This value—the minimum TTL in the SOA record. In the example, the value is `86400` (one day).
- The value of the TTL for the SOA record. The default value is 900 seconds. For information about changing this value, see Editing records.

When Route 53 responds to DNS queries with an NXDOMAIN or NODATA response (a negative response), you're charged the rate for standard queries. (See "Queries" in Amazon Route 53 Pricing.) If you're concerned about the cost of negative responses, one option is to change the TTL for the SOA record, the minimum TTL in the SOA record (this value), or both. Note that raising these TTLs, which apply to negative responses for the whole hosted zone, can have both good and bad effects:

- DNS resolvers on the internet cache the non-existence of records for longer periods, which reduces the number of queries that are forwarded to Route 53. This reduces the Route 53 charge for DNS queries.
- However, if you ever erroneously delete a valid record and later recreate it, DNS resolvers will cache the negative response (this record doesn't exist) for a longer period. This lengthens the amount of time that your customers or users can't reach the corresponding resource, such as a web server for acme.example.com.

To find your SOA records in Route 53

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.

3. Select the linked name of the domain for which you want to view records.
4. On the **Records** section you can see all the records listed and you can also filter records to find your SOA value.

Working with private hosted zones

A *private hosted zone* is a container that holds information about how you want Amazon Route 53 to respond to DNS queries for a domain and its subdomains within one or more VPCs that you create with the Amazon VPC service. Here's how private hosted zones work:

1. You create a private hosted zone, such as example.com, and specify the VPC that you want to associate with the hosted zone. After you create the hosted zone you can associate more VPCs with it.
2. You create records in the hosted zone that determine how Route 53 responds to DNS queries for your domain and subdomains within and among your VPCs. For example, suppose you have a database server that runs on an EC2 instance in the VPC that you associated with your private hosted zone. You create an A or AAAA record, such as db.example.com, and you specify the IP address of the database server.

   For more information about records, see Working with records. For information about the Amazon VPC requirements for using private hosted zones, see Using private hosted zones in the *Amazon VPC User Guide*.
3. When an application submits a DNS query for db.example.com, Route 53 returns the corresponding IP address. To get an answer from a private hosted zone, you must be running an EC2 instance in one of the associated VPCs or have an inbound endpoint from a hybrid setup. If you query a private hosted zone from outside the VPCs or your hybrid setup, the query is resolved recursively on the internet.
4. The application uses the IP address that it got from Route 53 to establish a connection with the database server.

When you create a private hosted zone, the following name servers are used:

- ns-0.awsdns-00.com
- ns-512.awsdns-00.net
- ns-1024.awsdns-00.org
- ns-1536.awsdns-00.co.uk

These name servers are used because the DNS protocol requires that every hosted zone must have an NS record set. These name servers are reserved and never used by Route 53 public hosted zones. You can only query those zones through VPC Resolver in a VPC that has been associated to the hosted zone by using an inbound endpoint connected to the VPCs specified in the private hosted zone.

While the name servers are visible on the internet, VPC Resolver doesn't connect to the name server addresses. Further, the private hosted zone information is not returned if you directly query the name servers over the internet. Instead, the VPC Resolver detects that queries are within a private namespace based on VPC to hosted zone associations and uses direct, private connectivity to reach the private DNS servers.

> **Note**
>
> You can change the NS record set in a private hosted zone if you want and private DNS resolution will still work. We don't recommend doing so, but if you choose to, you should use reserved domain names which are not used by public DNS servers.

If you want to route traffic for your domain on the internet, you use a Route 53 *public* hosted zone. For more information, see Working with public hosted zones.

Topics

- Considerations when working with a private hosted zone
- Creating a private hosted zone
- Listing private hosted zones
- Associating more VPCs with a private hosted zone
- Associating an Amazon VPC and a private hosted zone that you created with different AWS accounts
- Disassociating VPCs from a private hosted zone
- Deleting a private hosted zone
- VPC permissions

Considerations when working with a private hosted zone

When using private hosted zones, note the following considerations.

- Amazon VPC settings
- Route 53 health checks
- Supported routing policies for records in a private hosted zone
- Split-view DNS
- Public and private hosted zones that have overlapping namespaces
- Private hosted zones that have overlapping namespaces
- Private hosted zones and Route 53 VPC Resolver rules
- Delegating responsibility for a subdomain
- Custom DNS servers
- Required IAM permissions

Amazon VPC settings

To use private hosted zones, you must set the following Amazon VPC settings to `true`:

- `enableDnsHostnames`
- `enableDnsSupport`

For more information, see View and update DNS attributes for your VPC in the *Amazon VPC User Guide*.

Route 53 health checks

In a private hosted zone, you can associate Route 53 health checks only with az-affinity, failover, geolocation, geoproximity, latency, multivalue answer, and weighted records. For information about associating health checks with failover records, see Configuring failover in a private hosted zone.

Supported routing policies for records in a private hosted zone

You can use the following routing policies when you create records in a private hosted zone:

- Simple routing
- Failover routing
- Multivalue answer routing
- Weighted routing
- Latency-based routing
- Geolocation routing

- Geoproximity routing
- AZ-affinity routing

Creating records in a private hosted zone using other routing policies is not supported.

Split-view DNS

You can use Route 53 to configure split-view DNS, also known as split-horizon DNS. In split-view DNS, you use the same domain name (example.com) for internal uses (accounting.example.com) and external uses, such as your public website (www.example.com). You might also want to use the same subdomain name internally and externally, but serve different content or require different authentication for internal and external users.

To configure split-view DNS, you perform the following steps:

1. Create public and private hosted zones that have the same name. (Split-view DNS still works if you're using another DNS service for the public hosted zone.)
2. Associate one or more Amazon VPCs with the private hosted zone. Route 53 VPC Resolver uses the private hosted zone to route DNS queries in the specified VPCs.
3. Create records in each hosted zone. Records in the public hosted zone control how internet traffic is routed, and records in the private hosted zone control how traffic is routed in your Amazon VPCs.

If you need to perform name resolution of both your VPC and on-premises workloads, you can use Route 53 VPC Resolver. For more information, see What is Route 53 VPC Resolver?.

Public and private hosted zones that have overlapping namespaces

If you have private and public hosted zones that have overlapping namespaces, such as example.com and accounting.example.com, VPC Resolver routes traffic based on the most specific match. When users are logged into an EC2 instance in an Amazon VPC that you have associated with the private hosted zone, here's how Route 53 VPC Resolver handles DNS queries:

1. VPC Resolver evaluates whether the name of the private hosted zone matches the domain name in the request, such as accounting.example.com. A match is defined as either of the following:
   - An identical match
   - The name of the private hosted zone is a parent of the domain name in the request. For example, suppose the domain name in the request is the following:

seattle.accounting.example.com

The following hosted zones match because they're parents of seattle.accounting.example.com:

- **accounting.example.com**
- **example.com**

If there's no matching private hosted zone, then VPC Resolver forwards the request to a public DNS resolver, and your request is resolved as a regular DNS query.

2. If a private hosted zone name matches the domain name in the request, VPC Resolver searches the hosted zone for a record that matches the domain name and DNS type, such as an A record for accounting.example.com.

> **Note**
>
> If the queried name doesn't exist in the matching private hosted zone, VPC Resolver returns NXDOMAIN (non-existent domain) to the client and doesn't retry the query against public DNS. If the name exists but has no record of the requested type, VPC Resolver returns NODATA.
> If you enable public DNS fallback for the zone, VPC Resolver retries the query against public DNS instead of returning NXDOMAIN. For more information, see Public DNS fallback for private hosted zones.

Private hosted zones that have overlapping namespaces

If you have two or more private hosted zones that have overlapping namespaces, such as example.com and accounting.example.com, VPC Resolver routes traffic based on the most specific match.

> **Note**
>
> If you have a private hosted zone (example.com) and a Route 53 VPC Resolver rule that routes traffic to your network for the same domain name, the VPC Resolver rule takes precedence. See Private hosted zones and Route 53 VPC Resolver rules.

When users are logged into an EC2 instance in an Amazon VPC that you have associated with all of the private hosted zones, here's how VPC Resolver handles DNS queries:

1. VPC Resolver evaluates whether the domain name in the request, such as accounting.example.com, matches the name of one of the private hosted zones.
2. If there is no hosted zone that exactly matches the domain name in the request, VPC Resolver checks for a hosted zone that has a name that is the parent of the domain name in the request. For example, suppose the domain name in the request is the following:

   `seattle.accounting.example.com`

   The following hosted zones match because they're parents of `seattle.accounting.example.com`:

   - `accounting.example.com`
   - `example.com`

   VPC Resolver chooses `accounting.example.com` because it's more specific than `example.com`.
3. VPC Resolver searches the `accounting.example.com` hosted zone for a record that matches the domain name and DNS type in the request, such as an A record for `seattle.accounting.example.com`.

   If the queried name doesn't exist in the zone, VPC Resolver returns NXDOMAIN (non-existent domain). If the name exists but has no record of the requested type, VPC Resolver returns NODATA.

> **Note**
>
> If you enable public DNS fallback for `accounting.example.com`, VPC Resolver retries queries for names that don't exist in that zone against public DNS instead of returning NXDOMAIN. It doesn't check the `example.com` zone. For more information, see Public DNS fallback for private hosted zones.

Private hosted zones and Route 53 VPC Resolver rules

If you have a private hosted zone (example.com) and a VPC Resolver rule that routes traffic to your network for the same domain name, the VPC Resolver rule takes precedence.

For example, suppose you have the following configuration:

- You have a private hosted zone called example.com, and you associate it with a VPC.
- You create a Route 53 VPC Resolver rule that forwards traffic for example.com to your network, and you associate the rule with the same VPC.

In this configuration, the VPC Resolver rule takes precedence over the private hosted zone. DNS queries are forwarded to your network instead of being resolved based on the records in the private hosted zone.

Delegating responsibility for a subdomain

You can now create NS records in a private hosted zone to delegate responsibility for a subdomain. For more information, see Resolver delegation rules tutorial.

Custom DNS servers

If you have configured custom DNS servers on Amazon EC2 instances in your VPC, you must configure those DNS servers to route your private DNS queries to the IP address of the Amazon-provided DNS servers for your VPC. This IP address is the IP address at the base of the VPC network range "plus two." For example, if the CIDR range for your VPC is 10.0.0.0/16, the IP address of the DNS server is 10.0.0.2.

If you want to route DNS queries between VPCs and your network, you can use VPC Resolver. For more information, see What is Route 53 VPC Resolver?.

Required IAM permissions

To create private hosted zones, you need to grant IAM permissions for Amazon EC2 actions in addition to permissions for Route 53 actions. For more information, see Actions, resources, and condition keys for Route 53 in the *Service Authorization Reference*.

Creating a private hosted zone

A private hosted zone is a container for records for a domain that you host in one or more Amazon virtual private clouds (VPCs). You create a hosted zone for a domain (such as example.com), and then you create records to tell Amazon Route 53 how you want traffic to be routed for that domain within and among your VPCs.

> **Important**
>
> When you create a private hosted zone, you must associate a VPC with the hosted zone, and the VPC that you specify must have been created by using the same account that

you're using to create the hosted zone. After you create the hosted zone, you can associate additional VPCs with it, including VPCs that you created by using a different AWS account. To associate VPCs that you created by using one account with a private hosted zone that you created by using a different account, you must authorize the association and then make the association programmatically. For more information, see Associating an Amazon VPC and a private hosted zone that you created with different AWS accounts.

For information about creating a private hosted zone by using the Route 53 API, see the Amazon Route 53 API Reference.

To create a private hosted zone using the Route 53 console

1. For each VPC that you want to associate with the Route 53 hosted zone, change the following VPC settings to `true`:

   - `enableDnsHostnames`
   - `enableDnsSupport`

   For more information, see Updating DNS support for your VPC in the *Amazon VPC User Guide*.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. If you're new to Route 53, choose **Get started**

   If you're already using Route 53, choose **Hosted zones** in the navigation pane.
4. Choose **Create hosted zone**.
5. In the **Create private hosted zone** pane, enter a domain name and, optionally, a comment.

   For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.
6. In the **Type** list, choose **Private hosted zone**.
7. In the **VPC ID** list, choose the VPC that you want to associate with the hosted zone.

> **Note**
>
> If the console displays the following message, you're trying to associate a hosted zone that uses the same name space as that of another hosted zone within the same VPC:

"A conflicting domain is already associated with the given VPC or Delegation Set." For example, if hosted zone A and hosted zone B both have the same domain name, such as `example.com`, you can't associate both hosted zones with the same VPC.

8. Choose **Create hosted zone**.

Listing private hosted zones

You can use the Amazon Route 53 console to list all of the hosted zones that you created with the current AWS account. For information about how to list hosted zones using the Route 53 API, see ListHostedZones in the *Amazon Route 53 API Reference*.

To list the hosted zones associated with an AWS account

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.

   The **Hosted Zones** page automatically displays a list of all of the hosted zones that were created using the current AWS account. The **Type** column indicates whether a hosted zone is private or public. Choose the column heading to group all private hosted zones and all public hosted zones.

Associating more VPCs with a private hosted zone

You can use the Amazon Route 53 console to associate more VPCs with a private hosted zone if you created the hosted zone and the VPCs by using the same AWS account.

Important

If you want to associate VPCs that you created by using one account with a private hosted zone that you created by using a different account, you first must authorize the association. In addition, you can't use the AWS console either to authorize the association or associate the VPCs with the hosted zone. For more information, see Associating an Amazon VPC and a private hosted zone that you created with different AWS accounts.

For information about how to associate more VPCs with a private hosted zone using the Route 53 API, see AssociateVPCWithHostedZone in the *Amazon Route 53 API Reference*.

To associate additional VPCs with a private hosted zone using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the radio button for the private hosted zone that you want to associate more VPCs with.
4. Choose **Edit**.
5. Choose **Add VPC**.
6. Choose the Region and the ID of the VPC that you want to associate with this hosted zone.
7. To associate more VPCs with this hosted zone, repeat steps 5 and 6.
8. Choose **Save changes**.

Associating an Amazon VPC and a private hosted zone that you created with different AWS accounts

If you want to associate a VPC that you created with one AWS account with a private hosted zone that you created with a different account, perform the following procedure:

To associate an Amazon VPC and a private hosted zone that you created with different AWS accounts

1. Using the account that created the hosted zone, authorize the association of the VPC with the private hosted zone by using one of the following methods:

   - **AWS CLI** – See create-vpc-association-authorization in the *AWS CLI Command Reference*
   - **AWSSDK** or **AWS Tools for Windows PowerShell** – See the applicable documentation on the AWSDocumentation page
   - **Amazon Route 53 API** – See CreateVPCAssociationAuthorization in the *Amazon Route 53 API Reference*

   Note the following:

- If you want to associate multiple VPCs that you created with one account with a hosted zone that you created with a different account, you must submit one authorization request for each VPC.
- When you authorize the association, you must specify the hosted zone ID, so the private hosted zone must already exist.
- You can't use the Route 53 console either to authorize the association of a VPC with a private hosted zone or to make the association.

2. Using the account that created the VPC, associate the VPC with the hosted zone. As with authorizing the association, you can use the AWS SDK, Tools for Windows PowerShell, the AWS CLI, or the Route 53 API. If you're using the API, use the AssociateVPCWithHostedZone action.
3. *Recommended* – Delete the authorization to associate the VPC with the hosted zone. Deleting the authorization does not affect the association, it just prevents you from reassociating the VPC with the hosted zone in the future. If you want to reassociate the VPC with the hosted zone, you'll need to repeat steps 1 and 2 of this procedure.

> **Important**
>
> The `ListHostedZonesByVPC` returns the hosted zones given a VPC and `GetHostedZone` API returns the VPCs associated to the hosted zone. These APIs only consider the hosted zone to VPC association that are created by `AssociateVPCWithHostedZone` API or when the private hosted zone is created. If you want a complete list of hosted zone associations to a VPC, also call ListProfileResourceAssociations.

> **Note**
>
> For the maximum number of authorizations that you can create, see Quotas on entities.

Disassociating VPCs from a private hosted zone

You can use the Amazon Route 53 console to disassociate VPCs from a private hosted zone. This causes Route 53 to stop routing traffic using records in the hosted zone for DNS queries that originate in the VPC. For example, if the example.com hosted zone is associated with a VPC

and you disassociate the hosted zone from that VPC, Route 53 stops resolving DNS queries for example.com or any of the other records in the example.com hosted zone.

> **Note**
>
> You can't disassociate the last VPC from a private hosted zone. If you want to disassociate that VPC, you must first associate another VPC with the hosted zone.

To disassociate VPCs from a private hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the radio button for the private hosted zone that you want to disassociate one or more VPCs from.
4. Choose **Edit**.
5. Choose **Remove VPC** next to the VPC that you want to disassociate from this hosted zone.
6. Choose **Save changes**.

Deleting a private hosted zone

This section explains how to delete a private hosted zone using the Amazon Route 53 console.

You can delete a private hosted zone only if there are no records other than the default SOA and NS records. If your hosted zone contains other records, you must delete them before you can delete your hosted zone. This prevents you from accidentally deleting a hosted zone that still contains records.

Topics

- Deleting private hosted zones that were created by another service
- Using the Route 53 console to delete a private hosted zone

Deleting private hosted zones that were created by another service

If a private hosted zone was created by another service, you can't delete it using the Route 53 console. Instead, you need to use the applicable process for the other service:

- **AWS Cloud Map** – To delete a hosted zone that AWS Cloud Map created when you created a private DNS namespace, delete the namespace. AWS Cloud Map deletes the hosted zone automatically. For more information, see Deleting namespaces in the *AWS Cloud Map Developer Guide*.
- **Amazon Elastic Container Service (Amazon ECS) Service Discovery** – To delete a private hosted zone that Amazon ECS created when you created a service using service discovery, delete the Amazon ECS services that are using the namespace, and delete the namespace. For more information, see Deleting a service in the *Amazon Elastic Container Service Developer Guide*.

Using the Route 53 console to delete a private hosted zone

To use the Route 53 console to delete a private hosted zone, perform the following procedure.

To delete a private hosted zone using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. Confirm that the hosted zone that you want to delete contains only an NS and an SOA record. If it contains additional records, delete them:

   a. Choose the name of the hosted zone that you want to delete.

   b. On the **Record** page, if the list of records includes any records for which the value of the **Type** column is something other than **NS** or **SOA**, choose the row, and choose **Delete**.

      To select multiple, consecutive records, choose the first row, press and hold the **Shift** key, and choose the last row. To select multiple, non-consecutive records, choose the first row, press and hold the **Ctrl** key, and choose the remaining rows.
3. On the Hosted Zones page, choose the row for the hosted zone that you want to delete.
4. Choose **Delete**.
5. Type the confirmation key and choose **Delete**.

VPC permissions

VPC permissions use Identity and Access management (IAM) policy condition to allow you to set granular permissions for VPCs when using AssociateVPCWithHostedZone, DisassociateVPCFromHostedZone, CreateVPCAssociationAuthorization, DeleteVPCAssociationAuthorization, CreateHostedZone, and ListHostedZonesByVPC APIs.

With the IAM policy condition, `route53:VPCs`, you can grant granular administrative rights to other AWS users. This allows you to grant someone permissions to associate hosted zone with, disassociate hosted zone from, create VPC association authorization for, delete VPC association authorization for, create hosted zone with or list hosted zones for:

- A single VPC.
- Any VPCs within the same Region.
- Multiple VPCs.

For more information about VPC permissions, see Using IAM policy conditions for fine-grained access control.

To learn how to authenticate AWS users, see Authenticating with identities and to learn how to control access to Route 53 resources, see Access control.

Hosted zone tiers

A *hosted zone tier* is a bundle of features that Route 53 offers together. Each public or private hosted zone can only have one tier at a time.

Route 53 offers the following tiers:

- `STANDARD` – Includes standard Change Guardrails and Accelerated Recovery. The default tier, at no additional cost.
- `ADVANCED` – Includes all `STANDARD` features plus additional features, at additional cost. For more information, see Advanced hosted zones.

You can choose the tier when you create a hosted zone or change it at any time. For more information about features that each tier offers, see Hosted zone features.

> **Note**
>
> Changing the tier doesn't affect the records in a hosted zone or how Route 53 responds to DNS queries. Features that affect DNS responses, such as Public DNS Fallback, take effect only after you turn them on.

Topics

- Advanced hosted zones
- Choosing a tier when you create a hosted zone
- Changing the tier of a hosted zone
- Viewing the tier of a hosted zone
- Considerations for hosted zone tiers

Advanced hosted zones

A hosted zone that uses the ADVANCED tier is an *advanced hosted zone*. Advanced hosted zones help you prevent and investigate DNS changes that can take your applications offline. You can create a hosted zone with the ADVANCED tier, or change an existing public or private hosted zone to it.

Advanced hosted zones add the following features:

- Change Guardrails that block deleting records and delegations that are still in use (ALIAS_TARGET_CHECK, ACTIVE_RECORD_CHECK, and ACTIVE_DELEGATION_CHECK)
- Change History that tracks who changed the records in the hosted zone, what they changed, and when
- Public DNS Fallback that resolves queries against public DNS when a name doesn't exist in a private hosted zone

Choosing a tier when you create a hosted zone

You can specify the tier for any hosted zone. For general information about creating hosted zones, see Creating a public hosted zone and Creating a private hosted zone.

Using the Route 53 console

To choose a tier when you create a hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose **Create hosted zone**.
4. Enter the settings for your public or private hosted zone.

5. In the tier settings, select the tier for the hosted zone.
6. Choose **Create hosted zone**.

Using the AWS CLI

To choose a tier when you create a hosted zone

- Run the `create-hosted-zone` command. The following AWS Command Line Interface (AWS CLI) command creates a public hosted zone that uses the ADVANCED tier.

```
aws route53 create-hosted-zone \
    --name example.org \
    --caller-reference 2026-12-01-example \
    --hosted-zone-config HostedZoneTier=ADVANCED
```

Using an AWS SDK

To choose a tier when you create a hosted zone

- Call the CreateHostedZone API, and set `HostedZoneTier` in `HostedZoneConfig`. The following AWS SDK for Python (Boto3) example creates a public hosted zone that uses the ADVANCED tier. Replace *example.org* with your domain name.

```
import time
import boto3

route53 = boto3.client("route53")

route53.create_hosted_zone(
    Name="example.org",
    CallerReference=str(time.time()),
    HostedZoneConfig={"HostedZoneTier": "ADVANCED"},
)
```

Using CloudFormation

To choose a tier when you create a hosted zone

- In your template, set the `HostedZoneTier` property in `HostedZoneConfig`. The following example creates a public hosted zone that uses the `ADVANCED` tier.

```
Resources:
  ExampleHostedZone:
    Type: AWS::Route53::HostedZone
    Properties:
      Name: example.org
      HostedZoneConfig:
        HostedZoneTier: ADVANCED
```

  For more information, see AWS::Route53::HostedZone in the *CloudFormation Template Reference Guide*.

Changing the tier of a hosted zone

You can change an existing hosted zone to any tier.

When you change a hosted zone from `ADVANCED` to `STANDARD`:

- Before the change, you must turn off the `ADVANCED`-only optional features: `ALIAS_TARGET_CHECK`, `ACTIVE_RECORD_CHECK`, `ACTIVE_DELEGATION_CHECK`, and Public DNS Fallback. Otherwise the request fails.
- Route 53 turns off Change History. You can no longer read the existing change history.
- If you later change back to `ADVANCED`, Change History starts a new history. It doesn't restore the old one.
- `NS_FORMAT_CHECK` and `NS_DELEGATION_AUTHORITY_CHECK` stay on. `NS_OCCLUSION_CHECK` and Accelerated Recovery keep their current settings.
- Route 53 stops the additional charges for the `ADVANCED` tier.
- Records and DNS query responses don't change.

Before you change the tier, review Considerations for hosted zone tiers.

Using the Route 53 console

To change the tier of a hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the hosted zone.
4. Choose the option to change the tier.
5. Select the new tier.
6. When prompted, confirm the change.

Using the AWS CLI

To change the tier of a hosted zone

- Run the `update-hosted-zone-tier` command, and set `--hosted-zone-tier` to the new tier. The following AWS CLI command changes a hosted zone to the ADVANCED tier. Replace *Z111111122222222333333* with the ID of your hosted zone.

  ```
  aws route53 update-hosted-zone-tier \
      --hosted-zone-id Z111111122222222333333 \
      --hosted-zone-tier ADVANCED
  ```

  If the command succeeds, it returns no output. To validate the request without changing the tier, add `--dry-run`.

Using an AWS SDK

To change the tier of a hosted zone

- Call the UpdateHostedZoneTier API. The following AWS SDK for Python (Boto3) example changes a hosted zone to the ADVANCED tier. Replace *Z111111122222222333333* with the ID of your hosted zone.

  ```
  import boto3

  route53 = boto3.client("route53")
  ```

```
route53.update_hosted_zone_tier(
    HostedZoneId="Z11111112222222333333",
    HostedZoneTier="ADVANCED",
)
```

To validate the request without changing the tier, add `DryRun=True`.

Using CloudFormation

To change the tier of a hosted zone

- In your template, change the value of `HostedZoneTier` in `HostedZoneConfig`, and then update the stack. The following example changes the hosted zone to the `ADVANCED` tier.

```
      HostedZoneConfig:
        HostedZoneTier: ADVANCED
```

Viewing the tier of a hosted zone

Using the Route 53 console

To view the tier of a hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the hosted zone.
4. In the hosted zone details, review the tier.

Using the AWS CLI

To view the tier of a hosted zone

- Run the `get-hosted-zone` command with the `--query` option to return only the tier. Replace *Z11111112222222333333* with the ID of your hosted zone.

```
aws route53 get-hosted-zone --id Z11111112222222333333 --query
 'HostedZone.Config.HostedZoneTier'
```

The output looks similar to the following.

```
"ADVANCED"
```

Using an AWS SDK

To view the tier of a hosted zone

- Call the GetHostedZone API. The tier is in `HostedZone.Config.HostedZoneTier`. The following AWS SDK for Python (Boto3) example prints the tier. Replace *Z11111112222222333333* with the ID of your hosted zone.

```
import boto3

route53 = boto3.client("route53")

response = route53.get_hosted_zone(Id="Z11111112222222333333")
print(response["HostedZone"]["Config"]["HostedZoneTier"])
```

Considerations for hosted zone tiers

Before you choose or change a tier, review the following:

- Route 53 prorates the additional charges for the ADVANCED tier. For example, if a hosted zone uses the ADVANCED tier for half of a month, you pay about half of the monthly ADVANCED charge for that month. For more information, see Amazon Route 53 pricing.
- You can't change the tier of a hosted zone that's associated with any Amazon CloudFront flat-rate pricing plan, including paid plans. To change the tier, first disassociate the hosted zone from the plan. For more information, see CloudFront flat-rate plans in the *Amazon CloudFront Developer Guide*.

Hosted zone features

A *hosted zone feature* is a capability of a hosted zone, such as Change Guardrails or Change History. A *hosted zone tier* is a bundle of features. Each tier includes different features, and some features apply only to public or only to private hosted zones. For more information about tiers, see Hosted zone tiers.

Topics

- Available hosted zone features
- Updating optional features
- Viewing the features of a hosted zone
- Viewing the change history of a hosted zone
- Change Guardrails for hosted zones
- Public DNS fallback for private hosted zones
- Enabling accelerated recovery for managing public DNS records

Available hosted zone features

The following table lists each feature and what it does. It also shows the tiers that offer the feature, whether it's on by default, and the hosted zone types it applies to.

Hosted zone features by tier

Feature	Description	Tiers	Default	Hosted zone types
Change History	Tracks who changed the records in the hosted zone, what they changed, and when.	ADVANCED	On; can't be turned off	Public and private
Change Guardrails – NS_FORMAT_CHECK	Rejects an NS record whose value looks like a Route 53 name server but isn't one.	STANDARD, ADVANCED	On; can't be turned off	Public and private

Feature	Description	Tiers	Default	Hosted zone types
Change Guardrails – NS_DELEGATION_AUTHORITY_CHECK	Rejects an NS record that delegates a subdomain to Route 53 name servers that aren't authoritative for it.	STANDARD, ADVANCED	On; can't be turned off	Public
Change Guardrails – NS_OCCLUSION_CHECK	Rejects a record that a delegation would hide, or a delegation that would hide existing records.	STANDARD, ADVANCED	Off	Public and private
Change Guardrails – ALIAS_TARGET_CHECK	Rejects deleting a record that an alias record points to.	ADVANCED	Off	Public and private
Change Guardrails – ACTIVE_RECORD_CHECK	Rejects deleting a record that still receives DNS queries.	ADVANCED	Off	Public and private
Change Guardrails – ACTIVE_DELEGATION_CHECK	Rejects deleting a delegation that resolvers still use.	ADVANCED	Off	Public and private
Public DNS Fallback	Falls back to public DNS when the name doesn't exist in the private hosted zone.	ADVANCED	Off	Private

Feature	Description	Tiers	Default	Hosted zone types
Accelerated Recovery	Lets you resume making DNS changes within about 60 minutes if the US East (N. Virginia) Region is impaired.	STANDARD, ADVANCED	Off	Public

Updating optional features

You can turn optional features on or off with any of the following methods. You can also turn on optional features when you create a hosted zone. To do so, pass the same options to the `create-hosted-zone` command or the CreateHostedZone API.

Using the Route 53 console

To turn optional features on or off

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the hosted zone.
4. Choose **Edit features**.
5. Turn each optional feature on or off.
6. Choose **Save changes**.

Using the AWS CLI

To turn optional features on or off

- Run the `update-hosted-zone-features` command. Replace *Z1111111222222333333* with the ID of your hosted zone. The following example turns on the `ACTIVE_RECORD_CHECK` guardrail. To turn it off, set `Status` to `DISABLED`.

```
aws route53 update-hosted-zone-features \
    --hosted-zone-id Z1111111222222333333 \
```

```
    --change-guardrails Name=ACTIVE_RECORD_CHECK,Status=ENABLED
```

If the command succeeds, it returns no output. To validate the request without changing any features, add `--dry-run`.

Using an AWS SDK

To turn optional features on or off with an AWS SDK, call the UpdateHostedZoneFeatures API. The following AWS SDK for Python (Boto3) example turns on the `ACTIVE_RECORD_CHECK` guardrail. Replace *Z111111122222222333333* with the ID of your hosted zone.

```
import boto3

route53 = boto3.client("route53")

route53.update_hosted_zone_features(
    HostedZoneId="Z111111122222222333333",
    ChangeGuardrails=[
        {"Name": "ACTIVE_RECORD_CHECK", "Status": "ENABLED"}
    ],
)
```

To validate the request without changing any features, add `DryRun=True`.

Using CloudFormation

To turn on optional features in CloudFormation, use the `HostedZoneFeatures` property. The following example template creates an `ADVANCED` tier hosted zone. It also turns on the `ACTIVE_RECORD_CHECK` guardrail.

```
Resources:
  ExampleHostedZone:
    Type: AWS::Route53::HostedZone
    Properties:
      Name: example.org
      HostedZoneConfig:
        HostedZoneTier: ADVANCED
      HostedZoneFeatures:
        ChangeGuardrails:
          - Name: ACTIVE_RECORD_CHECK
            Status: ENABLED
```

In the `HostedZoneFeatures` property, you can also set `EnableAcceleratedRecovery` and `EnablePublicDNSFallback`. You can declare only the optional guardrails. For more information, see AWS::Route53::HostedZone in the *CloudFormation Template Reference Guide*.

For instructions on turning on specific features, see the following topics:

- Enabling and disabling guardrails for a hosted zone
- Enabling and disabling public DNS fallback
- How to enable accelerated recovery for managing public DNS records

Viewing the features of a hosted zone

Using the Route 53 console

To view the features of a hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the hosted zone.
4. In the hosted zone details, review the status of each feature.

Using the AWS CLI

To view the features of a hosted zone

- Run the `get-hosted-zone` command. Replace *Z11111112222222333333* with the ID of your hosted zone.

  ```
  aws route53 get-hosted-zone --id Z11111112222222333333
  ```

  The output looks similar to the following.

  ```
  {
      "HostedZone": {
          "Id": "/hostedzone/Z11111112222222333333",
          "Name": "example.org.",
          "CallerReference": "2026-12-01-example",
  ```

```
        "Config": {
            "PrivateZone": false,
            "HostedZoneTier": "ADVANCED"
        },
        "ResourceRecordSetCount": 2,
        "Features": {
            "ChangeHistory": {
                "Status": "ENABLED",
                "RetentionStartsAt": "2026-12-01T17:48:16.751Z"
            },
            "ChangeGuardrails": [
                {
                    "Name": "NS_FORMAT_CHECK",
                    "Status": "ENABLED"
                },
                {
                    "Name": "NS_DELEGATION_AUTHORITY_CHECK",
                    "Status": "ENABLED"
                },
                {
                    "Name": "NS_OCCLUSION_CHECK",
                    "Status": "DISABLED"
                },
                {
                    "Name": "ALIAS_TARGET_CHECK",
                    "Status": "DISABLED"
                },
                {
                    "Name": "ACTIVE_RECORD_CHECK",
                    "Status": "ENABLED"
                },
                {
                    "Name": "ACTIVE_DELEGATION_CHECK",
                    "Status": "DISABLED"
                }
            ]
        }
    },
    "DelegationSet": {
        "NameServers": [
            "ns-111.awsdns-11.com",
            "ns-222.awsdns-22.net",
            "ns-1333.awsdns-33.org",
            "ns-1444.awsdns-44.co.uk"
```

```
        ]
    }
}
```

In the output, `HostedZone.Features` shows the status of each feature. For a `STANDARD` tier hosted zone, Change History and the features that only `ADVANCED` offers show `DISABLED`.

Using an AWS SDK

To view the features of a hosted zone

- Call the GetHostedZone API. The following AWS SDK for Python (Boto3) example prints the features of a hosted zone. Replace *Z111111122222222333333* with the ID of your hosted zone.

```
import boto3

route53 = boto3.client("route53")

response = route53.get_hosted_zone(Id="Z111111122222222333333")
print(response["HostedZone"]["Features"])
```

The `Features` object shows the status of each feature.

Viewing the change history of a hosted zone

Change History shows the changes to the DNS records in a hosted zone. You can use it to answer questions such as the following:

- Who changed the records in this hosted zone last week?
- When did the value of `www.example.com` change, and who changed it?
- Which changes has the hosted zone received since the last change that my system processed?

Change History is available only for hosted zones that use the `ADVANCED` tier, and it's enabled by default for those hosted zones. For more information about tiers, see Hosted zone tiers.

Topics

- How Change History works

- Understanding how the tier affects Change History
- Viewing changes
- Change History compared with CloudTrail
- Reading new changes into another system
- Permissions for Change History

How Change History works

Route 53 records each change batch that it applies to the hosted zone, which is the set of changes that Route 53 records for one request, such as a `ChangeResourceRecordSets` request, and retains it for 365 days. For each change batch, Change History shows who submitted it, when it was submitted, and which records changed, in the same format that `ListResourceRecordSets` returns. For an UPSERT, Change History shows the new values, so to see the earlier values, look at the previous change to the same record.

Each change batch includes the following information:

`ChangeId`

The ID of the change batch. This is the same ID that `ChangeResourceRecordSets` returns in `ChangeInfo/Id` and that you pass to `GetChange`.

`Operation`

The Route 53 API operation that submitted the change batch. Valid values: `ChangeResourceRecordSets` | `CreateHostedZone`. A hosted zone that you create on the `ADVANCED` tier starts its history with a `CreateHostedZone` entry.

`SubmittedAt`

The date and time, in UTC, when the change batch was submitted. Route 53 lists change batches in the order in which it applied them, so `SubmittedAt` values aren't always in strict order.

`SubmittedBy`

The ARN of the principal that submitted the request, for example `arn:aws:sts::111122223333:assumed-role/Admin/my-session`.

`RequestId`

The request ID of the call that submitted the change batch.

`Comment`

The comment from the `ChangeResourceRecordSets` request, if any.

`ChangeCount`

The number of changes in the change batch. For `CreateHostedZone`, the value is 0, because the entry doesn't list the name server (NS) and start of authority (SOA) records that Route 53 creates with the hosted zone.

Change History shows only changes to records. It doesn't show changes to the hosted zone itself, such as its comment, tier, features, or VPC associations, or changes to other resources, such as health checks, CIDR collections, DNSSEC signing, and query logging configurations. To audit those changes, use CloudTrail. For more information, see Change History compared with CloudTrail.

When you delete a hosted zone, you can no longer read its change history. If you need the history, read it and store it before you delete the hosted zone.

Understanding how the tier affects Change History

Route 53 records changes whenever a hosted zone uses the `ADVANCED` tier. `RetentionStartsAt` is the earliest time that you can read changes from:

- When you create a hosted zone on the `ADVANCED` tier, `RetentionStartsAt` is the time when you created the hosted zone.
- When you upgrade a hosted zone to `ADVANCED`, `RetentionStartsAt` is the time of the upgrade. Changes from before the upgrade aren't available.

Warning

When you change a hosted zone from `ADVANCED` to `STANDARD`, you permanently lose access to its change history. If you upgrade the hosted zone to `ADVANCED` again, even right away, Route 53 starts a new history from the time of the upgrade and doesn't restore the earlier changes. Before you downgrade, read and store any history that you need.

To change the tier of a hosted zone, see Changing the tier of a hosted zone.

To see `RetentionStartsAt`, call GetHostedZone. The following AWS Command Line Interface (AWS CLI) command shows only the Change History element of the response:

```
aws route53 get-hosted-zone \
    --id Z1111111222222233333 \
    --query 'HostedZone.Features.ChangeHistory'
```

Example output:

```
{
    "Status": "ENABLED",
    "RetentionStartsAt": "2026-11-20T16:05:31.214000+00:00"
}
```

For a hosted zone that uses the STANDARD tier, `Status` is DISABLED and the response doesn't include `RetentionStartsAt`.

Viewing changes

You can view the change history of a hosted zone in the Route 53 console, with the AWS CLI, or with an AWS SDK.

Using the Route 53 console

To view the change history of a hosted zone

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of an advanced hosted zone.
4. Choose the **Change history** tab.

   The tab lists the change batches in the hosted zone, newest first.
5. (Optional) To show only some changes, do one or both of the following:

   a. To show the changes in a time range, select a time range.

   b. To show the changes to one record, enter its record name and select its type. The list then shows one row for each change to records with that name and type.
6. To view the changes in a batch, select it. The details panel shows each change in the batch, its action, and the record values.

Using the AWS CLI

Use the AWS CLI commands for the following actions:

- ListHostedZoneChangeBatches lists the change batches in a hosted zone, with a summary of each batch.
- GetHostedZoneChangeBatch returns one change batch, including every change in it.
- ListResourceRecordSetChanges lists the individual changes in a hosted zone, optionally only the changes to the records that have a specific name and type.

For the request parameters, such as the time range, sort order, and paging, see each action's page in the Amazon Route 53 API Reference. The AWS CLI follows `NextToken` for you and returns every page. To control paging, use `--page-size`, `--max-items`, and `--starting-token`. In the following examples, replace *Z1111111222222233333* with the ID of your hosted zone.

To list the change batches in a hosted zone

- The following AWS CLI command lists the change batches that were submitted to the hosted zone on December 1, 2026, newest first.

```
aws route53 list-hosted-zone-change-batches \
    --hosted-zone-id Z1111111222222233333 \
    --submitted-after 2026-12-01T00:00:00Z \
    --submitted-before 2026-12-02T00:00:00Z
```

Example output:

```
{
    "ChangeBatchSummaries": [
        {
            "ChangeId": "C2682N5HXP0BZ4",
            "Operation": "ChangeResourceRecordSets",
            "SubmittedAt": "2026-12-01T17:42:10.123000+00:00",
            "SubmittedBy": "arn:aws:sts::111122223333:assumed-role/Admin/my-
session",
            "RequestId": "b25f48e8-84fd-11e6-80d9-574EXAMPLE11",
            "Comment": "Move www to the new load balancer",
            "ChangeCount": 2
        },
        {
```

```
            "ChangeId": "C1PA6795UKMFR9",
            "Operation": "ChangeResourceRecordSets",
            "SubmittedAt": "2026-12-01T09:15:47.508000+00:00",
            "SubmittedBy": "arn:aws:sts::111122223333:assumed-role/Deploy/my-
pipeline-session",
            "RequestId": "0f9a8c2e-3b1d-4a5e-9c7f-2d6EXAMPLE22",
            "ChangeCount": 1
        }
    ]
}
```

To get the changes in a change batch

- The following AWS CLI command returns the change batch with the ID C2682N5HXP0BZ4. You can specify the ID with or without the /change/ prefix that ChangeResourceRecordSets returns.

```
aws route53 get-hosted-zone-change-batch \
    --hosted-zone-id Z1111111222222233333 \
    --change-id C2682N5HXP0BZ4
```

Example output:

```
{
    "ChangeBatchSummary": {
        "ChangeId": "C2682N5HXP0BZ4",
        "Operation": "ChangeResourceRecordSets",
        "SubmittedAt": "2026-12-01T17:42:10.123000+00:00",
        "SubmittedBy": "arn:aws:sts::111122223333:assumed-role/Admin/my-session",
        "RequestId": "b25f48e8-84fd-11e6-80d9-574EXAMPLE11",
        "Comment": "Move www to the new load balancer",
        "ChangeCount": 2
    },
    "ChangeBatch": {
        "Comment": "Move www to the new load balancer",
        "Changes": [
            {
                "Action": "UPSERT",
                "ResourceRecordSet": {
                    "Name": "www.example.com.",
                    "Type": "A",
```

```
                    "TTL": 60,
                    "ResourceRecords": [
                        {
                            "Value": "10.0.1.44"
                        }
                    ]
                }
            },
            {
                "Action": "DELETE",
                "ResourceRecordSet": {
                    "Name": "old.example.com.",
                    "Type": "A",
                    "TTL": 300,
                    "ResourceRecords": [
                        {
                            "Value": "10.0.2.7"
                        }
                    ]
                }
            }
        ]
    }
}
```

To list the changes to one record

- The following AWS CLI command lists the changes to the A records for `www.example.com`, newest first.

```
aws route53 list-resource-record-set-changes \
    --hosted-zone-id Z1111111222222233333 \
    --record-name www.example.com \
    --record-type A
```

Example output:

```
{
    "ResourceRecordSetChanges": [
        {
            "ChangeId": "C2682N5HXP0BZ4",
```

```
            "Operation": "ChangeResourceRecordSets",
            "SubmittedAt": "2026-12-01T17:42:10.123000+00:00",
            "SubmittedBy": "arn:aws:sts::111122223333:assumed-role/Admin/my-
session",
            "RequestId": "b25f48e8-84fd-11e6-80d9-574EXAMPLE11",
            "Action": "UPSERT",
            "ResourceRecordSet": {
                "Name": "www.example.com.",
                "Type": "A",
                "TTL": 60,
                "ResourceRecords": [
                    {
                        "Value": "10.0.1.44"
                    }
                ]
            }
        },
        {
            "ChangeId": "C0X4HN2R6YPLA7",
            "Operation": "ChangeResourceRecordSets",
            "SubmittedAt": "2026-11-30T21:15:03.402000+00:00",
            "SubmittedBy": "arn:aws:sts::111122223333:assumed-role/Admin/my-
session",
            "RequestId": "e4b8d2a6-0f71-4c39-a5e2-6d9EXAMPLE44",
            "Action": "UPSERT",
            "ResourceRecordSet": {
                "Name": "www.example.com.",
                "Type": "A",
                "TTL": 60,
                "ResourceRecords": [
                    {
                        "Value": "10.0.1.10"
                    }
                ]
            }
        },
        {
            "ChangeId": "C3QW7VKZ8M1TJ2",
            "Operation": "ChangeResourceRecordSets",
            "SubmittedAt": "2026-11-24T10:02:11.870000+00:00",
            "SubmittedBy": "arn:aws:sts::111122223333:assumed-role/Deploy/my-
pipeline-session",
            "RequestId": "7c1e9f04-5a2b-4d8e-b3f6-91aEXAMPLE33",
            "Action": "CREATE",
```

```
            "ResourceRecordSet": {
                "Name": "www.example.com.",
                "Type": "A",
                "TTL": 300,
                "ResourceRecords": [
                    {
                        "Value": "10.0.1.10"
                    }
                ]
            }
        }
    ]
}
```

Each entry shows one change, with the full record that the change wrote or deleted. Weighted, latency, and other records that have the same name and type appear as separate entries, each with its `SetIdentifier`.

Using an AWS SDK

Call the ListHostedZoneChangeBatches, GetHostedZoneChangeBatch, and ListResourceRecordSetChanges APIs. The following AWS SDK for Python (Boto3) example prints each change that was submitted to the hosted zone on December 1, 2026, with who submitted it. Replace *Z111111122222222333333* with the ID of your hosted zone.

```
import boto3

route53 = boto3.client("route53")
zone_id = "Z111111122222222333333"

paginator = route53.get_paginator("list_hosted_zone_change_batches")
for page in paginator.paginate(
    HostedZoneId=zone_id,
    SubmittedAfter="2026-12-01T00:00:00Z",
    SubmittedBefore="2026-12-02T00:00:00Z",
):
    for summary in page["ChangeBatchSummaries"]:
        batch = route53.get_hosted_zone_change_batch(
            HostedZoneId=zone_id, ChangeId=summary["ChangeId"]
        )
        for change in batch["ChangeBatch"]["Changes"]:
            record = change["ResourceRecordSet"]
```

```
            print(summary["SubmittedAt"], summary.get("SubmittedBy"),
                  change["Action"], record["Name"], record["Type"])
```

Change History compared with CloudTrail

Change History and CloudTrail both show who changed a hosted zone, but they answer different questions. Use Change History to see what changed in the records of a hosted zone. Use CloudTrail to see every call to the Route 53 API in your account. The following table compares them.

	Change History	CloudTrail
What it records	Each change batch that Route 53 applied to the records in a hosted zone, with the full record that each change wrote or deleted.	Calls to Route 53 API actions, including read calls, calls that fail, and changes to resources other than records, such as health checks and VPC associations.
How to find the changes to one record	Call `ListResourceRecordSetChanges` with the record name and type. The results include only changes that Route 53 applied.	Search the `ChangeResourceRecordSets` events for the hosted zone, and read the changes in `requestParameters.changeBatch`. Each event shows the request as it was sent, including requests that Route 53 rejected.
How long it's available	365 days after each change batch was submitted. Changes from before `RetentionStartsAt` aren't available.	The last 90 days in **Event history**. To keep events longer, create a trail that delivers them to an Amazon S3 bucket.
Setup	Enabled by default for every advanced hosted zone.	**Event history** is on for every AWS account. A trail must be created.
Reading new changes into another system	Read the change batches in the order that Route 53 applied them, and resume after the last change batch that you processed with `StartAfterChangeId`. For	Process the events that the trail delivers.

	Change History	CloudTrail
	more information, see Reading new changes into another system.	

To find the CloudTrail event for a change batch, match the change batch's `RequestId` to the event's `requestID`, or its `ChangeId` to `responseElements.changeInfo.id`. The event includes details that Change History doesn't, such as the IP address that the request came from.

For more information about CloudTrail, see Logging Amazon Route 53 API calls with AWS CloudTrail.

Reading new changes into another system

To keep a copy of a hosted zone in another system, such as an inventory or an audit store, read the records once and then read new change batches as they arrive:

1. Find the starting point. Call `ListHostedZoneChangeBatches` with `MaxResults` set to 1 and the default `SortOrder` (`DESCENDING`), and save the `ChangeId` of the newest change batch. If the response has no change batches, leave out `StartAfterChangeId` when you list new changes.
2. Read the current records with ListResourceRecordSets.
3. Call `ListHostedZoneChangeBatches` with `SortOrder` set to `ASCENDING` and `StartAfterChangeId` set to the saved `ChangeId`. Follow `NextToken` until the response doesn't include it.
4. For each change batch, call `GetHostedZoneChangeBatch` to get its changes, apply them, and save its `ChangeId`. Changes made while you read the records can already be in your copy, so apply each change so that applying it again has no effect.
5. To check for new changes later, repeat the previous two steps, starting after the saved `ChangeId`.

Save the last `ChangeId` that you processed rather than a `NextToken`, because a `NextToken` expires after 24 hours. If `StartAfterChangeId` returns an `InvalidInput` error because the change batch expired or the hosted zone was downgraded to `STANDARD` and upgraded again, read the records again with `ListResourceRecordSets` and start over.

Permissions for Change History

The following table lists each Change History API operation and the IAM actions that you must allow in the `Action` element of a policy statement to call it. A resource type marked with an asterisk (*) is required. The `hostedzone` resource type has the ARN `arn:aws:route53:::hostedzone/`*`Id`*. These actions don't support service-specific condition keys.

IAM actions for Change History

API operation	Description	IAM actions	Access level	Resource types (*required)
ListHostedZoneChangeBatches	Grants permission to get a list of the change batches in a specified hosted zone's change history	route53:ListHostedZoneChangeBatches	List	hostedzone*
GetHostedZoneChangeBatch	Grants permission to get the records that a specified change batch created, updated, or deleted in a hosted zone	route53:GetHostedZoneChangeBatch route53:ListResourceRecordSets	List	hostedzone*
ListResourceRecordSetChanges	Grants permission to list the changes to a specified record in a hosted zone's change history	route53:GetHostedZoneChangeBatch route53:ListResourceRecordSets	List	hostedzone*

`ListResourceRecordSetChanges` doesn't have an IAM action of its own.

Change History shows the ARN of the principal that submitted each change, including role session names. Grant these actions only to principals that can see that information. For an example policy, see Example 6: Allow reading the change history of a hosted zone.

Change Guardrails for hosted zones

Amazon Route 53 Change Guardrails are safety checks that stop a change to the records in a hosted zone when the change would take part of your domain offline. Before Route 53 applies a change, it runs each guardrail that is on for the hosted zone. If a guardrail finds a problem, Route 53 rejects the whole request with an `InvalidChangeBatch` error, applies none of its changes, and tells you which record and which guardrail caused the rejection and how to fix it.

Route 53 provides six guardrails. Two are always on. You turn the other four on for each hosted zone, and three of them are available only for hosted zones that use the `ADVANCED` tier. Guardrails apply to the changes that you make with the Route 53 console, the `ChangeResourceRecordSets` API, the AWS SDKs, the AWS CLI, and CloudFormation. The following table lists the six guardrails.

Guardrail	What it rejects	Default	Tier	Hosted zones
Name server format check (`NS_FORMAT_CHECK`)	An NS record value that looks like a Route 53 name server but isn't one.	Always on	`STANDARD` and `ADVANCED`	Public and private
Delegation authority check (`NS_DELEGATION_AUTHORITY_CHECK`)	An NS record that delegates a subdomain to Route 53 name servers that aren't authoritative for the subdomain.	Always on	`STANDARD` and `ADVANCED`	Public only
Name server occlusion check (`NS_OCCLUSION_CHECK`)	A record that an existing delegation hides from resolvers, or a delegation that would hide existing records.	Off	`STANDARD` and `ADVANCED`	Public and private

Guardrail	What it rejects	Default	Tier	Hosted zones
Alias target check (`ALIAS_TARGET_CHECK`)	Deleting a record that an alias record points to.	Off	`ADVANCED` only	Public and private
Active record check (`ACTIVE_RECORD_CHECK`)	Deleting a record that still receives DNS queries.	Off	`ADVANCED` only	Public and private
Active delegation check (`ACTIVE_DELEGATION_CHECK`)	Deleting a delegation that resolvers still use.	Off	`ADVANCED` only	Public and private

Topics

- What each guardrail checks
- How Route 53 decides that a record is in use
- Enabling and disabling guardrails for a hosted zone

What each guardrail checks

Each guardrail checks your change for one kind of mistake. It judges your hosted zone as it would be after the whole request, so a request that creates and deletes records is judged on its net result.

Name server format check

What it rejects

An NS record value that looks like a Route 53 name server but has a format error, such as a misspelled or missing top-level domain, a missing letter, a trailing comma, or several name servers in one value.

Why it matters

A delegation to a name server that doesn't exist takes the subdomain offline. If the misspelled name belongs to someone else, that party can answer the queries for your subdomain.

How it works

Route 53 compares every NS record value that looks like one of its own name servers with the formats that it uses. It doesn't check the name servers of other DNS providers. For example, the guardrail rejects `ns-1234.awsdns-56.og`, where `org` is misspelled, and `ns-1234.awsdns-56.org.example.com`, which has extra labels, and accepts `ns-1234.awsdns-56.org`.

Delegation authority check

What it rejects

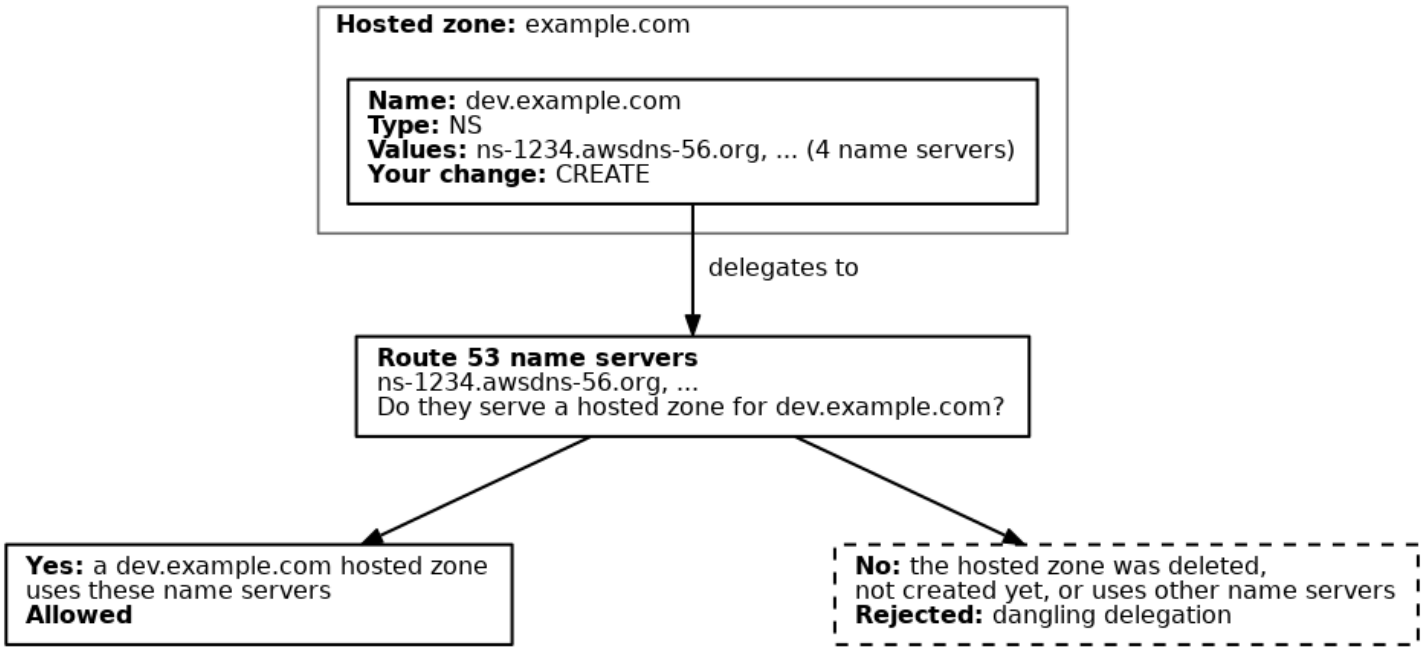
An NS record that delegates a subdomain to Route 53 name servers that aren't authoritative for the subdomain, which means that they don't serve a hosted zone for it. This happens when you create the NS record before the hosted zone for the subdomain exists, when you keep the NS record after that hosted zone is deleted, or when you copy name servers from another hosted zone.

Why it matters

Queries for the subdomain fail, and the subdomain is at risk of takeover: anyone who gets a hosted zone that uses those name servers can answer for your subdomain. This is a dangling delegation. For more information, see Protection from dangling delegation records in Route 53.

How it works

For example, an NS record for `dev.example.com` in your `example.com` hosted zone passes if the name servers in it serve a hosted zone for `dev.example.com`. If that hosted zone was deleted and the NS record still names its name servers, the guardrail rejects the record. Create the hosted zone first, and then put the four name servers that Route 53 assigns to it in the NS record. Route 53 checks only its own name servers, not the name servers of other DNS providers, and because the check verifies public delegations, it doesn't apply to private hosted zones.



Name server occlusion check

What it rejects

A change that occludes records. When an NS record delegates a subdomain to other name servers, the delegation occludes the records at or under that subdomain in your own hosted zone: resolvers that follow the delegation ask the other name servers, which don't have your records. The guardrail rejects whichever record you add second: a record at or under a subdomain that your hosted zone already delegates, or an NS record for a subdomain that already has records under it.

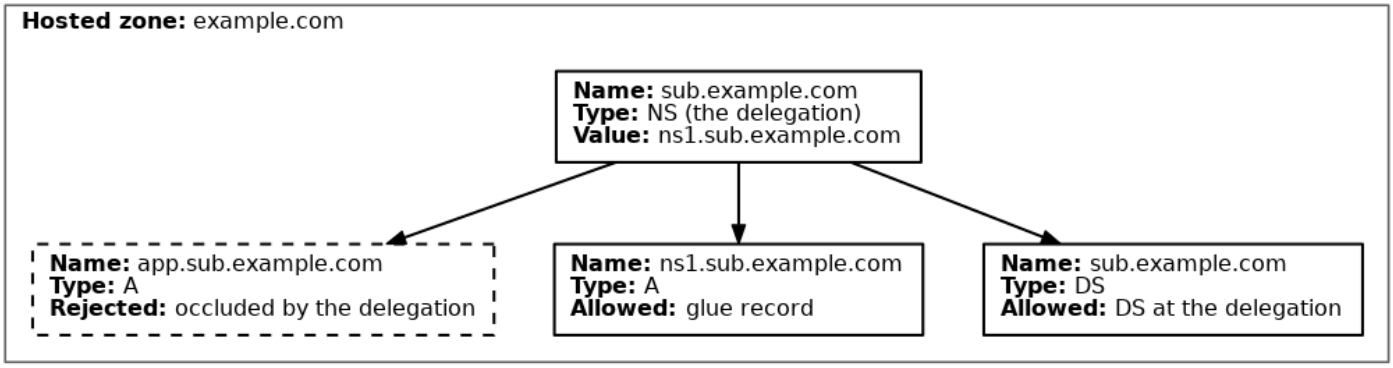
Why it matters

The occluded names stop resolving for the clients whose resolvers follow the delegation, even though the records are still in your hosted zone, which makes the problem hard to find.

How it works

For example, your `example.com` hosted zone delegates `sub.example.com` to other name servers. If you create an A record for `app.sub.example.com`, the guardrail rejects it. If `app.sub.example.com` already exists and you create the NS record for `sub.example.com`, the guardrail rejects the NS record. The guardrail allows the records that belong at a delegation: the NS record itself and a Delegation Signer (DS) record at the delegated name, glue records (A or AAAA records whose name is one of the delegation's name servers), and the NS record at

the zone apex, which names your hosted zone's own name servers. The guardrail doesn't check deletions, and it allows a request that fixes the conflict in the same step, such as deleting the records under `sub.example.com` and creating the NS record in one request. If more records would be occluded than one request can delete, delete them in earlier requests, and then create the NS record.



Alias target check

What it rejects

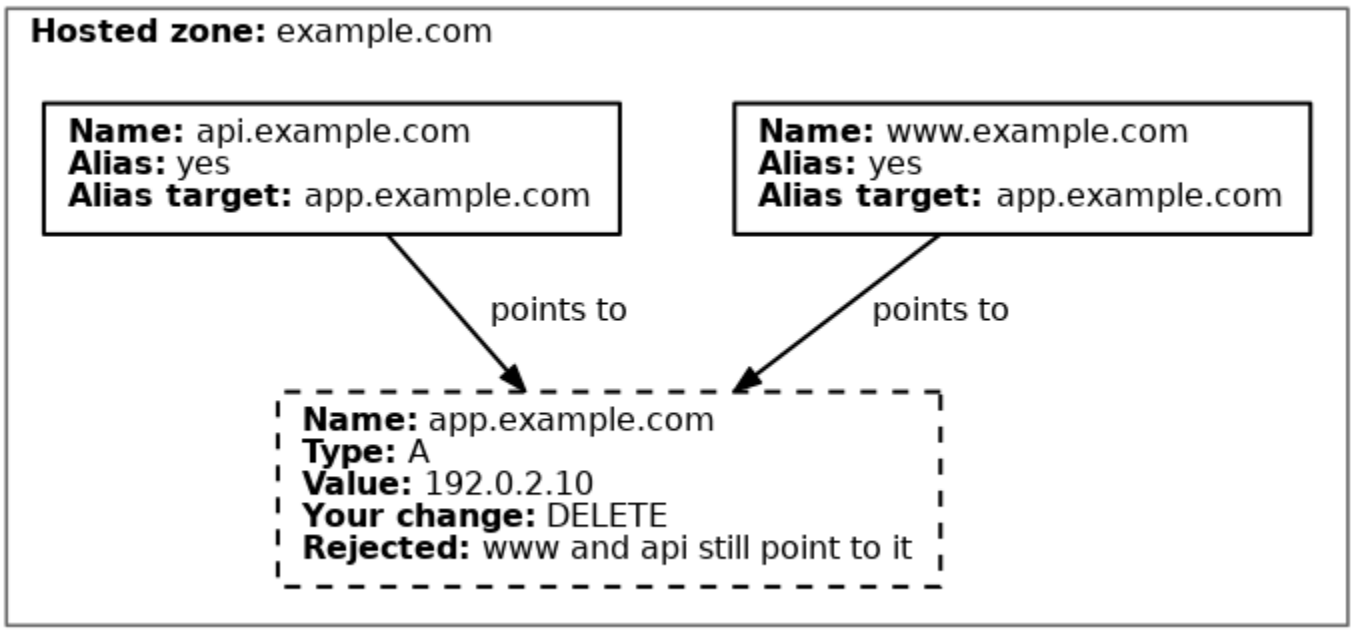
Deleting a record that an alias record in the same hosted zone points to. For example, `www.example.com` is an alias record that points to the A record `app.example.com`, and you delete `app.example.com`.

Why it matters

Without its target, the alias record stops resolving, and every client that queries it gets an error, even though you didn't change the alias record itself.

How it works

The guardrail allows the deletion if the same request deletes or retargets every alias record that points to the target, or if another record with the same name and type remains, such as the other members of a weighted record set. Changing a record's values isn't a deletion. The guardrail checks only alias records in the same hosted zone. To delete a chain of alias records in several requests, start at the top: delete each alias record before the record that it points to. If more alias records point to the target than one request can delete or retarget, delete or retarget them in earlier requests, and then delete the target.



Active record check

What it rejects

Deleting a record that is still receiving DNS queries when no record would remain at the same name and type to answer them. For example, you delete the A record for `api.example.com` during a cleanup, but clients queried it a few minutes ago and no other record for `api.example.com` remains.

Why it matters

Nothing fails right away, because resolvers keep the old answer until its TTL runs out. After that, every client that asks for the name gets no answer.

How it works

Route 53 checks whether the record answered a DNS query recently. For how long a record counts as in use, see How Route 53 decides that a record is in use. The guardrail judges only records that are gone after your request. Changing a record's values isn't a deletion, and NS records are judged by the active delegation check instead.

Deleting one record of a set that uses a routing policy is allowed when a remaining record still answers the deleted record's clients:

- Weighted, latency, failover, multivalue answer, and geoproximity sets – Any remaining record.

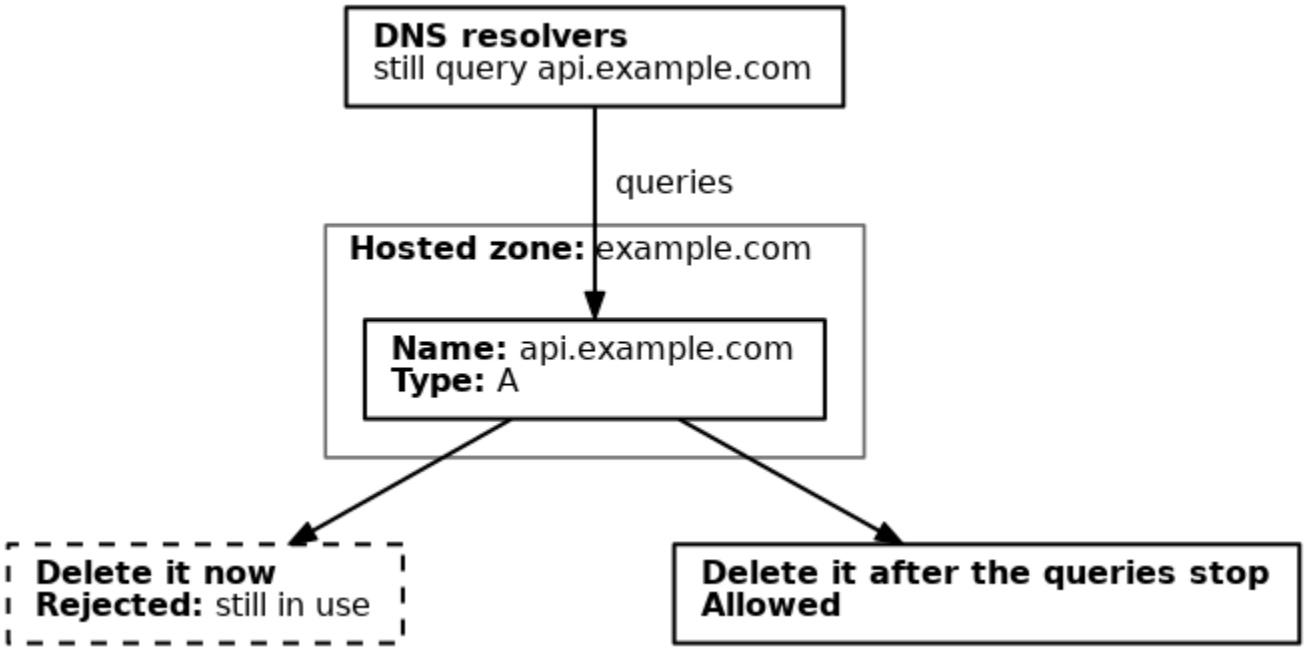
- Geolocation sets – A remaining record for a larger area, such as the continent or the default record.
- AZ-affinity sets – A remaining record for a larger area, such as the Region or the Global record.
- IP-based sets – The remaining default record of the same CIDR collection.

Route 53 rejects deleting the default record of a geolocation, AZ-affinity, or IP-based set while it is in use, because no other record answers its clients.

The guardrail counts only the queries for the record's own name, so it can't protect these records:

- The secondary record of a failover set while the primary record is healthy. It receives no queries.
- A record that clients reach only through a CNAME record. The queries count for the CNAME record, not for its target.

To protect a record regardless of its query activity, add a record set protection for it. For more information, see Protecting records from changes.



Active delegation check

What it rejects

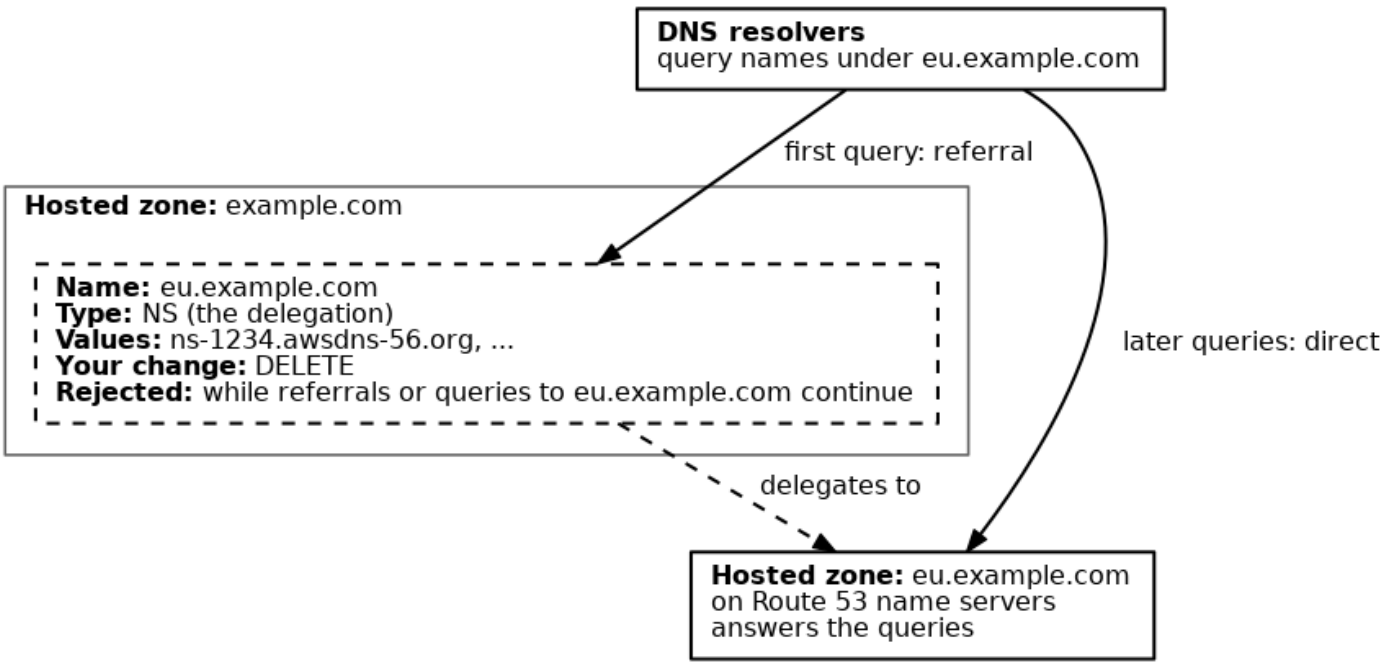
Deleting the NS record for a subdomain (a delegation) while resolvers are still being referred to it. For example, you delete the NS record that delegates `eu.example.com` to another hosted zone, but that hosted zone still answers queries for names under `eu.example.com`.

Why it matters

Deleting a delegation takes down every name under the subdomain at once, as soon as the resolvers' cached referrals expire, and nothing warns you until queries fail.

How it works

The delegation is in use if Route 53 recently referred resolvers to the subdomain. If the NS record names Route 53 name servers that serve a hosted zone for the subdomain, the delegation is also in use while that hosted zone receives DNS queries, because resolvers that have the delegation cached query the hosted zone directly. Delegations to name servers outside Route 53 are judged by the referrals alone. Changing the name servers in the NS record isn't a deletion. For how long a delegation counts as in use, see How Route 53 decides that a record is in use.



How Route 53 decides that a record is in use

Route 53 keeps track of the names and record types that recently answered DNS queries, and of the subdomains that it recently referred resolvers to. Because resolvers cache an answer for the record's time to live (TTL) and don't ask Route 53 again until it expires, Route 53 treats a record or a delegation as in use for 15 minutes plus its TTL after it last answered a query, up to a maximum of one hour. A record with a TTL of 60 seconds is in use for 16 minutes after its last query, so you can delete it 20 minutes after it stops receiving traffic. A record with a TTL of 24 hours is in use for one hour.

Because of the one-hour maximum, resolvers can keep using a record whose TTL is longer than one hour after Route 53 has stopped treating it as in use. Before you delete a record that has a long TTL, lower the TTL, wait for the old TTL to expire, and then delete the record.

The active record check and the active delegation check are based on the DNS queries that Route 53 receives, so they work best for hosted zones that receive regular traffic. A rarely queried record is in use only for a short time after each query, and it might not be in use at the moment you delete it, even if something still depends on it. To protect a record regardless of its query activity, add a record set protection for it. For more information, see Protecting records from changes.

Occasionally, Route 53 might not have current query activity data, for example during a service event that affects the system that collects it. The two checks then use the most recent activity that Route 53 has recorded. If that activity shows that your record is in use, Route 53 rejects the deletion. If Route 53 has no recorded activity for the record, or can't read its activity data at all, it allows the deletion, because it doesn't block a change that it can't verify. The other guardrails don't depend on query activity. If a service event affects the guardrails, you can see the status and the affected period in the AWS Health Dashboard. For more information, see the AWS Health User Guide.

Enabling and disabling guardrails for a hosted zone

You turn the four optional guardrails on or off for each hosted zone. Enabling a guardrail doesn't change your records. From then on, Route 53 applies the guardrail to the changes that you make in that hosted zone. The two always-on guardrails can't be turned off.

To turn a guardrail on or off, specify it in `ChangeGuardrails` when you create the hosted zone with CreateHostedZone, or later with UpdateHostedZoneFeatures. Each entry has a `Name` (`NS_OCCLUSION_CHECK`, `ALIAS_TARGET_CHECK`, `ACTIVE_RECORD_CHECK`, or `ACTIVE_DELEGATION_CHECK`) and a `Status` (`ENABLED` or `DISABLED`). To see the guardrails

that Route 53 applies to a hosted zone, call GetHostedZone and look at `ChangeGuardrails` in `Features`. The response lists every guardrail with its status. For the console and AWS CLI procedures, see Updating optional features.

The following AWS CLI command turns on the active record check for a hosted zone. Replace *Z123456789* with the ID of your hosted zone.

```
aws route53 update-hosted-zone-features \
    --hosted-zone-id Z123456789 \
    --change-guardrails Name=ACTIVE_RECORD_CHECK,Status=ENABLED
```

`ALIAS_TARGET_CHECK`, `ACTIVE_RECORD_CHECK`, and `ACTIVE_DELEGATION_CHECK` require the `ADVANCED` tier. If you turn one of them on for a hosted zone that uses the `STANDARD` tier, Route 53 rejects the request with an `InvalidInput` error. Route 53 also rejects a change of the hosted zone to the `STANDARD` tier while one of them is on. Turn them off first. For more information, see Hosted zone tiers.

If a guardrail rejects a change that you want to make anyway, turn the guardrail off, make the change, and then turn the guardrail on again. AWS CloudTrail records every change that you make to a hosted zone's guardrails, so you can see who turned a guardrail off, what changed while it was off, and when it was turned on again.

Public DNS fallback for private hosted zones

With public DNS fallback, when a query from your VPC asks for a name that doesn't exist in your private hosted zone, Route 53 retries the query against public DNS instead of returning NXDOMAIN (non-existent domain). Use it when your private hosted zone holds internal records, or records that override public names inside your VPCs, and you want every other name to resolve the way it does on the internet.

When public DNS fallback is disabled, Route 53 treats a private hosted zone as authoritative for its whole domain inside the associated VPCs. Suppose your private hosted zone is `example.com`. A query from the VPC for `www.example.com`, which exists only in public DNS, returns NXDOMAIN. The usual workaround is to copy the public record into the private hosted zone and keep the two in sync. When public DNS fallback is enabled, the query for `www.example.com` returns the public answer, so you don't need the copy.

Public DNS fallback is a setting on a private hosted zone, and it applies to every VPC that's associated with the zone, directly or through a Route 53 Profile. It's disabled by default and is

available only for private hosted zones that use the ADVANCED tier. For more information, see Hosted zone tiers.

Topics

- How public DNS fallback works
- When changes take effect
- Enabling and disabling public DNS fallback
- Checking the public DNS fallback status
- Considerations for public DNS fallback

How public DNS fallback works

Route 53 VPC Resolver, the DNS resolver that's available by default in all VPCs, always checks the private hosted zone first. What happens next depends on the answer that the zone returns:

- If the zone has a record of the requested type, VPC Resolver returns that record.
- If the name exists in the zone but has no record of the requested type, VPC Resolver returns NODATA. For example, an A record exists for `app.example.com`, and a query asks for the AAAA record.
- If the name doesn't exist in the zone:
  - If public DNS fallback is disabled, VPC Resolver returns NXDOMAIN.
  - If public DNS fallback is enabled, VPC Resolver retries the query against public DNS and returns the public answer.

Queries for the following names never fall back, because the zone doesn't return NXDOMAIN for them:

- The zone apex, such as `example.com`, which always exists because every zone has start of authority (SOA) and NS records. Create any other apex records that your VPC clients need, such as an A record for your website or MX records for email, in the private hosted zone as well.
- Names that match a wildcard record, such as `*.example.com`. Create any public records that your VPC clients need for those names in the private hosted zone as well.
- Names in a subdomain that the zone delegates with NS records, which get a referral instead. To resolve those names from the VPC, use a VPC Resolver delegation rule. For more information, see Resolver delegation rules tutorial.

When private hosted zones overlap, such as `example.com` and `accounting.example.com`, the public DNS fallback setting of the zone that's authoritative for the name applies. To use public DNS fallback for every name in a domain, enable it on each private hosted zone. For more information about which zone is authoritative, see Private hosted zones that have overlapping namespaces.

When changes take effect

After you enable or disable public DNS fallback, the change generally propagates to all Route 53 servers within 60 seconds. To confirm that it has, see Checking the public DNS fallback status.

A change to the public DNS fallback setting doesn't affect answers that are currently in resolver caches. With the default SOA record, a name that returned NXDOMAIN shortly before you enable fallback can keep returning NXDOMAIN for up to 15 minutes. For more information about how long negative answers are cached, see The start of authority (SOA) record. To test the change sooner, query a name that you haven't queried recently. The same applies when you disable fallback: cached public answers keep resolving until their TTLs expire.

Enabling and disabling public DNS fallback

You can enable or disable public DNS fallback on an existing private hosted zone that uses the `ADVANCED` tier by using the Route 53 console, API, AWS CLI, or SDK. If the zone uses the `STANDARD` tier, first change its tier. For more information, see Changing the tier of a hosted zone.

To enable or disable public DNS fallback using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the advanced private hosted zone that you want to update.
4. In **Hosted zone details**, choose **Edit hosted zone**.
5. Under **Public DNS fallback**, select **Enable public DNS fallback** to enable it, or clear it to disable it.
6. Choose **Save changes**.

To do this through the API, set `EnablePublicDNSFallback` to `true` or `false` in UpdateHostedZoneFeatures, or use one of the following AWS CLI commands, replacing *`Z1D633PJN98FT9`* with the ID of your hosted zone.

To enable public DNS fallback:

```
aws route53 update-hosted-zone-features --hosted-zone-id Z1D633PJN98FT9 --enable-
public-dns-fallback
```

To disable public DNS fallback:

```
aws route53 update-hosted-zone-features --hosted-zone-id Z1D633PJN98FT9 --no-enable-
public-dns-fallback
```

To enable public DNS fallback when you create a private hosted zone in the console, choose **Advanced** for **Tier**, and then select **Enable public DNS fallback**. With the API or AWS CLI, include it in the create request. For an example, see Updating optional features.

Checking the public DNS fallback status

To check the public DNS fallback status of a zone, look at the **Public DNS fallback** status on the hosted zone details page in the Route 53 console. You can also call GetHostedZone, or the `get-hosted-zone` AWS CLI command, and read the `PublicDNSFallbackStatus` field in the `Features` element of the response. The status is one of the following values:

`ENABLING`

The change to enable public DNS fallback is propagating to the Route 53 name servers.

`ENABLED`

Public DNS fallback is enabled, and the change has propagated.

`DISABLING`

The change to disable public DNS fallback is propagating to the Route 53 name servers.

`DISABLED`

Public DNS fallback is disabled, and any change has propagated.

Considerations for public DNS fallback

Keep the following in mind when you use public DNS fallback:

- If a VPC Resolver forwarding rule covers the same domain name as the private hosted zone, the rule takes precedence and public DNS fallback doesn't apply. For more information about VPC Resolver rules, see Private hosted zones and Route 53 VPC Resolver rules.

- Resolver DNS Firewall rules still apply to queries that fall back. If a rule blocks a domain name, queries for that name stay blocked. For more information about DNS Firewall, see Using DNS Firewall to filter outbound DNS traffic.
- Public answers can come from any public DNS provider. Your public hosted zone doesn't have to be in Route 53.
- Public DNS fallback also applies to VPCs that you associate with the zone after you enable it.
- If a CNAME record in the zone points to a name in another private hosted zone, VPC Resolver looks up the target in that zone. That zone's public DNS fallback setting decides whether a missing target falls back.

Enabling accelerated recovery for managing public DNS records

Route 53 accelerated recovery for managing public DNS records helps you achieve a 60-minute Recovery Time Objective (RTO) if the US East (N. Virginia) Region becomes unavailable. When you turn on this feature for a Route 53 public hosted zone, you can resume making DNS changes within about 60 minutes after AWS detects that the US East (N. Virginia) Region is impaired.

Important

Accelerated recovery is available only for public hosted zones. Private hosted zones are not supported.

Note

DNS query resolution from the Route 53 data plane continues to work normally during Regional service impairment. See Resilience in Route 53 for an understanding of data plane versus control plane operations.

Topics

- How accelerated recovery for public DNS records works
- Resubmitting DNS changes after failover
- Failback to the US East (N. Virginia) Region
- Additional considerations
- How to enable accelerated recovery for managing public DNS records

How accelerated recovery for public DNS records works

When accelerated recovery is on, Route 53 keeps a copy of your public hosted zone in the US West (Oregon) Region. If the US East (N. Virginia) Region becomes unavailable for a long time, Route 53 fails over within 60 minutes. It then routes control plane requests for your enabled hosted zones to the US West (Oregon) Region. You can then keep making DNS changes through the CLI, SDK, and API. Note that only some API methods work during failover. See the "Additional considerations" section for details. When the Region recovers, Route 53 fails back to the US East (N. Virginia) Region.

> **Note**
>
> You must turn on accelerated recovery before any issues with the US East (N. Virginia) Region occur. You can do this through the Console, CLI, SDK, or API (see the section titled *How to enable accelerated recovery for managing public DNS records* below). You cannot turn on accelerated recovery after a failover starts.

Resubmitting DNS changes after failover

Under normal conditions, changes to public hosted zones with accelerated recovery are accepted by the US East (N. Virginia) Region and then copied to the US West (Oregon) Region. However, when an outage occurs in the US East (N. Virginia) Region, some changes might be accepted there but not yet copied to the US West (Oregon) Region. These in-flight changes are called "stranded changes". After failover finishes, Route 53 suggests that you resubmit stranded changes before you continue your DNS work. You can do this with the API or with AWS CloudFormation, as described below.

Using the API to track and submit DNS changes

If you use the Route 53 API, AWS CLI, or AWS SDKs to manage DNS records, use the ChangeResourceRecordSets API to submit changes and the GetChange API to track them.

When you call ChangeResourceRecordSets, Route 53 returns an ID for the change (see ChangeInfo for details). You can then pass this ID to GetChange to check the status. A status of INSYNC means the change was copied to the US West (Oregon) Region and sent to all Route 53 DNS servers. You don't need to do anything more for these changes. However, during an outage in the US East (N. Virginia) Region, GetChange might return PENDING, meaning the change might not have been copied. If that happens, after failover finishes, GetChange returns NoSuchChange. This means

Route 53 could not copy that change. You can safely ignore these stranded changes and resubmit them as new changes. The failover is done when Route 53 posts a message to the AWS Health Dashboard.

Using AWS CloudFormation to track and submit changes

AWS CloudFormation tracks status for your DNS changes using the GetChange API. It only marks an update as done after changes are INSYNC. If the US East (N. Virginia) Region becomes unavailable while you are using CloudFormation to manage DNS records, your actions won't finish during the outage. After the Route 53 failover is done, retry the same actions to let CloudFormation resubmit the DNS changes.

Failback to the US East (N. Virginia) Region

Route 53 fails back control plane work for your public hosted zone to the US East (N. Virginia) Region after that Region recovers. During failback, you don't need to resubmit DNS changes, because no stranded changes are created during this process.

Additional considerations

There are a few more things to know about the accelerated recovery feature:

1. You can't create new hosted zones, delete hosted zones, turn on DNSSEC signing, or turn off DNSSEC signing during failover.
2. AWS PrivateLink links won't work after failover, but will work again after failback to the US East (N. Virginia) Region.
3. CloudFront flat-rate plans are not supported at this time.
4. Hosted zones with accelerated recovery can't be deleted. You must turn off accelerated recovery before you can delete the hosted zone.
5. Record set protections stay in effect when you turn accelerated recovery on or off. While you turn accelerated recovery on or off, requests to create or delete protections can fail with a `Throttling` error. Retry after a short wait. For more information about record set protections and accelerated recovery, see Record set protections and accelerated recovery.
6. During failover, these API methods will still work for public hosted zones with accelerated recovery. All other Route 53 API methods won't work until failback.
   - `ChangeResourceRecordSets`
   - `CreateResourceRecordSetProtections`
   - `DeleteResourceRecordSetProtections`

- GetChange
- GetGeoLocation
- GetHostedZone
- GetHostedZoneCount
- GetHostedZoneLimit
- GetReusableDelegationSet
- GetReusableDelegationSetLimit
- ListGeoLocations
- ListHostedZones
- ListHostedZonesByName
- ListResourceRecordSetProtections
- ListResourceRecordSets
- ListReusableDelegationSets

How to enable accelerated recovery for managing public DNS records

You can turn on accelerated recovery using the Route 53 console, API, CLI, or SDK. The time it takes depends on the size of your hosted zone and other factors. Plan for the process to take up to several hours. You can check the status in the Accelerated recovery tab of your hosted zone or through the GetHostedZone API. Near the end of the process, there is a brief period of up to several minutes where DNS changes are not accepted. After this, DNS changes work as normal.

To enable and disable accelerated recovery using the Route 53 console

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the public hosted zone for which you want to enable accelerated recovery.
4. In the **Accelerated recovery** tab, choose **Enable**.
5. Choose **Save changes**.
6. Monitor the hosted zone status. The status shows **Enabling accelerated recovery** during setup and changes to **Enabled** when complete.

You can disable accelerated recovery using the same steps above, but instead choosing **Disable**.

CLI example to enable

```
aws route53 update-hosted-zone-features --enable-accelerated-recovery --hosted-zone-id
 Z123456789
```

CLI example to disable

```
aws route53 update-hosted-zone-features --no-enable-accelerated-recovery --hosted-zone-
id Z123456789
```

Migrating a hosted zone to a different AWS account

When migrating a hosted zone to a different AWS account, follow these recommended steps.

These steps are most suitable for hosted zones with infrequent record changes. For hosted zones with frequent record updates, consider the following:

- Don't update any resource records during migration.
- Publish resource record changes in both old and new hosted zones after the delegation has been transferred.

Prerequisites

Install or upgrade the AWS CLI:

For information about downloading, installing, and configuring the AWS CLI, see the AWS Command Line Interface User Guide.

> **Note**
>
> Configure the CLI so that you can use it when you're using both the account that created the hosted zone and the account that you're migrating the hosted zone to. For more information, see Configure in the *AWS Command Line Interface User Guide*

If you're already using the AWS CLI, we recommend that you upgrade to the latest version of the CLI so that the CLI commands support the latest Route 53 features.

Topics

- Step 1: Prepare for migration
- Step 2: Create the new hosted zone
- Step 3: (Optional) Migrate health checks
- Step 4: Migrate records from the old hosted zone to the new hosted zone
- Step 5: Compare records in the old and new hosted zones
- Step 6: Update the domain registration to use name servers for the new hosted zone
- Step 7: Change the TTL for the NS record back to a higher value
- Step 8: Re-enable DNSSEC signing and establish the chain of trust (if required)
- Step 9: (Optional) delete the old hosted zone

Step 1: Prepare for migration

The preparation steps help you minimize the risks associated with migrating a hosted zone.

1. Monitor zone availability

You can monitor the zone for the availability of your domain names. This can help you address any issues that might lead to rolling back the migration. You can monitor for your domain names with most traffic by using CloudWatch or query logging. For more information about setting up query logging, see Monitoring Amazon Route 53.

The monitoring can be done through a shell script, or through a third party service. It shouldn't, however, be the only signal to determine if a rollback is required as you might also get feedback from your customers due to a domain not being available.

2. Lower the TTL setting

The TTL (time to live) setting for a record specifies how long you want DNS resolvers to cache the record and use the cached information. When the TTL expires, a resolver sends another query to the DNS service provider for a domain to get the latest information.

The typical TTL setting for the NS record is 172800 seconds, or two days. The NS record lists the name servers that the Domain Name System (DNS) can use to get information about how to route traffic for your domain. Lowering the TTL for the NS record, both with your current DNS service provider and with Route 53, reduces downtime for your domain if you discover a problem while you're migrating DNS to Route 53. If you don't lower the TTL, your domain could be unavailable on the internet for up to two days if something goes wrong.

To lower the TTL

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. Choose **Hosted zones** in the navigation pane.
3. Choose the name of the hosted zone.
4. Choose the NS record, and and in the **Record details** pane, choose **Edit record**.
5. Change the value of **TTL (Seconds)**. We recommend that you specify a value between 60 seconds and 900 seconds (15 minutes).
6. Choose **Save**.

3. Remove the DS record from the parent zone (If you have DNSSEC configured)

If you've configured DNSSEC for your domain, remove the Delegation Signer (DS) record from the parent zone before you migrate your domain to Route 53.

If the parent zone is hosted through Route 53, see Deleting public keys for a domain for more information. If the parent zone is hosted on another registrar, contact them to remove the DS record.

Route 53 does not currently support migrating the DNSSEC setting. As such, you will need to disable DNSSEC validation performed against your domain prior to the migration by removing the DS record from the parent zone. After the migration, you can re-enable DNSSEC validation by configuring DNSSEC on the new hosted zone and adding the respective DS record to the parent zone.

4. Make sure there are no other ongoing operations relying on the migrating hosted zone

Some operations will rely on DNS resolution in the migrating hosted zone, for example, the TLS/SSL certificate renewal process might require making DNS record changes and the provider will try to resolve the DNS record as the validation method. Before the migration, you should make sure there is no other operation happening, in order to avoid unexpected impact from the hosted zone migration.

Step 2: Create the new hosted zone

Create the new hosted zone in the account you want to migrate the hosted zone to.

Choose the tab for the instructions for either the AWS CLI or console.

- CLI
- Console

CLI

Enter the following command:

```
aws route53 create-hosted-zone \
            --name $hosted_zone_name \
            --caller-reference $unique_string
```

For more information, see create-hosted-zone.

Console

To create the new hosted zone using a different account

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   Sign in with the account credentials for the account that you want to migrate the hosted zone to.
2. Create a hosted zone. For more information, see Creating a public hosted zone.
3. Make note of the hosted zone ID. In some cases, you'll need this information later in the process.
4. Log out of the Route 53 console.

Lower the NS TTL in the new zone as well, similar to Lower TTL setting in preparation Step 1, Lower the TTL setting.

Step 3: (Optional) Migrate health checks

You can associate DNS records in the new account with Route 53 health checks from the account you're migrating from. To migrate a Route 53 health check, you need to create new health checks in your new account with the same configuration as your existing ones. For more information, see Creating Amazon Route 53 health checks.

Step 4: Migrate records from the old hosted zone to the new hosted zone

You can migrate records from an AWS account to another by using the console or the AWS CLI.

Console

If your zone contains just a few records, you can consider to use Route 53 console to list the records in your old zone, note them down, and create them in the new zone. If you have migrated the health check in Step 3: (Optional) Migrate health checks, when you create the records in the new hosted zone, you should specify the new health check ID. For more information, see the following topics:

- Listing records
- Creating records by using the Amazon Route 53 console
- Configuring DNS failover

You should lower the NS TTL in the new zone as well, similar to Lower TTL setting in Step 1.

CLI

If your zone contains a large number of records, you can export the records you want to migrate to a file, edit the file, and then use the edited file to create records in the new hosted zone. The following procedure uses AWS CLI commands, though third-party tools are also available for this purpose.

1. Run the following command:

```
aws route53 list-resource-record-sets --hosted-zone-id hosted-zone-id > path-to-output-file
```

Note the following:

- For *hosted-zone-id*, specify the ID of the old hosted zone that contains the records you want to migrate.
- For *path-to-output-file*, specify the directory path and file name that you want to save the output in.
- The > character sends the output to the specified file.

- The AWS CLI automatically handles pagination for hosted zones that contain more than 100 records. For more information, see Using the AWS Command Line Interface's pagination options in the *AWS Command Line Interface User Guide*.

  If you use another programmatic method to list records, such as one of the AWS SDKs, you can get a maximum of 100 records per page of results. If the hosted zone contains more than 100 records, you must submit multiple requests to list all records.

Make a copy of this output. After you create records in the new hosted zone, we recommend that you run the AWS CLI `list-resource-record-sets` command on the new hosted zone and compare the two outputs to make sure that all the records were created.

2. Edit the records that you want to migrate.

   Edit the exported file before you can use it with the `change-resource-record-sets` command. You can make these changes using the search and replace function in a text editor.

   > **Note**
   >
   > The following steps describe manual editing using a text editor. Advanced users can automate these transformations using programmatic tools such as jq, Python, or other scripting languages.

   Open a copy of the file that you created in step 1 of this procedure that contains the records that you want to migrate, and make the following changes:

   - Replace the ResourceRecordSets element at the top of the file with `Changes` element.
   - Optional – add a `Comment` element.
   - Delete the lines related to the NS and SOA records of the hosted zone name. The new hosted zone already has those records.
   - For each record, add an `Action` and a `ResourceRecordSets` element, add opening and closing brackets (`{ }`) as required to make the JSON code valid.

> **Note**
>
> You can use a JSON validator to verify that you have all the braces and brackets in the correct places. To find an online JSON validator, search "JSON validator" in your browser.

- If the hosted zone contains any aliases that refer to other records in the same hosted zone, make the following changes:
  - Change the hosted zone ID to the ID of the new hosted zone.

    > **Important**
    >
    > If the alias record is pointing to another resource, for example, a load balancer, do not change the hosted zone ID to the hosted zone ID of the domain. If you change the hosted zone ID accidentally, rollback the hosted zone ID to the hosted zone id of the resource itself, not the hosted zone ID of the domain. The resource hosted zone ID can be found in the AWS console where the resource was created.

  - Move the alias records to the bottom of the file. Route 53 must create the record that an alias record refers to before it can create the alias record.

    > **Important**
    >
    > If one or more alias records refer to other alias records, the records that are the alias target must appear in the file before the referencing alias records. For example, if `alias.example.com` is the alias target for `alias.alias.example.com`, `alias.example.com` must appear first in the file.

  - Delete any alias records that route traffic to a traffic policy instance. Make note of the records so you can recreate them later.
- If you migrated health checks in Step 3: (Optional) Migrate health checks, change the records to associate with the newly created health check IDs.

The following example shows the edited version of records for a hosted zone for example.com. The red, italicized text is new:

```
{
    "Comment": "string",
    "Changes": [
        {
            "Action": "CREATE",
            "ResourceRecordSet":{
                "ResourceRecords": [
                    {
                        "Value": "192.0.2.4"
                    },
                    {
                        "Value": "192.0.2.5"
                    },
                    {
                        "Value": "192.0.2.6"
                    }
                ],
                "Type": "A",
                "Name": "route53documentation.com.",
                "TTL": 300
            }
        },
        {
            "Action": "CREATE",
            "ResourceRecordSet":{
                "AliasTarget": {
                    "HostedZoneId": "Z3BJ6K6RIION7M",
                    "EvaluateTargetHealth": false,
                    "DNSName": "s3-website-us-west-2.amazonaws.com."
                },
                "Type": "A",
                "Name": "www.route53documentation.com."
            }
        }
    ]
}
```

3. Split large files into smaller files

If you have a lot of records or if you have records that have a lot of values (for example, a lot of IP addresses), you might need to split the file into smaller files. The max values are:

- Each file can contain a maximum of 1,000 records.
- The maximum combined length of the values in all `Value` elements is 32,000 bytes.

4. Create records in the new hosted zone

   Enter the following CLI:

   ```
   aws route53 change-resource-record-sets \
               --hosted-zone-id new-hosted-zone-id \
               --change-batch file://path-to-file-that-contains-records
   ```

   Specify the following values:

   - For *new-hosted-zone-id*, specify the ID of the new hosted zone.
   - For *path-to-file-that-contains-records*, specify the directory path and file name that you edited in the previous steps.

   If you deleted any alias records that route traffic to a traffic policy instance, recreate them using the Route 53 console. For more information, see Creating records by using the Amazon Route 53 console.

Step 5: Compare records in the old and new hosted zones

To confirm that you successfully created all of your records in the new hosted zone, enter the following CLI command to list the records in the new hosted zone and compare the output with the list of records from the old hosted zone.

```
aws route53 list-resource-record-sets \
            --hosted-zone-id new-hosted-zone-id \
            --output json > path-to-output-file
```

Specify the following values:

- For *new-hosted-zone-id*, specify the ID of the new hosted zone.

- For *path-to-output-file*, specify the directory path and file name that you want to save the output in. Use a file name that is different from the file name that you used in Step 4.

  The > character sends the output to the specified file.

Compare the output with the output from Step 4. Other than the values of the NS and SOA records and any changes that you made in Step 4 (such as different hosted zone IDs or domain names), the two outputs should be identical.

If the records in the new hosted zone don't match the records in the old hosted zone, do one of the following:

- Make minor corrections using the Route 53 console. For more information, see Editing records.
- Delete all the records except the NS and SOA records in the new hosted zone, and repeat the procedure in Step 4.

Step 6: Update the domain registration to use name servers for the new hosted zone

When you finish migrating the records to the new hosted zone, change the name servers for the domain registration to use the name servers for the new hosted zone. For more information, see Making Amazon Route 53 the DNS service for an existing domain.

If your hosted zone is in use - for example, if your users are using the domain name to browse to a website or access a web application - you should continue monitoring traffic and availability of the hosted zone, including website or application traffic, email, etc.

- **If the traffic slows or stops** – Change the name service for the domain registration back to the previous name servers of the old hosted zone. Then determine what went wrong.
- **If the traffic is unaffected** – Continue to the next step.

Step 7: Change the TTL for the NS record back to a higher value

In the new hosted zone, change the TTL for the NS record to a more typical value, for example, 172800 seconds (two days). This improves latency for your users because they don't have to wait as often for DNS resolvers to send a query for the name servers for your domain.

To change the TTL

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. Choose **Hosted zones** in the navigation pane.
3. Choose the name of the hosted zone.
4. Choose the NS record, and and in the **Record details** pane, choose **Edit record**.
5. Change the value of **TTL (Seconds)** to the number of seconds that you want DNS resolvers to cache the names of the name servers for your domain. We recommend a value of 172800 seconds.
6. Choose **Save**.

Step 8: Re-enable DNSSEC signing and establish the chain of trust (if required)

You can re-enable DNSSEC signing in two steps:

1. Enable DNSSEC signing for Route 53, and request that Route 53 create a key signing key (KSK) based on a customer managed key in AWS Key Management Service.
2. Create a chain of trust for the hosted zone by adding a Delegation Signer (DS) record to the parent zone, so DNS responses can be authenticated with trusted cryptographic signatures.

For instructions, see Enabling DNSSEC signing and establishing a chain of trust.

Step 9: (Optional) delete the old hosted zone

When you're confident that you don't need the old hosted zone any longer, you can optionally delete it. For instructions, see Deleting a public hosted zone.

> **Important**
>
> Don't delete the old hosted zone or any records in that hosted zone for at least 48 hours after you update the domain registration to use name servers for the new hosted zone. If you delete the old hosted zone before DNS resolvers stop using the records in that hosted zone, your domain could be unavailable on the internet until resolvers start using the new hosted zone.

Working with records

After you create a hosted zone for your domain, such as example.com, you create records to tell the Domain Name System (DNS) how you want traffic to be routed for that domain.

For example, you might create records that cause DNS to do the following:

- Route internet traffic for example.com to the IP address of a host in your data center.
- Route email for that domain (ichiro@example.com) to a mail server (mail.example.com).
- Route traffic for a subdomain called operations.tokyo.example.com to the IP address of a different host.

Each record includes the name of a domain or a subdomain, a record type (for example, a record with a type of MX routes email), and other information for that record type (for MX records, the host name of one or more mail servers and a priority for each server). For information about the different record types, see Supported DNS record types.

The name of each record in a hosted zone must end with the name of the hosted zone. For example, the example.com hosted zone can contain records for www.example.com and accounting.tokyo.example.com subdomains, but cannot contain records for a www.example.ca subdomain.

> **Note**
>
> To create records for complex routing setups, you can also use the Traffic Flow visual editor and save the setup as a traffic policy. You can then link the traffic policy to one or more domain names (such as example.com) or subdomain names (such as www.example.com), in the same hosted zone or in multiple hosted zones. You can also roll back updates if the new setup isn't working as expected. For more information, see Using Traffic Flow to route DNS traffic.

Amazon Route 53 doesn't charge for the records that you add to a hosted zone. For information about the maximum number of records that you can create in a hosted zone, see Quotas.

Topics

- Choosing a routing policy

- Choosing between alias and non-alias records
- Supported DNS record types
- Creating records by using the Amazon Route 53 console
- Resource record set permissions
- Values that you specify when you create or edit Amazon Route 53 records
- Creating records by importing a zone file
- Editing records
- Deleting records
- Listing records
- Protecting records from changes

Choosing a routing policy

When you create a record, you choose a routing policy, which determines how Amazon Route 53 responds to queries:

- **Simple routing policy** – Use for a single resource that performs a given function for your domain, for example, a web server that serves content for the example.com website. You can use simple routing to create records in a private hosted zone.
- **Failover routing policy** – Use when you want to configure active-passive failover. You can use failover routing to create records in a private hosted zone.
- **Geolocation routing policy** – Use when you want to route traffic based on the location of your users. You can use geolocation routing to create records in a private hosted zone.
- **Geoproximity routing policy** – Use when you want to route traffic based on the location of your resources and, optionally, shift traffic from resources in one location to resources in another location. You can use geoproximity routing to create records in a private hosted zone.
- **Latency routing policy** – Use when you have resources in multiple AWS Regions and you want to route traffic to the Region that provides the best latency. You can use latency routing to create records in a private hosted zone.
- **IP-based routing policy** – Use when you want to route traffic based on the location of your users, and have the IP addresses that the traffic originates from.
- **Multivalue answer routing policy** – Use when you want Route 53 to respond to DNS queries with up to eight healthy records selected at random. You can use multivalue answer routing to create records in a private hosted zone.

- **Weighted routing policy** – Use to route traffic to multiple resources in proportions that you specify. You can use weighted routing to create records in a private hosted zone.
- **AZ-affinity routing policy** – Use when you want to route traffic to resources in the same Availability Zone (AZ), Zone Group, or AWS Region that the query comes from, with automatic fallback to a less specific location. You can use AZ-affinity routing to create records in a private hosted zone.

Topics

- Simple routing
- Failover routing
- Geolocation routing
- Geoproximity routing
- Latency-based routing
- IP-based routing
- Multivalue answer routing
- Weighted routing
- AZ-affinity routing
- How Amazon Route 53 uses EDNS0 to estimate the location of a user

Simple routing

With simple routing, you can set up standard DNS records, with no special Route 53 routing such as weighted or latency. You typically route traffic to a single resource, for example, to a web server for your website.

You can use simple routing policy for records in a private hosted zone.

If you choose the simple routing policy in the Route 53 console, you can't create multiple records with the same name and type, but you can specify multiple values in the same record, such as multiple IP addresses. (If you choose the simple routing policy for an alias record, you can specify only one AWS resource or one record in the current hosted zone.) If you specify multiple values in a record, Route 53 returns all values to the recursive resolver in random order, and the resolver returns the values to the client (such as a web browser) that sent the DNS query. The client then chooses a value and resends the query. With simple routing, although you can specify multiple IP addresses, they are not health checked.

For information about values that you specify when you use the simple routing policy to create records, see the following topics:

- Values specific for simple records
- Values specific for simple alias records
- Values that are common for all routing policies
- Values that are common for alias records for all routing policies

Failover routing

With failover routing, you can route traffic to a resource when it's healthy or to a different resource when the first one is unhealthy. The primary and secondary records can route traffic to anything from an Amazon S3 bucket set up as a website to a complex tree of records. For more information, see Active-passive failover.

You can use failover routing policy for records in a private hosted zone.

For information about values that you specify when you use the failover routing policy to create records, see the following topics:

- Values specific for failover records
- Values specific for failover alias records
- Values that are common for all routing policies
- Values that are common for alias records for all routing policies

Geolocation routing

With geolocation routing, you can choose the resources that serve your traffic based on where your users are located, meaning the location that DNS queries come from. For example, you might want all queries from Europe to be routed to an Elastic Load Balancing load balancer in the Frankfurt Region.

When you use geolocation routing, you can show your content in the language of your users. You can also use geolocation routing to limit content to only the locations where you have distribution rights. Another use is to balance load across endpoints in a way that is easy to predict and manage, so that each user location always goes to the same endpoint.

You can specify geographic locations by continent, by country, or by state in the United States. If you create separate records for overlapping geographic regions—for example, one record for North America and one for Canada—priority goes to the smallest geographic region. This allows you to route some queries for a continent to one resource and to route queries for selected countries on that continent to a different resource. (For a list of the countries on each continent, see Location.)

Geolocation works by mapping IP addresses to locations. However, some IP addresses aren't mapped to a location, so even if you create geolocation records that cover all seven continents, Amazon Route 53 will get some DNS queries from locations that it can't identify. You can create a default record that handles both queries from unmapped IP addresses and queries from locations where you haven't created geolocation records. If you don't create a default record, Route 53 returns a "no answer" response for queries from those locations.

You can use geolocation routing for records in both public and private hosted zones.

For more information, see How Amazon Route 53 uses EDNS0 to estimate the location of a user.

For information about values that you specify when you use the geolocation routing policy to create records, see the following topics:

- Values specific for geolocation records
- Values specific for geolocation alias records
- Values that are common for all routing policies
- Values that are common for alias records for all routing policies

Geolocation routing in private hosted zones

For private hosted zones, Route 53 responds to DNS queries based on the AWS Region of the VPC that the query originated from. For the list of AWS Regions, see Regions and zones in the *Amazon EC2 user guide*.

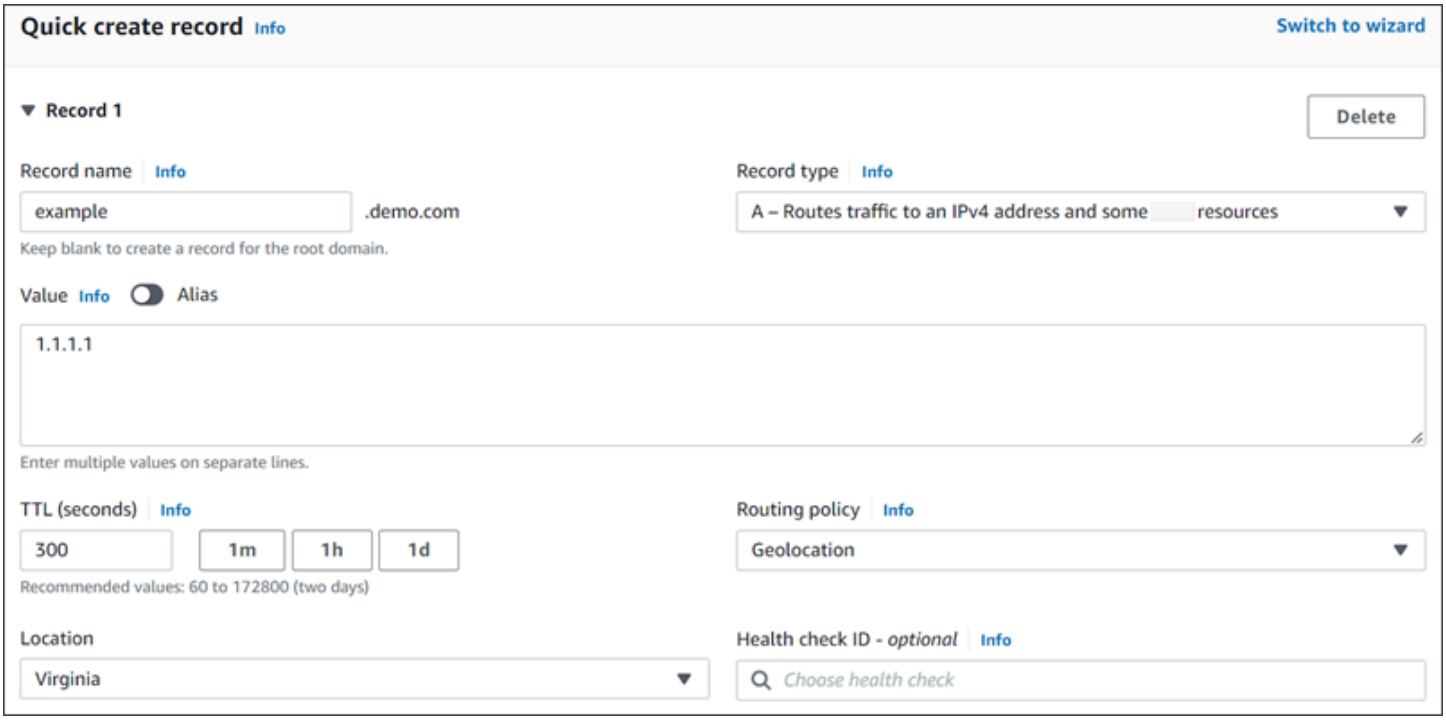
If the DNS query originates from an on-premises part of a hybrid network, it will be considered as having originated from the AWS Region that the VPC is located in.

If you include health checks, you can create default records for:

- IP addresses that aren't mapped to geographic locations.
- DNS queries that come from locations that you haven't created geolocation records for.

If the geolocation record for the DNS query's region is unhealthy, the default record will be returned (if it is healthy).

In the example configuration in the following figure, DNS queries coming from an us-east-1 AWS Region (Virginia) will be routed to the 1.1.1.1 endpoint.



Geoproximity routing

With geoproximity routing, Amazon Route 53 routes traffic to your resources based on the location of your users and your resources. It sends traffic to the closest available resource. You can also choose to send more or less traffic to a given resource by setting a value called a *bias*. A bias expands or shrinks the geographic region from which traffic goes to a resource.

You can use geoproximity routing for records in both public and private hosted zones. For private hosted zones, see Geoproximity routing in private hosted zones.

You create geoproximity rules for your resources and specify one of the following values for each rule:

- If you're using AWS resources, specify the AWS Region or Local Zone Group that you created the resource in.
- If you're using non-AWS resources, specify the latitude and longitude of the resource.

To use AWS Local Zones, you have to first enable them. For more information, see Getting started with Local Zones in the *AWS Local Zones User Guide*.

To learn about the difference between AWS Regions and Local Zones, see Regions and Zones in the *Amazon EC2 User Guide*.

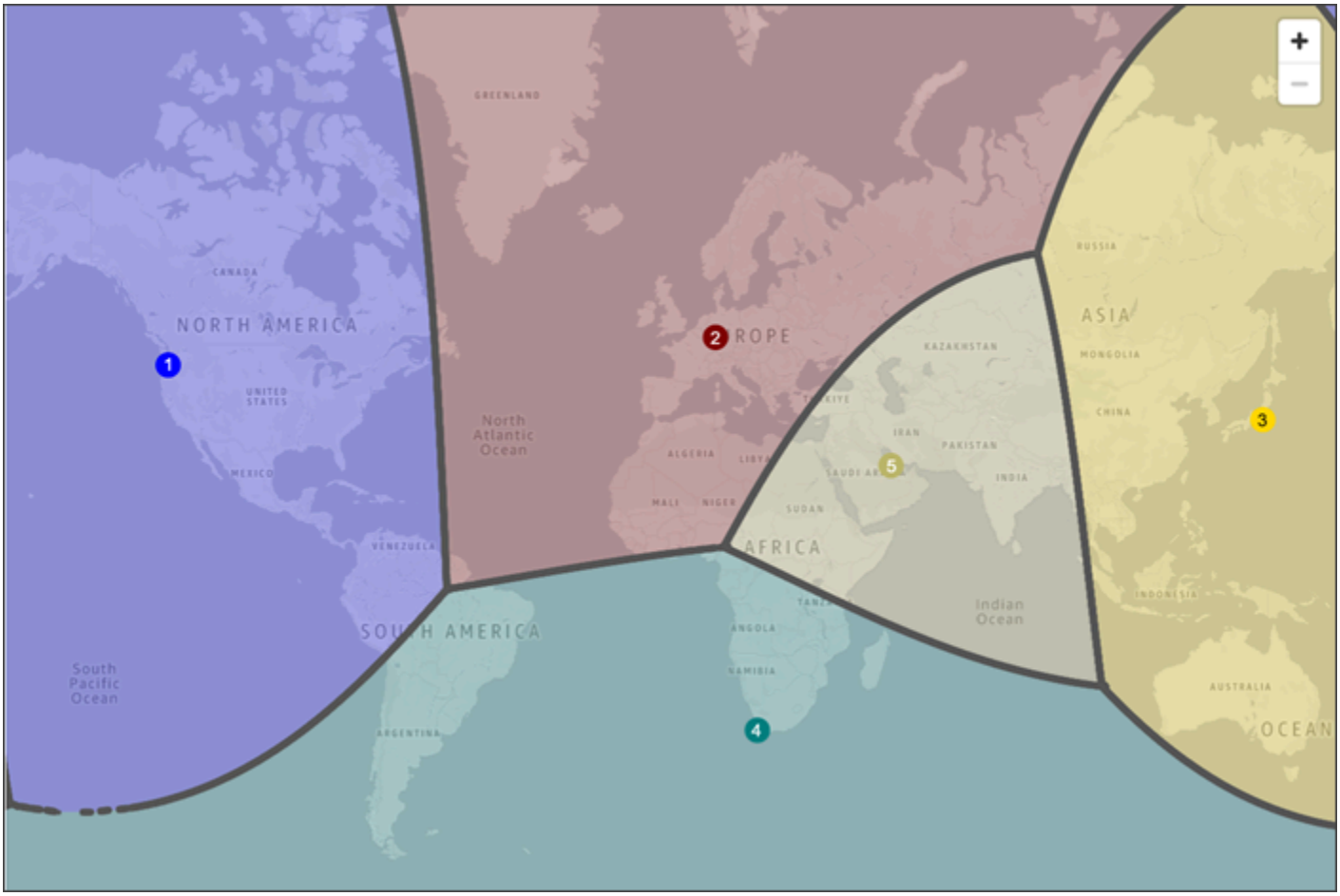
To optionally change the size of the geographic region from which Route 53 routes traffic to a resource, specify the applicable value for the bias:

- To expand the size of the geographic region from which Route 53 routes traffic to a resource, specify a positive integer from 1 to 99 for the bias. Route 53 shrinks the size of adjacent regions.
- To shrink the size of the geographic region from which Route 53 routes traffic to a resource, specify a negative bias of -1 to -99. Route 53 expands the size of adjacent regions.

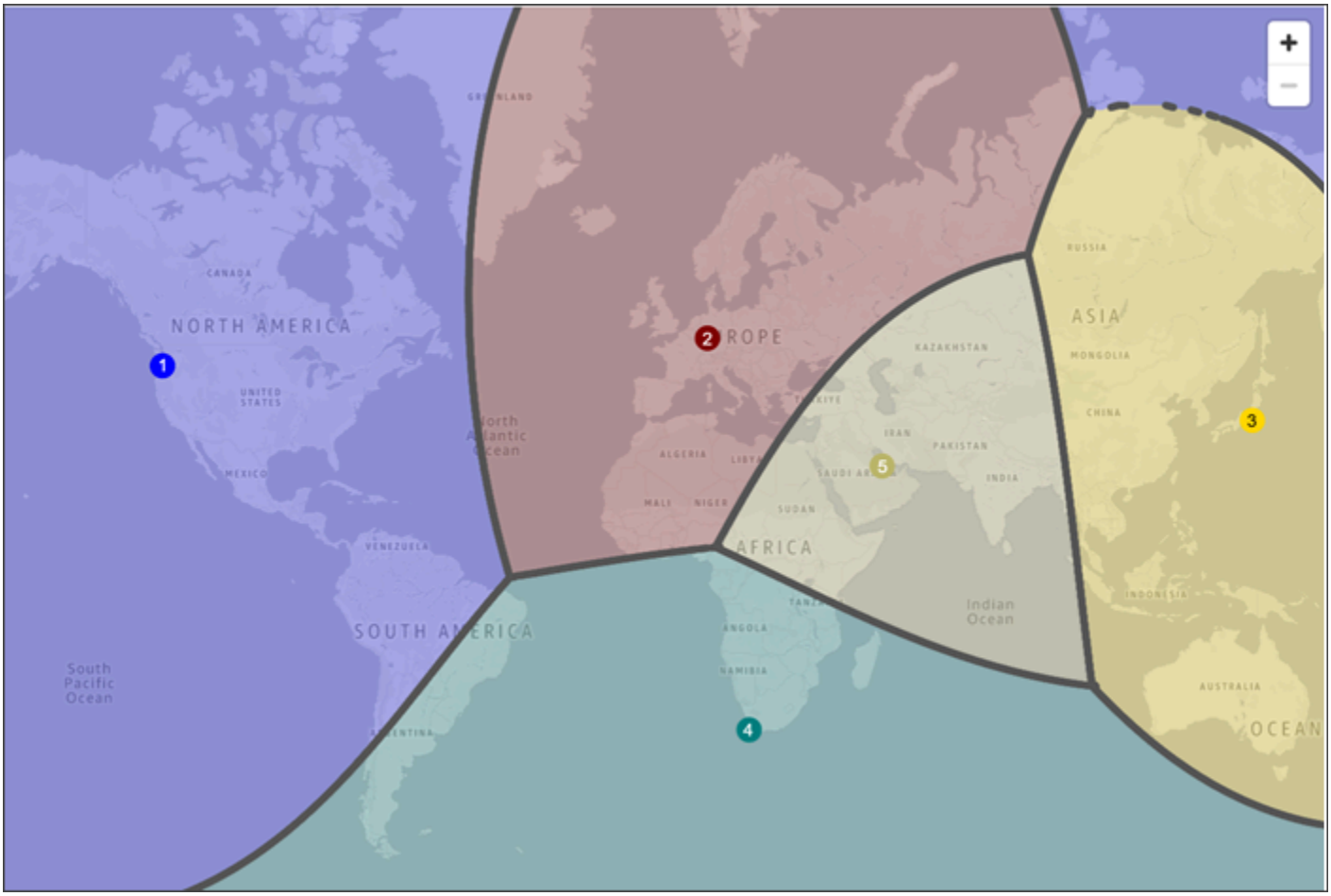
The following map shows four AWS Regions (numbered 1 through 5):

1. US West (Oregon)
2. Europe (Frankfurt)
3. Asia Pacific (Tokyo)
4. Africa (Cape Town)
5. Middle East (Bahrain)

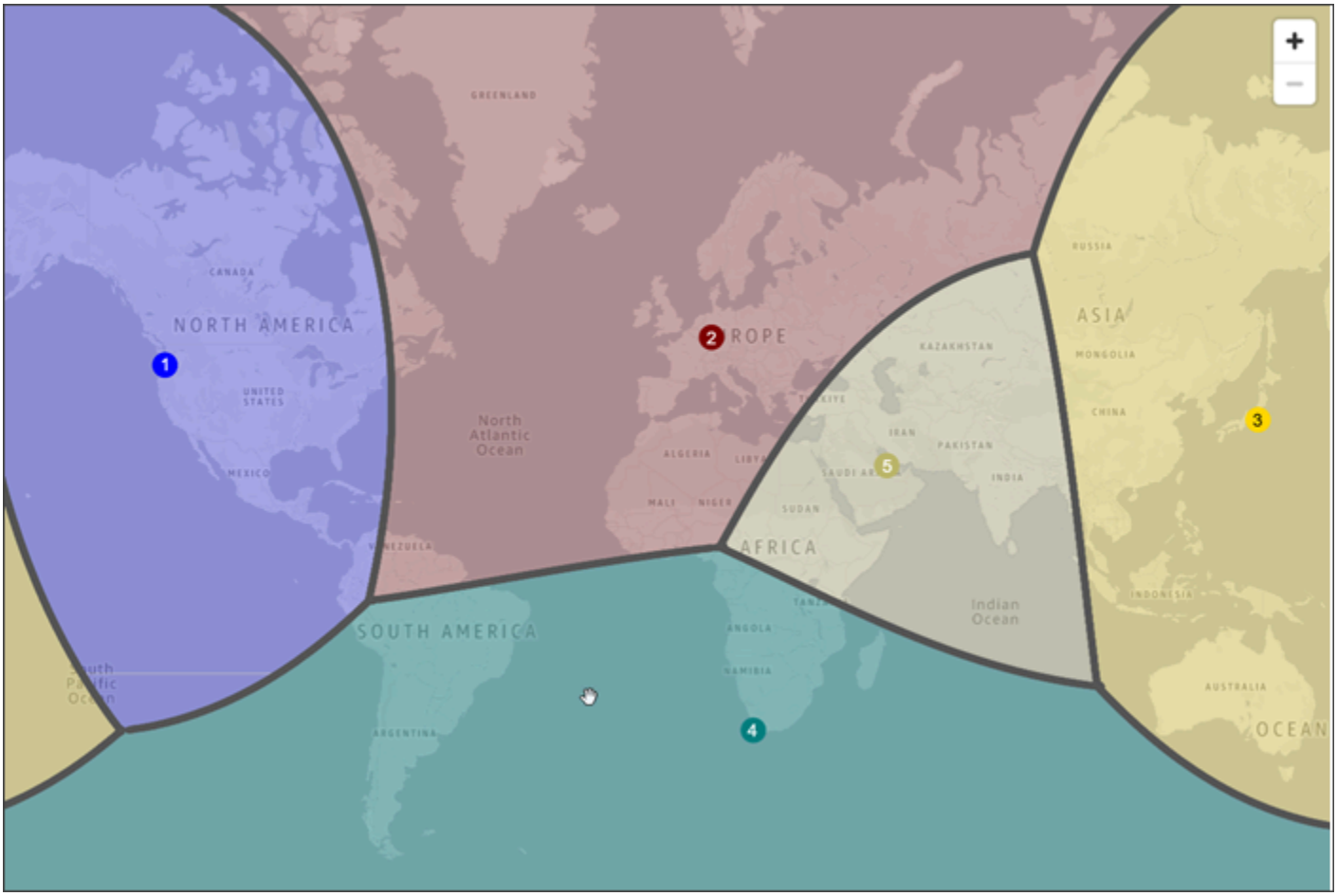
> **Note**
>
> The maps are available only with Traffic Flow.



The following map shows what happens if you add a bias of +25 for the US West (Oregon) Region (number **1** on the map). Traffic is routed to the resource in that Region from a larger portion of North America and from all of South America than previously.



The following map shows what happens if you change the bias to -25 for the US West (Oregon) Region. Traffic is routed to the resource in that Region from smaller portions of North and South America than previously, and more traffic is routed to resources in the adjacent regions **2**, **3**, and **4**.



The effect of changing the bias for your resources depends on several factors, including the following:

- The number of resources that you have.
- How close the resources are to each other.
- The number of users near the border between geographic regions. For example, suppose you have resources in the AWS Regions US East (N. Virginia) and US West (Oregon), and many users in Dallas, Austin, and San Antonio, Texas, USA. Those cities are about the same distance from both resources, so a small change in bias could shift a lot of traffic from one AWS Region to another.

We suggest that you change the bias in small steps to avoid overwhelming your resources due to a sudden shift in traffic.

For more information, see How Amazon Route 53 uses EDNS0 to estimate the location of a user.

How Amazon Route 53 uses bias to route traffic

Here's the formula that Amazon Route 53 uses to determine how to route traffic:

Bias

```
Biased distance = actual distance * [1 - (bias/100)]
```

When the value of the bias is positive, Route 53 treats the source of a DNS query and the resource in a geoproximity record (such as an EC2 instance in an AWS Region) as if they were closer together than they really are. For example, suppose you have these geoproximity records:

- A record for web server A, which has a positive bias of 50
- A record for web server B, which has no bias

When a geoproximity record has a positive bias of 50, Route 53 halves the distance between the source of a query and the resource for that record. Then Route 53 figures out which resource is closer to the source of the query. Suppose web server A is 150 km from the source of a query and web server B is 100 km from the source of the query. If neither record had a bias, Route 53 would route the query to web server B because it's closer. However, because the record for web server A has a positive bias of 50, Route 53 treats web server A as if it's 75 km from the source of the query. As a result, Route 53 routes the query to web server A.

Here's the calculation for a positive bias of 50:

```
Bias = 50
Biased distance = actual distance * [1 - (bias/100)]

Biased distance = 150 kilometers * [1 - (50/100)]
Biased distance = 150 kilometers * (1 - .50)
Biased distance = 150 kilometers * (.50)
Biased distance = 75 kilometers
```

Geoproximity routing in private hosted zones

For private hosted zones, Route 53 responds to DNS queries based on the AWS Region of the VPC that the query originated from, and routes traffic to the closest resource, taking bias into account.

> **Note**
>
> If you have an outbound endpoint forwarded to an inbound endpoint, the record resolves based on where the inbound endpoint is, not the outbound endpoint.

Latency-based routing

If your application is hosted in multiple AWS Regions, you can improve performance for your users by serving their requests from the AWS Region with the lowest latency.

> **Note**
>
> Data about the latency between users and your resources is based entirely on traffic between users and AWS data centers. If you aren't using resources in an AWS Region, the actual latency between your users and your resources can vary significantly from AWS latency data. This is true even if your resources are located in the same city as an AWS Region.

To use latency-based routing, you create latency records for your resources in multiple AWS Regions. When Route 53 gets a DNS query for your domain or subdomain (example.com or acme.example.com), it checks which AWS Regions you've created latency records for, finds which Region gives the user the lowest latency, and then picks a latency record for that Region. Route 53 responds with the value from the chosen record, such as the IP address for a web server.

For example, suppose you have Elastic Load Balancing load balancers in the US West (Oregon) Region and in the Asia Pacific (Singapore) Region. You create a latency record for each load balancer. Here's what happens when a user in London enters the name of your domain in a browser:

1. DNS routes the query to a Route 53 name server.
2. Route 53 refers to its data on latency between London and the Singapore Region and between London and the Oregon Region.
3. If latency is lower between the London and Oregon Regions, Route 53 responds to the query with the IP address for the Oregon load balancer. If latency is lower between London and the Singapore Region, Route 53 responds with the IP address for the Singapore load balancer.

Latency between hosts on the internet can change over time as network connections and routing change. Latency-based routing uses latency data taken over a period of time, and the data reflects these changes. A request routed to the Oregon Region this week might go to the Singapore Region next week.

> **Note**
>
> When a browser or other viewer uses a DNS resolver that supports the edns-client-subnet extension of EDNS0, the DNS resolver sends Route 53 a truncated version of the user's IP address. If you configure latency-based routing, Route 53 considers this value when routing traffic to your resources. For more information, see How Amazon Route 53 uses EDNS0 to estimate the location of a user.

You can use latency routing policy for records in a private hosted zone.

For information about values that you specify when you use the latency routing policy to create records, see the following topics:

- Values specific for latency records
- Values specific for latency alias records
- Values that are common for all routing policies
- Values that are common for alias records for all routing policies

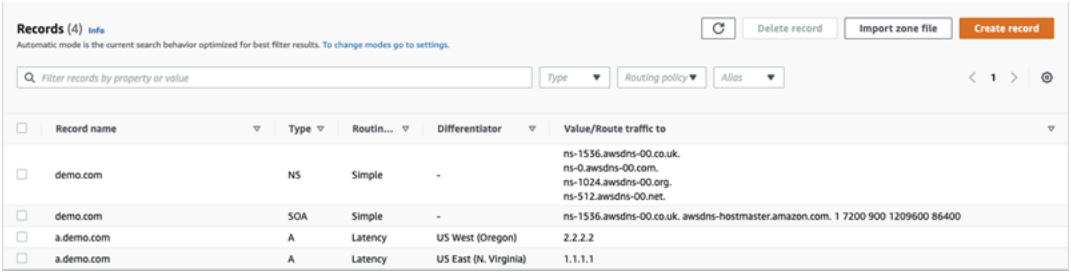
Latency-based routing in private hosted zones

For private hosted zones, Route 53 answers DNS queries with an endpoint that is in the same AWS Region, or is closest in distance to the AWS Region of the VPC that the query originated from.

> **Note**
>
> If you have an outbound endpoint forwarded to an inbound endpoint, the record will resolve based on where the inbound endpoint is, not the outbound endpoint.

If you include health checks, and the record with the lowest latency to the query's origin is unhealthy, a healthy endpoint with the next lowest latency is returned.

In the example configuration in the following figure, DNS queries coming from a us-east-1 AWS Region, or closest to it, will be routed to the 1.1.1.1 endpoint. DNS queries from us-west-2, or closest to it, will be routed to the 2.2.2.2 endpoint.



IP-based routing

With IP-based routing in Amazon Route 53, you can fine-tune your DNS routing based on your knowledge of your network, applications, and clients. This helps you make the best DNS routing choices for your end users. IP-based routing gives you fine-grained control to boost performance or cut network costs by uploading your data to Route 53 as user-IP-to-endpoint mappings.

Geolocation and latency-based routing uses data that Route 53 collects and keeps current. This approach works well for most customers. IP-based routing gives you the added ability to tune routing based on what you know about your customer base. For example, a global video provider might want to route end users from a given internet service provider (ISP).

Some common use cases for IP-based routing are the following:

- You want to route end users from certain ISPs to specific endpoints so you can optimize network transit costs or performance.
- You want to add overrides to existing Route 53 routing types, such as geolocation routing, based on your knowledge of your clients' physical locations.

Managing IP ranges and associating them to a resource record set (RRSet)

For IPv4, you can use CIDR blocks between 1 and 24 bits of length, inclusive, while for IPv6, you can use CIDR blocks between 1 and 48 bits of length, inclusive. To define a zero bit CIDR block (0.0.0.0/0 or ::/0), use the default ("*") location.

For DNS queries with a CIDR longer than the one specified in the CIDR collection, Route 53 will match it to the shorter CIDR. For example, if you specify 2001:0DB8::/32 as the CIDR block in your CIDR collection and a query originates from 2001:0DB8:0000:1234::/48, it will match. If, on the other hand, you specify 2001:0DB8:0000:1234::/48 in your CIDR collection and a query originates

from 2001:0DB8::/32, this will not match and Route 53 will answer with the record for the default ("*") location.

You can group sets of CIDR blocks (or IP ranges) into CIDR locations, which are in turn grouped into reusable entities called CIDR collections:

CIDR block

An IP range in CIDR notation, for example, 192.0.2.0/24 or 2001:DB8::/32.

CIDR location

A named list of CIDR blocks. For example, example-isp-seattle = [192.0.2.0/24, 203.0.113.0/22, 198.51.100.0/24, 2001:DB8::/32]. The blocks in a CIDR location list don't have to be adjacent or the same range.

A single location can have both IPv4 and IPv6 blocks, and this location can be associated to both A and AAAA record sets, respectively.

The location name is often a location by convention, but can be any string, for example, *Company-A*.

CIDR collection

A named collection of locations. For example, mycollection = [example-isp-seattle, example-isp-tokyo].

IP-based routing resource record sets reference a location in a collection, and all resource record sets for the same record set name and type must reference the same collection. For example, if you create websites in two Regions and want to direct DNS queries from two different CIDR locations to a specific website based on the originating IP addresses, then both of those locations must be listed in the same CIDR collection.

You cannot use IP-based routing policy for records in a private hosted zone.

For information about values that you specify when you use the IP-based routing policy to create records, see the following topics:

- Values specific for IP-based records
- Values specific for IP-based alias records
- Values that are common for all routing policies
- Values that are common for alias records for all routing policies

Topics

- Creating a CIDR collection with CIDR locations and blocks
- Working with CIDR locations and blocks
- Deleting a CIDR collection
- Moving geolocation routing to IP-based routing

Creating a CIDR collection with CIDR locations and blocks

To get started, create a CIDR collection and add CIDR blocks and locations to it.

To create a CIDR collection using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **IP-based routing**, and then **CIDR collections**.
3. Select **Create CIDR collection**.
4. In the **Create CIDR collection** pane, under **Details**, enter a name for the collection.
5. Choose **Create collection** to create an empty collection.

   - or -

   In the **Create CIDR locations** section, enter a name for the CIDR location in the **CIDR location** box. The location name can be any identifying string, for example **`company 1`**, or **`Seattle`**. It doesn't have to be an actual geographic location.

   > **Important**
   >
   > The CIDR location name has a maximum length of 16 characters.

   Enter the CIDR blocks in the **CIDR blocks** box one per line. These can be IPv4 or IPv6 addresses ranging from /0 to /24 for IPv4 and /0 to /48 for IPv6.
6. After you have entered the CIDR blocks, choose **Create CIDR collection**, or **Add another location** to keep entering locations and CIDR block. You can enter multiple CIDR locations per collection.
7. After you have entered CIDR locations, choose **Create CIDR collection**.

Working with CIDR locations and blocks

To work with CIDR locations by using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **IP-based routing**, **CIDR collections** and then, in the **CIDR collections** section, choose a link to a CIDR collection in the **Collection name** list.

   On the **CIDR locations** page, you can create a CIDR location, delete it, or edit a location and its blocks.

   - To create a location, choose **Create CIDR location**.
   - In the **Create CIDR location** pane, enter a name for the location, the CIDR blocks associated with the location, and then choose **Create**.
   - To view a CIDR location and the blocks within, choose the radio button next to a location to display its name and CIDR blocks in the location pane.

     In this pane, you can also choose **Edit** to update the name of the location or its CIDR blocks. Choose **Save** when you have finished editing.
   - To delete a CIDR location and the blocks within, choose the radio button next to the location you want to delete, and then choose **Delete**. To confirm deletion, enter the location name in the text input field and choose **Delete** again.

     > **Important**
     >
     > Deleting a CIDR location can't be undone. If you have any DNS records associated with the location, your domain might become unreachable.

Deleting a CIDR collection

To delete a CIDR collection, its locations, and blocks by using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **IP-based routing** and then **CIDR collections**.

3. In the **CIDR collections** section, choose the linked name of the collection that you want to delete.
4. On the **CIDR locations** page, select each location one at a time, choose **Delete**, enter its name in the dialog box, and then choose **Delete**. You must delete each location associated with a CIDR collection before you can delete the collection.
5. After the deletion of each CIDR location is complete, on the **CIDR locations** page, choose the radio button next to the collection you want to delete, and then choose **Delete**.

Moving geolocation routing to IP-based routing

If you use geolocation or geoproximity routing and specific clients are consistently routed to a non-optimal endpoint based on their physical location or network topology, you can target those clients' public IP ranges with IP-based routing instead.

To move geolocation routing to IP-based routing

1. Identify the geolocation records you want to override. The following example shows a geolocation configuration that you want to fine-tune for California IP ranges.

   Example geolocation configuration

Record set name	Routing policy and origin	IP address of the application endpoint
example.com	Geolocation-routing (US)	`198.51.100.1`
example.com	Geolocation-routing (EU)	`198.51.100.2`

2. To override IP ranges from California to a new application endpoint, recreate the geolocation routing records under a new record set name (for example, `geo.example.com` instead of `example.com`).

   Recreated geolocation records

Record set name	Routing policy and origin	IP address of the application endpoint
geo.example.com	Geolocation-routing (US)	`198.51.100.1`

Record set name	Routing policy and origin	IP address of the application endpoint
geo.example.com	Geolocation-routing (EU)	198.51.100.2

3. Create IP-based routing records and a default record that points to your recreated geolocation routing record set.

   IP-based routing records

Record set name	Routing policy and origin	IP address of the application endpoint
example.com	IP-based routing (default)	Alias record to geo.example.com. For example, 198.51.100.1 .
example.com	IP-based routing (California IP ranges)	198.51.100.3

Multivalue answer routing

With multivalue answer routing, you can set up Amazon Route 53 to return multiple values, such as IP addresses for your web servers, in response to DNS queries. You can specify multiple values for almost any record, but with multivalue answer routing, you can also check the health of each resource. Route 53 then returns only values for healthy resources. This isn't a substitute for a load balancer, but returning multiple health-checked IP addresses is a way to use DNS to improve uptime and load sharing.

To route traffic roughly at random to multiple resources, such as web servers, you create one multivalue answer record for each resource and, optionally, link a Route 53 health check with each record. Route 53 responds to DNS queries with up to eight healthy records and gives different answers to different DNS resolvers. If a web server becomes unavailable after a resolver caches a response, client software can try another IP address in the response.

Note the following:

- If you associate a health check with a multivalue answer record, Route 53 responds to DNS queries with the corresponding IP address only when the health check is healthy.

- If you don't associate a health check with a multivalue answer record, Route 53 always considers the record to be healthy.
- If you have eight or fewer healthy records, Route 53 responds to all DNS queries with all the healthy records.
- When all records are unhealthy, Route 53 responds to DNS queries with up to eight unhealthy records.

You can use multivalue answer routing policy for records in a private hosted zone.

For information about values that you specify when you use the multivalue answer routing policy to create records, see Values specific for multivalue answer records and Values that are common for all routing policies.

Weighted routing

With weighted routing, you can link multiple resources to a single domain name (example.com) or subdomain name (acme.example.com) and choose how much traffic goes to each resource. This can be useful for load balancing and testing new versions of software.

To set up weighted routing, you create records with the same name and type for each of your resources. You give each record a relative weight that matches how much traffic you want to send to each resource. Amazon Route 53 sends traffic to a resource based on the weight you assign to the record as a share of the total weight for all records in the group:

$$\frac{\text{Weight for a specified record}}{\text{Sum of the weights for all records}}$$

For example, if you want to send a tiny part of your traffic to one resource and the rest to another resource, you might set weights of 1 and 255. The resource with a weight of 1 gets 1/256th of the traffic (1/(1+255)), and the other resource gets 255/256ths (255/(1+255)). You can slowly change the balance by changing the weights. If you want to stop sending traffic to a resource, change the weight for that record to 0.

For information about values that you specify when you use the weighted routing policy to create records, see the following topics:

- Values specific for weighted records
- Values specific for weighted alias records

- Values that are common for all routing policies
- Values that are common for alias records for all routing policies

You can use weighted routing policy for records in a private hosted zone.

Health checks and weighted routing

If you add health checks to all the records in a group of weighted records, but you give nonzero weights to some records and zero weights to others, health checks work the same as when all records have nonzero weights with the following exceptions:

- Route 53 initially considers only the nonzero weighted records, if any.
- If all the records that have a weight greater than 0 are unhealthy, then Route 53 considers the zero-weighted records.

The following table details the behavior when the 0-weight record includes a health check:

Zero-weight record with health check

	Record 1	**Record 2**	**Record 3**
Weight	1	1	0
Includes health check?	Yes	Yes	Yes
Health check status	Unhealthy	Unhealthy	Healthy
DNS query answered?	No	No	Yes
Health check status	Unhealthy	Unhealthy	Unhealthy
DNS query answered?	Yes	Yes	No

	Record 1	Record 2	Record 3
Health check status	Unhealthy	Healthy	Unhealthy
DNS query answered?	No	Yes	No
Health check status	Healthy	Healthy	Unhealthy
DNS query answered?	Yes	Yes	No
Health check status	Healthy	Healthy	Healthy
DNS query answered?	Yes	Yes	No

The following table details the behavior when the 0-weight record doesn't include a health check:

Zero-weight record without health check

	Record 1	Record 2	Record 3
Weight	1	1	0
Includes health check?	Yes	Yes	No
Health check status	Healthy	Healthy	N/A
DNS query answered?	Yes	Yes	No
Health check status	Unhealthy	Unhealthy	N/A

	Record 1	Record 2	Record 3
DNS query answered?	No	No	Yes
Health check status	Unhealthy	Healthy	N/A
DNS query answered?	No	Yes	No

AZ-affinity routing

With AZ-affinity routing, you can route traffic to resources based on the Availability Zone (AZ) that the query comes from. This helps you reduce latency and inter-zone data transfer costs by keeping traffic within an Availability Zone, and still route traffic to another location when the closest resource is unhealthy. If you haven't created a record for that Availability Zone, or if the record is unhealthy, Amazon Route 53 falls back to a record for a less specific location.

To set up AZ-affinity routing, you create records that have the same name and type, one for each client location that you want to route traffic for. When Route 53 receives a query for an AZ-affinity record set, it responds with the first healthy record in the following order of location types, from most specific to least specific:

1. **Availability Zone** – The Availability Zone that the query comes from. Specify the AZ ID, such as `use1-az1`, not the Availability Zone name, such as `us-east-1a`. An Availability Zone name can map to a different Availability Zone in each AWS account.
2. **Zone Group** – Any Availability Zone in the Zone Group. Specify the Zone Group ID, such as `us-east-1-zg-1`.
3. **Region** – Any Availability Zone in the AWS Region. Specify the AWS Region code, such as `us-east-1`.
4. **Partition** – Any AWS Region in the AWS Partition. Specify the AWS Partition for your account, such as `aws`. Use a Partition record as the fallback for queries from any AWS Region.
5. **Global** – All other queries, such as queries from the internet. Specify `global`. If no record matches where the query comes from and you didn't create a Global record, Route 53 responds with no records (a NOERROR response that contains no answers, also known as NODATA). We recommend that you always create a Global record so that every query gets an answer.

You don't have to create a record for every location type. Route 53 skips a level when you didn't create a record for the location at that level.

You can use AZ-affinity routing for records in both public and private hosted zones. In a private hosted zone, Route 53 responds only to queries from VPCs that you associated with the hosted zone, the same as for other records in private hosted zones.

Route 53 routes each query based on where it comes from:

- When a resource in a VPC, such as an Amazon EC2 instance, uses Route 53 VPC Resolver to resolve a DNS name, VPC Resolver tells Route 53 which Availability Zone the query comes from. This applies to both public and private hosted zones.
- Queries that don't come through VPC Resolver match only the Global record. Examples are queries from the internet and queries from DNS resolvers that send queries directly to Route 53 name servers.

For information about values that you specify when you use the AZ-affinity routing policy to create records, see the following topics:

- Values specific for AZ-affinity records
- Values specific for AZ-affinity alias records
- Values that are common for all routing policies
- Values that are common for alias records for all routing policies

Finding AZ IDs and Zone Groups

To find the AZ IDs for the Availability Zones in your account, check the **Service health** panel on the Amazon EC2 Dashboard, or use the describe-availability-zones CLI command. On an Amazon EC2 instance in an Availability Zone, you can also get the AZ ID of the instance from the instance metadata item `placement/availability-zone-id`. For more information, see AZ IDs.

To find the Zone Group of each Availability Zone in a Region, run the following command, replacing `us-east-1` with your Region. The output lists the AZ ID and the Zone Group (`GroupName`) of each Availability Zone.

```
aws ec2 describe-availability-zones \
    --region us-east-1 \
```

```
    --filters Name=zone-type,Values=availability-zone \
    --query "AvailabilityZones[].[ZoneId,GroupName]" \
    --output text
```

For `us-east-1`, the output includes a line that lists `use1-az1` and `us-east-1-zg-1`, which shows that the Availability Zone `use1-az1` is in the Zone Group `us-east-1-zg-1`.

If you omit the `zone-type` filter, the output can also include Local Zones and Wavelength Zones, which AZ-affinity routing doesn't support.

Health checks and AZ-affinity routing

To have Route 53 fall back to a less specific location when a resource is unhealthy, associate a health check with each AZ-affinity record, or, for an alias record, set **Evaluate target health** to **Yes**. If a non-alias record doesn't have a health check, Route 53 always considers the record healthy, which will prevent falling back to a less specific location. If all the records that match where the query comes from are unhealthy, Route 53 responds with the most specific of those records, even though it's unhealthy.

If an AZ-affinity alias record routes traffic to non-alias records in the same hosted zone, also associate health checks with those records (see Evaluate target health in Values specific for AZ-affinity alias records).

For more information, see the following topics:

- How Amazon Route 53 chooses records when health checking is configured
- Configuring failover in a private hosted zone
- Best practices for Amazon Route 53 DNS

Considerations for AZ-affinity routing

Note the following when you use AZ-affinity routing:

- **Local Zones, Wavelength Zones, and Outposts** are not supported location types.
- **Lowercase location IDs** – Location IDs are case sensitive. Specify them in lowercase, for example, `use1-az1`, `us-east-1-zg-1`, `aws`, and `global`.
- **Not an access control** – AZ-affinity routing chooses which record to respond with, but it doesn't restrict who can get a record. Don't use it to keep a record's value private.

- **Traffic Flow** – AZ-affinity routing isn't supported in traffic policies.
- **Testing** – The DNS checking tool in the Route 53 console and the `TestDNSAnswer` API don't support AZ-affinity records. To test your configuration, run a command such as `dig` from Amazon EC2 instances in the Availability Zones that you created records for. Send the queries to VPC Resolver (the default DNS server for the VPC), not directly to the Route 53 name servers for the hosted zone.

How Amazon Route 53 uses EDNS0 to estimate the location of a user

To improve the accuracy of geolocation, geoproximity, IP-based, and latency routing, Amazon Route 53 supports the edns-client-subnet extension of EDNS0. (EDNS0 adds optional extensions to the DNS protocol.) Route 53 can use edns-client-subnet only when DNS resolvers support it:

- When a browser or other viewer uses a DNS resolver that does not support edns-client-subnet, Route 53 uses the source IP address of the DNS resolver to guess where the user is and responds to geolocation queries with the DNS record for the resolver's location.
- When a browser or other viewer uses a DNS resolver that does support edns-client-subnet, the DNS resolver sends Route 53 a shortened version of the user's IP address. Route 53 finds the location of the user from the shortened IP address instead of the resolver's source IP address; this usually gives a more accurate location. Route 53 then responds to geolocation queries with the DNS record for the user's location.
- EDNS0 is not applicable to private hosted zones. For private hosted zones Route 53 uses data from the VPC Resolvers in the AWS Region that the private hosted zone is in to make geolocation, geoproximity, and latency routing decisions.

For more information about edns-client-subnet, see the EDNS Client Subnet RFC, Client Subnet in DNS Requests.

Choosing between alias and non-alias records

Amazon Route 53 *alias records* provide a Route 53–specific extension to DNS. Alias records let you route traffic to selected AWS resources, such as CloudFront distributions and Amazon S3 buckets. They also let you route traffic from one record in a hosted zone to another record.

Unlike a CNAME record, you can create an alias record at the top node of a DNS namespace, also known as the *zone apex*. For example, if you register the DNS name example.com, the zone apex

is example.com. You can't create a CNAME record for example.com, but you can create an alias record for example.com that routes traffic to www.example.com (as long as the record type for www.example.com is not CNAME).

When Route 53 receives a DNS query for an alias record, Route 53 responds with the applicable value for that resource:

- **An Amazon API Gateway custom regional API or edge-optimized API** – Route 53 responds with one or more IP addresses for your API.
- **An Amazon VPC interface endpoint** – Route 53 responds with one or more IP addresses for your interface endpoint.
- **A CloudFront distribution** – Route 53 responds with one or more IP addresses for CloudFront edge servers that can serve your content.
- **App Runner service** – Route 53 responds with one or more IP addresses.
- **An Elastic Beanstalk environment** – Route 53 responds with one or more IP addresses for the environment.
- **An Elastic Load Balancing load balancer** – Route 53 responds with one or more IP addresses for the load balancer. This includes Application Load Balancer, Classic Load Balancer and, Network Load Balancer.
- **An AWS Global Accelerator accelerator** – Route 53 responds with the IP addresses for the accelerator.
- **An OpenSearch Service** – Route 53 responds with one or more IP addresses for the OpenSearch Service custom domain.
- **An Amazon S3 bucket that is configured as a static website** – Route 53 responds with one IP address for the Amazon S3 bucket.
- **Another Route 53 record of the same type in the same hosted zone** – Route 53 responds as if the query is for the record that is referenced by the alias record (see Comparison of alias and CNAME records).
- **AWS AppSync domain name** – Route 53 responds with one or more IP addresses for your interface endpoint.

For more information, see Routing internet traffic to your AWS resources.

When you use an alias record to route traffic to an AWS resource, Route 53 detects changes in the resource on its own. For example, suppose an alias record for example.com points to an Elastic

Load Balancing load balancer at lb1-1234.us-east-2.elb.amazonaws.com. If the IP address of the load balancer changes, Route 53 starts responding to DNS queries with the new IP address.

If an alias record points to an AWS resource, you can't set the time to live (TTL); Route 53 uses the default TTL for the resource. If an alias record points to another record in the same hosted zone, Route 53 uses the TTL of the target record. For more information about the current TTL value for Elastic Load Balancing, go to Request routing in the *Elastic Load Balancing User Guide* and search for "ttl".

For information about creating records by using the Route 53 console, see Creating records by using the Amazon Route 53 console. For information about the values that you specify for alias records, see the applicable topic in Values that you specify when you create or edit Amazon Route 53 records:

- Values specific for simple alias records
- Values specific for weighted alias records
- Values specific for latency alias records
- Values specific for failover alias records
- Values specific for geolocation alias records
- Values specific for geoproximity alias records
- Values specific for AZ-affinity alias records
- Values that are common for alias records for all routing policies

Comparison of alias and CNAME records

Alias records are similar to CNAME records, but there are some important differences. The following list compares alias records and CNAME records.

Resources that you can redirect queries to

Alias records

An alias record can only redirect queries to selected AWS resources, including but not limited to the following:

- Amazon S3 buckets
- CloudFront distributions
- Another record in the same Route 53 hosted zone

For example, you can create an alias record named acme.example.com that redirects queries to an Amazon S3 bucket that is also named acme.example.com. You can also create an acme.example.com alias record that redirects queries to a record named zenith.example.com in the example.com hosted zone.

CNAME records

A CNAME record can redirect DNS queries to any DNS record. For example, you can create a CNAME record that redirects queries from acme.example.com to zenith.example.com or to acme.example.org. You don't need to use Route 53 as the DNS service for the domain that you're redirecting queries to.

Creating records that have the same name as the domain (records at the zone apex)

Alias records

In most configurations, you can create an alias record that has the same name as the hosted zone (the zone apex). The one exception is when you want to redirect queries from the zone apex (such as example.com) to a record in the same hosted zone that has a type of CNAME (such as zenith.example.com). The alias record must have the same type as the record you're routing traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

CNAME records

You can't create a CNAME record that has the same name as the hosted zone (the zone apex). This is true both for hosted zones for domain names (example.com) and for hosted zones for subdomains (zenith.example.com).

Pricing for DNS queries

Alias records

Route 53 doesn't charge for alias queries to AWS resources. For more information, see Amazon Route 53 Pricing.

CNAME records

Route 53 charges for CNAME queries.

> **Note**
>
> If you create a CNAME record that redirects to the name of another record in a Route 53 hosted zone (the same hosted zone or another hosted zone), each DNS query is charged as two queries:

- Route 53 responds to the first DNS query with the name of the record that you want to redirect to.
- Then the DNS resolver must submit another query for the record in the first response to get information about where to direct traffic, for example, the IP address of a web server.

If the CNAME record redirects to the name of a record that is hosted with another DNS service, Route 53 charges for one query. The other DNS service might charge for the second query.

Record type specified in the DNS query

Alias records

Route 53 responds to a DNS query only when the name of the alias record (such as acme.example.com) and the type of the alias record (such as A or AAAA) match the name and type in the DNS query.

CNAME records

A CNAME record redirects DNS queries for a record name regardless of the record type specified in the DNS query, such as A or AAAA.

How records are listed in dig or nslookup queries

Alias records

In the response to a dig or nslookup query, an alias record is listed as the record type that you specified when you created the record, such as A or AAAA. (The record type that you specify for an alias record depends on the resource that you're routing traffic to. For example, to route traffic to an S3 bucket, you specify A for the type.) The alias property is visible only in the Route 53 console or in the response to a programmatic request, such as an AWS CLI `list-resource-record-sets` command.

CNAME records

A CNAME record is listed as a CNAME record in response to dig or nslookup queries.

Supported DNS record types

Amazon Route 53 supports the DNS record types that are listed in this section. Each record type also includes an example of how to format the `Value` element when you are accessing Route 53 using the API.

> **Note**
>
> For record types that include a domain name, enter a fully qualified domain name, for example, *www.example.com*. The trailing dot is optional; Route 53 assumes that the domain name is fully qualified. This means that Route 53 treats *www.example.com* (without a trailing dot) and *www.example.com.* (with a trailing dot) as identical.

Route 53 provides an extension to DNS functionality known as alias records. Similar to CNAME records, alias records let you route traffic to selected AWS resources, such as CloudFront distributions and Amazon S3 buckets. For more information, including a comparison of alias and CNAME records, see Choosing between alias and non-alias records.

Topics

- A record type
- AAAA record type
- CAA record type
- CNAME record type
- DS record type
- HTTPS record type
- MX record type
- NAPTR record type
- NS record type
- PTR record type
- SOA record type
- SPF record type
- SRV record type

- SSHFP record type
- SVCB record type
- TLSA record type
- TXT record type

A record type

You use an A record to route traffic to a resource, such as a web server, using an IPv4 address in dotted decimal notation.

Example for the Amazon Route 53 console

```
192.0.2.1
```

Example for the Route 53 API

```
<Value>192.0.2.1</Value>
```

AAAA record type

You use an AAAA record to route traffic to a resource, such as a web server, using an IPv6 address in colon-separated hexadecimal format.

Example for the Amazon Route 53 console

```
2001:0db8:85a3:0:0:8a2e:0370:7334
```

Example for the Route 53 API

```
<Value>2001:0db8:85a3:0:0:8a2e:0370:7334</Value>
```

CAA record type

A CAA record specifies which certificate authorities (CAs) are allowed to issue certificates for a domain or subdomain. Creating a CAA record helps to prevent the wrong CAs from issuing certificates for your domains. A CAA record isn't a substitute for the security requirements that are

specified by your certificate authority, such as the requirement to validate that you're the owner of a domain.

You can use CAA records to specify the following:

- Which certificate authorities (CAs) can issue SSL/TLS certificates, if any
- The email address or URL to contact when a CA issues a certificate for the domain or subdomain

When you add a CAA record to your hosted zone, you specify three settings separated by spaces:

```
flags tag "value"
```

Note the following about the format for CAA records:

- The value of `tag` can contain only the characters A-Z, a-z, and 0-9.
- Always enclose `value` in quotation marks ("").
- Some CAs allow or require additional values for `value`. Specify additional values as name-value pairs, and separate them with semicolons (;), for example:

  ```
  0 issue "ca.example.net; account=123456"
  ```
- If a CA receives a request for a certificate for a subdomain (such as www.example.com) and if no CAA record for the subdomain exists, the CA submits a DNS query for a CAA record for the parent domain (such as example.com). If a record for the parent domain exists and if the certificate request is valid, the CA issues the certificate for the subdomain.
- We recommend that you consult with your CA to determine what values to specify for a CAA record.
- You can't create a CAA record and a CNAME record that have the same name because DNS doesn't allow using the same name for both a CNAME record and any other type of record.

Topics

- Authorize a CA to issue a certificate for a domain or subdomain
- Authorize a CA to issue a wildcard certificate for a domain or subdomain
- Prevent any CA from issuing a certificate for a domain or subdomain
- Request that any CA contacts you if the CA receives an invalid certificate request
- Use another setting that is supported by the CA
- Examples

Authorize a CA to issue a certificate for a domain or subdomain

To authorize a CA to issue a certificate for a domain or subdomain, create a record that has the same name as the domain or subdomain, and specify the following settings:

- **flags** – `0`
- **tag** – `issue`
- **value** – the code for the CA that you authorize to issue a certificate for the domain or subdomain

For example, suppose you want to authorize ca.example.net to issue a certificate for example.com. You create a CAA record for example.com with the following settings:

```
0 issue "ca.example.net"
```

For information about how to authorize AWS Certificate Manager to issue a certificate, see Configure a CAA record in the *AWS Certificate Manager User Guide*.

Authorize a CA to issue a wildcard certificate for a domain or subdomain

To authorize a CA to issue a wildcard certificate for a domain or subdomain, create a record that has the same name as the domain or subdomain, and specify the following settings. A wildcard certificate applies to the domain or subdomain and all of its subdomains.

- **flags** – `0`
- **tag** – `issuewild`
- **value** – the code for the CA that you authorize to issue a certificate for a domain or subdomain, and its subdomains

For example, suppose you want to authorize ca.example.net to issue a wildcard certificate for example.com, which applies to example.com and all of its subdomains. You create a CAA record for example.com with the following settings:

```
0 issuewild "ca.example.net"
```

When you want to authorize a CA to issue a wildcard certificate for a domain or subdomain, create a record that has the same name as the domain or subdomain, and specify the following settings. A wildcard certificate applies to the domain or subdomain and all of its subdomains.

Prevent any CA from issuing a certificate for a domain or subdomain

To prevent any CA from issuing a certificate for a domain or subdomain, create a record that has the same name as the domain or subdomain, and specify the following settings:

- **flags** – `0`
- **tag** – `issue`
- **value** – `";"`

For example, suppose you don't want any CA to issue a certificate for example.com. You create a CAA record for example.com with the following settings:

```
0 issue ";"
```

If you don't want any CA to issue a certificate for example.com or its subdomains, you create a CAA record for example.com with the following settings:

```
0 issuewild ";"
```

> **Note**
>
> If you create a CAA record for example.com and specify both of the following values, a CA that is using the value ca.example.net can issue the certificate for example.com:
>
> ```
> 0 issue ";"
> 0 issue "ca.example.net"
> ```

Request that any CA contacts you if the CA receives an invalid certificate request

If you want any CA that receives an invalid request for a certificate to contact you, specify the following settings:

- **flags** – `0`
- **tag** – `iodef`
- **value** – the URL or email address that you want the CA to notify if the CA receives an invalid request for a certificate. Use the applicable format:

  `"mailto:`*`email-address`*`"`

```
"http://URL"

"https://URL"
```

For example, if you want any CA that receives an invalid request for a certificate to send email to admin@example.com, you create a CAA record with the following settings:

```
0 iodef "mailto:admin@example.com"
```

Use another setting that is supported by the CA

If your CA supports a feature that isn't defined in the RFC for CAA records, specify the following settings:

- **flags** – 128 (This value prevents the CA from issuing a certificate if the CA doesn't support the specified feature.)
- **tag** – the tag that you authorize the CA to use
- **value** – the value that corresponds with the value of tag

For example, suppose your CA supports sending a text message if the CA receives an invalid certificate request. (We aren't aware of any CAs that support this option.) Settings for the record might be the following:

```
128 exampletag "15555551212"
```

Examples

Example for the Route 53 console

```
0 issue "ca.example.net"
0 iodef "mailto:admin@example.com"
```

Example for the Route 53 API

```
<ResourceRecord>
   <Value>0 issue "ca.example.net"</Value>
   <Value>0 iodef "mailto:admin@example.com"</Value>
</ResourceRecord>
```

CNAME record type

A CNAME record maps DNS queries for the name of the current record, such as acme.example.com, to another domain (example.com or example.net) or subdomain (acme.example.com or zenith.example.org).

> **Important**
>
> The DNS protocol does not allow you to create a CNAME record for the top node of a DNS namespace, also known as the zone apex. For example, if you register the DNS name example.com, the zone apex is example.com. You cannot create a CNAME record for example.com, but you can create CNAME records for www.example.com, newproduct.example.com, and so on.
>
> In addition, if you create a CNAME record for a subdomain, you cannot create any other records for that subdomain. For example, if you create a CNAME for www.example.com, you cannot create any other records for which the value of the **Name** field is www.example.com.

Amazon Route 53 also supports alias records, which allow you to route queries to selected AWS resources, such as CloudFront distributions and Amazon S3 buckets. Aliases are similar in some ways to the CNAME record type; however, you can create an alias for the zone apex. For more information, see Choosing between alias and non-alias records.

Example for the Route 53 console

```
hostname.example.com
```

Example for the Route 53 API

```
<Value>hostname.example.com</Value>
```

DS record type

A delegation signer (DS) record refers a zone key for a delegated subdomain zone. You might create a DS record when you establish a chain of trust when you configure DNSSEC signing. For more information about configuring DNSSEC in Route 53, see Configuring DNSSEC signing in Amazon Route 53.

The first three values are decimal numbers representing the key tag, algorithm, and digest type. The fourth value is the digest of the zone key. For more information about the DS record format, see RFC 4034.

Example for the Route 53 console

```
123 4 5 1234567890abcdef1234567890absdef
```

Example for the Route 53 API

```
<Value>123 4 5 1234567890abcdef1234567890absdef</Value>
```

HTTPS record type

An HTTPS resource record is a form of the Service Binding (SVCB) DNS record that provides extended configuration information, enabling a client to easily and securely connect to a service with an HTTP protocol. The configuration information is provided in parameters that allow the connection in one DNS query, rather than necessitating multiple DNS queries.

The format for an HTTPS resource record is:

```
SvcPriority TargetName SvcParams(optional)
```

The following parameters are described in RFC 9460, section 9.1.

SvcPriority

An integer that represents the priority. 0 priority means alias mode, and is generally intended for aliasing at the zone apex. This value is an integer 0-32767 for Route 53 of which 1-32767 are service mode records. Lower the priority, higher the preference.

TargetName

The domain name of either the alias target (for alias mode) or the alternate endpoint (for ServiceMode).

SvcParams (optional)

A whitespace-separated list, with each parameter consisting of a Key=Value pair or a standalone key. If there is more than one value, they are presented as a comma-separated list. The following are the defined SvcParams:

- `1:alpn` – Application Layer Protocol Negotiation Protocol IDs. Default is HTTP/1.1, h2 is HTTP/2 over TLS, and h3 is HTTP/3 (HTTP over QUIC protocol).
- `2:no-default-alpn` – The default is not supported and you must provide an `alpn` parameter.
- `3:port` – the alternate endpoint, or the port where the service can be reached.
- `4:ipv4hint` – IPv4 address hints.
- `5:ech` – Encrypted Client Hello.
- `6:ipv6hint` – IPv6 address hints.
- `7:dohpath` – DNS over HTTPS template
- `8:ohttp` – The service operates an Oblivious HTTP target

Example for the Amazon Route 53 console for alias mode

```
0 example.com
```

Example for the Amazon Route 53 console for service mode

```
16 example.com alpn="h2,h3" port=808
```

Example for the Amazon Route 53 API for alias mode

```
<Value>0 example.com</Value>
```

Example for the Route 53 API for service mode

```
<Value>16 example.com alpn="h2,h3" port=808</Value>
```

For more information, see RFC 9460, Service Binding and Parameter Specification via the DNS (SVCB and HTTPS Resource Records).

> **Note**
>
> Route 53 does not support the arbitrary unknown-key presentation format `keyNNNNN`

MX record type

An MX record specifies the names of your mail servers and, if you have two or more mail servers, the priority order. Each value for an MX record contains two values, priority and domain name.

Priority

An integer that represents the priority for an email server. If you specify only one server, the priority can be any integer between 0 and 65535. If you specify multiple servers, the value that you specify for the priority indicates which email server you want email to be routed to first, second, and so on. The server with the lowest value for the priority takes precedence. For example, if you have two email servers and you specify values of 10 and 20 for the priority, email always goes to the server with a priority of 10 unless it's unavailable. If you specify values of 10 and 10, email is routed to the two servers approximately equally.

Domain name

The domain name of the email server. Specify the name (such as mail.example.com) of an A or AAAA record. In RFC 2181, Clarifications to the DNS Specification, section 10.3 forbids specifying the name of a CNAME record for the domain name value. (When the RFC mentions "alias," it means a CNAME record, not a Route 53 alias record.)

Example for the Amazon Route 53 console

```
10 mail.example.com
```

Example for the Route 53 API

```
<Value>10 mail.example.com</Value>
```

NAPTR record type

A Name Authority Pointer (NAPTR) is a type of record that is used by Dynamic Delegation Discovery System (DDDS) applications to convert one value to another or to replace one value with another. For example, one common use is to convert phone numbers into SIP URIs.

The `Value` element for an NAPTR record consists of six space-separated values:

Order

When you specify more than one record, the sequence that you want the DDDS application to evaluate records in. Valid values: 0-65535.

Preference

When you specify two or more records that have the same **Order**, your preference for the sequence that those records are evaluated in. For example, if two records have an **Order** of 1, the DDDS application first evaluates the record that has the lower **Preference**. Valid values: 0-65535.

Flags

A setting that is specific to DDDS applications. Values currently defined in RFC 3404 are uppercase- and lowercase letters **"A"**, **"P"**, **"S"**, and **"U"**, and the empty string, **""**. Enclose **Flags** in quotation marks.

Service

A setting that is specific to DDDS applications. Enclose **Service** in quotation marks.

For more information, see the applicable RFCs:

- **URI DDDS application** – https://tools.ietf.org/html/rfc3404#section-4.4
- **S-NAPTR DDDS application** – https://tools.ietf.org/html/rfc3958#section-6.5
- **U-NAPTR DDDS application** – https://tools.ietf.org/html/rfc4848#section-4.5

Regexp

A regular expression that the DDDS application uses to convert an input value into an output value. For example, an IP phone system might use a regular expression to convert a phone number that is entered by a user into a SIP URI. Enclose **Regexp** in quotation marks. Specify either a value for **Regexp** or a value for **Replacement**, but not both.

The regular expression can include any of the following printable ASCII characters:

- a-z
- 0-9
- - (hyphen)
- (space)
- ! # $ % & ' ( ) * + , - / : ; < = > ? @ [ ] ^ _ ` { | } ~ .
- " (quotation mark). To include a literal quote in a string, precede it with a \ character: \".

- \ (backslash). To include a backslash in a string, precede it with a \ character: \\.

Specify all other values, such as internationalized domain names, in octal format.

For the syntax for **Regexp**, see RFC 3402, section 3.2, Substitution Expression Syntax

Replacement

The fully qualified domain name (FQDN) of the next domain name that you want the DDDS application to submit a DNS query for. The DDDS application replaces the input value with the value that you specify for **Replacement**, if any. Specify either a value for **Regexp** or a value for **Replacement**, but not both. If you specify a value for **Regexp**, specify a dot (**.**) for **Replacement**.

The domain name can include a-z, 0-9, and - (hyphen).

For more information about DDDS applications and about NAPTR records, see the following RFCs:

- RFC 3401
- RFC 3402
- RFC 3403
- RFC 3404

Example for the Amazon Route 53 console

```
100 50 "u" "E2U+sip" "!^(\\+441632960083)$!sip:\\1@example.com!" .
100 51 "u" "E2U+h323" "!^\\+441632960083$!h323:operator@example.com!" .
100 52 "u" "E2U+email:mailto" "!^.*$!mailto:info@example.com!" .
```

Example for the Route 53 API

```
<ResourceRecord>
   <Value>100 50 "u" "E2U+sip" "!^(\\+441632960083)$!sip:\\1@example.com!" .</Value>
   <Value>100 51 "u" "E2U+h323" "!^\\+441632960083$!h323:operator@example.com!" .</
Value>
   <Value>100 52 "u" "E2U+email:mailto" "!^.*$!mailto:info@example.com!" .</Value>
</ResourceRecord>
```

NS record type

An NS record identifies the name servers for the hosted zone. Note the following:

- The most common use for an NS record is to control how internet traffic is routed for a domain. To use the records in a hosted zone to route traffic for a domain, you update the domain registration settings to use the four name servers in the default NS record. (This is the NS record that has the same name as the hosted zone.)
- You can create a separate hosted zone for a subdomain (acme.example.com) and use that hosted zone to route internet traffic for the subdomain and its subdomains (subdomain.acme.example.com). You set up this configuration, known as "delegating responsibility for a subdomain to a hosted zone" by creating another NS record in the hosted zone for the root domain (example.com). For more information, see Routing traffic for subdomains.
- You also use NS records to configure white-label name servers. For more information, see Configuring white-label name servers.
- Another use for an NS record is for private hosted zones when you create a delegate rule to delegate the authority for a subdomain to your on-premises resolver. You must create this NS record before you create a delegate rule. For more information, see How Resolver endpoints forward DNS queries from your VPCs to your network.

For more information about NS records, see NS and SOA records that Amazon Route 53 creates for a public hosted zone.

Example for the Amazon Route 53 console

```
ns-1.example.com
```

Example for the Route 53 API

```
<Value>ns-1.example.com</Value>
```

PTR record type

A PTR record maps an IP address to the corresponding domain name.

Example for the Amazon Route 53 console

```
hostname.example.com
```

Example for the Route 53 API

```
<Value>hostname.example.com</Value>
```

SOA record type

A start of authority (SOA) record provides information about a domain and the corresponding Amazon Route 53 hosted zone. For information about the fields in an SOA record, see NS and SOA records that Amazon Route 53 creates for a public hosted zone.

Example for the Route 53 console

```
ns-2048.awsdns-64.net hostmaster.awsdns.com 1 1 1 1 60
```

Example for the Route 53 API

```
<Value>ns-2048.awsdns-64.net hostmaster.awsdns.com 1 1 1 1 60</Value>
```

SPF record type

SPF records were formerly used to verify the identity of the sender of email messages. However, we no longer recommend that you create records for which the record type is SPF. RFC 7208, *Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1*, has been updated to say, "...[I]ts existence and mechanism defined in [RFC4408] have led to some interoperability issues. Accordingly, its use is no longer appropriate for SPF version 1; implementations are not to use it." In RFC 7208, see section 14.1, The SPF DNS Record Type.

Instead of an SPF record, we recommend that you create a TXT record that contains the applicable value. For more information about valid values, see the Wikipedia article Sender Policy Framework.

Example for the Amazon Route 53 console

```
"v=spf1 ip4:192.168.0.1/16 -all"
```

Example for the Route 53 API

```
<Value>"v=spf1 ip4:192.168.0.1/16 -all"</Value>
```

SRV record type

An SRV record `Value` element consists of four space-separated values. The first three values are decimal numbers representing priority, weight, and port. The fourth value is a domain name.

SRV records are used for accessing services, such as a service for email or communications. For information about SRV record format, refer to the documentation for the service that you want to connect to.

Example for the Amazon Route 53 console

```
10 5 80 hostname.example.com
```

Example for the Route 53 API

```
<Value>10 5 80 hostname.example.com</Value>
```

SSHFP record type

A Secure Shell fingerprint record (SSHFP) identifies SSH keys associated with the domain name. SSHFP records must be secured with DNSSEC for a chain of trust to be established. For more information about DNSSEC, see Configuring DNSSEC signing in Amazon Route 53

The format for an SSHFP resource record is:

```
[Key Algorithm] [Hash Type] Fingerprint
```

The following parameters are defined in RFC 4255.

Key Algorithm

Algorithm type:

- `0` – Reserved and not used.
- `1`: RSA – Rivest–Shamir–Adleman algorithm is one of the first public-key cryptosystems and is still in use for secure data transmission.
- `2`: DSA – Digital Signature Algorithm is a Federal Information Processing Standard for digital signatures. DSA is based on modular exponentiation and the discrete logarithm mathematical models.
- `3`: ECDSA – Elliptic Curve Digital Signature Algorithm is a variant of the DSA that uses elliptic curve cryptography.
- `4`: Ed25519 – Ed25519 algorithm is the EdDSA signature scheme that uses SHA-512 (SHA-2) and Curve25519.

- 6: Ed448 – Ed448 is the EdDSA signature scheme that uses SHAKE256 and Curve448.

Hash Type

Algorithm used to create the public key hash:

- 0 –Reserved and not used.
- 1: SHA-1
- 2: SHA-256

Fingerprint

Hexadecimal representation of the hash.

Example for the Amazon Route 53 console

```
1 1 09F6A01D2175742B257C6B98B7C72C44C4040683
```

Example for the Route 53 API

```
<Value>1 1 09F6A01D2175742B257C6B98B7C72C44C4040683</Value>
```

For more information, see RFC 4255: Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints.

SVCB record type

You use an SVCB record to deliver configuration information for accessing service endpoints. The SVCB is a generic DNS record and can be used to negotiate parameters for a variety of application protocols.

The format for an SVCB resource record is:

```
SvcPriority TargetName SvcParams(optional)
```

The following parameters are described in RFC 9460, section 2.3.

SvcPriority

An integer that represents the priority. 0 priority means alias mode, and is generally intended for aliasing at the zone apex. Lower the priority, higher the preference.

TargetName

The domain name of either the alias target (for alias mode) or the alternate endpoint (for ServiceMode).

SvcParams (optional)

A whitespace-separated list, with each parameter consisting of a Key=Value pair or a standalone key. If there is more than one value, they are presented as a comma-separated list. This value is an integer 0-32767 for Route 53 of which 1-32767 are service mode records. The following are the defined SvcParams:

- `1:alpn` – Application Layer Protocol Negotiation Protocol IDs. Default is HTTP/1.1, h2 is HTTP/2 over TLS, and h3 is HTTP/3 (HTTP over QUIC protocol).
- `2:no-default-alpn` – The default is not supported and you must provide an `alpn` parameter.
- `3:port` – the port for the alternate endpoint where the service can be reached.
- `4:ipv4hint` – IPv4 address hints.
- `5:ech` – Encrypted Client Hello.
- `6:ipv6hint` – IPv6 address hints.
- `7:dohpath` – DNS over HTTPS template
- `8:ohttp` – The service operates an Oblivious HTTP target

Example for the Amazon Route 53 console for alias mode

```
0 example.com
```

Example for the Amazon Route 53 console for service mode

```
16 example.com alpn="h2,h3" port=808
```

Example for the Amazon Route 53 API for alias mode

```
<Value>0 example.com</Value>
```

Example for the Route 53 API for service mode

```
<Value>16 example.com alpn="h2,h3" port=808</Value>
```

For more information, see RFC 9460, Service Binding and Parameter Specification via the DNS (SVCB and HTTPS Resource Records).

> **Note**
>
> Route 53 does not support the arbitrary unknown-key presentation format `keyNNNNN`

TLSA record type

You use a TLSA record to use DNS-Based Authentication of Named Entities (DANE). A TLSA record associates a certificate/public key with a Transport Layer Security (TLS) endpoint, and clients can validate the certificate/public key using a TLSA record signed with DNSSEC.

TLSA records can only be trusted if DNSSEC is enabled on your domain. For more information about DNSSEC, see Configuring DNSSEC signing in Amazon Route 53

The format for a TLSA resource record is:

```
[Certificate usage] Selector [Matching type] [Certificate association data]
```

The following parameters are specified in RFC 6698, section 3.

Certificate usage

Specifies the provided association that will be used to match the certificate presented in the TLS handshake:

- 0: CA Constraint – The certificate or public key must be found in any of the Public Key Infrastructure (PKIX) certification paths for the end entity certificate provided by the server in TLS. This constraint limits which CAs can be used to issue certificates for a specified service.
- 1: Service Certificate Constraint – Specifies an end entity certificate (or the public key) that must match with the end entity certificate given by the server in TLS. This certification limits which end entity certificate can be used by a specified service on a host.
- 2: A trust Anchor Assertion – Specifies a certificate (or the public key) that must be used as the “trust anchor” when validating the end entity certificate given by the server in TLS. Allows a domain administrator to specify a trust anchor.
- 3: Domain-Issued Certification – Specifies a certificate (or the public key) that must match the end entity certificate given by the server in TLS. This certification allows for a domain

administrator to issue certificates for a domain without involving a third-party CA. This certificate does not need to pass PKIX validation.

Selector

Specifies which part of the certificate presented by the server in the handshake is matched against the association value:

- 0: The entire certificate must be matched.
- 1: The Subject Public Key, or the DER-encoded binary structure, must be matched.

Matching type

Specifies the presentation (as determined by the Selector field) of the certificate match:

- 0: Exact match of the content.
- 1: SHA-256 hash.
- 2: SHA-512 hash.

Certificate association data

The data to be matched based on the settings of the other fields.

Example for the Amazon Route 53 console

```
0 0 1 d2abde240d7cd3ee6b4b28c54df034b97983a1d16e8a410e4561cb106618e971
```

Example for the Route 53 API

```
<Value>0 0 1 d2abde240d7cd3ee6b4b28c54df034b97983a1d16e8a410e4561cb106618e971</Value>
```

For more information, see RFC 6698, The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA.

TXT record type

A TXT record contains one or more strings that are enclosed in double quotation marks ("). When you use the simple routing policy, include all values for a domain (example.com) or subdomain (www.example.com) in the same TXT record.

Topics

- Entering TXT record values

- Special characters in a TXT record value
- Uppercase and lowercase in a TXT record value
- Examples

Entering TXT record values

A single string can include up to 255 characters, including the following:

- a-z
- A-Z
- 0-9
- Space
- - (hyphen)
- ! " # $ % & ' ( ) * + , - / : ; < = > ? @ [\ ] ^ _ ` { | } ~ .

If you need to enter a value longer than 255 characters, break the value into strings of 255 characters or fewer, and enclose each string in double quotation marks ("). In the console, list all the strings on the same line:

```
"String 1" "String 2" "String 3"
```

For the API, include all the strings in the same `Value` element:

```
<Value>"String 1" "String 2" "String 3"</Value>
```

The maximum length of a value in a TXT record is 4,000 characters.

To enter more than one TXT value, enter one value per row.

Special characters in a TXT record value

If your TXT record contains any of the following characters, you must specify the characters by using escape codes in the format *three-digit octal code*:

- Characters 000 to 040 octal (0 to 32 decimal, 0x00 to 0x20 hexadecimal)
- Characters 177 to 377 octal (127 to 255 decimal, 0x7F to 0xFF hexadecimal)

For example, if the value of your TXT record is `"exämple.com"`, you specify `"ex\344mple.com"`.

For a mapping between ASCII characters and octal codes, perform an internet search for "ASCII octal codes." One useful reference is ASCII Code - The extended ASCII table.

To include a quotation mark (") in a string, put a backslash (\) character before the quotation mark: `\"`.

Uppercase and lowercase in a TXT record value

Case is preserved, so `"Ab"` and `"aB"` are different values.

Examples

Example for the Amazon Route 53 console

Put each value on a separate line:

```
"This string includes \"quotation marks\"."
"The last character in this string is an accented e specified in octal format: \351"
"v=spf1 ip4:192.168.0.1/16 -all"
```

Example for the Route 53 API

Put each value in a separate `Value` element:

```
<Value>"This string includes \"quotation marks\"."</Value>
<Value>"The last character in this string is an accented e specified in octal format:
 \351"</Value>
<Value>"v=spf1 ip4:192.168.0.1/16 -all"</Value>
```

Creating records by using the Amazon Route 53 console

The following procedure explains how to create records using the Amazon Route 53 console. For information about how to create records using the Route 53 API, see ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.

> **Note**
>
> To create records for complex routing setups, you can also use the Traffic Flow visual editor and save the setup as a traffic policy. You can then link the traffic policy to one or more

domain names (such as example.com) or subdomain names (such as www.example.com), in the same hosted zone or in multiple hosted zones. You can also roll back updates if the new setup isn't working as expected. For more information, see Using Traffic Flow to route DNS traffic.

To create a record using the Route 53 console

1. If you're not creating an alias record, go to step 2.

   Also go to step 2 if you're creating an alias record that routes DNS traffic to an AWS resource other than an Elastic Load Balancing load balancer or another Route 53 record.

   If you're creating an alias record that routes traffic to an Elastic Load Balancing load balancer, and if you created your hosted zone and your load balancer using different accounts, perform the procedure Getting the DNS name for an Elastic Load Balancing load balancer to get the DNS name for the load balancer.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Hosted zones**.
4. If you already have a hosted zone for your domain, skip to step 5. If you don't, perform the applicable procedure to create a hosted zone:

   - To route internet traffic to your resources, such as Amazon S3 buckets or Amazon EC2 instances, see Creating a public hosted zone.
   - To route traffic in your VPC, see Creating a private hosted zone.
5. On the **Hosted zones** page, choose the name of the hosted zone that you want to create records in.
6. Choose **Create record**.
7. Choose and define the applicable routing policy and values. For more information, see the topic for the kind of record that you want to create:

   - Values that are common for all routing policies
   - Values that are common for alias records for all routing policies
   - Values specific for simple records
   - Values specific for simple alias records

- Values specific for failover records
- Values specific for failover alias records
- Values specific for geolocation records
- Values specific for geolocation alias records
- Values specific for geoproximity records
- Values specific for geoproximity alias records
- Values specific for latency records
- Values specific for latency alias records
- Values specific for IP-based records
- Values specific for IP-based alias records
- Values specific for multivalue answer records
- Values specific for weighted records
- Values specific for weighted alias records
- Values specific for AZ-affinity records
- Values specific for AZ-affinity alias records

8. Choose **Create records**.

> **Note**
>
> Your new records take time to propagate to the Route 53 DNS servers. Currently, the only way to verify that changes have propagated is to use the GetChange API action. Changes generally propagate to all Route 53 name servers within 60 seconds.

9. If you're creating multiple records, repeat steps 7 through 8.

Getting the DNS name for an Elastic Load Balancing load balancer

1. Sign in to the AWS Management Console using the AWS account that was used to create the Classic, Application, or Network Load Balancer that you want to create an alias record for.
2. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
3. In the navigation pane, choose **Load Balancers**.
4. In the list of load balancers, select the load balancer for which you want to create an alias record.

5. On the **Description** tab, get the value of **DNS name**.
6. If you want to create alias records for other Elastic Load Balancing load balancers, repeat steps 4 and 5.
7. Sign out of the AWS Management Console.
8. Sign in to the AWS Management Console again using the AWS account that you used to create the Route 53 hosted zone.
9. Return to step 3 of the procedure Creating records by using the Amazon Route 53 console.

Resource record set permissions

Resource record set permissions use Identity and Access management (IAM) policy conditions to let you set fine-grained permissions for actions on the Route 53 console or when using the ChangeResourceRecordSets API.

A resource record set is a group of DNS records with the same name and type (and class, but for most uses the class is always IN, or internet), but they hold different data. For example, if you choose geolocation routing, you can have multiple A or AAAA records pointing to different endpoints for the same domain. All of these A or AAAA records form a resource record set. For more information about DNS terms, see RFC 7719.

With the IAM policy conditions, `route53:ChangeResourceRecordSetsNormalizedRecordNames`, `route53:ChangeResourceRecordSetsRecordTypes`, and `route53:ChangeResourceRecordSetsActions`, you can grant granular administrative rights to other AWS users in any other AWS account. This allows you to grant someone permissions to:

- A single resource record set.
- All resource record sets of a specific DNS record type.
- Resource record sets where the names contain a specific string.
- Perform any, or all of the `CREATE | UPSERT | DELETE` actions when using the ChangeResourceRecordSets API, or the Route 53 console.

You can also create access permissions that combine any of the Route 53 policy conditions. For example, you can grant someone permissions to modify the A record data for marketing-example.com, but not allow that user to delete any records.

For more information about resource record set permissions and examples of how to use them, see Using IAM policy conditions for fine-grained access control.

To learn how to authenticate AWS users, see Authenticating with identities and to learn how to control access to Route 53 resources, see Access control.

Values that you specify when you create or edit Amazon Route 53 records

When you create records using the Amazon Route 53 console, the values that you specify depend on the routing policy that you want to use and on whether you're creating alias records, which route traffic to AWS resources.

Alias records that route traffic to certain AWS resources for which you specify the target resource (for example, Elastic Load Balancing, CloudFront distribution, Amazon S3 bucket). You can also optionally associate health checks and configure target health evaluation. The following topics provide detailed information on the values required for each routing policy and record type, helping you configure your Route 53 records effectively.

Topics

- Values that are common for all routing policies
- Values that are common for alias records for all routing policies
- Values specific for simple records
- Values specific for simple alias records
- Values specific for failover records
- Values specific for failover alias records
- Values specific for geolocation records
- Values specific for geolocation alias records
- Values specific for geoproximity records
- Values specific for geoproximity alias records
- Values specific for latency records
- Values specific for latency alias records
- Values specific for IP-based records
- Values specific for IP-based alias records

- Values specific for multivalue answer records
- Values specific for weighted records
- Values specific for weighted alias records
- Values specific for AZ-affinity records
- Values specific for AZ-affinity alias records

Values that are common for all routing policies

These are the common values that you can specify when you create or edit Amazon Route 53 records. These values are used by all routing policies.

Topics

- Record name
- Value/Route traffic to
- TTL (seconds)

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Name** field.

CNAME records

If you're creating a record that has a value of **CNAME** for **Record type**, the name of the record can't be the same as the name of the hosted zone.

Special characters

For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.

Wildcard characters

You can use an asterisk (*) character in the name. DNS treats the * character either as a wildcard or as the * character (ASCII 42), depending on where it appears in the name. For more information, see Using an asterisk (*) in the names of hosted zones and records.

> **Important**
>
> You can't use the * wildcard for resource records sets that have a type of **NS**.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

A — IPv4 address

An IP address in IPv4 format, for example, **192.0.2.235**.

AAAA — IPv6 address

An IP address in IPv6 format, for example, **2001:0db8:85a3:0:0:8a2e:0370:7334**.

CAA — Certificate Authority Authorization

Three space-separated values that control which certificate authorities are allowed to issue certificates or wildcard certificates for the domain or subdomain that is specified by **Record name**. You can use CAA records to specify the following:

- Which certificate authorities (CAs) can issue SSL/TLS certificates, if any
- The email address or URL to contact when a CA issues a certificate for the domain or subdomain

CNAME — Canonical name

The fully qualified domain name (for example, *www.example.com*) that you want Route 53 to return in response to DNS queries for this record. A trailing dot is optional; Route 53 assumes that the domain name is fully qualified. This means that Route 53 treats *www.example.com* (without a trailing dot) and *www.example.com.* (with a trailing dot) as identical.

MX — Mail exchange

A priority and a domain name that specifies a mail server, for example, **10 mailserver.example.com**. The trailing dot is treated as optional.

NAPTR — Name Authority Pointer

Six space-separated settings that are used by Dynamic Delegation Discovery System (DDDS) applications to convert one value to another or to replace one value with another. For more information, see NAPTR record type.

PTR — Pointer

The domain name that you want Route 53 to return.

NS — Name server

The domain name of a name server, for example, **ns1.example.com**.

> **Note**
>
> You can specify an NS record with only simple routing policy.

SPF — Sender Policy Framework

An SPF record enclosed in quotation marks, for example, **"v=spf1 ip4:192.168.0.1/16-all"**. SPF records are not recommended. For more information, see Supported DNS record types.

SRV — Service locator

An SRV record. SRV records are used for accessing services, such as a service for email or communications. For information about SRV record format, refer to the documentation for the service that you want to connect to. Trailing dot is treated as optional.

The format of an SRV record is:

[priority] [weight] [port] [server host name]

For example:

1 10 5269 xmpp-server.example.com.

TXT — Text

A text record. Enclose text in quotation marks, for example, **"Sample text entry"**.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect. Recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, we recommend that you specify a shorter value first, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

If you're associating this record with a health check, see Best practices for Amazon Route 53 DNS for TTL recommendations.

Values that are common for alias records for all routing policies

These are the common alias values that you can specify when you create or edit Amazon Route 53 records. These values are used by all routing policies.

Topics

- Record name
- Value/route traffic to

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Name** field.

CNAME records

If you're creating a record that has a value of **CNAME** for **Type**, the name of the record can't be the same as the name of the hosted zone.

Aliases to CloudFront distributions and Amazon S3 buckets

The value that you specify depends in part on the AWS resource that you're routing traffic to:

- **CloudFront distribution** – Your distribution must include an alternate domain name that matches the name of the record. For example, if the name of the record is **acme.example.com**, your CloudFront distribution must include **acme.example.com** as one of the alternate domain names. For more information, see Using alternate domain names (CNAMEs) in the *Amazon CloudFront Developer Guide*.
- **Amazon S3 bucket** – The name of the record must match the name of your Amazon S3 bucket. For example, if the name of your bucket is **acme.example.com**, the name of this record must also be **acme.example.com**.

  In addition, you must configure the bucket for website hosting. For more information, see Configure a bucket for website hosting in the *Amazon Simple Storage Service User Guide*.

Special characters

For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.

Wildcard characters

You can use an asterisk (*) character in the name. DNS treats the * character either as a wildcard or as the * character (ASCII 42), depending on where it appears in the name. For more information, see Using an asterisk (*) in the names of hosted zones and records.

Value/route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

Important

If you used the same AWS account to create your hosted zone and the resource that you're routing traffic to, and if your resource doesn't appear in the **Endpoint** list, check the following:

- Confirm that you chose a supported value for **Record type**. Supported values are specific to the resource that you're routing traffic to. For example, to route traffic to an S3 bucket, you must choose **A — IPv4 address** for **Record type**.
- Confirm that the account has the IAM permissions that are required to list the applicable resources. For example, for CloudFront distributions to appear in the **Endpoint** list, the account must have permission to perform the following action: `cloudfront:ListDistributions`.

  For an example IAM policy, see Permissions required to use the Amazon Route 53 console.

If you used different AWS accounts to create the hosted zone and the resource, the **Endpoint** list doesn't display your resource. See the following documentation for your resource type to determine what value to type in **Endpoint**.

API Gateway custom regional APIs and edge-optimized APIs

For API Gateway custom regional APIs and edge-optimized APIs, do one of the following:

- **If you used the same account to create your Route 53 hosted zone and your API** – Choose **Endpoint**, and then choose an API from the list. If you have a lot of APIs, you can enter the first few characters of the API endpoint to filter the list.

  > **Note**
  >
  > The name of this record must match a custom domain name for your API, such as **api.example.com**.

- **If you used different accounts to create your Route 53 hosted zone and your API** – Enter the API endpoint for the API, such as **api.example.com**.

  If you used one AWS account to create the current hosted zone and a different account to create an API, the API won't appear in the **Endpoints** list under **API Gateway APIs**.

  If you used one account to create the current hosted zone and one or more different accounts to create all of your APIs, the **Endpoints** list shows **No targets available** under **API Gateway APIs**. For more information, see Routing traffic to an Amazon API Gateway API by using your domain name.

CloudFront distributions

For CloudFront distributions, do one of the following:

- **If you used the same account to create your Route 53 hosted zone and your CloudFront distribution** – Choose **Endpoint** and choose a distribution from the list. If you have a lot of distributions, you can enter the first few characters of the domain name for your distribution to filter the list.

  If your distribution doesn't appear in the list, note the following:

  - The name of this record must match an alternate domain name in your distribution.
  - If you just added an alternate domain name to your distribution, it might take 15 minutes for your changes to propagate to all CloudFront edge locations. Until changes have propagated, Route 53 can't know about the new alternate domain name.

- **If you used different accounts to create your Route 53 hosted zone and your distribution** – Enter the CloudFront domain name for the distribution, such as **d111111abcdef8.cloudfront.net**.

If you used one AWS account to create the current hosted zone and a different account to create a distribution, the distribution will not appear in the **Endpoints** list.

If you used one account to create the current hosted zone and one or more different accounts to create all of your distributions, the **Endpoints** list shows **No targets available** under **CloudFront distributions**.

> **Important**
>
> Do not route queries to a CloudFront distribution that has not propagated to all edge locations, or your users won't be able to access the applicable content.

Your CloudFront distribution must include an alternate domain name that matches the name of the record. For example, if the name of the record is **acme.example.com**, your CloudFront distribution must include **acme.example.com** as one of the alternate domain names. For more information, see Using alternate domain names (CNAMEs) in the *Amazon CloudFront Developer Guide*.

If IPv6 is enabled for the distribution, create two records, one with a value of **A — IPv4 address** for **Record type**, and one with a value of **AAAA — IPv6 address**. For more information, see Routing traffic to an Amazon CloudFront distribution by using your domain name.

App Runner service

For App Runner service, do one of the following:

- **If you used the same account to create your Route 53 hosted zone and your App Runner service** – choose the AWS Region, then choose the domain name of the environment that you want to route traffic to from the list.
- **If you used different accounts to create your Route 53 hosted zone and your App Runner** – Enter the custom domain name. For more information, see Managing custom domain names for App Runner.

  If you used one AWS account to create the current hosted zone and a different account to create a App Runner, the App Runner will not appear in the **Endpoints** list.

For more information, see Configuring Amazon Route 53 to route traffic to an App Runner service.

Elastic Beanstalk environments that have regionalized subdomains

If the domain name for your Elastic Beanstalk environment includes the Region that you deployed the environment in, you can create an alias record that routes traffic to the environment. For example, the domain name `my-environment.us-west-2.elasticbeanstalk.com` is a regionalized domain name.

> **Important**
>
> For environments that were created before early 2016, the domain name doesn't include the Region. To route traffic to these environments, you must create a CNAME record instead of an alias record. Note that you can't create a CNAME record for the root domain name. For example, if your domain name is example.com, you can create a record that routes traffic for acme.example.com to your Elastic Beanstalk environment, but you can't create a record that routes traffic for example.com to your Elastic Beanstalk environment.

For Elastic Beanstalk environments that have regionalized subdomains, do one of the following:

- **If you used the same account to create your Route 53 hosted zone and your Elastic Beanstalk environment** – Choose **Endpoint**, and then choose an environment from the list. If you have a lot of environments, you can enter the first few characters of the CNAME attribute for the environment to filter the list.
- **If you used different accounts to create your Route 53 hosted zone and your Elastic Beanstalk environment** – Enter the CNAME attribute for the Elastic Beanstalk environment.

For more information, see Routing traffic to an AWS Elastic Beanstalk environment.

ELB Load Balancers

For ELB load balancers, do one of the following:

- **If you used the same account to create your Route 53 hosted zone and your load balancer** – Choose **Endpoint** and choose a load balancer from the list. If you have a lot of load balancers, you can enter the first few characters of the DNS name to filter the list.
- **If you used different accounts to create your Route 53 hosted zone and your load balancer** – Enter the value that you got in the procedure Getting the DNS name for an Elastic Load Balancing load balancer.

If you used one AWS account to create the current hosted zone and a different account to create a load balancer, the load balancer will not appear in the **Endpoints** list.

If you used one account to create the current hosted zone and one or more different accounts to create all of your load balancers, the **Endpoints** list shows **No targets available** under **Elastic Load Balancers**.

The console prepends **dualstack.** for Application and Classic Load Balancer from a different account. When a client, such as a web browser, requests the IP address for your domain name (example.com) or subdomain name (www.example.com), the client can request an IPv4 address (an A record), an IPv6 address (a AAAA record), or both IPv4 and IPv6 addresses (in separate requests). The **dualstack.** designation allows Route 53 to respond with the appropriate IP address for your load balancer based on which IP address format the client requested.

For more information, see Routing traffic to an ELB load balancer.

AWS Global Accelerator accelerators

For AWS Global Accelerator accelerators, enter the DNS name for the accelerator. You can enter the DNS name of an accelerator that you created using the current AWS account or using a different AWS account.

Amazon S3 Buckets

For Amazon S3 buckets that are configured as website endpoints, do one of the following:

- **If you used the same account to create your Route 53 hosted zone and your Amazon S3 bucket** – Choose **Endpoint** and choose a bucket from the list. If you have a lot of buckets, you can enter the first few characters of the DNS name to filter the list.

  The value of **Endpoint** changes to the Amazon S3 website endpoint for your bucket.
- **If you used different accounts to create your Route 53 hosted zone and your Amazon S3 bucket** – Enter the name of the Region that you created your S3 bucket in. Use the value that appears in the **Website endpoint** column in the table Amazon S3 website endpoints in the *Amazon Web Services General Reference*.

  If you used AWS accounts other than the current account to create your Amazon S3 buckets, the bucket won't appear in the **Endpoints** list.

You must configure the bucket for website hosting. For more information, see Configure a bucket for website hosting in the *Amazon Simple Storage Service User Guide*.

The name of the record must match the name of your Amazon S3 bucket. For example, if the name of your Amazon S3 bucket is **acme.example.com**, the name of this record must also be **acme.example.com**.

In a group of weighted alias, latency alias, failover alias, or geolocation alias records, you can create only one record that routes queries to an Amazon S3 bucket because the name of the record must match the name of the bucket and bucket names must be globally unique.

Amazon OpenSearch Service

For OpenSearch Service, do one of the following:

- **OpenSearch Service custom domain**: The name of the record must match the custom domain. For example, if the name of your custom domain is test.example.com, the name of this record must also be test.example.com.
- **If you used the same account to create your Route 53 hosted zone and your OpenSearch Service domain** – choose the AWS Region, then choose the domain name.
- **If you used different accounts to create your Route 53 hosted zone and your OpenSearch Service domain** – Enter the custom domain name. For more information, see Create a custom endpoint.

  If you used one AWS account to create the current hosted zone and a different account to create a OpenSearch Service domain, the domain will not appear in the **Endpoints** list.

  If you used one account to create the current hosted zone and one or more different accounts to create all of your OpenSearch Service domains, the **Endpoints** list shows **No targets available** under **OpenSearch Service**.

For more information, see Configuring Amazon Route 53 to route traffic to an Amazon OpenSearch Service domain endpoint.

Amazon VPC interface endpoints

For Amazon VPC interface endpoints, do one of the following:

- **If you used the same account to create your Route 53 hosted zone and your interface endpoint** – Choose **Endpoint**, and then choose an interface endpoint from the list. If you have a lot of interface endpoints, you can enter the first few characters of the DNS hostname to filter the list.
- **If you used different accounts to create your Route 53 hosted zone and your interface endpoint** – Enter the DNS hostname for the interface endpoint, such

as **vpce-123456789abcdef01-example-us-east-1a.elasticloadbalancing.us-east-1.vpce.amazonaws.com**.

If you used one AWS account to create the current hosted zone and a different account to create an interface endpoint, the interface endpoint won't appear in the **Endpoint** list under **VPC endpoints**.

If you used one account to create the current hosted zone and one or more different accounts to create all of your interface endpoints, the **Endpoint** list shows **No targets available** under **VPC endpoints**.

For more information, see Routing traffic to an Amazon Virtual Private Cloud interface endpoint by using your domain name.

Records in this Hosted Zone

For records in this hosted zone, choose **Endpoint** and choose the applicable record. If you have a lot of records, you can enter the first few characters of the name to filter the list.

If the hosted zone contains only the default NS and SOA records, the **Endpoints** list shows **No targets available**.

> **Note**
>
> If you're creating an alias record that has the same name as the hosted zone (known as the *zone apex*), you can't choose a record for which the value of **Record type** is **CNAME**. This is because the alias record must have the same type as the record you're routing traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

Values specific for simple records

When you create simple records, you specify the following values.

Topics

- Routing policy
- Record name
- Value/Route traffic to
- Record type
- TTL (seconds)

Routing policy

Choose **Simple routing**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Name** field.

For more information about record names, see Record name.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**
- **AAAA — IPv6 address**

- **CAA — Certificate Authority Authorization**
- **CNAME — Canonical name**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**
- **NS — Name server**

  The domain name of a name server, for example, **ns1.example.com**.

  > **Note**
  >
  > You can specify an NS record with only simple routing policy.

- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**
- **TXT — Text**

For more information about the above values, see common values for Value/Route traffic to.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the value for **Record type** based on how you want Route 53 to respond to DNS queries.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect. Recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, specify a shorter value first, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

Values specific for simple alias records

When you create alias records, you specify the following values. For more information, see Choosing between alias and non-alias records.

> **Note**
>
> If you are using Route 53 in the AWS GovCloud (US) Region, this feature has some restrictions. For more information, see the Amazon Route 53 page in the *AWS GovCloud (US) User Guide*.

Topics

- Routing policy
- Record name
- Value/route traffic to
- Record type
- Evaluate target health

Routing policy

Choose **Simple routing**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Name** field.

For more information about record names, see Record name.

Value/route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to.

For information about what AWS resources you can target, see common values for alias records for value/route traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the applicable value based on the AWS resource that you're routing traffic to:

API Gateway custom regional API or edge-optimized API

Select **A — IPv4 address**.

Amazon VPC interface endpoints

Select **A — IPv4 address**.

CloudFront distribution

Select **A — IPv4 address**.

If IPv6 is enabled for the distribution, create two records, one with a value of **A — IPv4 address** for **Type**, and one with a value of **AAAA — IPv6 address**.

App Runner service

Select **A — IPv4 address**

Elastic Beanstalk environment that has regionalized subdomains

Select **A — IPv4 address**

ELB load balancer

Select **A — IPv4 address** or **AAAA — IPv6 address**

Amazon S3 bucket

Select **A — IPv4 address**

OpenSearch Service

Select **A — IPv4 address** or **AAAA — IPv6 address**

Another record in this hosted zone

Select the type of the record that you're creating the alias for. All types are supported except **NS** and **SOA**.

> **Note**
>
> If you're creating an alias record that has the same name as the hosted zone (known as the *zone apex*), you can't route traffic to a record for which the value of **Type** is **CNAME**. This is because the alias record must have the same type as the record you're routing traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

Evaluate target health

When the value of **Routing policy** is **Simple**, you can choose either **No** or the default **Yes** because **Evaluate target health** has no effect for **Simple** routing. If you have only one record that has a given name and type, Route 53 responds to DNS queries by using the values in that record regardless of whether the resource is healthy.

For other routing policies, **Evaluate target health** determines whether Route 53 checks the health of the resource that the alias record refers to:

- **Services where Evaluate target health provides operational benefit**: For load balancers (ELB) and AWS Elastic Beanstalk environments with load balancers, setting **Evaluate target health** to **Yes** allows Route 53 to route traffic away from unhealthy resources.
- **Highly available services**: For services like Amazon S3 buckets, VPC interface endpoints, Amazon API Gateway, AWS Global Accelerator, Amazon OpenSearch Service, and Amazon VPC Lattice, **Evaluate target health** provides no operational benefit because these services are designed for high availability. For failover scenarios with these services, use Route 53 health checks instead.

For detailed information about how **Evaluate target health** works with different AWS services, see the EvaluateTargetHealth documentation in the API reference.

Values specific for failover records

When you create failover records, you specify the following values.

> **Note**
>
> For information about creating failover records in a private hosted zone, see Configuring failover in a private hosted zone.

Topics

- Routing policy
- Record name
- Record type
- TTL (seconds)
- Value/Route traffic to
- Failover record type
- Health check
- Record ID

Routing policy

Choose **Failover**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for both of the records in the group of failover records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the same value for both the primary and secondary failover records.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect. Recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, specify a shorter value first, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

If you're associating this record with a health check, specify a TTL of 60 seconds or less so clients respond quickly to changes in health status.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**
- **AAAA — IPv6 address**
- **CAA — Certificate Authority Authorization**
- **CNAME — Canonical name**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**

- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**
- **TXT — Text**

For more information about the above values, see common values for Value/Route traffic to.

Failover record type

Choose the applicable value for this record. For failover to function correctly, you must create one primary and one secondary failover record.

You can't create non-failover records that have the same values for **Record name** and **Record type** as failover records.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on health check status. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate Target Health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for

Evaluate Target Health, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain Name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain Name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Record ID

Enter a value that uniquely identifies the primary and secondary records.

Values specific for failover alias records

When you create failover alias records, you specify the following values.

For information, see the following topics:

- For information about creating failover records in a private hosted zone, see Configuring failover in a private hosted zone.
- For information about alias records, see Choosing between alias and non-alias records.

Topics

- Routing policy
- Record name
- Record type
- Value/Route traffic to
- Failover record type
- Health check
- Evaluate target health
- Record ID

Routing policy

Choose **Failover**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for both of the records in the group of failover records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the applicable value based on the AWS resource that you're routing traffic to. Select the same value for both the primary and secondary failover records:

API Gateway custom regional API or edge-optimized API

Select **A — IPv4 address**.

Amazon VPC interface endpoints

Select **A — IPv4 address**.

CloudFront distribution

Select **A — IPv4 address**.

If IPv6 is enabled for the distribution, create two records, one with a value of **A — IPv4 address** for **Type**, and one with a value of **AAAA — IPv6 address**.

App Runner service

Select **A — IPv4 address**

Elastic Beanstalk environment that has regionalized subdomains

Select **A — IPv4 address**

ELB load balancer

Select **A — IPv4 address** or **AAAA — IPv6 address**

Amazon S3 bucket

Select **A — IPv4 address**

OpenSearch Service

Select **A — IPv4 address** or **AAAA — IPv6 address**

Another record in this hosted zone

Select the type of the record that you're creating the alias for. All types are supported except **NS** and **SOA**.

> **Note**
>
> If you're creating an alias record that has the same name as the hosted zone (known as the *zone apex*), you can't route traffic to a record for which the value of **Type** is **CNAME**. This is because the alias record must have the same type as the record you're routing traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

Value/Route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to.

For information about what AWS resources you can target, see common values for alias records for value/route traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

> **Note**
>
> When you create primary and secondary failover records, you can optionally create one failover and one failover *alias* record that have the same values for **Name** and **Record type**. If you mix failover and failover alias records, either one can be the primary record.

Failover record type

Choose the applicable value for this record. For failover to function correctly, you must create one primary and one secondary failover record.

You can't create non-failover records that have the same values for **Record name** and **Record type** as failover records.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate Target Health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain Name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain Name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Evaluate target health

Select **Yes** if you want Route 53 to determine whether to respond to DNS queries using this record by checking the health of the resource specified by **Endpoint**.

Note the following:

API Gateway custom regional APIs and edge-optimized APIs

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an API Gateway custom regional API or an edge-optimized API.

CloudFront distributions

You can't set **Evaluate target health** to **Yes** when the endpoint is a CloudFront distribution.

Elastic Beanstalk environments that have regionalized subdomains

If you specify an Elastic Beanstalk environment in **Endpoint** and the environment contains an ELB load balancer, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. (An environment automatically contains an ELB load balancer if it includes more than one Amazon EC2 instance.) If you set **Evaluate target health** to **Yes** and either no Amazon EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other available resources that are healthy, if any.

If the environment contains a single Amazon EC2 instance, there are no special requirements.

ELB load balancers

Health checking behavior depends on the type of load balancer:

- **Classic load balancers** – If you specify an ELB Classic Load Balancer in **Endpoint**, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. If you set **Evaluate target health** to **Yes** and either no EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other resources.
- **Application and Network Load Balancers** – If you specify an ELB Application or Network Load Balancer and you set **Evaluate Target health** to **Yes**, Route 53 routes queries to the load balancer based on the health of the target groups that are associated with the load balancer:
  - For an Application or Network Load Balancer to be considered healthy, a target group that contains targets must contain at least one healthy target. If any target group contains only unhealthy targets, the load balancer is considered unhealthy, and Route 53 routes queries to other resources.
  - A target group that has no registered targets is considered unhealthy.

> **Note**
>
> When you create a load balancer, you configure settings for Elastic Load Balancing health checks; they're not Route 53 health checks, but they perform a similar function. Do not create Route 53 health checks for the EC2 instances that you register with an ELB load balancer.

S3 buckets

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an S3 bucket.

Amazon VPC interface endpoints

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an Amazon VPC interface endpoint.

Other records in the same hosted zone

If the AWS resource that you specify in **Endpoint** is a record or a group of records (for example, a group of weighted records) but is not another alias record, we recommend that you associate a health check with all of the records in the endpoint. For more information, see What happens when you omit health checks?.

Record ID

Enter a value that uniquely identifies the primary and secondary records.

Values specific for geolocation records

When you create geolocation records, you specify the following values.

Topics

- Routing policy
- Record name
- Record type
- TTL (seconds)
- Value/Route traffic to
- Location
- U.S. states
- Health check
- Record ID

Routing policy

Choose **Geolocation**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Name** field.

Enter the same name for all of the records in the group of geolocation records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the same value for all of the records in the group of geolocation records.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect because recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, we recommend that you initially specify a shorter value, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

If you're associating this record with a health check, we recommend that you specify a TTL of 60 seconds or less so clients respond quickly to changes in health status.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**
- **AAAA — IPv6 address**
- **CAA — Certificate Authority Authorization**
- **CNAME — Canonical name**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**
- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**

- **TXT — Text**

For more information about the above values, see common values for Value/Route traffic to.

Location

When you configure Route 53 to respond to DNS queries based on the location that the queries originated from, select the continent or country for which you want Route 53 to respond with the settings in this record. If you want Route 53 to respond to DNS queries for individual states in the United States, select **United States** from the **Location** list, and then select the state under the **Sublocation** group.

For a private hosted zone, select the continent, country, or sub-division closest to the AWS Region that your resource is in. For example, if your resource is in us-east-1, you can specify North America, United States, or Virginia.

> **Important**
>
> We recommend that you create one geolocation record that has a value of **Default** for **Location**. This covers geographic locations that you haven't created records for and IP addresses that Route 53 can't identify a location for.

You can't create non-geolocation records that have the same values for **Record name** and **Record type** as geolocation records.

For more information, see Geolocation routing.

Here are the countries that Amazon Route 53 associates with each continent. The country codes are from ISO 3166. For more information, see the Wikipedia article ISO 3166-1 alpha-2:

Africa (AF)

AO, BF, BI, BJ, BW, CD, CF, CG, CI, CM, CV, DJ, DZ, EG, ER, ET, GA, GH, GM, GN, GQ, GW, KE, KM, LR, LS, LY, MA, MG, ML, MR, MU, MW, MZ, NA, NE, NG, RE, RW, SC, SD, SH, SL, SN, SO, SS, ST, SZ, TD, TG, TN, TZ, UG, YT, ZA, ZM, ZW

Antarctica (AN)

AQ, GS, TF

Asia (AS)

AE, AF, AM, AZ, BD, BH, BN, BT, CC, CN, GE, HK, ID, IL, IN, IO, IQ, IR, JO, JP, KG, KH, KP, KR, KW, KZ, LA, LB, LK, MM, MN, MO, MV, MY, NP, OM, PH, PK, PS, QA, SA, SG, SY, TH, TJ, TM, TW, UZ, VN, YE

Europe (EU)

AD, AL, AT, AX, BA, BE, BG, BY, CH, CY, CZ, DE, DK, EE, ES, FI, FO, FR, GB, GG, GI, GR, HR, HU, IE, IM, IS, IT, JE, LI, LT, LU, LV, MC, MD, ME, MK, MT, NL, NO, PL, PT, RO, RS, RU, SE, SI, SJ, SK, SM, TR, UA, VA, XK

> **Note**
>
> Some providers consider TR to be in Asia and the IP-addresses will reflect that.

North America (NA)

AG, AI, AW, BB, BL, BM, BQ, BS, BZ, CA, CR, CU, CW, DM, DO, GD, GL, GP, GT, HN, HT, JM, KN, KY, LC, MF, MQ, MS, MX, NI, PA, PM, PR, SV, SX, TC, TT, US, VC, VG, VI

Oceania (OC)

AS, AU, CK, FJ, FM, GU, KI, MH, MP, NC, NF, NR, NU, NZ, PF, PG, PN, PW, SB, TK, TL, TO, TV, UM, VU, WF, WS

South America (SA)

AR, BO, BR, CL, CO, EC, FK, GF, GY, PE, PY, SR, UY, VE

> **Note**
>
> Route 53 doesn't support creating geolocation records for the following countries: Bouvet Island (BV), Christmas Island (CX), Western Sahara (EH), and Heard Island and McDonald Islands (HM). No data is available about IP addresses for these countries.

U.S. states

When you configure Route 53 to respond to DNS queries based on the state of the United States that the queries originated from, select the state from the **U.S. states** list. United States territories (for example, Puerto Rico) are listed as countries in the **Location** list.

> **Important**
>
> Some IP addresses are associated with the United States, but not with an individual state. If you create records for all of the states in the United States, we recommend that you also create a record for the United States to route queries for these unassociated IP addresses. If you don't create a record for the United States, Route 53 responds to DNS queries from unassociated United States IP addresses with settings from the default geolocation record (if you created one) or with a "no answer" response.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate Target Health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for

Evaluate Target Health, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain Name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain Name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

For geolocation records, if an endpoint is unhealthy, Route 53 looks for a record for the larger, associated geographic Region. For example, suppose you have records for a state in the United States, for the United States, for North America, and for all locations (**Location** is **Default**). If the endpoint for the state record is unhealthy, Route 53 checks the records for the United States, for North America, and for all locations, in that order, until it finds a record that has a healthy endpoint. If all applicable records are unhealthy, including the record for all locations, Route 53 responds to the DNS query using the value for the record for the smallest geographic region.

Record ID

Enter a value that uniquely identifies this record in the group of geolocation records.

Values specific for geolocation alias records

When you create geolocation alias records, you specify the following values.

For more information, see Choosing between alias and non-alias records.

Topics

- Routing policy
- Record name
- Record type
- Value/Route traffic to
- Location
- U.S. states
- Health check
- Evaluate target health
- Record ID

Routing policy

Choose **Geolocation**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of geolocation records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the applicable value based on the AWS resource that you're routing traffic to. Select the same value for all of the records in the group of geolocation records:

API Gateway custom regional API or edge-optimized API

Select **A — IPv4 address**.

Amazon VPC interface endpoints

Select **A — IPv4 address**.

CloudFront distribution

Select **A — IPv4 address**.

If IPv6 is enabled for the distribution, create two records, one with a value of **A — IPv4 address** for **Type**, and one with a value of **AAAA — IPv6 address**.

App Runner service

Select **A — IPv4 address**

Elastic Beanstalk environment that has regionalized subdomains

Select **A — IPv4 address**

ELB load balancer

Select **A — IPv4 address** or **AAAA — IPv6 address**

Amazon S3 bucket

Select **A — IPv4 address**

OpenSearch Service

Select **A — IPv4 address** or **AAAA — IPv6 address**

Another record in this hosted zone

Select the type of the record that you're creating the alias for. All types are supported except **NS** and **SOA**.

> **Note**
>
> If you're creating an alias record that has the same name as the hosted zone (known as the *zone apex*), you can't route traffic to a record for which the value of **Type** is **CNAME**. This is because the alias record must have the same type as the record you're routing

traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

Value/Route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to.

For information about what AWS resources you can target, see Value/route traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

Location

When you configure Route 53 to respond to DNS queries based on the location that the queries originated from, select the continent or country for which you want Route 53 to respond with the settings in this record. If you want Route 53 to respond to DNS queries for individual states in the United States, select **United States** from the **Location** list, and then select the state from the **U.S. states** list.

For a private hosted zone, select the continent, country, or sub-division closest to the AWS Region that your resource is in. For example, if your resource is in us-east-1, you can specify North America, United States, or Virginia.

Important

We recommend that you create one geolocation record that has a value of **Default** for **Location**. This covers geographic locations that you haven't created records for and IP addresses that Route 53 can't identify a location for.

You can't create non-geolocation records that have the same values for **Record name** and **Record type** as geolocation records.

For more information, see Geolocation routing.

Here are the countries that Amazon Route 53 associates with each continent. The country codes are from ISO 3166. For more information, see the Wikipedia article ISO 3166-1 alpha-2:

Africa (AF)

AO, BF, BI, BJ, BW, CD, CF, CG, CI, CM, CV, DJ, DZ, EG, ER, ET, GA, GH, GM, GN, GQ, GW, KE, KM, LR, LS, LY, MA, MG, ML, MR, MU, MW, MZ, NA, NE, NG, RE, RW, SC, SD, SH, SL, SN, SO, SS, ST, SZ, TD, TG, TN, TZ, UG, YT, ZA, ZM, ZW

Antarctica (AN)

AQ, GS, TF

Asia (AS)

AE, AF, AM, AZ, BD, BH, BN, BT, CC, CN, GE, HK, ID, IL, IN, IO, IQ, IR, JO, JP, KG, KH, KP, KR, KW, KZ, LA, LB, LK, MM, MN, MO, MV, MY, NP, OM, PH, PK, PS, QA, SA, SG, SY, TH, TJ, TM, TW, UZ, VN, YE

Europe (EU)

AD, AL, AT, AX, BA, BE, BG, BY, CH, CY, CZ, DE, DK, EE, ES, FI, FO, FR, GB, GG, GI, GR, HR, HU, IE, IM, IS, IT, JE, LI, LT, LU, LV, MC, MD, ME, MK, MT, NL, NO, PL, PT, RO, RS, RU, SE, SI, SJ, SK, SM, TR, UA, VA, XK

> **Note**
>
> Some providers consider TR to be in Asia and the IP-addresses will reflect that.

North America (NA)

AG, AI, AW, BB, BL, BM, BQ, BS, BZ, CA, CR, CU, CW, DM, DO, GD, GL, GP, GT, HN, HT, JM, KN, KY, LC, MF, MQ, MS, MX, NI, PA, PM, PR, SV, SX, TC, TT, US, VC, VG, VI

Oceania (OC)

AS, AU, CK, FJ, FM, GU, KI, MH, MP, NC, NF, NR, NU, NZ, PF, PG, PN, PW, SB, TK, TL, TO, TV, UM, VU, WF, WS

South America (SA)

AR, BO, BR, CL, CO, EC, FK, GF, GY, PE, PY, SR, UY, VE

> **Note**
>
> Route 53 doesn't support creating geolocation records for the following countries: Bouvet Island (BV), Christmas Island (CX), Western Sahara (EH), and Heard Island and McDonald Islands (HM). No data is available about IP addresses for these countries.

U.S. states

When you configure Route 53 to respond to DNS queries based on the state of the United States that the queries originated from, select the state from the **U.S. states** list. United States territories (for example, Puerto Rico) are listed as countries in the **Location** list.

> **Important**
>
> Some IP addresses are associated with the United States, but not with an individual state. If you create records for all of the states in the United States, we recommend that you also create a record for the United States to route queries for these unassociated IP addresses. If you don't create a record for the United States, Route 53 responds to DNS queries from unassociated United States IP addresses with settings from the default geolocation record (if you created one) or with a "no answer" response.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

For geolocation records, if an endpoint is unhealthy, Route 53 looks for a record for the larger, associated geographic Region. For example, suppose you have records for a state in the United States, for the United States, for North America, and for all locations (**Location** is **Default**). If the endpoint for the state record is unhealthy, Route 53 checks the records for the United States, for North America, and for all locations, in that order, until it finds a record that has a healthy endpoint. If all applicable records are unhealthy, including the record for all locations, Route 53 responds to the DNS query using the value for the record for the smallest geographic region.

Evaluate target health

Select **Yes** if you want Route 53 to determine whether to respond to DNS queries using this record by checking the health of the resource specified by **Endpoint**.

Note the following:

API Gateway custom regional APIs and edge-optimized APIs

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an API Gateway custom Regional API or an edge-optimized API.

CloudFront distributions

You can't set **Evaluate target health** to **Yes** when the endpoint is a CloudFront distribution.

Elastic Beanstalk environments that have regionalized subdomains

If you specify an Elastic Beanstalk environment in **Endpoint** and the environment contains an ELB load balancer, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. (An environment automatically contains an ELB load balancer if it includes more than one Amazon EC2 instance.) If you set **Evaluate target health** to **Yes** and either no Amazon EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other available resources that are healthy, if any.

If the environment contains a single Amazon EC2 instance, there are no special requirements.

ELB load balancers

Health checking behavior depends on the type of load balancer:

- **Classic Load Balancers** – If you specify an ELB Classic Load Balancer in **Endpoint**, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. If you set **Evaluate target health** to **Yes** and either no EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other resources.
- **Application and Network Load Balancers** – If you specify an ELB Application or Network Load Balancer and you set **Evaluate target health** to **Yes**, Route 53 routes queries to the load balancer based on the health of the target groups that are associated with the load balancer:
  - For an Application or Network Load Balancer to be considered healthy, every target group that contains targets must contain at least one healthy target. If any target group contains only unhealthy targets, the load balancer is considered unhealthy, and Route 53 routes queries to other resources.
  - A target group that has no registered targets is considered unhealthy.

> **Note**
>
> When you create a load balancer, you configure settings for Elastic Load Balancing health checks; they're not Route 53 health checks, but they perform a similar function.

Do not create Route 53 health checks for the EC2 instances that you register with an ELB load balancer.

S3 buckets

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an S3 bucket.

Amazon VPC interface endpoints

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an Amazon VPC interface endpoint.

Other records in the same hosted zone

If the AWS resource that you specify in **Endpoint** is a record or a group of records (for example, a group of weighted records) but is not another alias record, we recommend that you associate a health check with all of the records in the endpoint. For more information, see What happens when you omit health checks?.

Record ID

Enter a value that uniquely identifies this record in the group of geolocation records.

Values specific for geoproximity records

When you create geoproximity records, you specify the following values.

Topics

- Routing policy
- Record name
- Record type
- TTL (seconds)
- Value/Route traffic to
- Endpoint location
- Bias
- Health check
- Record ID

Routing policy

Choose **Geoproximity**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Name** field.

Enter the same name for all of the records in the group of geoproximity records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the same value for all of the records in the group of geoproximity records.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect because recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, we recommend that you initially specify a shorter value, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

If you're associating this record with a health check, we recommend that you specify a TTL of 60 seconds or less so clients respond quickly to changes in health status.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**
- **AAAA — IPv6 address**
- **CAA — Certificate Authority Authorization**
- **CNAME — Canonical name**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**
- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**

- **TXT — Text**

For more information about the above values, see common values for Value/Route traffic to.

Endpoint location

You can specify the resource endpoint location by using one of the following:

Custom coordinates

Specify the longitude and lattitude for a geopgraphic area.

AWS Region

Choose an available Region from the **Location** list.

For more information about the Regions, see AWS Global Infrastructure.

AWS Local Zone Group

Choose an available Local Zone Group from the **Location** list.

For more information about Local Zones, see Available Local Zones in the *AWS Local Zones User Guide*. A local Zone Group is usually the Local Zone without the ending character. For example, if the Local Zone is `us-east-1-bue-1a` the Local Zone Group is `us-east-1-bue-1`.

You can also identify the Local Zones Group for a specific Local Zone by using the describe-availability-zones CLI command:

```
aws ec2 describe-availability-zones \
    --region us-west-2 \
    --all-availability-zones \
    --query "AvailabilityZones[?ZoneName=='us-west-2-den-1a']" \
    | grep "GroupName"
```

This command returns: `"GroupName": "us-west-2-den-1"`, specifying that the Local Zone `us-west-2-den-1a` belongs to the Local Zone Group `us-west-2-den-1`.

You can't create non-geoproximity records that have the same values for **Record name** and **Record type** as geoproximity records.

You also can't create two geoproximity resource record sets that specify the same location for the same record name and record type.

Bias

A bias either expands or shrinks a geographic area from which Route 53 routes traffic to a resource. A positive bias expands the area, and a negative bias shrinks it. For more information, see How Amazon Route 53 uses bias to route traffic.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate Target Health** for an alias record or the records in a group of failover alias, geolocation alias, geoproximity alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain Name**, specify the

domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain Name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

For geoproximity records, if an endpoint is unhealthy, Route 53 looks for a closest endpoint that is still healthy.

Record ID

Enter a value that uniquely identifies this record in the group of geoproximity records.

Values specific for geoproximity alias records

When you create geoproximity alias records, you specify the following values.

For more information, see Choosing between alias and non-alias records.

Topics

- Routing policy
- Record name
- Record type
- Value/Route traffic to
- Endpoint location
- Bias
- Health check
- Evaluate target health
- Record ID

Routing policy

Choose **Geoproximity**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of geoproximity records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the applicable value based on the AWS resource that you're routing traffic to. Select the same value for all of the records in the group of geoproximity records:

API Gateway custom regional API or edge-optimized API

Select **A — IPv4 address**.

Amazon VPC interface endpoints

Select **A — IPv4 address**.

CloudFront distribution

Select **A — IPv4 address**.

If IPv6 is enabled for the distribution, create two records, one with a value of **A — IPv4 address** for **Type**, and one with a value of **AAAA — IPv6 address**.

App Runner service

Select **A — IPv4 address**

Elastic Beanstalk environment that has regionalized subdomains

Select **A — IPv4 address**

ELB load balancer

Select **A — IPv4 address** or **AAAA — IPv6 address**

Amazon S3 bucket

Select **A — IPv4 address**

OpenSearch Service

Select **A — IPv4 address** or **AAAA — IPv6 address**

Another record in this hosted zone

Select the type of the record that you're creating the alias for. All types are supported except **NS** and **SOA**.

> **Note**
>
> If you're creating an alias record that has the same name as the hosted zone (known as the *zone apex*), you can't route traffic to a record for which the value of **Type** is **CNAME**. This is because the alias record must have the same type as the record you're routing

traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

Value/Route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to.

For information about what AWS resources you can target, see Value/route traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

Endpoint location

You can specify the resource endpoint location by using one of the following:

Custom coordinates

Specify the longitude and lattitude for a geopgraphic area.

AWS Region

Choose an available Region from the **Location** list.

For more information about the Regions, see AWS Global Infrastructure.

AWS Local Zone Group

Choose an available Local Zone Region from the **Location** list.

For more information about Local Zones, see Available Local Zones in the *AWS Local Zones User Guide*. A local Zone Group is usually the Local Zone without the ending character. For example, if the Local Zone is `us-east-1-bue-1a` the Local Zone Group is `us-east-1-bue-1`.

You can also identify the Local Zones Group for a specific Local Zone by using the describe-availability-zones CLI command:

```
aws ec2 describe-availability-zones \
    --region us-west-2 \
    --all-availability-zones \
    --query "AvailabilityZones[?ZoneName=='us-west-2-den-1a']" \
```

```
    | grep "GroupName"
```

This command returns: `"GroupName": "us-west-2-den-1"`, specifying that the Local Zone `us-west-2-den-1a` belongs to the Local Zone Group `us-west-2-den-1`.

You can't create non-geoproximity records that have the same values for **Record name** and **Record type** as geoproximity records.

You also can't create two geoproximity resource record sets that specify the same location for the same record name and record type.

For more information, see available-local-zones.html

Bias

A bias either expands or shrinks a geographic area from which Route 53 routes traffic to a resource. A positive bias expands the area, and a negative bias shrinks it. For more information, see How Amazon Route 53 uses bias to route traffic.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, geoproximity alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify

health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

For geoproximity records, if an endpoint is unhealthy, Route 53 looks for a closest endpoint that is still healthy.

Evaluate target health

Select **Yes** if you want Route 53 to determine whether to respond to DNS queries using this record by checking the health of the resource specified by **Endpoint**.

Note the following:

API Gateway custom regional APIs and edge-optimized APIs

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an API Gateway custom Regional API or an edge-optimized API.

CloudFront distributions

You can't set **Evaluate target health** to **Yes** when the endpoint is a CloudFront distribution.

Elastic Beanstalk environments that have regionalized subdomains

If you specify an Elastic Beanstalk environment in **Endpoint** and the environment contains an ELB load balancer, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. (An environment automatically contains

an ELB load balancer if it includes more than one Amazon EC2 instance.) If you set **Evaluate target health** to **Yes** and either no Amazon EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other available resources that are healthy, if any.

If the environment contains a single Amazon EC2 instance, there are no special requirements.

ELB load balancers

Health checking behavior depends on the type of load balancer:

- **Classic Load Balancers** – If you specify an ELB Classic Load Balancer in **Endpoint**, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. If you set **Evaluate target health** to **Yes** and either no EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other resources.
- **Application and Network Load Balancers** – If you specify an ELB Application or Network Load Balancer and you set **Evaluate target health** to **Yes**, Route 53 routes queries to the load balancer based on the health of the target groups that are associated with the load balancer:
  - For an Application or Network Load Balancer to be considered healthy, every target group that contains targets must contain at least one healthy target. If any target group contains only unhealthy targets, the load balancer is considered unhealthy, and Route 53 routes queries to other resources.
  - A target group that has no registered targets is considered unhealthy.

> **Note**
>
> When you create a load balancer, you configure settings for Elastic Load Balancing health checks; they're not Route 53 health checks, but they perform a similar function. Do not create Route 53 health checks for the EC2 instances that you register with an ELB load balancer.

S3 buckets

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an S3 bucket.

Amazon VPC interface endpoints

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an Amazon VPC interface endpoint.

Other records in the same hosted zone

If the AWS resource that you specify in **Endpoint** is a record or a group of records (for example, a group of weighted records) but is not another alias record, we recommend that you associate a health check with all of the records in the endpoint. For more information, see What happens when you omit health checks?.

Record ID

Enter a value that uniquely identifies this record in the group of geoproximity records.

Values specific for latency records

When you create latency records, you specify the following values.

Topics

- Routing policy
- Record name
- Record type
- TTL (seconds)
- Value/Route traffic to
- Region
- Health check
- Record ID

Routing policy

Choose **Latency**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of latency records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the value for **Type** based on how you want Route 53 to respond to DNS queries.

Select the same value for all of the records in the group of latency records.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect. Recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, specify a shorter value first, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

If you're associating this record with a health check, specify a TTL of 60 seconds or less so clients respond quickly to changes in health status.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**
- **AAAA — IPv6 address**
- **CAA — Certificate Authority Authorization**
- **CNAME — Canonical name**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**
- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**
- **TXT — Text**

For more information about the above values, see common values for Value/Route traffic to.

Region

The Amazon EC2 Region where the resource that you specified in this record resides. Route 53 recommends an Amazon EC2 Region based on other values that you've specified. This also applies to private hosted zones. Don't change this value.

Note the following:

- You can only create one latency record for each Amazon EC2 Region.
- You aren't required to create latency records for all Amazon EC2 Regions. Route 53 chooses the Region with the best latency from among the Regions that you create latency records for.
- You can't create non-latency records that have the same values for **Record name** and **Record type** as latency records.
- If you create a record tagged with the Region **cn-north-1**, Route 53 always responds to queries from within China using this record, regardless of the latency.

For more information about using latency records, see Latency-based routing.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on health check status. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.

- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Record ID

Enter a value that uniquely identifies this record in the group of latency records.

Values specific for latency alias records

When you create latency alias records, you specify the following values.

For more information, see Choosing between alias and non-alias records.

Topics

- Routing policy
- Record name
- Record type
- Value/Route traffic to
- Region
- Health check
- Evaluate target health
- Record ID

Routing policy

Choose **Latency**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of latency records.

For more information about record names, see Record name

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the applicable value based on the AWS resource that you're routing traffic to:

API Gateway custom regional API or edge-optimized API

Select **A — IPv4 address**.

Amazon VPC interface endpoints

Select **A — IPv4 address**.

CloudFront distribution

Select **A — IPv4 address**.

If IPv6 is enabled for the distribution, create two records, one with a value of **A — IPv4 address** for **Type**, and one with a value of **AAAA — IPv6 address**.

App Runner service

Select **A — IPv4 address**

Elastic Beanstalk environment that has regionalized subdomains

Select **A — IPv4 address**

ELB load balancer

Select **A — IPv4 address** or **AAAA — IPv6 address**

Amazon S3 bucket

Select **A — IPv4 address**

OpenSearch Service

Select **A — IPv4 address** or **AAAA — IPv6 address**

Another record in this hosted zone

Select the type of the record that you're creating the alias for. All types are supported except **NS** and **SOA**.

> **Note**
>
> If you're creating an alias record that has the same name as the hosted zone (known as the *zone apex*), you can't route traffic to a record for which the value of **Type** is **CNAME**. This is because the alias record must have the same type as the record you're routing traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

Select the same value for all of the records in the group of latency records.

Value/Route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to.

For information about what AWS resources you can target, see common values for alias records for value/route traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

Region

The Amazon EC2 region where the resource that you specified in this record resides. Route 53 recommends an Amazon EC2 Region based on other values that you've specified. This also applies to private hosted zones. We recommend that you not change this value.

Note the following:

- You can only create one latency record for each Amazon EC2 Region.
- You aren't required to create latency records for all Amazon EC2 Regions. Route 53 chooses the Region with the best latency from among the Regions that you create latency records for.
- You can't create non-latency records that have the same values for **Record name** and **Record type** as latency records.
- If you create a record tagged with the Region **cn-north-1**, Route 53 always responds to queries from within China using this record, regardless of the latency.

For more information about using latency records, see Latency-based routing.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record,

Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain Name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Evaluate target health

Select **Yes** if you want Route 53 to determine whether to respond to DNS queries using this record by checking the health of the resource specified by **Endpoint**.

Note the following:

API Gateway custom regional APIs and edge-optimized APIs

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an API Gateway custom regional API or an edge-optimized API.

CloudFront distributions

You can't set **Evaluate Target Health** to **Yes** when the endpoint is a CloudFront distribution.

Elastic Beanstalk environments that have regionalized subdomains

If you specify an Elastic Beanstalk environment in **Endpoint** and the environment contains an ELB load balancer, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. (An environment automatically contains an ELB load balancer if it includes more than one Amazon EC2 instance.) If you set **Evaluate target health** to **Yes** and either no Amazon EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other available resources that are healthy, if any.

If the environment contains a single Amazon EC2 instance, there are no special requirements.

ELB load balancers

Health checking behavior depends on the type of load balancer:

- **Classic Load Balancers** – If you specify an ELB Classic Load Balancer in **Endpoint**, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. If you set **Evaluate target health** to **Yes** and either no EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other resources.
- **Application and Network Load Balancers** – If you specify an ELB Application or Network Load Balancer and you set **Evaluate target health** to **Yes**, Route 53 routes queries to the load balancer based on the health of the target groups that are associated with the load balancer:
  - For an Application or Network Load Balancer to be considered healthy, every target group that contains targets must contain at least one healthy target. If any target group contains only unhealthy targets, the load balancer is considered unhealthy, and Route 53 routes queries to other resources.
  - A target group that has no registered targets is considered unhealthy.

> **Note**
>
> When you create a load balancer, you configure settings for Elastic Load Balancing health checks; they're not Route 53 health checks, but they perform a similar function.

Do not create Route 53 health checks for the EC2 instances that you register with an ELB load balancer.

S3 buckets

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an S3 bucket.

Amazon VPC interface endpoints

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an Amazon VPC interface endpoint.

Other records in the same hosted zone

If the AWS resource that you specify in **Endpoint** is a record or a group of records (for example, a group of weighted records) but is not another alias record, we recommend that you associate a health check with all of the records in the endpoint. For more information, see What happens when you omit health checks?.

Record ID

Enter a value that uniquely identifies this record in the group of latency records.

Values specific for IP-based records

When you create IP-based records, you specify the following values.

> **Note**
>
> Although creating IP-based records in a private hosted zone is allowed, it's not supported.

Topics

- Routing policy
- Record name
- Record type
- TTL (seconds)
- Value/Route traffic to
- Location
- Health check
- Record ID

Routing policy

Choose **IP-based**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of IP-based records.

CNAME records

If you're creating a record that has a value of **CNAME** for **Record type**, the name of the record can't be the same as the name of the hosted zone.

Special characters

For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.

Wildcard characters

You can use an asterisk (*) character in the name. DNS treats the * character either as a wildcard or as the * character (ASCII 42), depending on where it appears in the name. For more information, see Using an asterisk (*) in the names of hosted zones and records.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the value for **Type** based on how you want Route 53 to respond to DNS queries.

Select the same value for all of the records in the group of IP-based records.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect. Recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, specify a shorter value first, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

If you're associating this record with a health check, specify a TTL of 60 seconds or less so clients respond quickly to changes in health status.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**
- **AAAA — IPv6 address**
- **CAA — Certificate Authority Authorization**
- **CNAME — Canonical name**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**
- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**
- **TXT — Text**

For more information about the above values, see Value/Route traffic to common values for Value/Route traffic to.

Location

The name of the CIDR location where the resource that you specified in this record is specified by the CIDR block values within the CIDR location.

For more information about using IP-based records, see IP-based routing.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information

about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on health check status. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, IP-based alias, latency alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Record ID

Enter a value that uniquely identifies this record in the group of IP-based records.

Values specific for IP-based alias records

When you create IP-based alias records, you specify the following values.

> **Note**
>
> You can create IP-based alias records only in public hosted zones.

For more information, see Choosing between alias and non-alias records.

Topics

- Routing policy
- Record name
- Record type
- Value/Route traffic to
- Location
- Health check
- Evaluate target health
- Record ID

Routing policy

Choose **IP-based**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of IP-based records.

CNAME records

If you're creating a record that has a value of **CNAME** for **Record type**, the name of the record can't be the same as the name of the hosted zone.

Aliases to CloudFront distributions and Amazon S3 buckets

The value that you specify depends in part on the AWS resource that you're routing traffic to:

- **CloudFront distribution** – Your distribution must include an alternate domain name that matches the name of the record. For example, if the name of the record is **acme.example.com**, your CloudFront distribution must include **acme.example.com** as one of the alternate domain names. For more information, see Using alternate domain names (CNAMEs) in the *Amazon CloudFront Developer Guide*.
- **Amazon S3 bucket** – The name of the record must match the name of your Amazon S3 bucket. For example, if the name of your bucket is **acme.example.com**, the name of this record must also be **acme.example.com**.

  In addition, you must configure the bucket for website hosting. For more information, see Configure a bucket for website hosting in the *Amazon Simple Storage Service User Guide*.

Special characters

For information about how to specify characters other than a-z, 0-9, and - (hyphen) and how to specify internationalized domain names, see DNS domain name format.

Wildcard characters

You can use an asterisk (*) character in the name. DNS treats the * character either as a wildcard or as the * character (ASCII 42), depending on where it appears in the name. For more information, see Using an asterisk (*) in the names of hosted zones and records.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the applicable value based on the AWS resource that you're routing traffic to. Select the same value for all of the records in the group of IP-based records:

API Gateway custom regional API or edge-optimized API

Select **A — IPv4 address**.

Amazon VPC interface endpoints

Select **A — IPv4 address**.

CloudFront distribution

Select **A — IPv4 address**.

If IPv6 is enabled for the distribution, create two records, one with a value of **A — IPv4 address** for **Type**, and one with a value of **AAAA — IPv6 address**.

App Runner service

Select **A — IPv4 address**

Elastic Beanstalk environment that has regionalized subdomains

Select **A — IPv4 address**

ELB load balancer

Select **A — IPv4 address** or **AAAA — IPv6 address**

Amazon S3 bucket

Select **A — IPv4 address**

OpenSearch Service

Select **A — IPv4 address** or **AAAA — IPv6 address**

Another record in this hosted zone

Select the type of the record that you're creating the alias for. All types are supported except **NS** and **SOA**.

> **Note**
>
> If you're creating an alias record that has the same name as the hosted zone (known as the *zone apex*), you can't route traffic to a record for which the value of **Type** is **CNAME**. This is because the alias record must have the same type as the record you're routing traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

Value/Route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to.

For information about what AWS resources you can target, see common values for alias records for value/route traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

Location

When you configure Route 53 to respond to DNS queries based on the location that the queries originated from, select the CIDR location for which you want Route 53 to respond with the settings in this record.

> **Important**
>
> We recommend that you create one IP-based record that has a value of **Default** for **Location**. This covers locations that you haven't created records for and IP addresses that Route 53 can't identify a location for.

You can't create non-IP-based records that have the same values for **Record name** and **Record type** as IP-based records.

For more information, see IP-based routing.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, IP-based routing alias, latency alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

For IP-based alias records, if an endpoint is unhealthy, Route 53 looks for a record within the larger, associated location. For example, suppose you have records for a state in the United States, for the United States, for North America, and for all locations (**Location** is **Default**). If the endpoint for the state record is unhealthy, Route 53 checks the records for the United States, for North America, and for all locations, in that order, until it finds a record that has a healthy endpoint. If all applicable records are unhealthy, including the record for all locations, Route 53 responds to the DNS query using the value for the record for the smallest geographic region.

Evaluate target health

Select **Yes** if you want Route 53 to determine whether to respond to DNS queries using this record by checking the health of the resource specified by **Endpoint**.

Note the following:

API Gateway custom regional APIs and edge-optimized APIs

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an API Gateway custom regional API or an edge-optimized API.

CloudFront distributions

You can't set **Evaluate target health** to **Yes** when the endpoint is a CloudFront distribution.

Elastic Beanstalk environments that have regionalized subdomains

If you specify an Elastic Beanstalk environment in **Endpoint** and the environment contains an ELB load balancer, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. (An environment automatically contains an ELB load balancer if it includes more than one Amazon EC2 instance.) If you set **Evaluate target health** to **Yes** and either no Amazon EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other available resources that are healthy, if any.

If the environment contains a single Amazon EC2 instance, there are no special requirements.

ELB load balancers

Health checking behavior depends on the type of load balancer:

- **Classic Load Balancers** – If you specify an ELB Classic Load Balancer in **Endpoint**, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. If you set **Evaluate target health** to **Yes** and either no EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other resources.
- **Application and Network Load Balancers** – If you specify an ELB Application or Network Load Balancer and you set **Evaluate target health** to **Yes**, Route 53 routes queries to the load balancer based on the health of the target groups that are associated with the load balancer:
  - For an Application or Network Load Balancer to be considered healthy, every target group that contains targets must contain at least one healthy target. If any target group contains only unhealthy targets, the load balancer is considered unhealthy, and Route 53 routes queries to other resources.

- A target group that has no registered targets is considered unhealthy.

> **Note**
>
> When you create a load balancer, you configure settings for Elastic Load Balancing health checks; they're not Route 53 health checks, but they perform a similar function. Do not create Route 53 health checks for the EC2 instances that you register with an ELB load balancer.

S3 buckets

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an S3 bucket.

Amazon VPC interface endpoints

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an Amazon VPC interface endpoint.

Other records in the same hosted zone

If the AWS resource that you specify in **Endpoint** is a record or a group of records (for example, a group of weighted records) but is not another alias record, we recommend that you associate a health check with all of the records in the endpoint. For more information, see What happens when you omit health checks?.

Record ID

Enter a value that uniquely identifies this record in the group of IP-based records.

Values specific for multivalue answer records

When you create multivalue answer records, you specify the following values.

> **Note**
>
> Creating multivalue answer alias records is not supported.

Topics

- Routing policy
- Record name
- Record type
- TTL (seconds)
- Value/Route traffic to
- Health check
- Record ID

Routing policy

Choose **Multivalue answer**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of multivalue records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select any value except **NS** or **CNAME**.

Select the same value for all of the records in the group of multivalue answer records.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect because recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, we recommend that you initially specify a shorter value, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

If you're associating this record with a health check, we recommend that you specify a TTL of 60 seconds or less so clients respond quickly to changes in health status.

> **Note**
>
> If you create two or more multivalue answer records that have the same name and type, you are using the console, and you specify different values for **TTL**, Route 53 changes the value of **TTL** for all of the records to the last value that you specified.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. If you enter more than one value, enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**
- **AAAA — IPv6 address**
- **CAA — Certificate Authority Authorization**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**
- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**
- **TXT — Text**

For more information about the above values, see common values for Value/Route traffic to.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate**

Target Health, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Record ID

Enter a value that uniquely identifies this record in the group of multivalue answer records.

Values specific for weighted records

When you create weighted records, you specify the following values.

Topics

- Routing policy
- Record name
- Record type
- TTL (seconds)
- Value/Route traffic to
- Weight
- Health check
- Record ID

Routing policy

Select **Weighted**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of weighted records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the same value for all of the records in the group of weighted records.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect. Recursive resolvers use the values in their cache for longer periods before they ask Route 53 for the latest information. If you're changing settings for a domain or subdomain that's already in use, specify a shorter value first, such as 300 seconds, and increase the value after you confirm that the new settings are correct.

If you're associating this record with a health check, specify a TTL of 60 seconds or less so clients respond quickly to changes in health status.

You must specify the same value for **TTL** for all of the records in this group of weighted records.

> **Note**
>
> If you create two or more weighted records that have the same name and type, and you specify different values for **TTL**, Route 53 changes the value of **TTL** for all of the records to the last value that you set.

If a group of weighted records includes one or more weighted alias records that route traffic to an ELB load balancer, specify a TTL of 60 seconds for all of the non-alias weighted records that have the same name and type. Values other than 60 seconds (the TTL for load balancers) will change the effect of the values that you specify for **Weight**.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**

- **AAAA — IPv6 address**
- **CAA — Certificate Authority Authorization**
- **CNAME — Canonical name**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**
- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**
- **TXT — Text**

For more information about the above values, see common values for Value/Route traffic to.

Weight

A value that determines the proportion of DNS queries that Route 53 responds to using the current record. Route 53 calculates the sum of the weights for the records that have the same combination of DNS name and type. Route 53 then responds to queries based on the ratio of a resource's weight to the total.

You can't create non-weighted records that have the same values for **Record name** and **Record type** as weighted records.

Enter an integer between 0 and 255. To stop routing to a resource, set **Weight** to 0. If you set **Weight** to 0 for all of the records in the group, traffic is routed to all resources with equal probability. This makes sure you don't disable routing for a group of weighted records.

The effect of setting **Weight** to 0 is different when you associate health checks with weighted records. For more information, see How Amazon Route 53 chooses records when health checking is configured.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information

about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on health check status. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an unhealthy endpoint, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Record ID

Enter a value that uniquely identifies this record in the group of weighted records.

Values specific for weighted alias records

When you create weighted alias records, you specify the following values. For more information, see Choosing between alias and non-alias records.

Topics

- Routing policy
- Record name
- Record type
- Value/Route traffic to
- Weight
- Health check
- Evaluate target health
- Record ID

Routing policy

Choose **Weighted**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Name** field.

Enter the same name for all of the records in the group of weighted records.

For more information about record names, see Record name

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the applicable value based on the AWS resource that you're routing traffic to:

API Gateway custom regional API or edge-optimized API

Select **A — IPv4 address**.

Amazon VPC interface endpoints

Select **A — IPv4 address**.

CloudFront distribution

Select **A — IPv4 address**.

If IPv6 is enabled for the distribution, create two records, one with a value of **A — IPv4 address** for **Type**, and one with a value of **AAAA — IPv6 address**.

App Runner service

Select **A — IPv4 address**

Elastic Beanstalk environment that has regionalized subdomains

Select **A — IPv4 address**

ELB load balancer

Select **A — IPv4 address** or **AAAA — IPv6 address**

Amazon S3 bucket

Select **A — IPv4 address**

OpenSearch Service

Select **A — IPv4 address** or **AAAA — IPv6 address**

Another record in this hosted zone

Select the type of the record that you're creating the alias for. All types are supported except **NS** and **SOA**.

> **Note**
>
> If you're creating an alias record that has the same name as the hosted zone (known as the *zone apex*), you can't route traffic to a record for which the value of **Type** is **CNAME**. This is because the alias record must have the same type as the record you're routing traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.

Select the same value for all of the records in the group of weighted records.

Value/Route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to.

For information about what AWS resources you can target, see common values for alias records for value/route traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

Weight

A value that determines the proportion of DNS queries that Route 53 responds to using the current record. Route 53 calculates the sum of the weights for the records that have the same combination of DNS name and type. Route 53 then responds to queries based on the ratio of a resource's weight to the total.

You can't create non-weighted records that have the same values for **Record name** and **Record type** as weighted records.

Enter an integer between 0 and 255. To disable routing to a resource, set **Weight** to 0. If you set **Weight** to 0 for all of the records in the group, traffic is routed to all resources with equal probability. This ensures that you don't accidentally disable routing for a group of weighted records.

The effect of setting **Weight** to 0 is different when you associate health checks with weighted records. For more information, see How Amazon Route 53 chooses records when health checking is configured.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Associating a health check with a record is useful only when Route 53 is choosing between two or more records to respond to a DNS query, and you want Route 53 to base the choice in part on the status of a health check. Use health checks only in the following configurations:

- You're checking the health of all of the records in a group of records that have the same name, type, and routing policy (such as failover or weighted records), and you specify health check IDs for all the records. If the health check for a record specifies an endpoint that is not healthy, Route 53 stops responding to queries using the value for that record.
- You select **Yes** for **Evaluate target health** for an alias record or the records in a group of failover alias, geolocation alias, latency alias, IP-based alias, or weighted alias record. If the alias records reference non-alias records in the same hosted zone, you must also specify health checks for the referenced records. If you associate a health check with an alias record and also select **Yes** for **Evaluate Target Health**, both must evaluate to true. For more information, see What happens when you associate a health check with an alias record?.

If your health checks specify the endpoint only by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Evaluate target health

Select **Yes** if you want Route 53 to determine whether to respond to DNS queries using this record by checking the health of the resource specified by **Endpoint**.

Note the following:

API Gateway custom regional APIs and edge-optimized APIs

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an API Gateway custom Regional API or an edge-optimized API.

CloudFront distributions

You can't set **Evaluate target health** to **Yes** when the endpoint is a CloudFront distribution.

Elastic Beanstalk environments that have regionalized subdomains

If you specify an Elastic Beanstalk environment in **Endpoint** and the environment contains an ELB load balancer, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. (An environment automatically contains an ELB load balancer if it includes more than one Amazon EC2 instance.) If you set **Evaluate target health** to **Yes** and either no Amazon EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other available resources that are healthy, if any.

If the environment contains a single Amazon EC2 instance, there are no special requirements.

ELB load balancers

Health checking behavior depends on the type of load balancer:

- **Classic Load Balancers** – If you specify an ELB Classic Load Balancer in **Endpoint**, Elastic Load Balancing routes queries only to the healthy Amazon EC2 instances that are registered with the load balancer. If you set **Evaluate Target Health** to **Yes** and either no EC2 instances are healthy or the load balancer itself is unhealthy, Route 53 routes queries to other resources.
- **Application and Network Load Balancers** – If you specify an ELB Application or Network Load Balancer and you set **Evaluate Target Health** to **Yes**, Route 53 routes queries to the load balancer based on the health of the target groups that are associated with the load balancer:
  - For an Application or Network Load Balancer to be considered healthy, every target group that contains targets must contain at least one healthy target. If any target group contains only unhealthy targets, the load balancer is considered unhealthy, and Route 53 routes queries to other resources.
  - A target group that has no registered targets is considered unhealthy.

> **Note**
>
> When you create a load balancer, you configure settings for Elastic Load Balancing health checks; they're not Route 53 health checks, but they perform a similar function. Do not create Route 53 health checks for the EC2 instances that you register with an ELB load balancer.

S3 buckets

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an S3 bucket.

Amazon VPC interface endpoints

There are no special requirements for setting **Evaluate target health** to **Yes** when the endpoint is an Amazon VPC interface endpoint.

Other records in the same hosted zone

If the AWS resource that you specify in **Endpoint** is a record or a group of records (for example, a group of weighted records) but is not another alias record, we recommend that you associate a health check with all of the records in the endpoint. For more information, see What happens when you omit health checks?.

Record ID

Enter a value that uniquely identifies this record in the group of weighted records.

Values specific for AZ-affinity records

When you create AZ-affinity records, you specify the following values. For more information, see AZ-affinity routing.

Topics

- Routing policy
- Record name
- Record type
- TTL (seconds)
- Value/Route traffic to
- Location type
- Health check
- Record ID

Routing policy

Choose **AZ affinity**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of AZ-affinity records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the same value for all of the records in the group of AZ-affinity records.

TTL (seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you reduce the number of calls that DNS recursive resolvers must make to Route 53. This reduces latency and lowers your bill for Route 53 service. For more information, see How Amazon Route 53 routes traffic for your domain.

However, if you specify a longer value for TTL, it takes longer for changes to the record (for example, a new IP address) to take effect, and it takes longer for clients to stop using a record that Route 53 has stopped responding with because it's unhealthy.

If you're associating this record with a health check, see Best practices for Amazon Route 53 DNS for TTL recommendations.

Specify the same value for **TTL** for all of the records in this group of AZ-affinity records.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter a value that is appropriate for the value of **Record type**. For all types except **CNAME**, you can enter more than one value. Enter each value on a separate line.

You can route traffic to, or specify the following values:

- **A — IPv4 address**
- **AAAA — IPv6 address**
- **CAA — Certificate Authority Authorization**
- **CNAME — Canonical name**
- **MX — Mail exchange**
- **NAPTR — Name Authority Pointer**
- **PTR — Pointer**
- **SPF — Sender Policy Framework**
- **SRV — Service locator**
- **TXT — Text**

For more information about the above values, see common values for Value/Route traffic to.

To route traffic to an AWS resource, create an AZ-affinity alias record instead. For more information, see Values specific for AZ-affinity alias records.

Location type

The level of the AWS location hierarchy that this record answers for. Route 53 responds to each query with the most specific record that matches where the query comes from. Choose one of the following values, listed from most specific to least specific:

Availability Zone

Answers queries from one Availability Zone. Specify the AZ ID, such as `use1-az1`, not the Availability Zone name, such as `us-east-1a`.

Zone Group

Answers queries from any Availability Zone in a Zone Group, such as `us-east-1-zg-1`. You can specify only Zone Groups of Availability Zones, not Zone Groups of Local Zones or Wavelength Zones.

Region

Answers queries from any Availability Zone in a Region, such as `us-east-1`.

Partition

Answers queries from any AWS Region. Use it as the fallback for queries that come from AWS, and use a Global record for queries from the internet. In the API, specify `Partition` for `LocationType` and the partition name in lowercase, such as `aws`, for `LocationId`.

Global

Answers all other queries, such as queries from the internet. We recommend that you always create a Global record so that every query gets an answer. If no record matches a query and you didn't create a Global record, Route 53 responds with no answer (NODATA).

If the matching record is unhealthy, Route 53 falls back to the record for the next less specific location that you created a record for, which contains the query's Availability Zone: its Zone Group, its Region, its partition, and then Global.

You also choose a location that the record answers for. In the console, you choose the location from a list. In the API, you specify a location ID. For the ID formats, see AZ-affinity routing. To find the AZ IDs and Zone Groups for your account, see Finding AZ IDs and Zone Groups.

You can't create non-AZ-affinity records that have the same values for **Record name** and **Record type** as AZ-affinity records.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy. When the endpoint is unhealthy, Route 53 falls back to the record for the next less specific location.

If you don't select a health check, Route 53 always considers the record healthy and doesn't fall back past it to a less specific record. We recommend that you associate a health check with each AZ-affinity record.

Route 53 doesn't check the health of the endpoint specified in the record, for example, the endpoint specified by the IP address in the **Value** field. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

If your health checks specify the endpoint only by domain name, create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

Record ID

Enter a value that uniquely identifies this record in the group of AZ-affinity records, such as a timestamp or a sequential number.

Values specific for AZ-affinity alias records

When you create AZ-affinity alias records, you specify the following values. For more information, see Choosing between alias and non-alias records and AZ-affinity routing.

Topics

- Routing policy
- Record name
- Record type
- Value/Route traffic to
- Location type
- Health check
- Evaluate target health
- Record ID

Routing policy

Choose **AZ affinity**.

Record name

Enter the name of the domain or subdomain that you want to route traffic for. The default value is the name of the hosted zone.

> **Note**
>
> If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the **Record name** field.

Enter the same name for all of the records in the group of AZ-affinity records.

For more information about record names, see Record name.

Record type

The DNS record type. For more information, see Supported DNS record types.

Select the applicable value based on the AWS resource that you're routing traffic to. If you're routing traffic to another record in this hosted zone, select the type of that record. All types are supported except **NS** and **SOA**.

Select the same value for all of the records in the group of AZ-affinity records.

Value/Route traffic to

The value that you choose from the list or that you type in the field depends on the AWS resource that you're routing traffic to. You can route traffic to an AWS resource or to another record in this hosted zone. You can't route traffic to a record in a different hosted zone.

For information about what AWS resources you can target, see common values for alias records for value/route traffic to.

For more information about how to configure Route 53 to route traffic to specific AWS resources, see Routing internet traffic to your AWS resources.

Location type

The level of the AWS location hierarchy that this record answers for: **Availability Zone**, **Zone Group**, **Region**, **Partition**, or **Global**. You specify the same values as for non-alias AZ-affinity records. For more information, see Location type.

If the matching record is unhealthy, Route 53 falls back to the record for the next less specific location that you created a record for, which contains the query's Availability Zone: its Zone Group, its Region, its partition, and then Global. For an alias record, Route 53 treats the record as unhealthy because of the alias target only if you set **Evaluate target health** to **Yes**. For more information, see Evaluate target health.

You can't create non-AZ-affinity records that have the same values for **Record name** and **Record type** as AZ-affinity records.

Health check

Select a health check if you want Route 53 to check the health of a specified endpoint and to respond to DNS queries using this record only when the endpoint is healthy.

For alias records, we recommend that you set **Evaluate target health** to **Yes** instead of associating a health check with the alias record. If you associate a health check with an alias record and also select **Yes** for **Evaluate target health**, both must evaluate to true. If the alias record references

non-alias records in the same hosted zone, associate health checks with the referenced records. For more information, see What happens when you associate a health check with an alias record?.

Route 53 doesn't check the health of the endpoint specified in the record. When you select a health check for a record, Route 53 checks the health of the endpoint that you specified in the health check. For information about how Route 53 determines whether an endpoint is healthy, see How Amazon Route 53 determines whether a health check is healthy.

Evaluate target health

Select **Yes** if you want Route 53 to determine whether to respond to DNS queries using this record by checking the health of the alias target. If the alias target is unhealthy, Route 53 falls back to the record for the next less specific location.

If you select **No**, Route 53 doesn't check the health of the alias target, so it doesn't fall back past this record when the alias target is unhealthy. If you also associate a health check with this record, Route 53 falls back when that health check is unhealthy.

If the alias target is a record or a group of records (for example, a group of weighted records) but is not another alias record, we recommend that you associate a health check with all of the records in the alias target. For more information, see What happens when you omit health checks?.

Some AWS resources have special requirements for **Evaluate target health**. For more information, see `EvaluateTargetHealth` in AliasTarget in the *Amazon Route 53 API Reference*.

Record ID

Enter a value that uniquely identifies this record in the group of AZ-affinity records, such as a timestamp or a sequential number.

Creating records by importing a zone file

If you're moving from another DNS service provider, and if that provider lets you export your current DNS settings to a zone file, you can quickly create all the records for an Amazon Route 53 hosted zone by importing that zone file.

> **Note**
>
> A zone file uses a standard format known as BIND to show records in a text format. For information about the format of a zone file, see the Wikipedia entry Zone file. More

information is available in RFC 1034, Domain Names—Concepts and Facilities section 3.6.1, and RFC 1035, Domain Names—Implementation and Specification section 5.

If you want to create records by importing a zone file, note the following:

- The zone file must be in RFC-compliant format.
- The domain name of the records in the zone file must match the name of the hosted zone.
- Route 53 supports the `$ORIGIN` and `$TTL` keywords. If the zone file includes `$GENERATE` or `$INCLUDE` keywords, the import fails and Route 53 returns an error.
- When you import the zone file, Route 53 ignores the SOA record in the zone file. Route 53 also ignores any NS records that have the same name as the hosted zone.
- You can import a maximum of 1000 records.
- If the hosted zone already contains records that appear in the zone file, the import process fails, and no records are created.
- For TXT records that contain backslash characters, the zone file import process interprets certain backslash sequences as control characters. To include literal backslash characters in TXT record values:
  - Use double backslashes (`\\\\`) in the zone file to represent a single literal backslash in the final TXT record.
  - For example, if your TXT record should contain `\\jYTDWqH...` (with a literal backslash and j), specify `\\\\jYTDWqH...` in the zone file.

  This is particularly important for ACME challenge records and other TXT records that contain literal backslash characters.
- For long TXT records (such as DKIM records), the zone file import process supports splitting the content into multiple strings. To create TXT records with multiple strings:
  - Use separate lines in your zone file with the same record name and type.

    Example

```
example.com. 300 IN TXT "v=DKIM1; k=rsa; p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC"
example.com. 300 IN TXT
 "7fCC6C13dM9tXuJmUBH7D4Vw8y1ByJ8z9QX2fvLm3pN4sR5tU6vW7xY8zA9bC0dE1f"
example.com. 300 IN TXT
 "G2hI3jK4lM5nO6pQ7rS8tU9vW0xY1zA2bC3dE4fG5hI6jK7lM8nO9pQ0rS1tU2vW3x"
```

The import process automatically combines these into a single TXT record with multiple strings. Each individual string can contain up to 65,535 characters. Do not concatenate long strings into a single quoted value.

- We suggest that you review the zone file to confirm that record names include or exclude a trailing dot as needed:
  - When the name of a record in the zone file includes a trailing dot (`example.com.`), the import process interprets the name as a fully qualified domain name and creates a Route 53 record with that name.
  - When the name of a record in the zone file does not include a trailing dot (`www`), the import process concatenates that name with the domain name in the zone file (`example.com`) and creates a Route 53 record with the concatenated name (`www.example.com`).

  If the export process doesn't add a trailing dot to the fully qualified domain names of a record, the Route 53 import process adds the domain name to the record's name. For example, suppose you're importing records into the hosted zone `example.com` and the name of an MX record in the zone file is `mail.example.com`, with no trailing dot. The Route 53 import process creates an MX record named `mail.example.com.example.com`.

  Important

  For CNAME, MX, PTR, and SRV records, this behavior also applies to the domain name that is included in the RDATA value. For example, suppose you have a zone file for `example.com`. If a CNAME record in the zone file (`support`, without a trailing dot) has an RDATA value of `www.example.com` (also without a trailing dot), the import process creates a Route 53 record with the name `support.example.com` that routes traffic to `www.example.com.example.com`. Before you import your zone file, review RDATA values and update as applicable. For TXT records containing backslash characters, use double backslashes (\\\\) in the zone file to represent literal backslashes.

Route 53 doesn't support exporting records to a zone file.

Note

If you're creating a record that has the same name as the hosted zone, don't enter a value (for example, an @ symbol) in the Name field.

To create records by importing a zone file

1. Get a zone file from the DNS service provider that currently handles your domain. The process and terms vary from one provider to another. Refer to your provider's interface and documentation for information about exporting or saving your records in a zone file or a BIND file.

   If the process isn't clear, try asking your current DNS provider's support for your *records list* or *zone file* data.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Hosted zones**.
4. On the **Hosted zones** page, create a new hosted zone:

   a. Choose **Create hosted zone**.

   b. Enter the name of your domain and, optionally, a comment.

   c. Choose **Create**.
5. Choose **Import zone file**.
6. In the **Import zone file** pane, paste the contents of your zone file into the **Zone file** text box.
7. Choose **Import**.

   > **Note**
   >
   > Depending on the number of records in your zone file, you might have to wait a few minutes for the records to be created.
8. If you're using another DNS service for the domain (which is common if you registered the domain with another registrar), move DNS service to Route 53. When that step is done, your registrar will identify Route 53 as your DNS service for DNS queries for your domain, and the queries will start going to Route 53 DNS servers. (Typically, there's a day or two of delay before DNS queries start going to Route 53 because your previous DNS service's data is cached on DNS resolvers for that long.) For more information, see Making Amazon Route 53 the DNS service for an existing domain.

Editing records

The following procedure explains how to edit records using the Amazon Route 53 console. For information about how to edit records using the Route 53 API, see ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.

> **Note**
>
> Your changes to records take time to propagate to the Route 53 DNS servers. Currently, the only way to verify that changes have propagated is to use the GetChange API action. Changes generally propagate to all Route 53 name servers within 60 seconds.

To edit records using the Route 53 console

1. If you're not editing alias records, skip to step 2.

   If you're editing alias records that route traffic to an Elastic Load Balancing Classic Load Balancer, Application Load Balancer, or Network Load Balancer, and if you created your Route 53 hosted zone and your load balancer using different accounts, perform the procedure Getting the DNS name for an Elastic Load Balancing load balancer to get the DNS name for the load balancer.

   If you're editing alias records for any other AWS resource, skip to step 2.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Hosted zones**.
4. On the **Hosted Zones** page, choose the row for the hosted zone that contains the records that you want to edit.
5. Select the row for the record that you want to edit, and then enter your changes in the **Edit record** pane.
6. Enter the applicable values. For more information, see Values that you specify when you create or edit Amazon Route 53 records.
7. Choose **Save changes**.
8. If you're editing multiple records, repeat steps 5 through 7.

Deleting records

The following procedure explains how to delete records using the Route 53 console. For information about how to delete records using the Route 53 API, see ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.

> **Note**
>
> Your changes to records take time to propagate to the Route 53 DNS servers. Currently, the only way to verify that changes have propagated is to use the GetChange API action. Changes generally propagate to all Route 53 name servers within 60 seconds.

To delete records

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the Hosted Zones page, choose the row for the hosted zone that contains records that you want to delete.
3. In the list of records, select the record that you want to delete.

   To select multiple, consecutive records, choose the first row, hold the **Shift** key, and choose the last row. To select multiple, nonconsecutive records, choose the first row, hold the **Ctrl** key, and choose additional rows.

   You can't delete the records that have a value of **NS** or **SOA** for **Type**.
4. Choose **Delete**.
5. Choose **Delete** to close the dialog box.

Listing records

The following procedure explains how to use the Amazon Route 53 console to list the records in a hosted zone. For information about how to list records using the Route 53 API, see ListResourceRecordSets in the *Amazon Route 53 API Reference*.

To list records

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. On the **Hosted Zones** page, choose the name of a hosted zone.
4. To change the search mode, choose the gear icon on the upper right of the **Records** table. Choose one of:

   - **Automatic**

     In this mode, the service uses a filter based on a number of records. Full for less than 2000 and fast for more than 2000 records.

   - **Full**

     In this mode, all the search filters are available, but the search performance might be slower.

   - **Fast**

     In this mode, some advanced features aren't available, but the search performance will be faster.

To show only certain records, enter search terms above the list of records. In automatic mode the search behavior depends on whether the hosted zone has up to 2,000 records or more than 2,000 records:

Up to 2,000 records and full mode

- To display the records that have specific values, enter a value in the search bar and press **Enter**. For example, to display the records that have an IP address beginning with **192.0**, enter that value in the **Search** field and press **Enter**.
- To display only the records that have the same DNS record type, select **Record type** in the dropdown list, and enter the record type.
- To display only alias records, select **Alias** in the dropdown list, and enter **`Yes`**.
- To display only weighted records, select **Routing policy** in the dropdown list, and enter **`WEIGHTED`**.

More than 2,000 records and fast mode

- You can search only on record names, not on record values. You also can't filter based on the record type, or on alias or weighted records.

  To do this, put your cursor into the **Filter** textbox, select **Properties** and then **Record name**.

- For records that have three labels (three parts separated by dots), when you enter a value in the search field and press **Enter**, the Route 53 console automatically performs a wildcard search on the third label from the right in the record name. For example, suppose the hosted zone example.com contains 100 records named record1.example.com through record100.example.com. (Record1 is the third label from the right.) Here's what happens when you search on the following values:

  - **record1** – The Route 53 console searches for **record1*.example.com**, which returns **record1.example.com**, **record10.example.com** through **record19.example.com**, and **record100.example.com**.
  - **record1.example.com** – As in the preceding example, the console searches for **record1*.example.com** and returns the same records.
  - **1** – The console searches for **1*.example.com** and returns no records.
  - **example** – The console searches for **example*.example.com** and returns no records.
  - **example.com** – In this example, the console doesn't perform a wildcard search. It returns all the records in the hosted zone.
  - **Automatic search mode** – When using this search mode, you must first provide a property, such as record name, to be able to search.

  > **Note**
  >
  > If the third label from the right contains one or more hyphens (such as `third-label.example.com`), and if you search for the part of the third label immediately before the hyphen (`third` in this example), Route 53 won't return any records. Instead, either include the hyphen (search for `third-`) or omit the character immediately before the hyphen (search for `third`).

- For records that have four or more labels, you must specify the exact name of the record. No wildcard searches are supported. For example, if the hosted zone includes a record named label4.record1.example.com, you can find that record only if you specify **label4.record1.example.com** in the search field.

Protecting records from changes

To prevent a critical record from being changed or deleted by mistake, add a record set protection for its name and type. While the protection exists, Route 53 rejects any `ChangeResourceRecordSets` request that would create, update, or delete those records. The protection applies to every user and role, including users with full Route 53 permissions. To change a protected record, you first delete its protection.

Use protections for records that rarely change and that would cause an outage if someone changed them by mistake, such as the apex records of a production domain, mail exchanger (MX) records, or records that route traffic to critical endpoints.

Topics

- How record set protections work
- Managing record set protections by using the Route 53 console
- Managing record set protections by using the AWS CLI
- Changing records that are protected
- Deleting a hosted zone that has protections
- Record set protections and accelerated recovery
- Permissions for record set protections
- Troubleshooting record set protection errors

How record set protections work

- You identify a protection by a record name and a record type in one hosted zone, for example `www.example.com` and `A`.
- A protection covers every record with that name and type, whatever its set identifier. For example, if `www.example.com` has three weighted `A` records, one protection on `www.example.com` `A` protects all three. You can't protect only some of those records.
- You can create a protection for a name and type that has no records yet. The protection reserves that name and type: no one can create a record with that name and type until you delete the protection. Reserved names appear in the list of protections but not in the list of records.
- Route 53 checks protections for every change in a `ChangeResourceRecordSets` request, for `CREATE`, `DELETE`, and `UPSERT`. If any change in the request affects a protected name and type, Route 53 rejects the whole request and makes none of the changes.

- Route 53 doesn't reject a change that would not modify anything. For example, an UPSERT whose values are identical to the existing record succeeds without changing the record.
- You can add an optional comment of up to 256 characters to each protection. Route 53 includes the comment in the error message when it rejects a change, so use the comment to tell people why the record is protected and whom to contact.
- You can't update a protection. To change its comment, delete the protection and create it again.
- You can use protections in public and private hosted zones.
- You can't delete a hosted zone that has protections. Delete the protections first. For more information about deleting a hosted zone that has protections, see Deleting a hosted zone that has protections.
- Requests to create and delete protections are all-or-nothing. Each request either adds or removes every protection in the request, or none of them.
- Protections don't change how Route 53 answers DNS queries. They only control changes to records.

The following details explain how record set protections behave in specific situations:

Record types that you can protect

You can protect every record type that `ChangeResourceRecordSets` accepts, including the NS and SOA records at the zone apex.

Who can remove a protection

Any principal that has the `route53:DeleteResourceRecordSetProtections` permission on the hosted zone can delete any protection in that zone. A protection is only as strong as the IAM policies that control that permission. For more information about these permissions, see Permissions for record set protections.

Records that Route 53 manages

Protections apply to changes that you request. A protection on the apex NS or SOA records blocks your changes to them, but it doesn't prevent Route 53 from updating those records as it manages the hosted zone.

Other services and tools

When an AWS CloudFormation stack, the AWS CLI, or an AWS SDK changes a protected record, the change fails with the same `InvalidChangeBatch` error.

Logging

CloudTrail logs requests to create, delete, and list protections. For more information, see Logging Amazon Route 53 API calls with AWS CloudTrail.

Managing record set protections by using the Route 53 console

The hosted zone details page has a **Protections** tab. On the **Records** tab, the **Protection** column shows whether each record is protected. When a record is protected, the console disables **Edit** and **Delete** for it and explains that you must remove the protection first.

To add a protection

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and then choose the name of the hosted zone.
3. On the **Protections** tab, choose **Add protection**.
4. Enter the following values:
   - **Record name** – The subdomain to protect, without the hosted zone name. Leave it blank to protect the zone apex.
   - **Record types** – One or more record types. The console creates one protection for each type.
   - (Optional) **Comment** – Why the record is protected. This text appears in error messages when Route 53 blocks a change.
5. Choose **Add protection**.

If a protection already exists for a selected type, the console reports A protection already exists for *name* (*type*) and adds none of the protections.

To protect records when you create them – On the create record page, in the **Protection** section, select **Enable protection**. The console creates the records and then adds the protections. If you see a message that the records were created but aren't protected, add the protection on the **Protections** tab. You can add protection to at most 100 records in one submission.

To remove a protection

1. On the hosted zone's **Protections** tab, select one or more protections.

2. Choose **Remove protection**.
3. Review the records that will be unprotected. A protection covers all records with the same name and type.
4. To confirm, type **`confirm`**, and then choose **Remove protection**.

The console removes the selected protections in one request of up to 100. If any one of them fails, the console removes none of them.

To view protections – Choose the **Protections** tab. The table shows **Record name**, **Record type**, **Protection type**, and **Comment**.

Managing record set protections by using the AWS CLI

To manage protections programmatically, use the CreateResourceRecordSetProtections, DeleteResourceRecordSetProtections, and ListResourceRecordSetProtections actions. The following examples use the AWS CLI.

Create protections

The following command creates two protections at the apex of the `example.com` hosted zone: one for the `A` records, with a comment, and one for the `MX` records.

```
aws route53 create-resource-record-set-protections \
    --hosted-zone-id Z1D633PJN98FT9 \
    --protections '[{"RecordName":"example.com","RecordType":"A","Comment":"Production
 apex. Contact the DNS team before changing."},
{"RecordName":"example.com","RecordType":"MX"}]'
```

Each element of `--protections` has the following values:

`RecordName`

The name of the records to protect. Specify the full name, such as `www.example.com`.

`RecordType`

The type of the records to protect, such as `A` or `MX`.

`Comment`

(Optional) Up to 256 characters that explain why the records are protected. Route 53 includes the comment in the error message when it rejects a change.

A successful request returns an empty response.

List protections

The following command lists the protections in the hosted zone.

```
aws route53 list-resource-record-set-protections \
    --hosted-zone-id Z1D633PJN98FT9
```

Example output:

```
{
    "ResourceRecordSetProtections": [
        {
            "RecordName": "example.com.",
            "RecordType": "A",
            "Comment": "Production apex. Contact the DNS team before changing."
        },
        {
            "RecordName": "example.com.",
            "RecordType": "MX"
        }
    ]
}
```

Route 53 returns `RecordName` in fully qualified form with a trailing dot, as `ListResourceRecordSets` does.

Delete protections

The following command deletes the protection for the A records at the zone apex.

```
aws route53 delete-resource-record-set-protections \
    --hosted-zone-id Z1D633PJN98FT9 \
    --protections '[{"RecordName":"example.com","RecordType":"A"}]'
```

A successful request returns an empty response.

Note the following about listing protections:

- Route 53 sorts results by record name and then by record type.

- Route 53 returns at most 300 protections per page. If you omit `MaxResults`, or specify a value greater than 300, the page size is 300.
- If the response contains `NextToken`, more protections exist. Pass the token in the next request. The AWS CLI pages through all results automatically. Use `--page-size`, `--max-items`, and `--starting-token` to control paging.
- Use `StartRecordName` and `StartRecordType` to start the listing at that position, inclusive. `StartRecordType` requires `StartRecordName`. When you specify `NextToken`, it takes precedence over both.
- Use a `NextToken` only with the hosted zone that it was returned for.

Changing records that are protected

If any change in a `ChangeResourceRecordSets` request has the same record name and type as a protection, Route 53 rejects the whole request with an `InvalidChangeBatch` error (HTTP 400) and applies none of the changes. The error message has one entry for each blocked change, in the following form:

```
RRSet with DNS name www.example.com., type A is protected and cannot be changed.
 Protection comment: "Production apex. Contact the DNS team before changing.". Call
 DeleteResourceRecordSetProtections for this record set before changing it.
```

If the change specifies a set identifier, the message includes the set identifier after the type. If the protection has no comment, the message omits the protection comment.

To change a protected record, delete the protection with `DeleteResourceRecordSetProtections`, make the change, and then create the protection again if you still need it.

Deleting a hosted zone that has protections

You can't delete a hosted zone while it has any record set protections, including protections on names that have no records. Route 53 returns an error with the following message:

```
Hosted Zone hosted-zone-id has resource record set protections. All resource record set
 protections need to be deleted (DeleteResourceRecordSetProtections) before deleting
 hosted zone.
```

To delete the hosted zone, list the protections with `ListResourceRecordSetProtections`, delete them with `DeleteResourceRecordSetProtections` (up to 100 per request), delete the records, and then delete the hosted zone.

Record set protections and accelerated recovery

- Protections stay in effect when you turn accelerated recovery on or off.
- While you turn accelerated recovery on or off, requests to create or delete protections can fail with a `Throttling` error. Retry after a short wait.
- To delete a hosted zone that has accelerated recovery and protections, turn off accelerated recovery, delete the protections, and then delete the hosted zone.
- During failover, you can still create, delete, and list record set protections. Protections that exist before failover remain in effect.

For more information about accelerated recovery, see Enabling accelerated recovery for managing public DNS records.

Permissions for record set protections

The record set protection actions `route53:CreateResourceRecordSetProtections`, `route53:DeleteResourceRecordSetProtections`, and `route53:ListResourceRecordSetProtections` support only the hosted zone resource (`arn:aws:route53:::hostedzone/`*`Id`*). They don't support any service-specific condition keys.

- A protection is only as strong as your control over `route53:DeleteResourceRecordSetProtections`. Grant it to a small set of administrators, and consider denying it to everyone else, for example with a service control policy.
- Conditions in a policy on `route53:ChangeResourceRecordSets` don't override a protection. Even a principal with `route53:*` can't change a protected record until the protection is deleted.

For more information about an example policy that denies removing protections, see Example 5: Deny removal of record set protections.

Troubleshooting record set protection errors

The following table lists the errors that the record set protection actions return, their causes, and what to do.

Error	Cause	What to do
`InvalidInput` (400)	A create or delete request contains more than 100 protections, or a list request has an invalid `MaxResults`, `StartRecordName`, or `StartRecordType`.	Split the request into batches of 100 or fewer, or fix the parameter.
`InvalidResourceRecordSetProtectionBatch` (400)	One or more entries are invalid, already have a protection (create), or have no protection (delete). The `Errors` list has one entry for each failing record, with a `Code` of `InvalidInput`, `ProtectionAlreadyExists`, or `ProtectionNotFound`.	Fix or remove the listed entries and send the request again. Route 53 didn't change any protections.
`ProtectionAlreadyExists` (409)	Every entry in a create request already has a protection.	No action is needed if you only wanted the records protected. To change a comment, delete the protection and create it again.
`LimitsExceeded` (400)	The request would take the hosted zone over its protection quota.	Delete protections that you no longer need, or request a higher quota. For more information about the quota, see Quotas on records.
`NoSuchHostedZone` (404)	The hosted zone doesn't exist or isn't in your account.	Check the hosted zone ID.
`PriorRequestNotComplete` (400), `Throttling` (400)	A previous request for the hosted zone hasn't completed, you exceeded the request rate, or accelerated recovery is being turned on or off.	Retry with exponential backoff.

Error	Cause	What to do
InvalidPaginationToken (400)	NextToken is invalid or was returned for a different hosted zone.	Start the listing again without NextToken .

Treat an unknown ProtectionError code as an error, because new values can be added.

Configuring DNSSEC signing in Amazon Route 53

Domain Name System Security Extensions (DNSSEC) signing lets DNS resolvers validate that a DNS response came from Amazon Route 53 and has not been tampered with. When you use DNSSEC signing, every response for a hosted zone is signed using public key cryptography. For an overview of DNSSEC, see the DNSSEC section of AWS re:Invent 2021 - Amazon Route 53: A year in review.

In this chapter, you can learn how to enable DNSSEC signing for Route 53, how to work with key-signing keys (KSKs), and how to troubleshoot issues. You can work with DNSSEC signing in the AWS Management Console or programmatically with the API. For more information about using the CLI or SDKs to work with Route 53, see Set up Amazon Route 53.

Before you enable DNSSEC signing, note the following:

- DNSSEC signing is supported only for public hosted zones. You can't create key-signing keys (KSKs) or enable DNSSEC signing for a private hosted zone. If you need DNSSEC validation for DNS queries from your VPCs, see Enabling DNSSEC validation in Amazon Route 53.
- To help prevent a zone outage and avoid problems with your domain becoming unavailable, you must quickly address and resolve DNSSEC errors. We strongly recommend that you set up a CloudWatch alarm that alerts you whenever a DNSSECInternalFailure or DNSSECKeySigningKeysNeedingAction error is detected. For more information, see Monitoring hosted zones using Amazon CloudWatch.
- There are two kinds of keys in DNSSEC: a key-signing key (KSK) and a zone-signing key (ZSK). In Route 53 DNSSEC signing, each KSK is based on an asymmetric customer managed key in AWS KMS that you own. You are responsible for KSK management, which includes rotating it if needed. ZSK management is performed by Route 53.
- When you enable DNSSEC signing for a hosted zone, Route 53 limits the TTL to one week. If you set a TTL of more than one week for records in the hosted zone, you don't get an error. However, Route 53 enforces a TTL of one week for the records. Records that have a TTL of less

than one week and records in other hosted zones that do not have DNSSEC signing enabled are not affected.

- When you use DNSSEC signing, multi-vendor configurations are not supported. If you have configured white-label name servers (also known as vanity name servers or private name servers), make sure those name servers are provided by a single DNS provider.
- Some DNS providers do not support Delegation Signer (DS) records in their authoritative DNS. If your parent zone is hosted by a DNS provider who does not support DS queries (not setting AA flag in the DS query response), then when you enable DNSSEC in its child zone, the child zone will become unresolvable. Make sure your DNS provider supports DS records.
- It can be helpful to set up IAM permissions to allow another user, besides the zone owner, to add or remove records in the zone. For example, a zone owner can add a KSK and enable signing, and might also be responsible for key rotation. However, someone else might be responsible for working with other records for the hosted zone. For an example IAM policy, see Example permissions for a domain record owner.
- To check to see if the TLD has DNSSEC support, see Domains that you can register with Amazon Route 53.

Topics

- Enabling DNSSEC signing and establishing a chain of trust
- Disabling DNSSEC signing
- Working with customer managed keys for DNSSEC
- Working with key-signing keys (KSKs)
- KMS key and ZSK management in Route 53
- DNSSEC proofs of nonexistence in Route 53
- Troubleshooting DNSSEC signing

Enabling DNSSEC signing and establishing a chain of trust

The incremental steps apply to the hosted zone owner and the parent zone maintainer. This can be the same person, but if not, the zone owner should notify and work with the parent zone maintainer.

We recommend following the steps in this article to have your zone signed and included in the chain of trust. The following steps minimize the risk of onboarding onto DNSSEC.

> **Note**
>
> Make sure you read the prerequisites before you start in Configuring DNSSEC signing in Amazon Route 53.

There are three steps to take to enable DNSSEC signing, as described in the following sections.

Topics

- Step 1: Prepare for enabling DNSSEC signing
- Step 2: Enable DNSSEC signing and create a KSK
- Step 3: Establish chain of trust

Step 1: Prepare for enabling DNSSEC signing

The preparation steps help you minimize the risk of onboarding to DNSSEC by monitoring zone availability and lowering wait times between enabling signing and the insertion of the Delegation Signer (DS) record.

To prepare for enabling DNSSEC signing

1. Monitor zone availability.

   You can monitor the zone for the availability of your domain names. This can help you address any issues that might warrant rolling a step back after you enable DNSSEC signing. You can monitor for your domain names with most traffic by using query logging. For more information about setting up query logging, see Monitoring Amazon Route 53.

   The monitoring can be done through a shell script, or through a third party service. It shouldn't, however, be the only signal to determine if a rollback is required. You might also get feedback from your customers due to a domain not being available.

2. Lower the zone's maximum TTL.

   The zone's maximum TTL is the longest TTL record in the zone. In the following example zone, the zone's maximum TTL is 1 day (86400 seconds).

Example zone records

Name	TTL	Record class	Record type	Record data
example.com.	900	IN	SOA	ns1.example.com. hostmaster.example.com. 2002022401 10800 15 604800 300
example.com.	900	IN	NS	ns1.example.com.
route53.example.com.	86400	IN	TXT	`some txt record`

Lowering the zone's maximum TTL helps reduce the wait time between enabling signing and the insertion of the Delegation Signer (DS) record. We recommend lowering the zone's maximum TTL to 1 hour (3600 seconds). You can then roll back after only an hour if any resolver has problems with caching signed records.

Rollback: undo the TTL changes.

3. Lower the SOA TTL and SOA minimum field.

The SOA minimum field is the last field in the SOA record data. In the following example SOA record, the minimum field has the value of 5 minutes (300 seconds).

Example SOA record

Name	TTL	Record class	Record type	Record data
example.com.	900	IN	SOA	ns1.example.com.

Name	TTL	Record class	Record type	Record data
				hostmaster.example.com. 2002022401 10800 15 604800 300

The SOA TTL and SOA minimum field determines how long resolvers remember negative answers. After you enable signing, Route 53 name servers start returning NSEC records for negative answers. The NSEC contains information that resolvers might use to synthesize a negative answer. If you have to roll back because the NSEC information caused a resolver to assume a negative answer for a name, then you only have to wait for the maximum of the SOA TTL and SOA minimum field for the resolver to stop the assumption.

Rollback: undo the SOA changes.

4. Make sure the TTL and SOA minimum field changes are effective.

   Use GetChange to make sure your changes so far have been propagated to all Route 53 DNS servers.

Step 2: Enable DNSSEC signing and create a KSK

You can enable DNSSEC signing and create a key-signing key (KSK) by using AWS CLI or on the Route 53 console.

- CLI
- Console

When you provide or create a KMS key, there are several requirements. For more information, see Working with customer managed keys for DNSSEC.

CLI

You can use a key that you already have, or create one by running an AWS CLI command like the following using your own values for `hostedzone_id`, `cmk_arn`, `ksk_name`, and `unique_string` (to make the request unique):

```
aws --region us-east-1 route53 create-key-signing-key \
   --hosted-zone-id $hostedzone_id \
   --key-management-service-arn $cmk_arn --name $ksk_name \
   --status ACTIVE \
   --caller-reference $unique_string
```

For more information about customer managed keys, see Working with customer managed keys for DNSSEC. See also CreateKeySigningKey.

To enable DNSSEC signing, run an AWS CLI command like the following, using your own value for the `hostedzone_id`:

```
aws --region us-east-1 route53 enable-hosted-zone-dnssec \
   --hosted-zone-id $hostedzone_id
```

For more information, see enable-hosted-zone-dnssec and EnableHostedZoneDNSSEC.

Console

To enable DNSSEC signing and create a KSK

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and then choose a hosted zone that you want to enable DNSSEC signing for.
3. On the **DNSSEC signing** tab, choose **Enable DNSSEC signing**.

   > **Note**
   >
   > If the option in this section is **Disable DNSSEC signing**, you have already completed the first step in enabling DNSSEC signing. Be sure that you establish, or that there already exists, a chain of trust for the hosted zone for DNSSEC, and then you're done. For more information, see Step 3: Establish chain of trust.

4. In the **Key-signing key (KSK) creation** section, choose **Create new KSK**, and under **Provide KSK name**, enter a name for the KSK that Route 53 creates for you. The name can include numbers, letters, and underscores (_). It must be unique.

5. Under **Customer managed CMK**, choose the customer managed key for Route 53 to use when it creates the KSK for you. You can use an existing customer managed key that applies to DNSSEC signing, or create a new customer managed key.

   When you provide or create a customer managed key, there are several requirements. For more information, see Working with customer managed keys for DNSSEC.

6. Enter the alias for an existing customer managed key. If you want to use a new customer managed key, enter an alias for the customer managed key, and Route 53 creates one for you.

   > **Note**
   >
   > If you choose to have Route 53 create a customer managed key, be aware that separate charges apply for each customer managed key. For more information, see AWS Key Management Service pricing.

7. Choose **Enable DNSSEC signing**.

After you enable zone signing, complete the following steps (whether you used the console or the CLI):

1. Ensure zone signing is effective.

   If you used AWS CLI, you can use the operation Id from the output of the `EnableHostedZoneDNSSEC()` call to run get-change or GetChange to make sure that all Route 53 DNS Servers are signing responses (status = `INSYNC`).

2. Wait for at least the previous zone's maximum TTL.

   Wait for resolvers to flush all unsigned records from their cache. To achieve this you should wait for at least the previous zone's maximum TTL. In the `example.com` zone above, the wait time would be 1 day.

3. Monitor for reports of customer issues.

   After you have enabled zone signing, your customers might start seeing issues related to network devices and resolvers. The recommended monitoring period is 2 weeks.

   The following are examples of issues that you might see:

- Some network devices can limit DNS response size to under 512 bytes, which is too small for some signed responses. These network devices should be reconfigured to allow larger DNS response sizes.
- Some network devices do a deep inspection on DNS responses and strips certain records it doesn't understand, like the ones used for DNSSEC. These devices should be reconfigured.
- Some customers' resolvers claim that they can accept a larger UDP response than their network supports. You can test your network capability and configure your resolvers appropriately. For more information see, DNS Reply Size Test Server.

Rollback: call DisableHostedZoneDNSSEC then rollback the steps in Step 1: Prepare for enabling DNSSEC signing.

Step 3: Establish chain of trust

After you enable DNSSEC signing for a hosted zone in Route 53, establish a chain of trust for the hosted zone to complete your DNSSEC signing setup. You do this by creating a Delegation Signer (DS) record in the *parent* hosted zone, for your hosted zone, using the information that Route 53 provides. Depending on where your domain is registered, you add the record to the parent hosted zone in Route 53 or at another domain registrar.

To establish a chain of trust for DNSSEC signing

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and then choose a hosted zone that you want to establish a DNSSEC chain of trust for. *You must enable DNSSEC signing first.*
3. On the **DNSSEC signing** tab, under **DNSSEC signing**, choose **View information to create DS record**.

> **Note**
>
> If you don't see **View information to create DS record** in this section, then you must enable DNSSEC signing before you establish the chain of trust. Choose **Enable DNSSEC signing** and complete the steps as described in Step 2: Enable DNSSEC signing and create a KSK, and then return to these steps to establish the chain of trust.

4. Under **Establish a chain of trust**, choose either **Route 53 registrar** or **Another domain registrar**, depending on where your domain is registered.
5. Use the provided values from step 3 to create a DS record for the parent hosted zone in Route 53. If your domain is not hosted at Route 53, use the provided values to create a DS record at your domain registrar website.

   Establish a chain of trust for the parent zone:

   - If your domain is managed through Route 53, follow these steps:

     Make sure that you configure the correct signing algorithm (ECDSAP256SHA256 and type 13) and digest algorithm (SHA-256 and type 2).

     If Route 53 is your registrar, do the following in the Route 53 console:

     1. Note the **Key type**, **Signing algorithm**, and **Public key** values. In the navigation pane, choose **Registered domains**.
     2. Select a domain, and then, within the **DNSSEC keys** tab, choose **Add key**.
     3. In the **Manage DNSSEC keys** dialog box, choose the appropriate **Key type** and **Algorithm** for the **Route 53 registrar** from the dropdown menus.
     4. Copy the **Public key** for the Route 53 registrar. In the **Manage DNSSEC keys** dialog box, paste the value into the **Public key** box.
     5. Choose **Add**.

        Route 53 adds the DS record to the parent zone from the public key. For example, if your domain is `example.com`, the DS record is added to the .com DNS zone.

   - If your domain is managed on another registry, follow the instructions in the **Another domain registrar** section.

     To make sure the following steps go smoothly, introduce a low DS TTL to the parent zone. We recommend setting the DS TTL to 5 minutes (300 seconds) for faster recovery if you need to roll your changes back.

     - Establish a chain of trust for the child zone:

       If your parent zone is administered by another registry, contact your registrar to introduce the DS record for your zone. Typically you can't adjust the TTL of the DS record.

     - If your parent zone is hosted on Route 53, contact the parent zone owner to introduce the DS record for your zone.

Provide the `$ds_record_value` to the parent zone owner. You can get it by clicking on the **View Information to create DS record** in the console and copying the **DS record** field, or by calling GetDNSSEC API and retrieving the value of the 'DSRecord' field:

```
aws --region us-east-1 route53 get-dnssec
            --hosted-zone-id $hostedzone_id
```

The parent zone owner can insert the record through the Route 53 console or CLI.

- To insert the DS record by using AWS CLI, the parent zone owner creates and names a JSON file similar to the following example. The parent zone owner might name the file something like `inserting_ds.json`.

```
{
    "HostedZoneId": "$parent_zone_id",
    "ChangeBatch": {
        "Comment": "Inserting DS for zone $zone_name",
        "Changes": [
            {
                "Action": "UPSERT",
                "ResourceRecordSet": {
                    "Name": "$zone_name",
                    "Type": "DS",
                    "TTL": 300,
                    "ResourceRecords": [
                        {
                            "Value": "$ds_record_value"
                        }
                    ]
                }
            }
        ]
    }
}
```

Then run the following command:

```
aws --region us-east-1 route53 change-resource-record-sets
            --cli-input-json file://inserting_ds.json
```

- To insert the DS record by using the console,

Open the Route 53 console at https://console.aws.amazon.com/route53/.

In the navigation pane, choose **Hosted zones**, the name of your hosted zone and then **Create record** button. Make sure you choose Simple routing for the **Routing policy**.

In the **Record name** field enter the same name as the `$zone_name`, select DS from the **Record type** drop-down, and enter the value of `$ds_record_value` into the **Value** field, and choose **Create records**.

Rollback: remove the DS from the parent zone, wait for the DS TTL, and then roll back the steps for establishing trust. If the parent zone is hosted on Route 53, the parent zone owner can change the `Action` from UPSERT to `DELETE` in the JSON file, and re-run the example CLI above.

6. Wait for the updates to propagate, based on the TTL for your domain records.

   If the parent zone is on Route 53 DNS service, the parent zone owner can confirm full propagation through the GetChange API.

   Otherwise, you can periodically probe the parent zone for the DS record, and then wait another 10 minutes afterwards to increase the probability of the DS record insertion being fully propagated. Do note that some registrars have scheduled DS insertion, for example, once a day.

When you introduce the Delegation Signer (DS) record in the parent zone, the validated resolvers that have picked up the DS start validating responses from the zone.

To make sure the steps for establishing trust go smoothly, complete the following:

1. Find the maximum NS TTL.

   There are 2 sets of NS records associated with your zones:

   - The delegation NS record — this is the NS record for your zone held by the parent zone. You can find this by running the following Unix commands (if your zone is example.com, the parent zone is com):

     ```
     dig -t NS com
     ```

Pick one of the NS records and then run the following:

```
dig @one of the NS records of your parent zone -t NS example.com
```

For example:

```
dig @b.gtld-servers.net. -t NS example.com
```

- The in-zone NS record — this is the NS record in your zone. You can find this by running the following Unix command:

  ```
  dig @one of the NS records of your zone -t NS example.com
  ```

  For example:

  ```
  dig @ns-0000.awsdns-00.co.uk. -t NS example.com
  ```

  Note the maximum TTL for both zones.

2. Wait for the maximum NS TTL.

   Prior to the DS insertion, resolvers are getting a signed response, but aren't validating the signature. When the DS record is inserted, resolvers don't see it until the NS record for the zone expires. When resolvers re-fetch the NS record, the DS record is then also returned.

   If your customer is running a resolver on a host with an out of sync clock, make sure the clock is within 1 hour of the correct time.

   After completing this step, all DNSSEC-aware resolvers validate your zone.

3. Observe name resolution.

   You should observe that there are no issues with resolvers validating your zone. Make sure you also account for the time needed for your customers to report problems to you.

   We recommend monitoring for up to 2 weeks.

4. (Optional) Lengthen the DS and NS TTLs.

   If you are satisfied with the setup, you can save the TTL and SOA changes you made. Note that Route 53 limits the TTL to 1 week for signed zones. For more information, see Configuring DNSSEC signing in Amazon Route 53.

   If you can change the DS TTL, we recommend that you set it to 1 hour.

Disabling DNSSEC signing

The steps for disabling DNSSEC signing in Route 53 vary, depending on the chain of trust that your hosted zone is part of.

For example, your hosted zone might have a parent zone that has a Delegation Signer (DS) record, as part of a chain of trust. Your hosted zone might also be itself a parent zone for child zones that have enabled DNSSEC signing, which is another part of the chain of trust. Investigate and determine the full chain of trust for your hosted zone before you take the steps to disable DNSSEC signing.

The chain of trust for your hosted zone that enables DNSSEC signing must be carefully undone as you disable signing. To remove your hosted zone from the chain of trust, you remove all DS records that are in place for the chain of trust that includes this hosted zone. This means that you must do the following, in order:

1. Remove any DS records that this hosted zone has for child zones that are part of a chain of trust.
2. Remove the DS record from the parent zone. Skip this step if you have an island of trust (there are no DS records in the parent zone and no DS records for child zones in this zone).
3. If you are not able to remove DS records, in order to remove the zone from the chain of trust, remove NS records from the parent zone. For more information, see Adding or changing name servers and glue records for a domain.

The following incremental steps allow you to monitor the effectiveness of the individual steps to avoid DNS availability issues in your zone.

To disable DNSSEC signing

1. Monitor zone availability.

   You can monitor the zone for the availability of your domain names. This can help you address any issues that might warrant rolling a step back after you enable DNSSEC signing. You can monitor for your domain names with most traffic by using query logging. For more information about setting up query logging, see Monitoring Amazon Route 53.

   The monitoring can be done through a shell script, or through a paid service. It shouldn't, however, be the only signal to determine if a rollback is required. You might also get feedback from your customers due to a domain not being available.

2. Find the current DS TTL.

   You can find the DS TTL by running the following Unix command:

   ```
   dig -t DS example.com example.com
   ```

3. Find the maximum NS TTL.

   There are 2 sets of NS records associated with your zones:

   - The delegation NS record — this is the NS record for your zone held by the parent zone. You can find this by running the following Unix commands:

     First find the NS of your parent zone (if your zone is example.com, the parent zone is com):

     ```
     dig -t NS com
     ```

     Pick one of the NS records and then run the following:

     ```
     dig @one of the NS records of your parent zone -t NS example.com
     ```

     For example:

     ```
     dig @b.gtld-servers.net. -t NS example.com
     ```

   - The in-zone NS record — this is the NS record in your zone. You can find this by running the following Unix command:

     ```
     dig @one of the NS records of your zone -t NS example.com
     ```

     For example:

     ```
     dig @ns-0000.awsdns-00.co.uk. -t NS example.com
     ```

     Note the maximum TTL for both zones.

4. Remove the DS record from the parent zone.

   Contact the parent zone owner to remove the DS record.

   Rollback: re-insert the DS record, confirm DS insertion is effective, and wait for the maximum NS (not DS) TTL before all resolvers will start validating again.

5. Confirm the DS removal is effective.

If the parent zone is on Route 53 DNS service, the parent zone owner can confirm full propagation through the GetChange API.

Otherwise, you can periodically probe the parent zone for the DS record, and then wait another 10 minutes afterwards to increase the probability of the DS record removal being fully propagated. Do note that some registrars have scheduled DS removal, for example once a day.

6. Wait for the DS TTL.

   Wait until all resolvers have expired the DS record from their caches.

7. Disable DNSSEC signing and deactivate the key-signing key (KSK).

   - CLI
   - Console

   CLI

   Call DisableHostedZoneDNSSEC and DeactivateKeySigningKey APIs.

   For example:

```
aws --region us-east-1 route53 disable-hosted-zone-dnssec \
            --hosted-zone-id $hostedzone_id

aws --region us-east-1 route53 deactivate-key-signing-key \
            --hosted-zone-id $hostedzone_id --name $ksk_name
```

   Console

   To disable DNSSEC signing

   1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
   2. In the navigation pane, choose **Hosted zones**, and then choose a hosted zone that you want to disable DNSSEC signing for.
   3. On the **DNSSEC signing** tab, choose **Disable DNSSEC signing**.

4. On the **Disable DNSSEC signing** page, choose one of the following options, depending on your scenario for the zone that you're disabling DNSSEC signing for.
   - **Parent zone only** — This zone has a parent zone with a DS record. In this scenario, you must remove the parent zone's DS record.
   - **Child zones only** — This zone has a DS record for a chain of trust with one or more child zones. In this scenario, you must remove the zone's DS records.
   - **Parent and child zones** — This zone has both a DS record for a chain of trust with one or more child zones *and* a parent zone with a DS record. In this scenario, do the following, in order:

     a. Remove the zone's DS records.

     b. Remove the parent zone's DS record.

     If you have an island of trust, you can skip this step.
5. Determine what the TTL is for each DS record that you remove in Step 4.,Make sure that the longest TTL period has expired.
6. Select the check box to confirm that you have followed the steps in order.
7. Type *disable* in the field, as shown, and then choose **Disable**.

To deactivate the key-signing key (KSK)

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and then choose a hosted zone that you want to deactivate the key-signing key (KSK) for.
3. In the **Key-signing keys (KSKs)** section, choose the KSK you want to deactivate, and under **Actions**, choose **Edit KSK**, set **KSK status** to **Inactive**, and then choose **Save KSK**.

Rollback: call ActivateKeySigningKey and EnableHostedZoneDNSSEC APIs.

For example:

```
aws --region us-east-1 route53 activate-key-signing-key \
            --hosted-zone-id $hostedzone_id --name $ksk_name
```

```
aws --region us-east-1 route53 enable-hosted-zone-dnssec \
            --hosted-zone-id $hostedzone_id
```

8. Confirm disabling zone signing is effective.

   Use the Id from the `EnableHostedZoneDNSSEC()` call to run GetChange to make sure that all Route 53 DNS Servers have stopped signing responses (status = `INSYNC`).

9. Observe name resolution.

   You should observe that there are no issues resulting in resolvers validating your zone. Allow 1-2 weeks to also account for the time needed for your customers to report problems to you.

10. (Optional) Clean up.

    If you don't re-enable signing, you can clean up the KSKs through DeleteKeySigningKey and delete the corresponding customer managed key to save costs.

Working with customer managed keys for DNSSEC

When you enable DNSSEC signing in Amazon Route 53, Route 53 creates a key-signing key (KSK) for you. To create a KSK, Route 53 must use a customer managed key in AWS Key Management Service that supports DNSSEC. This section describes the details and requirements for the customer managed key that are helpful to know as you work with DNSSEC.

Keep the following in mind when you work with customer managed keys for DNSSEC:

- The customer managed key that you use with DNSSEC signing must be in the US East (N. Virginia) Region.
- The customer managed key must be an asymmetric customer managed key with an ECC_NIST_P256 key spec. These customer managed keys are used only for signing and verification. For help creating an asymmetric customer managed key, see Creating asymmetric customer managed keys in the AWS Key Management Service Developer Guide. For help finding the cryptographic configuration of an existing customer managed key, see Viewing the cryptographic configuration of customer managed keys in the AWS Key Management Service Developer Guide.

- If you create a customer managed key yourself to use with DNSSEC in Route 53, you must include specific key policy statements that give Route 53 the required permissions. Route 53 must be able to access your customer managed key so that it can create a KSK for you. For more information, see Route 53 customer managed key permissions required for DNSSEC signing.
- Route 53 can create a customer managed key for you in AWS KMS to use with DNSSEC signing without additional AWS KMS permissions. However, you must have specific permissions if you want to edit the key after it's created. The specific permissions that you must have are the following: `kms:UpdateKeyDescription`, `kms:UpdateAlias`, and `kms:PutKeyPolicy`.
- Be aware that separate charges apply for each customer managed key that you have, whether you create the customer managed key or Route 53 creates it for you. For more information, see AWS Key Management Service pricing.

Working with key-signing keys (KSKs)

When you enable DNSSEC signing, Route 53 creates a key-signing key (KSK) for you. You can have up to two KSKs per hosted zone in Route 53. After you enable DNSSEC signing, you can add, remove, or edit your KSKs.

Note the following when you work with your KSKs:

- Before you can delete a KSK, you must edit the KSK to set its status to **Inactive**.
- When DNSSEC signing is enabled for a hosted zone, Route 53 limits the TTL to one week. If you set a TTL for records in the hosted zone to more than one week, you don't get an error, but Route 53 enforces a TTL of one week.
- To help prevent a zone outage and avoid problems with your domain becoming unavailable, you must quickly address and resolve DNSSEC errors. We strongly recommend that you set up a CloudWatch alarm that alerts you whenever a `DNSSECInternalFailure` or `DNSSECKeySigningKeysNeedingAction` error is detected. For more information, see Monitoring hosted zones using Amazon CloudWatch.
- The KSK operations described in this section allow you to rotate your zone's KSKs. For more information and a step-by-step example, see *DNSSEC Key Rotation* in the blog post Configuring DNSSEC signing and validation with Amazon Route 53.

To work with KSKs in the AWS Management Console, follow the guidance in the following sections.

Add a key-signing key (KSK)

When you enable DNSSEC signing, Route 53 creates a key-signing (KSK) for you. You can also add KSKs separately. You can have up to two KSKs per hosted zone in Route 53.

When you create a KSK, you must provide or request Route 53 to create a customer managed key to use with the KSK. When you provide or create a customer managed key, there are several requirements. For more information, see Working with customer managed keys for DNSSEC.

Follow these steps to add a KSK in the AWS Management Console.

To add a KSK

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and then choose a hosted zone.
3. On the **DNSSEC signing** tab, under **Key-signing keys (KSKs)**, choose **Switch to advanced view**, and then, under **Actions**, choose **Add KSK**.
4. Under **KSK**, enter a name for the KSK that Route 53 creates for you. The name can include numbers, letters, and underscores (_). It must be unique.
5. Enter the alias for a customer managed key that applies to DNSSEC signing, or enter an alias for a new customer managed key that Route 53 creates for you.

   > **Note**
   >
   > If you choose to have Route 53 create a customer managed key, be aware that separate charges apply for each customer managed key. For more information, see AWS Key Management Service pricing.

6. Choose **Create KSK**.

Edit a key-signing key (KSK)

You can edit the status of a KSK to be **Active** or **Inactive**. When a KSK is active, Route 53 uses that KSK for DNSSEC signing. Before you can delete a KSK, you must edit the KSK to set its status to **Inactive**.

Follow these steps to edit a KSK in the AWS Management Console.

To edit a KSK

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and then choose a hosted zone.
3. On the **DNSSEC signing** tab, under **Key-signing keys (KSKs)**, choose **Switch to advanced view**, and then, under **Actions**, choose **Edit KSK**.
4. Make the desired updates to the KSK, and then choose **Save**.

Delete a key-signing key (KSK)

Before you can delete a KSK, you must edit the KSK to set its status to **Inactive**.

One reason that you might delete a KSK is as part of routine key rotation. It's a best practice to rotate cryptographic keys periodically. Your organization might have standard guidance for how often to rotate keys.

Follow these steps to delete a KSK in the AWS Management Console.

To delete a KSK

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**, and then choose a hosted zone.
3. On the **DNSSEC signing** tab, under **Key-signing keys (KSKs)**, choose **Switch to advanced view**, and then under **Actions**, choose **Delete KSK**.
4. Follow the guidance to confirm deleting the KSK.

KMS key and ZSK management in Route 53

This section describes the current practice Route 53 uses for your DNSSEC signing enabled zones.

> **Note**
>
> Route 53 uses the following rule which might change. Any future changes will not reduce your zone's or Route 53's security posture.

How Route 53 uses the AWS KMS associated with your KSK

In DNSSEC, the KSK is used to generate the resource record signature (RRSIG) for the DNSKEY resource record set. All `ACTIVE` KSKs are used in the RRSIG generation. Route 53 generates an RRSIG by calling the `Sign` AWS KMS API on the associated KMS key. For more information, see Sign in the *AWS KMS API guide*. These RRSIGs do not count towards the zone's resource record set limit.

RRSIG has an expiration. To prevent the RRSIGs from expiring, the RRSIGs are refreshed regularly by regenerating them every one to seven days.

The RRSIGs are also refreshed every time you call any of these APIs:

- ActivateKeySigningKey
- CreateKeySigningKey
- DeactivateKeySigningKey
- DeleteKeySigningKey
- DisableHostedZoneDNSSEC
- EnableHostedZoneDNSSEC

Every time Route 53 performs a refresh, we generate 15 RRSIGs to cover the next few days in case the associated KMS key becomes inaccessible. For KMS key cost estimation, you can assume a once a day regular refresh. A KMS key might become inaccessible by inadvertent changes to the KMS key policy. Inaccessible KMS key will set the associated KSK's status to `ACTION_NEEDED`. We strongly recommend that you monitor this condition by setting up a CloudWatch alarm whenever a `DNSSECKeySigningKeysNeedingAction` error is detected because validating resolvers will start failing lookups after the last RRSIG expires. For more information, see Monitoring hosted zones using Amazon CloudWatch.

How Route 53 manages your zone's ZSK

Each new hosted zone with DNSSEC signing enabled will have one `ACTIVE` zone signing key (ZSK). The ZSK is generated separately for each hosted zone and is owned by Route 53. The current key algorithm is ECDSAP256SHA256.

We will start performing regular ZSK rotation on the zone within 7–30 days of the start of signing. Currently, Route 53 uses the Pre-Publish Key Rollover method. For more information, see Pre-Publish Zone Signing Key Rollover. This method will introduce another ZSK to the zone. The rotation will be repeated every 7–30 days.

Route 53 will suspend ZSK rotation if any of the zone's KSK is in `ACTION_NEEDED` status because Route 53 will not be able to regenerate the RRSIGs for DNSKEY resource record sets to account for the changes in the zone's ZSK. ZSK rotation will automatically resume after the condition is cleared.

DNSSEC proofs of nonexistence in Route 53

> **Note**
>
> Route 53 uses the following rule which might change. Any future changes will not reduce your zone's or Route 53's security posture.

There are three kinds of proof of nonexistence in DNSSEC:

- Proof of nonexistence of a record matching the query name.
- Proof of nonexistence of a type matching the query type.
- Proof of existence of a wildcard record used to generate the record in response.

Route 53 implements the proof of nonexistence of a record matching the query name using the BL method. For more information, see BL. It's a method that produces a compact representation of the proof and prevents zone walking.

In cases where there is a record matching the query name but not the query type (such as querying for web.example.com/AAAA but there is only web.example.com/A present), we return a minimal NSEC (next secure) record containing all supported resource record types.

When Route 53 synthesizes an answer from a wildcard record, the response will not be accompanied with a next secure record, or NSEC record for the wildcard. Such NSEC record is used in some implementations, typically those performing offline signing, to prevent the resource record signatures (RRSIG) in the response from being re-used to spoof a different response. Route 53 uses online signing for non-DNSKEY records to generate RRSIGs specific to the response which cannot be re-used for a different response.

Troubleshooting DNSSEC signing

The information in this section can help you address issues with DNSSEC signing, including enabling, disabling, and with your key-signing keys (KSKs).

Enabling DNSSEC

Make sure you have read the prerequisites in Configuring DNSSEC signing in Amazon Route 53 before you start enabling DNSSEC signing.

Disabling DNSSEC

In order to safely disable DNSSEC, Route 53 will check whether the target zone is in the chain of trust. It checks if the parent of the target zone has any NS records of the target zone and DS records of the target zone. If the target zone is not publicly resolvable, for example, getting a SERVFAIL response when querying for NS and DS, Route 53 cannot determine whether it is safe to disable DNSSEC. You can contact your parent zone to fix those issues, and retry disabling DNSSEC later.

KSK status is **Action needed**

A KSK can change its status to **Action needed** (or `ACTION_NEEDED` in a KeySigningKey status), when Route 53 DNSSEC loses access to a corresponding AWS KMS key (due to a change in permissions or AWS KMS key deletion).

If the status of a KSK is **Action needed**, it means that eventually it'll cause a zone outage for clients using DNSSEC validating resolvers and you must act fast to prevent a production zone becoming un-resolvable.

To correct the problem, make sure that the customer managed key that your KSK is based on is enabled and has the correct permissions. For more information about the required permissions, see Route 53 customer managed key permissions required for DNSSEC signing.

After you have fixed the KSK, activate it again by using the console or the AWS CLI, as described in Step 2: Enable DNSSEC signing and create a KSK.

To prevent this issue in the future, consider adding an Amazon CloudWatch metric to track the state of the KSK as suggested in Configuring DNSSEC signing in Amazon Route 53.

KSK status is **Internal failure**

When a KSK has a status of **Internal failure** (or `INTERNAL_FAILURE` in a KeySigningKey status), you can't work with any other DNSSEC entities until the problem is resolved. You must take action before you can work with DNSSEC signing, including working with this KSK or your other KSK.

To correct the problem, try again to activate or deactivate the KSK.

To correct the problem when working with the APIs, try enabling signing (EnableHostedZoneDNSSEC) or disabling signing (DisableHostedZoneDNSSEC).

It's important that you correct **Internal failure** problems promptly. You can't make any other changes to the hosted zone until you correct the problem, except the operations to fix the **Internal failure**.

Using AWS Cloud Map to create records and health checks

To route internet traffic or Amazon VPC traffic to application components or microservices, use AWS Cloud Map. It automatically creates DNS records and, optionally, health checks. For more information, see the AWS Cloud Map Developer Guide.

DNS constraints and behaviors

DNS messaging is subject to factors that affect how you create and use hosted zones and records. This section explains these factors.

Maximum response size

To comply with DNS standards, responses sent over UDP are no more than 512 bytes in size. Responses exceeding 512 bytes are truncated and the resolver must re-issue the request over TCP. If the resolver supports EDNS0 (as defined in RFC 2671), and advertises the EDNS0 option to Amazon Route 53, Route 53 permits responses up to 4096 bytes over UDP, without truncation.

Authoritative section processing

For successful queries, Route 53 appends name server (NS) records for the relevant hosted zone to the Authority section of the DNS response. For names that are not found (NXDOMAIN responses), Route 53 appends the start of authority (SOA) record (as defined in RFC 1035) for the relevant hosted zone to the Authority section of the DNS response.

Additional section processing

Route 53 appends records to the Additional section. If the records are known and appropriate, the service appends A or AAAA records for any target of an MX, CNAME, NS, or SRV record cited in the Answer section. For more information about these DNS record types, see Supported DNS record types.

Related topics

For information about transferring domain registration (not just DNS hosting) to Route 53, see the following topics:

- Pre-transfer checklist for domain transfers – Finish this checklist before transferring your domain to avoid common transfer failures.
- Transferring registration for a domain to Amazon Route 53 – Step-by-step process for transferring domain registration from another registrar to Route 53.
- Transferring domains – Overview of all domain transfer options, including transfers between AWS accounts.

Using Traffic Flow to route DNS traffic

Traffic Flow greatly simplifies the process of creating and maintaining records in large and complex configurations.

Managing related records in a hosted zone can be challenging in the following circumstances:

- You have a lot of resources that perform the same operation, such as web servers that serve traffic for the same domain.
- You want to create a complex tree of records using alias records and a combination of Route 53 routing policies, such as latency, failover, and weighted.

Traffic Flow advantages

To make it easier to track the records and their relationships, Traffic Flow simplifies DNS record creation with the following features:

Visual editor

With the Traffic Flow visual editor, you can create complex trees of records and see the relationships among the records. For example, you might create a configuration in which latency alias records reference weighted records, and the weighted records reference your resources in multiple AWS Regions. Each configuration is known as a *traffic policy*. You can create as many traffic policies as you want at no charge.

Versioning

You can create multiple versions of a traffic policy so you don't have to start all over when your configuration changes. Old versions continue to exist until you delete them; there's a default limit of 1000 versions per traffic policy. You can optionally give each version a description.

Automatic record creation and updating

A traffic policy can represent dozens or even hundreds of records. With Traffic Flow, you can create all those records automatically by creating a *traffic policy record*. You specify the hosted zone and the name of the record at the root of the tree, such as example.com or www.example.com, and Route 53 automatically creates all the other records in the tree. The root record—the traffic policy record—appears in the list of records for your hosted zone. Route 53 hides all the other records.

When you create a new version of a traffic policy, you can selectively update traffic policy records that you created using the previous traffic policy version. When you update a traffic policy record, Route 53 automatically updates all the other records in the tree. You can also quickly roll back changes by updating a traffic policy record again to use a previous version of a traffic policy.

> **Note**
>
> You can use Traffic Flow to create records only in public hosted zones.

Geoproximity routing policy

When using Traffic Flow, you can more intuitively understand how traffic is routed to each of your global endpoints by using the geoproximity map on the Traffic Flow visual editor. For more information, see Geoproximity routing.

Reuse for multiple records in different hosted zones

You can use a traffic policy to automatically create records in multiple public hosted zones. For example, if you're using the same web servers for multiple domain names, you can use the same traffic policy to create traffic policy records in the hosted zones for example.com, example.org, and example.net.

When a client submits a query for the name of the root record, such as example.com or www.example.com, Route 53 responds to the query based on the configuration in the traffic policy that you used to create the corresponding traffic policy record.

There's a monthly charge for each traffic policy record. For more information, see the "Traffic Flow" section of Amazon Route 53 pricing.

To minimize these charges, you can create one or more alias records in a hosted zone that reference a traffic policy record in that hosted zone. For example, you can create a traffic policy record for example.com and then create an alias record for www.example.com that references the traffic policy record.

Creating and managing traffic policies

Topics

- Creating a traffic policy
- Values that you specify when you create a traffic policy
- Viewing a map that shows the effect of geoproximity settings
- Creating additional versions of a traffic policy
- Creating a traffic policy by using a JSON document
- Viewing traffic policy versions and the associated policy records
- Deleting traffic policy versions and traffic policies

Creating a traffic policy

To create a traffic policy, perform the following procedure.

To create a traffic policy

1. Design your configuration. For information about how complex DNS routing configurations work, see Configuring DNS failover in Creating Amazon Route 53 health checks.
2. Based on the design for your configuration, create the health checks that you want to use for your endpoints.
3. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
4. In the navigation pane, choose **Traffic policies**.
5. Choose **Create traffic policy**.
6. Enter a name and an optional description, and then choose **Next**.
7. On the editor, choose the DNS resource record type in the **Properties** pane. This type will be assigned to all resource records that you create for this traffic policy.

   Choose **Confirm**. For more information, see Values that you specify when you create a traffic policy.
8. In the editor page, choose **Connect to** and then choose one of **New routing rule**, **New endpoint**, **Existing routing rule**, or **Existing endpoint**. If you selected a **New routing rule**, in the **Properties** pane select the routing rule and choose **Confirm**. Then, in the **Properties** pane, enter the appropriate values. Repeat this process until your policy is done. For more

information about the values for each connection type, see Values that you specify when you create a traffic policy.

You can delete rules, endpoints, and branches of a traffic policy in the following ways:

- To delete a rule or an endpoint, select the rule or the endpoint in the editor, and then choose **Delete** in **Properties** pane.

  > **Important**
  >
  > If you delete a rule that has child rules and endpoints, Amazon Route 53 also deletes all of the children.

- If you connect two rules to the same child rule or endpoint and you want to delete one of the connections, select the connection that you want to delete, and chose the **Delete** in the **Properties** pane for that connection.

9. Choose **Create policy**.
10. On the **Create policy records** page, use the new traffic policy to create one or more policy records in one hosted zone. For more information, see Values that you specify when you create or update a policy record. You can also create policy records later, either in the same hosted zone or in additional hosted zones.

    If you don't want to create policy records now, choose **Cancel**, and the console displays the list of traffic policies and policy records that you have created by using the current AWS account.
11. If you specified settings for policy records in the preceding step, choose **Create policy records**.

Values that you specify when you create a traffic policy

When you create a traffic policy, you specify the following values.

-
-
-
-
-
-

-
-

Policy name

Enter a name that describes the traffic policy. This value appears in the list of traffic policies in the console. You can't change the name of a traffic policy after you create it.

Version

This value is assigned automatically by Amazon Route 53 when you create a traffic policy or a new version of an existing policy.

Version description

Enter a description that applies to this version of the traffic policy. This value appears in the list of traffic policy versions in the console.

DNS type

Choose the DNS type that you want Amazon Route 53 to assign to all of the records when you create a policy record by using this traffic policy version. For a list of supported types, see Supported DNS record types.

> **Important**
>
> If you're creating a new version of an existing traffic policy, you can change the DNS type. However, you can't edit a policy record and choose a traffic policy version that has a DNS type that is different from the traffic policy version that you used to create the policy record. For example, if you created a policy record by using a traffic policy version that has a **DNS type** of A, you can't edit the policy record and choose a traffic policy version that has any other value for **DNS type**.

If you want to route traffic to the following AWS resources, choose the applicable value:

- **CloudFront distribution** – Choose **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.

> **Note**
>
> CloudFront alias endpoints don't support evaluate health check. You can however, create a health check to monitor CloudFront endpoints. For more information, see Creating and updating health checks.

- **ELB Application load balancer** – Choose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.
- **ELB Classic load balancer** – Choose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.
- **ELB Network load balancer** – Choose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.
- **Elastic Beanstalk environment**: Choose **A: IP address in IPv4 format**.
- **Amazon S3 bucket configured as a website endpoint**: Choose **A: IP address in IPv4 format**.

Connect to

Choose the applicable rule or endpoint based on the design for your configuration.

Failover rule

Choose this option when you want to configure active-passive failover, in which one resource takes all traffic when it's available and the other resource takes all traffic when the first resource isn't available.

For more information, see Active-passive failover.

> **Note**
>
> Evaluate target health is checked by default. It evaluates the health of the target endpoint to which traffic is routed through an alias record. If your endpoint doesn't receive DNS traffic through an alias record, clear this option and create a health check to monitor the endpoint health. For more information, see Creating and updating health checks.

Geolocation rule

Choose this option when you want Amazon Route 53 to respond to DNS queries based on the location of your users.

For more information, see Geolocation routing.

When you choose **Geolocation rule**, you also choose the country or the state in the United States that requests originate from.

> **Note**
>
> Evaluate target health is checked by default. It evaluates the health of the target endpoint to which traffic is routed through an alias record. If your endpoint doesn't receive DNS traffic through an alias record, clear this option and create a health check to monitor the endpoint health. For more information, see Creating and updating health checks.

Latency rule

Choose this option when you have resources in multiple Amazon EC2 data centers that perform the same function, and you want Route 53 to respond to DNS queries with the resources that provide the best latency.

When you choose **Latency rule**, you also choose an AWS Region.

For more information, see Latency-based routing.

> **Note**
>
> Evaluate target health is checked by default. It evaluates the health of the target endpoint to which traffic is routed through an alias record. If your endpoint doesn't receive DNS traffic through an alias record, clear this option and create a health check to monitor the endpoint health. For more information, see Creating and updating health checks.

Geoproximity rule

Choose this option when you want Route 53 to respond to DNS queries based on the location of your resources and, optionally, on a bias that you specify. The bias allows you to send more traffic to a resource or more traffic away from a resource.

When you choose **Geoproximity rule**, enter the following values:

Endpoint location

Choose the applicable value:

- **Custom (enter coordinates)** – If your endpoint is not an AWS resource, choose **Custom (enter coordinates)**.
- **An AWS Region** – If your endpoint is an AWS resource, choose the AWS Region that you created the resource in.
- **An AWS Local Zone** – If your endpoint is an AWS resource, choose the AWS Local Zone that you created the resource in.

If you use AWS Local Zones, you must first enable them. For more information, see Getting started with Local Zones in the *AWS Local Zones User Guide*.

For available Local Zones, see AWS Local Zones locations.

To learn about the difference between AWS Regions and Local Zones, see Regions and Zones in the *Amazon EC2 User Guide*.

> **Important**
>
> A single geoproximity routing policy cannot contain two or more locations that are geographically situated within the same metropolitan area.
> Additionally, some AWS Regions and Local Zones, such as US West (Oregon) and Portland, US, are situated too close to one another to be used within the same geoproximity routing policy. If you need to route traffic to more than one location within the same metropolitan area, define a geoproximity routing policy that creates a 50/50 weighted routing rule (WRR) for two endpoints in the area. This distributes traffic evenly between those endpoints.

Coordinates

If you chose **Custom (enter coordinates)** for **Endpoint location**, enter the latitude and longitude of the location of the resource. Note the following:

- Latitude represents the location south (negative) or north (positive) of the equator. Valid values are -90 degrees to 90 degrees.
- Longitude represents the location west (negative) or east (positive) of the prime meridian. Valid values are -180 degrees to 180 degrees.
- You can get latitude and longitude from some online mapping applications. For example, in Google Maps, the URL for a location specifies the latitude and longitude:

  https://www.google.com/maps/@**47.6086111**,**-122.3409953**,20z
- You can enter up to two decimals of precision, for example, **47.63**. If you specify a value with greater precision, Route 53 truncates the value to two places after the decimal. For latitude and for longitude at the equator, 0.01 degree is approximately 0.69 miles.

Bias

To optionally change the size of the geographic region from which Route 53 routes traffic to a resource, specify the applicable value for **Bias**:

- To expand the size of the geographic region from which Route 53 routes traffic to a resource, specify a positive integer from 1 to 99 for the bias. Route 53 shrinks the size of adjacent regions.
- To shrink the size of the geographic region from which Route 53 routes traffic to a resource, specify a negative bias of -1 to -99. Route 53 expands the size of adjacent regions.

> **Important**
>
> The effect of changing the value of **Bias** is relative, based on the location of other resources, rather than absolute, based on distance. As a result, the effect of a change is difficult to predict. For example, depending on where your resources are, changing the bias from 10 to 15 can mean the difference between adding or subtracting a significant amount of traffic from the New York City metropolitan area. We recommend that you change the bias in small increments and evaluate the results, and then make additional changes if appropriate.

For more information, see Geoproximity routing.

> **Note**
>
> Evaluate target health is checked by default. It evaluates the health of the target endpoint to which traffic is routed through an alias record. If your endpoint doesn't receive DNS traffic through an alias record, clear this option and create a health check to monitor the endpoint health. For more information, see Creating and updating health checks.

Multivalue answer rule

Choose this option when you want Route 53 to respond to DNS queries with up to eight healthy answers, selected approximately at random.

For more information, see Multivalue answer routing.

> **Note**
>
> Evaluate target health is checked by default. It evaluates the health of the target endpoint to which traffic is routed through an alias record. If your endpoint doesn't receive DNS traffic through an alias record, clear this option and create a health check to monitor the endpoint health. For more information, see Creating and updating health checks.

Weighted rule

Choose this option when you have multiple resources that perform the same function (for example, web servers that serve the same website) and you want Route 53 to route traffic to those resources in proportions that you specify (for example, 1/3 to one server and 2/3 to the other).

When you choose **Weighted rule**, enter the weight that you want to apply to this rule.

For more information, see Weighted routing.

> **Note**
>
> Evaluate target health is checked by default. It evaluates the health of the target endpoint to which traffic is routed through an alias record. If your endpoint doesn't receive DNS traffic through an alias record, clear this option and create a health check to monitor the endpoint health. For more information, see Creating and updating health checks.

Endpoint

Choose this option to specify the resource, such as a CloudFront distribution or an Elastic Load Balancing load balancer, that you want to route DNS queries to.

Existing rule

Choose this option when you want to route DNS queries to an existing rule in this traffic policy. For example, you might create two or more geolocation rules that route queries for different countries to the same failover rule. The failover rule might then routes queries to two Elastic Load Balancing load balancers.

This option isn't available if the traffic policy doesn't include any rules.

Existing endpoint

Choose this option when you want to route DNS queries to an existing endpoint. For example, if you have two failover rules, you might want to route DNS queries for both **On failover** (secondary) options to the same Elastic Load Balancing load balancer.

This option isn't available if the traffic policy doesn't include any endpoints.

Value type

Choose the applicable option:

CloudFront distribution

Choose this option if you want to route traffic to a CloudFront distribution. The option is available only if you chose **A: IP address in IPv4 format** for **DNS type** or **AAAA: IP address in IPv6 format** for **DNS type**.

> **Note**
>
> CloudFront alias endpoints don't support evaluate health check. You can however, create a health check to monitor CloudFront endpoints. For more information, see Creating and updating health checks.

ELB Application load balancer

Choose this option if you want to route traffic to an Elastic Load Balancing Application load balancer. The option is available only if you chose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format** for **DNS type**.

ELB Classic load balancer

Choose this option if you want to route traffic to an Elastic Load Balancing Classic load balancer. The option is available only if you chose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format** for **DNS type**.

ELB Network load balancer

Choose this option if you want to route traffic to an Elastic Load Balancing Network load balancer. The option is available only if you chose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format** for **DNS type**.

Elastic Beanstalk environment

Choose this option if you want to route traffic to an Elastic Beanstalk environment. The option is available only if you chose **A: IP address in IPv4 format** for **DNS type**.

S3 website endpoint

Choose this option if you want to route traffic to an Amazon S3 bucket that is configured as a website endpoint. The option is available only if you chose **A: IP address in IPv4 format** for **DNS type**.

Type *DNS type* value

Choose this option if you want Route 53 to respond to DNS queries using the value in the **Value** field. For example, if you chose **A** for the value of **DNS type** when you created this traffic policy, this option in the **Value type** list will be **Type A value**. This requires that you enter an IP address in IPv4 format in the **Value** field. Route 53 will respond to DNS queries that are routed to this endpoint with the IP address in the **Value** field.

Value

Choose or enter a value based on the option that you chose for **Value type**:

CloudFront distribution

Choose a CloudFront distribution from the list of distributions that are associated with the current AWS account.

ELB Application load balancer

Choose an Elastic Load Balancing Application load balancer from the list of load balancers that are associated with the current AWS account.

ELB Classic load balancer

Choose an Elastic Load Balancing Classic load balancer from the list of load balancers that are associated with the current AWS account.

ELB Network load balancer

Choose an Elastic Load Balancing Network load balancer from the list of load balancers that are associated with the current AWS account.

Elastic Beanstalk environment

Choose an Elastic Beanstalk environment from the list of environments that are associated with the current AWS account.

S3 website endpoint

Choose an Amazon S3 bucket from the list of Amazon S3 buckets that are configured as website endpoints and that are associated with the current AWS account.

> **Important**
>
> When you create a policy record based on this traffic policy, the bucket that you choose here must match the domain name (such as www.example.com) that you specify for Policy record DNS name in the policy record. If **Value** and **Policy record DNS name** don't match, Amazon S3 won't respond to DNS queries for the domain name.

Type *DNS type* value

Enter a value that corresponds with the value that you specified for **DNS type** when you started this traffic policy. For example, if you chose **MX** for **DNS type**, enter two values: the priority that you want to assign to a mail server and the domain name of the mail server, such as `10 sydney.mail.example.com`.

For more information about supported DNS types, see Supported DNS record types.

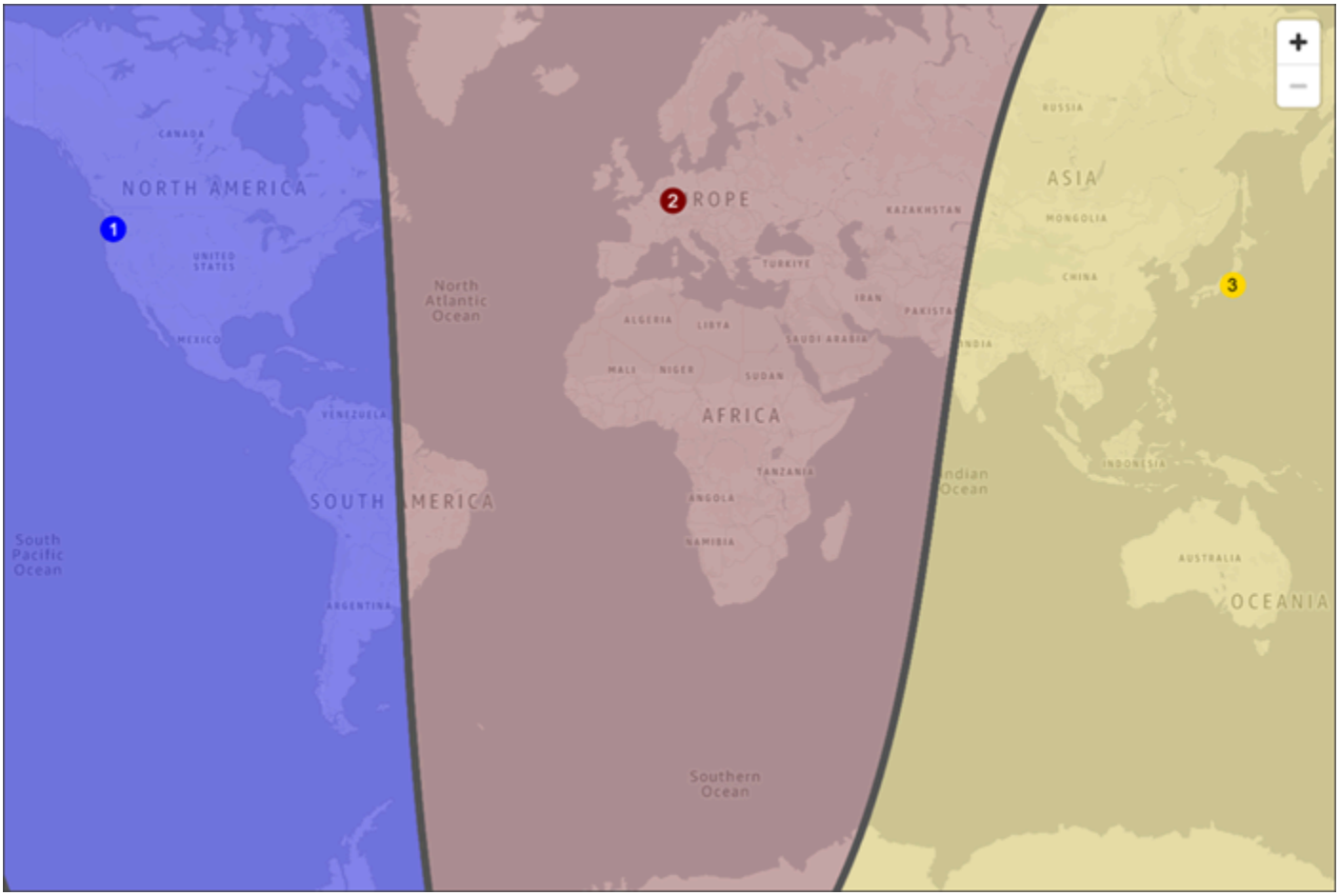
Key

Enter a friendly name for a routing rule or an endpoint for **Key**. This value displays as the name of a node in the editor.

Viewing a map that shows the effect of geoproximity settings

A *geoproximity rule* lets you specify the locations of your resources, both in AWS Regions , or Local Zones, and, using latitude and longitude, in non-AWS locations. When you create a geoproximity rule, by default, Route 53 routes internet traffic to the resource that is closest to your users. You can also choose to route more traffic or less traffic to a resource by specifying a *bias*, which expands or shrinks the geographic area from which traffic is routed to a resource. For more information about geoproximity routing, see Geoproximity routing.

You can display a map that shows the effect of your current geoproximity settings. For example, if you have resources in the US West (Oregon), Europe (Frankfurt), and Asia Pacific (Tokyo) Regions, and if you don't specify a bias, the map looks like this.



The map for a geoproximity rule automatically displays in the **Properties** and updates as you add or delete regions.

Note the following:

- The map is accurate to within approximately 10 miles (16 kilometers).
- The map automatically adjusts when you add, edit, or delete Regions, or when you change the bias setting for a Region.
- The Region number and color in each rule definition correspond with numbers and colors on the map.
- You can zoom in and zoom out to see more or less detail. Use the + and - buttons on the map, a touchpad, or the wheel on a mouse to change the zoom level. If you don't see the Region, or the Region number, you can enlarge the **Properties** pane and they will appear.
- You can move the map around inside the map window to see specific areas. Use a touchpad or drag the map to navigate. You can also move the map window in a browser window.

- If you have more than one geoproximity rule in a policy, you can view the map for only one rule at a time.

Creating additional versions of a traffic policy

When you edit a traffic policy, Amazon Route 53 automatically creates another version of the traffic policy and retains the previous versions unless you choose to delete them. The new version has the same name as the traffic policy that you're editing; it's distinguished from the original version by a version number that Route 53 increments automatically. You can base the new version of a traffic policy on any existing version of a traffic policy that has the same name.

Route 53 doesn't reuse version numbers for new versions of a given traffic policy. For example, if you create three versions of **MyTrafficPolicy**, delete the last two versions, and then create another version, the new version is version 4. By retaining the previous versions, Route 53 ensures that you can roll back to a previous configuration if a new configuration doesn't route traffic as you wanted it to.

To create a new traffic policy version, perform the following procedure.

To create another version of a traffic policy

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Traffic policies**.
3. Choose the name of the traffic policy that you want to create a new version of.
4. In the **Traffic policy versions** table at the top of the page, select the check box for the traffic policy version that you want to use as a basis for the new traffic policy version.
5. Choose **Edit policy as new version**.
6. On the **Edit policy as a new version** dialog box, enter a description for the new traffic policy version. Specify a description that distinguishes this version from other versions of the same traffic policy. When you create a new policy record, the value that you specify appears in the list of available versions for this traffic policy.
7. Choose **Next**.
8. Update the configuration as applicable. For more information, see Values that you specify when you create a traffic policy.

   You can delete rules, endpoints, and branches of a traffic policy in the following ways:

- To delete a rule or an endpoint, choose it in the editor and then choose **Delete** in the **Properties** pane.

  > **Important**
  >
  > If you delete a rule that has child rules and endpoints, Route 53 also deletes all of the children.

- If you connect two rules to the same child rule or endpoint and you want to delete one of the connections, select the connection that you want to delete, and then choose **Delete** in the **Properties** pane.

9. When you're finished editing, choose **Save as new version**.
10. *Optional*: Specify the settings to create one or more policy records in one hosted zone by using the new traffic policy version. For more information, see Values that you specify when you create or update a policy record. You can also create policy records later, either in the same hosted zone or in additional hosted zones.

    If you don't want to create policy records now, choose **Cancel**, and the console displays the list of traffic policies and policy records that you have created by using the current AWS account.
11. If you specified settings for policy records in the preceding step, choose **Create policy record**.

Creating a traffic policy by using a JSON document

You can create a new traffic policy or a new version of an existing traffic policy by importing a document in JSON format that describes all of the endpoints and rules that you want to include in the traffic policy. For information about the format of the JSON document and several examples that you can copy and revise, see Traffic policy document format in the *Amazon Route 53 API Reference*.

The easiest way to get the JSON-formatted document for an existing traffic policy version is to use the `get-traffic-policy` command in the AWS CLI. For more information, see get-traffic-policy in the *AWS CLI Command Reference*.

The JSON file created by the `get-traffic-policy` command includes backward slashes (\) as escape characters. Before you import the JSON file, replace all the backward slashes with null characters.

To create a traffic policy by importing a JSON document

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. To create a new traffic policy by importing a JSON document, perform the following steps:
   a. In the navigation pane, choose **Traffic policies**.
   b. Choose **Create traffic policy**.
   c. On the **Name policy** page, specify the applicable values. For more information, see Values that you specify when you create a traffic policy.
   d. Skip to step 4.
3. To create a new version of an existing traffic policy by importing a JSON document, perform the following steps:
   a. In the navigation pane, choose **Traffic policies**.
   b. Choose the name of the traffic policy that you want to base the new version on.
   c. In the **Traffic policy versions** table, select the check box for the version that you want to base the new version on.
   d. Choose **Edit policy as new version**.
   e. On the **Update description** page, enter a description for the new version.
   f. Skip to step 4.
4. Choose **Next**.
5. On the editor page choose the json editor
   ( )
   icon on the upper right corner of the editor.
6. Enter a new traffic policy, paste an example traffic policy, or paste an existing traffic policy.

   For example policies, see Traffic Flow policy JSON examples.
7. Choose **Update**.

Viewing traffic policy versions and the associated policy records

You can view all of the versions that you've created for a traffic policy as well as all of the policy records that you've created by using each of the versions of the traffic policy.

To view traffic policy versions and the associated policy records

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Traffic policies**.
3. Choose the linked name of a traffic policy.
4. The top table lists all of the versions that you've created of a traffic policy. The table includes the following information:

   Version number

   The number of each version of a traffic policy that you've created. If you choose the version number, the console displays the configuration for that version.

   Number of policy records

   The number of policy records that you've created by using this traffic policy version.

   DNS type

   The DNS type that you specified when you created the traffic policy version.

   Version descriptions

   The description that you specified when you created the traffic policy version.
5. The bottom table lists all of the policy records that you've created by using the traffic policy versions in the top table. The table includes the following information:

   DNS name

   The DNS names that you've associated the traffic policy with.

   Status

   Possible values include the following:

   Applied

   Route 53 has finished creating or updating a policy record and the corresponding records.

   Creating

   Route 53 is creating the records for a new policy record.

Updating

You have updated a policy record and Route 53 is in the process of creating a new group of records that will replace the existing group of records for the specified DNS name.

Deleting

Route 53 is in the process of deleting a policy record and the associated records.

Failed

Route 53 wasn't able to create or update the policy record and the associated records.

DNS type

The DNS type of all of the records that Route 53 created for this policy record. When you edit a policy record, you must specify a traffic policy version that has the same DNS type as the DNS type for the policy record that you're editing.

Version

Indicates the version of the traffic policy that you used to create the policy record.

TTL (in seconds)

The amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172,800 seconds, or two days), you pay less for Route 53 service because recursive resolvers send requests to Route 53 less often. However, it takes longer for changes to the records (for example, a new IP address) to take effect because recursive resolvers use the values in their cache for longer periods instead of asking Route 53 for the latest information.

Deleting traffic policy versions and traffic policies

To delete a traffic policy, you must delete all of the versions (including the original) that you've created for the traffic policy. In addition, to delete a traffic policy version, you must delete all of the policy records that you created by using the traffic policy version.

Important

If you delete policy records that Amazon Route 53 is using to respond to DNS queries, Route 53 stops responding to queries for the corresponding DNS names. For example, if Route 53 is using the policy record for www.example.com to respond to DNS queries for

www.example.com and you delete the policy record, users can't access your website or web application by using the domain name www.example.com.

To delete traffic policy versions and, optionally, a traffic policy, perform the following procedure:

To delete traffic policy versions and a traffic policy

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Traffic policies**.
3. Choose the linked name of the traffic policy for which you want to delete traffic policy versions and that, optionally, you want to delete completely.
4. If the traffic policy versions that you want to delete in the top table appear in the **Version** column in the **Policy records** table, select the check boxes for the corresponding policy records in the bottom table.

   For example, if you want to delete version 3 of a traffic policy but you created one of the policy records in the bottom table by using version 3, select the check box for that policy record.
5. Choose **Delete policy records** and in the **Delete policy record** dialog box, choose **Confirm**.
6. Refresh the page until the policy records that you deleted no longer appear in the table.
7. In the top table, select the check boxes for the traffic policy versions that you want to delete.
8. Choose **Delete**.
9. If you deleted all traffic policy versions in the preceding step and you want to delete the traffic policy, too, refresh the page to refresh the display until the table is empty.
10. In the navigation pane, choose **Traffic policies**.
11. In the list of traffic policies, select the check box for the traffic policy that you want to delete.
12. Choose **Delete traffic policy**.

Creating and managing policy records

To route internet traffic to the resources that you specified when you created a traffic policy, you create one or more policy records. Each policy record identifies the hosted zone where you want to create the policy record and the domain or subdomain name that you want to route traffic for. For

example, if you want to route traffic for www.example.com, you specify the hosted zone ID for the example.com hosted zone, and you specify www.example.com for the **Policy record DNS name**.

If you want to use the same traffic policy to route traffic for more than one domain or subdomain name, you have two options:

- You can create a policy record for each domain or subdomain name.
- You can create one policy record and then create CNAME or alias records that refer to the policy record.

For example, if you want to use the same traffic policy for example.com, example.net, and example.org, you can do either of the following:

- Create one policy record for each of them.
- Create a policy record for one of them and then create CNAME records in the hosted zones for the other two. In the two CNAME records, you specify the record name that you created a policy record for.

If you want to use the same traffic policy for a domain and its subdomains, such as example.com and www.example.com, you can create a policy record for one name and then create alias records for the rest. For example, you can create a policy record for example.com and then create an alias record for www.example.com that has the example.com record as the alias target.

> **Note**
>
> There's a monthly charge for each policy record that you create. If you want to use the same traffic policy for multiple domain or subdomain names, you can use CNAME or alias records to reduce your charges:
>
> - If you create one policy record and one or more CNAME records that refer to the policy record, you pay only for the policy record and for DNS queries for the CNAME records.
> - If you create one policy record and one or more alias records in the same hosted zone that refer to the policy record, you pay only for the policy record and for DNS queries for the alias records.

Topics

- Creating policy records
- Values that you specify when you create or update a policy record
- Updating policy records
- Deleting policy records

Creating policy records

To create a policy record, perform the following procedure.

> **Important**
>
> For each policy record that you create, you incur a monthly charge. If you later delete the policy record, the charge is prorated. For more information, see the section "Traffic Flow" on the Amazon Route 53 Pricing page.

To create a policy record

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Policy records**.
3. On the **Policy records** page, choose **Create policy records**.
4. On the **Create policy records** page, specify the applicable values. For more information, see Values that you specify when you create or update a policy record.
5. To add additional records to the same hosted zone, Choose **Add policy record**.
6. Choose **Create policy records**.

   It can take several minutes for the status of created policy record to display as **Applied**.
7. If you want to create policy records in another hosted zone, repeat steps 3 through 5.

> **Note**
>
> If the policy record status is **Failed**, choose the **info** button next to the status to get more information about the failure. If you need further help and want to contact AWS support, see How do I get technical support from AWS?

Values that you specify when you create or update a policy record

When you create or update a policy record, you specify the following values

- Traffic policy
- Version
- Hosted zone
- Policy record DNS name
- TTL

Traffic policy

Choose the traffic policy whose configuration you want to use for this policy record.

Version

Choose the version of the traffic policy whose configuration you want to use for this policy record.

If you're updating an existing policy record, you must choose a version for which the DNS type matches the current DNS type of the policy record. For example, if the DNS type of the policy record is **A**, you must choose a version for which the DNS type is **A**.

Hosted zone

Choose the hosted zone in which you want to create a policy record by using the specified traffic policy and version. You can't change the value of **Hosted zone** after you create a policy record.

DNS name

When you're creating a policy record, enter the domain name or subdomain name for which you want Route 53 to respond to DNS queries by using the configuration in the specified traffic policy and version.

To use the same configuration for more than one domain name or subdomain name in the specified hosted zone, choose **Add policy record**, and enter the applicable domain name or subdomain name and TTL.

You can't change the value of **Policy record DNS name** after you create a policy record.

TTL (in seconds)

Enter the amount of time, in seconds, that you want DNS recursive resolvers to cache information about this record. If you specify a longer value (for example, 172800 seconds, or two days), you pay less for Route 53 service because recursive resolvers send requests to Route 53 less often. However, it takes longer for changes to the records (for example, a new IP address) to take effect because recursive resolvers use the values in their cache for longer periods instead of asking Route 53 for the latest information.

Updating policy records

To update the settings in a policy record, perform the following procedure.

To update a policy record

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Policy records**.
3. On the **Policy records** page, select the check box for the policy record that you want to update, and choose **Edit**.
4. On the **Edit policy record** page, specify the applicable values. For more information, see Values that you specify when you create or update a policy record.
5. Choose **Edit policy record**.

   It can take several minutes for the status of created policy record to display as **Applied**.
6. If you want to update another policy record, repeat steps 3 through 5.

> **Note**
>
> If the policy record status is **Failed**, choose the **info** button next to the status to get more information about the failure. If you need further help and want to contact AWS support, see How do I get technical support from AWS?

Deleting policy records

To delete policy records, perform the following procedure.

> **Important**
>
> If you delete policy records that Amazon Route 53 is using to respond to DNS queries, Route 53 stops responding to queries for the corresponding DNS names. For example, if Route 53 is using the policy record for www.example.com to respond to DNS queries for www.example.com and you delete the policy record, users can't access your website or web application by using the domain name www.example.com.

To delete a policy record

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Policy records**.
3. On the **Policy records** page, select the check boxes for the policy records that you want to delete, and choose **Delete**.
4. On the **Delete policy record** dialog box, choose **Confirm**.

   Wait several minutes and refresh the page to make sure the policy record disappears from the list.

Traffic Policy Document Format

When you create a traffic policy programmatically by using the Amazon Route 53 API, one of the AWS SDKs, the AWS CLI, or AWS Tools for Windows PowerShell, you specify the definition of the traffic policy in a `Document` element in JSON format.

For more information about traffic policies, see Using Traffic Flow to Route DNS Traffic in the *Amazon Route 53 Developer Guide*.

Topics

- Basic Syntax
- Syntax for Endpoint Definitions
- Syntax for Rule Definitions
- Examples

Basic Syntax

Here is the basic syntax for a traffic policy document:

```
{
   "AWSPolicyFormatVersion": "2015-10-01 | 2023-05-09",
   "RecordType": "DNS type for all resource record sets created by this traffic
 policy",
   "StartEndpoint | StartRule": "ID that you assign to an endpoint or rule",
   "Endpoints": {
      "Endpoint ID that you assign": {
         Endpoint definition
      },
      ...
   },
   "Rules": {
      "Rule ID that you assign": {
         Rule definition
      },
      ...
   }
}
```

The basic syntax for a traffic policy document contains the following objects:

AWSPolicyFormatVersion

The version of the traffic policy format, the latest version that supports AWS Local Zones; `2023-05-09`.

Previous version is also supported; `2015-10-01`.

RecordType

The DNS type of all of the resource record sets that Amazon Route 53 will create based on this traffic policy. If you want to route traffic to the following AWS resources, choose the applicable value:

- **CloudFront distribution** – Choose **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.
- **ELB Application load balancer** – Choose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.

- **ELB Classic load balancer** – Choose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.
- **ELB Network load balancer** – Choose either **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.
- **Elastic Beanstalk environment** – Choose **A: IP address in IPv4 format** or **AAAA: IP address in IPv6 format**.
- **Amazon S3 bucket configured as a website endpoint**: Choose **A: IP address in IPv4 format**.

If you want to route traffic to other resources, choose the applicable type for the resource. For example, if you want to route traffic to mail servers, specify `MX`. For the list of DNS types that Route 53 supports, see Supported DNS Resource Record Types in the *Amazon Route 53 Developer Guide*.

StartEndpoint | StartRule

Whether you want the starting point for the traffic policy to be an endpoint or a rule, and the ID that you assigned to the endpoint or rule elsewhere in the traffic policy document.

Endpoints

The definitions of the endpoints that you want to use in this traffic policy. For more information, see Syntax for Endpoint Definitions.

Rules

The definitions of the rules that you want to use in this traffic policy. For more information, see Syntax for Rule Definitions.

Syntax for Endpoint Definitions

Here is the syntax for the endpoint definitions that you specify in a traffic policy document:

```
{
   "Type": value | cloudfront | application-load-balancer | elastic-load-balancer |
 network-load-balancer | elastic-beanstalk | s3-website,
   "Region": "AWS region that you created your Amazon S3 bucket in"
   "Value": "value applicable to the type of endpoint"
}
```

The syntax for an endpoint definition contains the following objects:

Type

Specify the applicable value:

value

To route traffic to a resource other than the ones in the following list, specify `value` for `Type`.

cloudfront

To route traffic to a CloudFront distribution, specify `cloudfront` for `Type`.

application-load-balancer

To route traffic to an Application Load Balancer, specify `application-load-balancer` for `Type`.

elastic-load-balancer

To route traffic to an ELB Classic load balancer, specify `elastic-load-balancer` for `Type`.

network-load-balancer

To route traffic to an ELB Network load balancer, specify `network-load-balancer` for `Type`.

elastic-beanstalk

To route traffic to an Elastic Beanstalk environment, specify `elastic-beanstalk` for `Type`.

s3-website

To route traffic to an Amazon S3 bucket that is configured as a website endpoint, specify `s3-website` for `Type`.

Region

To route traffic to an Amazon S3 bucket that is configured as a website endpoint, specify the Region in which you created the bucket for `Region`. For any other resource, omit `Region`.

Value

Specify the applicable value:

value

To route traffic to a resource other than the ones listed below, specify the value that corresponds with the value that you specified for `RecordType`. For example, if you specified `A` for `RecordType`, specify an IP address in IPv4 format for `Value`.

cloudfront

If you specified `cloudfront` for `Type`, specify the domain name that CloudFront assigned to your CloudFront distribution when you created it, for example, `d111111abcdef8.cloudfront.net`.

application-load-balancer

If you specified `application-load-balancer` for `Type`, specify the DNS name for your load balancer. For example, `exampleAlb-1283637735.us-east-1.elb.amazonaws.com`.

elastic-load-balancer

If you specified `elastic-load-balancer` for `Type`, specify the DNS name for your load balancer. Use the value that begins with `dualstack`, for example, `dualstack.my-load-balancer-1234567890.us-west-2.elb.amazonaws.com`.

network-load-balancer

If you specified `network-load-balancer` for `Type`, specify the DNS name for your load balancer. For example, `exampleNlb-74f87b114e21682a.elb.us-east-1.amazonaws.com`.

elastic-beanstalk

If you specified `elastic-beanstalk` for `Type`, specify the domain name for your Elastic Beanstalk environment. For example, `example-beanstalk-env-1.eba-ns4qrife.us-east-1.elasticbeanstalk.com`.

s3-website

If you specified `s3-website` for `Type`, specify the name of your Amazon S3 bucket, for example, `example.com.s3-website-us-east-1.amazonaws.com`.

> **Important**
>
> When you create a traffic policy instance based on this traffic policy, the bucket that you specify here must match the domain name (such as www.example.com) that you

specify for `Name` in the CreateTrafficPolicyInstance request. If `Value` and `Name` don't match, Amazon S3 won't respond to DNS queries for the domain name.

Syntax for Rule Definitions

There are different syntaxes for the rule definitions that you specify in a traffic policy document, depending on the type of routing policy that you want to use: failover, geolocation, geoproximity, latency, multivalue answer, or weighted.

Topics

- Failover Rules
- Geolocation Rules
- Geoproximity Rules
- Latency Rules
- Multivalue Answer Rules
- Weighted Rules

Failover Rules

For more information, see Configuring DNS Failover in the *Amazon Route 53 Developer Guide*.

```
{
   "RuleType": "failover",
   "Primary": {
      "EndpointReference | RuleReference": "ID that you assigned to the rule or
 endpoint that this rule routes traffic to",
      "EvaluateTargetHealth": "true" | "false",
      "HealthCheck": "optional health check ID"
   },
   "Secondary": {
      "EndpointReference | RuleReference": "ID that you assigned to the rule or
 endpoint that this rule routes traffic to",
      "EvaluateTargetHealth": "true" | "false",
      "HealthCheck": "optional health check ID"
   }
}
```

When you define a failover rule, you specify the following objects:

RuleType

Specify `failover`.

Primary | Secondary

For the `Primary` object, specify settings for the rule or endpoint that you want to route traffic to whenever the corresponding resources are available.

For the `Secondary` object, specify settings for the rule or endpoint that you want to route traffic to whenever the primary resources are not available.

EndpointReference | RuleReference

Whether you want to route traffic to an endpoint or to another rule, and the ID that you assigned to the endpoint or rule elsewhere in the traffic policy document.

EvaluateTargetHealth

A Boolean that indicates whether you want Amazon Route 53 to evaluate the health of the endpoint and route traffic only to healthy endpoints. For more information, see EvaluateTargetHealth in the documentation about ChangeResourceRecordSets.

HealthCheck

If you want to associate a health check with the endpoint or rule, specify the ID of the health check. For more information, see HealthCheckId in the documentation about ChangeResourceRecordSets.

Geolocation Rules

When you add a geolocation rule, you configure your traffic policy to route your traffic based on the geographic location of your users. For more information, see Geolocation Routing in the *Amazon Route 53 Developer Guide*.

```
{
   "RuleType": "geo",
   "Locations": [
      {
         "EndpointReference | RuleReference": "ID that you assigned to the rule or
 endpoint that this rule routes traffic to",
```

```
         "IsDefault": "true" | "false",
         "Continent": "continent name,
         "Country": "country name,
         "Subdivision": "subdivision name,
         "EvaluateTargetHealth": "true" | "false",
         "HealthCheck": "optional health check ID"
      },
      ...
   ]
}
```

When you define a geolocation rule, you specify the following objects:

RuleType

Specify geo.

Locations

Specify one set of values (EndpointReference | RuleReference, IsDefault, Continent, Country, Subdivision, EvaluateTargetHealth, and HealthCheck) for each of the geographic locations that you want to route traffic to.

EndpointReference | RuleReference

Whether you want to route traffic to an endpoint or to another rule, and the ID that you assigned to the endpoint or rule elsewhere in the traffic policy document.

IsDefault

A Boolean that indicates whether this set of values represents the default location. If you specify * as the value of the CountryCode element in a geographic location, the value of IsDefault element for this location is set to True by default and you can't specify True as the value for the IsDefault element in other geographic locations for the same geolocation rule. For more information about specifying * as the value for the CountryCode element, see the following entry for Continent, Country, Subdivision.

Continent, Country, Subdivision

Values that indicate the geographic location of users whose traffic you want to route to a rule or endpoint. For more information, see the following element descriptions in the documentation about ChangeResourceRecordSets:

- GeoLocation

- `ContinentCode`
- `CountryCode`
- `SubdivisionCode`

For more information, see Location in the *Amazon Route 53* Developer Guide.

> **Important**
>
> Geolocation works by mapping IP addresses to locations. However, some IP addresses aren't mapped to geographic locations, so even if you create geographic locations that cover all seven continents, Route 53 will receive some DNS queries from locations that it can't identify. We recommend that you create a geographic location for which the value of `Country` is `*`. Two groups of queries are routed to the resource that you specify in this record: queries that come from locations for which you haven't created geographic locations and queries from IP addresses that aren't mapped to a location. If you don't create a `*` geographic location, Route 53 returns a "no answer" response for queries from un-mapped locations.

EvaluateTargetHealth

A Boolean that indicates whether you want Amazon Route 53 to evaluate the health of the endpoint and route traffic only to healthy endpoints. For more information, see EvaluateTargetHealth in the documentation about ChangeResourceRecordSets.

HealthCheck

If you want to associate a health check with the endpoint or rule, specify the ID of the health check. For more information, see HealthCheckId in the documentation about ChangeResourceRecordSets.

Geoproximity Rules

When you add a geoproximity rule, you configure Amazon Route 53 to route traffic to your resources based on the geographic location of your resources. You can also optionally choose to route more traffic or less to a given endpoint or rule by specifying a bias. You use a bias to expand or shrink the size of the geographic region from which traffic is routed to an endpoint or rule. For more information, see Geoproximity Routing in the *Amazon Route 53 Developer Guide*.

The following example is for `AWSPolicyFormatVersion 2023-05-09` only:

```
{
   "RuleType": "geoproximity",
   "GeoproximityLocations": [
      {
         "EndpointReference | RuleReference": "ID that you assigned to the endpoint or
 rule that this rule routes traffic to",
         "Location":{
            "Type": "Region | LocalZone",
            "LocationName": "AWS Region | AWS Local Zone"
         },
         "Bias": "optional value to expand or shrink the geographic region for this
 rule, -99 to 99",
         "EvaluateTargetHealth": "true | false",
         "HealthCheck": "optional health check ID"
      },
      {
         "EndpointReference | RuleReference": "ID that you assigned to the endpoint or
 rule that this rule routes traffic to",
         "Location":{
            "Type": "Coordinate",
            "Latitude": "location south (negative) or north (positive) of the equator,
 -90 to 90 degrees",
            "Longitude": "location west (negative) or east (positive) of the prime
 meridian, -180 to 180 degrees"
         },
         "Bias": "optional value to expand or shrink the geographic region for this
 rule, -99 to 99",
         "EvaluateTargetHealth": "true | false",
         "HealthCheck": "optional health check ID"
      }
   ]
}
```

The following example is for `AWSPolicyFormatVersion 2015-10-01` only:

```
{
   "RuleType": "geoproximity",
   "GeoproximityLocations": [
      {
         "EndpointReference | RuleReference": "ID that you assigned to the endpoint or
 rule that this rule routes traffic to",
```

```
         "Region": "AWS Region",
         "Bias": "optional value to expand or shrink the geographic region for this
 rule, -99 to 99",
         "EvaluateTargetHealth": "true | false",
         "HealthCheck": "optional health check ID"
      },
      {
         "EndpointReference | RuleReference": "ID that you assigned to the endpoint or
 rule that this rule routes traffic to",
         "Latitude": "location south (negative) or north (positive) of the equator, -90
 to 90 degrees",
         "Longitude": "location west (negative) or east (positive) of the prime
 meridian, -180 to 180 degrees",
         "Bias": "optional value to expand or shrink the geographic region for this
 rule, -99 to 99",
         "EvaluateTargetHealth": "true | false",
         "HealthCheck": "optional health check ID"
      }
   ]
}
```

When you define a geoproximity rule, you specify the following objects:

RuleType

Specify `geoproximity`.

GeoproximityLocations

Specify one set of values for each resource that you want to route traffic to. You can specify a **Location** field, which supports AWS Regions, Local Zones, and latitudes and longitudes in `AWSPolicyFormatVersion 2023-05-09`. You can also specify either an AWS Region or the latitude and longitude of a geographic location in `AWSPolicyFormatVersion 2015-10-01`.

EndpointReference | RuleReference

Whether you want to route traffic to an endpoint or to another rule, and the ID that you assigned to the endpoint or rule elsewhere in the traffic policy document.

Location (`AWSPolicyFormatVersion 2023-05-09` only)

The Location field supports AWS Regions, Local Zones, and latitudes and longitudes with the following format:

For Local Zones: `"Location": { "Type": "LocalZone", "LocationName": "aws:route53:us-east-1-atl-1a" }`

For Regions: `"Location": { "Type": "Region", "LocationName": "aws:route53:us-east-1" },`

For coordinates: `"Location": { "Type": "Coordinate", "Latitude": "38.9", "Longitude": "-77.01" },`

For a list of valid region codes, see Region in the documentation about ChangeResourceRecordSets.

For available Local Zones, see AWS Local Zones locations.

Region (`AWSPolicyFormatVersion 2015-10-01` only)

If your endpoint is an AWS resource, specify the AWS Region that you created the resource in. Use the following format:

`aws:route53:`*`region-code`*

For a list of valid region codes, see Region in the documentation about ChangeResourceRecordSets.

Latitude and Longitude (`AWSPolicyFormatVersion 2015-10-01` only)

If your endpoint is not an AWS resource, enter the latitude and longitude of the location of the resource. Note the following:

- Latitude represents the location south (negative) or north (positive) of the equator. Valid values are -90 degrees to 90 degrees.
- Longitude represents the location west (negative) or east (positive) of the prime meridian. Valid values are -180 degrees to 180 degrees.
- You can get latitude and longitude from some online mapping applications. For example, in Google Maps, the URL for a location specifies the latitude and longitude:

  https://www.google.com/maps/@**47.6086111**,**-122.3409953**,20z
- You can enter up to two decimals of precision, for example, `47.61`. If you specify a value with greater precision, Route 53 returns an error. For latitude and for longitude at the equator, 0.01 degree is approximately 0.69 miles.

Bias

Specify a value for `Bias` if you want to route more traffic to an endpoint from nearby endpoints (positive values) or route less traffic to an endpoint (negative values). The range of valid values is -99 to 99; the default value is 0.

> **Important**
>
> The value of `Bias` is relative, based on the location of other resources, rather than absolute, based on distance. As a result, the effect of a change is difficult to predict. For example, depending on where your resources are, changing the bias from `10` to `15` can mean the difference between adding or subtracting a significant amount of traffic from the New York City metropolitan area. We recommend that you change the bias in small increments and evaluate the results, and then make additional changes if appropriate.

EvaluateTargetHealth

A Boolean that indicates whether you want Route 53 to evaluate the health of the endpoint and route traffic only to healthy endpoints. For more information, see EvaluateTargetHealth in the documentation about ChangeResourceRecordSets.

HealthCheck

If you want to associate a health check with the endpoint or rule, specify the ID of the health check. For more information, see HealthCheckId in the documentation about ChangeResourceRecordSets.

Latency Rules

When you add a latency rule, you configure your traffic policy to route your traffic based on the latency (the time delay) between your users and the AWS regions where you've created AWS resources such as ELB load balancers and Amazon S3 buckets. For more information, see Latency Routing in the *Amazon Route 53 Developer Guide*.

```
{
   "RuleType": "latency",
   "Regions": [
      {
```

```
        "EndpointReference | RuleReference": "ID that you assigned to the rule or
 endpoint that this rule routes traffic to",
        "Region": "AWS region that you want to route traffic to",
        "EvaluateTargetHealth": "true" | "false",
        "HealthCheck": "optional health check ID"
      },
      ...
   ]
}
```

When you define a latency rule, you specify the following objects:

RuleType

Specify `latency`.

Regions

Specify one set of values (`EndpointReference` | `RuleReference`, `Region`, `EvaluateTargetHealth`, and `HealthCheck`) for each of the Regions that you want to route traffic to.

EndpointReference | RuleReference

Whether you want to route traffic to an endpoint or to another rule, and the ID that you assigned to the endpoint or rule elsewhere in the traffic policy document.

Region

The region code for the AWS Region that you created the resource in. For a list of valid region codes, see Region in the documentation about ChangeResourceRecordSets.

EvaluateTargetHealth

A Boolean that indicates whether you want Amazon Route 53 to evaluate the health of the endpoint and route traffic only to healthy endpoints. For more information, see EvaluateTargetHealth in the documentation about ChangeResourceRecordSets.

HealthCheck

If you want to associate a health check with the endpoint or rule, specify the ID of the health check. For more information, see HealthCheckId in the documentation about ChangeResourceRecordSets.

Multivalue Answer Rules

When you add a multivalue answer rule, you configure your traffic policy to route traffic approximately randomly to your healthy resources. Amazon Route 53 responds to DNS queries with up to eight healthy records; if you have eight or fewer healthy records, Route 53 responds to all DNS queries with all the healthy records. For more information, see MultiValueAnswer.

```
{
   "RuleType": "multivalue",
   "Items": [
      {
         "EndpointReference": "ID that you assigned to the endpoint that this rule
 routes traffic to",
         "HealthCheck": "optional health check ID"
      },
      ...
   ]
}
```

When you define a multivalue answer rule, you specify the following objects:

RuleType

Specify `multivalue`.

Items

Specify one set of values (`EndpointReference` and `HealthCheck`) for each of the multivalue answer rules or endpoints that you want to route traffic to.

EndpointReference

The ID that you assigned to the endpoint elsewhere in the traffic policy document.

HealthCheck

If you want to associate a health check with the endpoint, specify the ID of the health check. For more information, see HealthCheckId.

Weighted Rules

When you add a weighted rule, you configure your traffic policy to route traffic based on proportions that you specify. For example, you might specify weights of 4, 5, 5, and 6 (sum=20).

The result is that 4/20ths of your traffic, on average, is routed to the first endpoint or rule, 5/20ths is routed both to the second and third endpoints or rules, and 6/20ths is routed to the last endpoint or rule. For more information, see Weighted Routing in the *Amazon Route 53 Developer Guide*.

```
{
   "RuleType": "weighted",
   "Items": [
      {
         "EndpointReference | RuleReference": "ID that you assigned to the rule or
 endpoint that this rule routes traffic to",
         "Weight": "value between 0 and 255",
         "EvaluateTargetHealth": "true" | "false",
         "HealthCheck": "optional health check ID"
      },
      ...
   ]
}
```

When you define a weighted rule, you specify the following objects:

RuleType

Specify `weighted`.

Items

Specify one set of values (`EndpointReference` | `RuleReference`, `Weight`, `EvaluateTargetHealth`, and `HealthCheck`) for each of the weighted rules or endpoints that you want to route traffic to.

EndpointReference | RuleReference

Whether you want to route traffic to an endpoint or to another rule, and the ID that you assigned to the endpoint or rule elsewhere in the traffic policy document.

Weight

A value between 0 and 255 that determines the proportion of traffic that is routed to the corresponding endpoint or rule. For more information, see Weight in the documentation about ChangeResourceRecordSets.

EvaluateTargetHealth

A Boolean that indicates whether you want Amazon Route 53 to evaluate the health of the endpoint and route traffic only to healthy endpoints. For more information, see EvaluateTargetHealth in the documentation about ChangeResourceRecordSets.

HealthCheck

If you want to associate a health check with the endpoint or rule, specify the ID of the health check. For more information, see HealthCheckId in the documentation about ChangeResourceRecordSets.

Examples

The following examples show how to use failover, geolocation, geoproximity, latency, and weighted rules, and how to use multiple types of rules in the same traffic policy.

Topics

- Failover Example
- Geolocation Example
- Geoproximity Examples
- Latency Example
- Weighted Example
- Example with Failover, Latency, and Geolocation Rules

Failover Example

```
{
   "AWSPolicyFormatVersion":"2015-10-01",
   "RecordType":"A",
   "StartRule":"site_switch",
   "Endpoints":{
      "my_elb":{
         "Type":"elastic-load-balancer",
         "Value":"elb-111111.us-east-1.elb.amazonaws.com"
      },
      "site_down_banner":{
         "Type":"s3-website",
```

```
         "Region":"us-east-1",
         "Value":"www.example.com"
      }
   },
   "Rules":{
      "site_switch":{
         "RuleType":"failover",
         "Primary":{
            "EndpointReference":"my_elb"
         },
         "Secondary":{
            "EndpointReference":"site_down_banner"
         }
      }
   }
}
```

Geolocation Example

```
{
  "AWSPolicyFormatVersion":"2015-10-01",
  "RecordType":"A",
  "StartRule":"geo_dest",
  "Endpoints":{
    "english":{
      "Type":"value",
      "Value":"192.0.2.1"
    },
    "french":{
      "Type":"value",
      "Value":"192.0.2.2"
    },
    "german":{
      "Type":"value",
      "Value":"192.0.2.3"
    }
  },
  "Rules":{
    "geo_dest":{
      "RuleType":"geo",
      "Locations":[
        {
          "EndpointReference":"english",
```

```
          "Country":"*",
          "HealthCheck":"11111111-1111-1111-1111-111111111111"
        },
        {
          "EndpointReference":"french",
          "Country":"FR",
          "IsDefault":false,
          "HealthCheck":"22222222-2222-2222-2222-222222222222"
        },
        {
          "EndpointReference":"french",
          "Country":"BE",
          "IsDefault":false,
          "HealthCheck":"22222222-2222-2222-2222-222222222222"
        },
        {
          "EndpointReference":"german",
          "Country":"DE",
          "IsDefault":false,
          "HealthCheck":"33333333-3333-3333-3333-333333333333"
        }
      ]
    }
  }
}
```

Geoproximity Examples

```
{
  "AWSPolicyFormatVersion":"2015-10-01",
  "RecordType":"A",
  "StartRule":"geoprox-rule",
  "Endpoints":{
    "aws-us-west-1-region":{
      "Type":"elastic-load-balancer",
      "Value":"elb-123456.us-east-1.elb.amazonaws.com"
    },
    "london-data-center":{
      "Type":"value",
      "Value":"192.0.2.1"
    }
  },
  "Rules":{
```

```
    "geoprox-rule":{
      "RuleType": "geoproximity",
      "GeoproximityLocations": [
        {
          "EndpointReference": "aws-us-west-1-region",
          "Region": "aws:route53:us-west-1",
          "Bias": "10",
          "HealthCheck": "11111111-1111-1111-1111-111111111111"
        },
        {
          "EndpointReference": "london-data-center",
          "Latitude": "51.50",
          "Longitude": "-0.16",
          "Bias": "0",
          "HealthCheck": "22222222-2222-2222-2222-222222222222"
        }
      ]
    }
  }
}
```

```
{
  "AWSPolicyFormatVersion":"2023-05-09",
  "RecordType":"A",
  "StartRule":"geoprox-rule",
  "Endpoints":{
    "aws-us-west-1-region":{
      "Type":"elastic-load-balancer",
      "Value":"elb-123456.us-west-1.elb.amazonaws.com"
    },
    "aws-us-east-1-atl-local-zone":{
      "Type":"elastic-load-balancer",
      "Value":"elb-123456.us-east-1.elb.amazonaws.com"
    },
    "london-data-center":{
      "Type":"value",
      "Value":"192.0.2.1"
    }
  },
  "Rules":{
    "geoprox-rule":{
      "RuleType": "geoproximity",
      "GeoproximityLocations": [
```

```
        {
          "EndpointReference": "aws-us-west-1-region",
          "Location": {
            "Type": "Region",
            "LocationName": "aws:route53:us-west-1"
          },
          "Bias": "-10",
          "HealthCheck": "11111111-1111-1111-1111-111111111111"
        },
        {
          "EndpointReference": "aws-us-east-1-atl-local-zone",
          "Location": {
            "Type": "LocalZone",
            "LocationName": "aws:route53:us-east-1-atl-1a"
          },
          "Bias": "10",
          "HealthCheck": "22222222-2222-2222-2222-222222222222"
        },
        {
          "EndpointReference": "london-data-center",
          "Location": {
            "Type": "Coordinate",
            "Latitude": "51.50",
            "Longitude": "-0.16"
          },
          "Bias": "0",
          "HealthCheck": "33333333-3333-3333-3333-333333333333"
        }
      ]
    }
  }
}
```

Latency Example

```
{
  "AWSPolicyFormatVersion":"2015-10-01",
  "RecordType":"A",
  "StartRule":"region_selector",
  "Endpoints":{
    "us_lb":{
      "Type":"elastic-load-balancer",
      "Value":"elb-123456.us-east-1.elb.amazonaws.com"
```

```
    },
    "europe_lb":{
      "Type":"elastic-load-balancer",
      "Value":"elb-654321.eu-west-1.elb.amazonaws.com"
    }
  },
  "Rules":{
    "region_selector":{
      "RuleType":"latency",
      "Regions":[
        {
          "Region":"us-east-1",
          "EndpointReference":"us_lb"
        },
        {
          "Region":"eu-west-1",
          "EndpointReference":"europe_lb"
        }
      ]
    }
  }
}
```

Weighted Example

```
{
  "AWSPolicyFormatVersion":"2015-10-01",
  "RecordType":"A",
  "StartRule":"round_robin",
  "Endpoints":{
    "srv1":{
      "Type":"value",
      "Value":"192.0.2.1"
    },
    "srv2":{
      "Type":"value",
      "Value":"192.0.2.2"
    },
    "srv3":{
      "Type":"value",
      "Value":"192.0.2.3"
    }
  },
```

```
  "Rules":{
    "round_robin":{
      "RuleType":"weighted",
      "Items":[
        {
          "EndpointReference":"srv1",
          "Weight":"3",
          "HealthCheck":"11111111-1111-1111-1111-111111111111"
        },
        {
          "EndpointReference":"srv2",
          "Weight":"1",
          "HealthCheck":"22222222-2222-2222-2222-222222222222"
        },
        {
          "EndpointReference":"srv3",
          "Weight":"1",
          "HealthCheck":"33333333-3333-3333-3333-333333333333"
        }
      ]
    }
  }
}
```

Example with Failover, Latency, and Geolocation Rules

```
{
  "AWSPolicyFormatVersion":"2015-10-01",
  "RecordType":"A",
  "StartRule":"geo_restriction",
  "Endpoints":{
    "east_coast_lb1":{
      "Type":"elastic-load-balancer",
      "Value":"elb-111111.us-east-1.elb.amazonaws.com"
    },
    "east_coast_lb2":{
      "Type":"elastic-load-balancer",
      "Value":"elb-222222.us-east-1.elb.amazonaws.com"
    },
    "west_coast_lb1":{
      "Type":"elastic-load-balancer",
      "Value":"elb-111111.us-west-1.elb.amazonaws.com"
    },
```

```
    "west_coast_lb2":{
      "Type":"elastic-load-balancer",
      "Value":"elb-222222.us-west-1.elb.amazonaws.com"
    },
    "denied_message":{
      "Type":"s3-website",
      "Region":"us-east-1",
      "Value":"video.example.com"
    }
  },
  "Rules":{
    "geo_restriction":{
      "RuleType":"geo",
      "Locations":[
        {
          "EndpointReference":"denied_message",
          "IsDefault":true
        },
        {
          "RuleReference":"region_selector",
          "Country":"US"
        }
      ]
    },
    "region_selector":{
      "RuleType":"latency",
      "Regions":[
        {
          "Region":"us-east-1",
          "RuleReference":"east_coast_region"
        },
        {
          "Region":"us-west-1",
          "RuleReference":"west_coast_region"
        }
      ]
    },
    "east_coast_region":{
      "RuleType":"failover",
      "Primary":{
        "EndpointReference":"east_coast_lb1"
      },
      "Secondary":{
        "EndpointReference":"east_coast_lb2"
```

```
      }
    },
    "west_coast_region":{
      "RuleType":"failover",
      "Primary":{
        "EndpointReference":"west_coast_lb1"
      },
      "Secondary":{
        "EndpointReference":"west_coast_lb2"
      }
    }
  }
}
```

What is Route 53 VPC Resolver?

Route 53 VPC Resolver responds recursively to DNS queries from AWS resources for public records, Amazon VPC-specific DNS names, and Amazon Route 53 private hosted zones, and is available by default in all VPCs.

> **Note**
>
> Route 53 VPC Resolver was previously called Route 53 Resolver, but was renamed when Route 53 Global Resolver was introduced.

An Amazon VPC connects to a VPC Resolver at a VPC+2 IP address. This VPC+2 address connects to a VPC Resolver within an Availability Zone.

A VPC Resolver automatically answers DNS queries for:

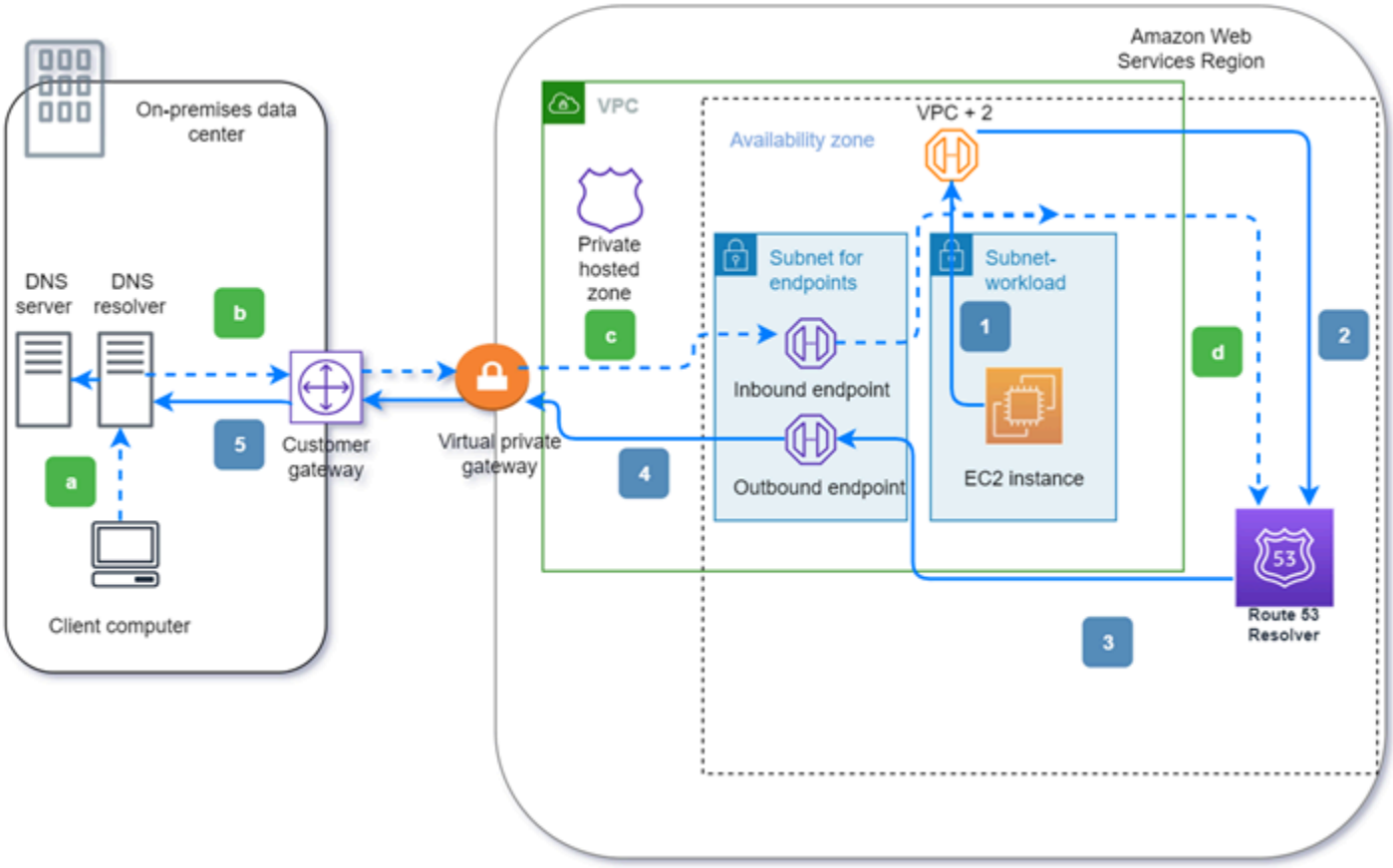
- Local VPC domain names for EC2 instances (for example, ec2-192-0-2-44.compute-1.amazonaws.com).
- Records in private hosted zones (for example, acme.example.com).
- For public domain names, VPC Resolver performs recursive lookups against public name servers on the internet.

If you have workloads that leverage both VPCs and on-premises resources, you also need to resolve DNS records hosted on-premises. Similarly, these on-premises resources might need to resolve names hosted on AWS. Through Resolver endpoints and conditional forwarding rules, you can resolve DNS queries between your on-premises resources and VPCs to create a hybrid cloud setup over VPN or Direct Connect (DX). Specifically:

- Inbound Resolver endpoints allow DNS queries to your VPC from your on-premises network or another VPC.
- Outbound Resolver endpoints allow DNS queries from your VPC to your on-premises network or another VPC.
- Resolver rules enable you to create one forwarding rule for each domain name and specify the name of the domain for which you want to forward DNS queries from your VPC to an on-

premises DNS resolver and from your on-premises to your VPC. Rules are applied directly to your VPC and can be shared across multiple accounts.

The following diagram shows hybrid DNS resolution with Resolver endpoints. Note that the diagram is simplified to show only one Availability Zone.



The diagram illustrates the following steps:

Outbound (solid arrows 1–5):

1. An Amazon EC2 instance needs to resolve a DNS query to the domain internal.example.com. The authoritative DNS server is in the on-premises data center. This DNS query is sent to the VPC+2 in the VPC that connects to VPC Resolver.
2. A VPC Resolver forwarding rule is configured to forward queries to internal.example.com in the on-premises data center.
3. The query is forwarded to an outbound endpoint.

4. The outbound endpoint forwards the query to the on-premises DNS resolver through a private connection between AWS and the data center. The connection can be either Direct Connect or AWS Site-to-Site VPN, depicted as a virtual private gateway.
5. The on-premises DNS resolver resolves the DNS query for internal.example.com and returns the answer to the Amazon EC2 instance through the same path in reverse.

Inbound (dashed arrows a–d):

a. A client in the on-premises data center needs to resolve a DNS query to an AWS resource for the domain dev.example.com. It sends the query to the on-premises DNS resolver.
b. The on-premises DNS resolver has a forwarding rule that points queries to dev.example.com to an inbound endpoint.
c. The query arrives at the inbound endpoint through a private connection, such as Direct Connect or AWS Site-to-Site VPN, depicted as a virtual gateway.
d. The inbound endpoint sends the query to VPC Resolver, and VPC Resolver resolves the DNS query for dev.example.com and returns the answer to the client through the same path in reverse.

Topics

- Resolving DNS queries between VPCs and your network
- Route 53 VPC Resolver availability and scaling
- Getting started with Route 53 VPC Resolver
- Forwarding inbound DNS queries to your VPCs
- Forwarding outbound DNS queries to your network
- Resolver delegation rules tutorial
- Enabling DNSSEC validation in Amazon Route 53

Resolving DNS queries between VPCs and your network

The VPC Resolver contains endpoints that you configure to answer DNS queries to and from your on-premises environment.

> **Note**
>
> Forwarding private DNS queries to any VPC CIDR + 2 address from on-premises or other VPC DNS servers is not supported, and can cause unstable results. Instead, we recommend that you use a Resolver inbound endpoint.

You also can integrate DNS resolution between VPC Resolver and DNS resolvers on your network by configuring forwarding rules. *Your network* can include any network that is reachable from your VPC, such as the following:

- The VPC itself
- Another peered VPC
- An on-premises network that is connected to AWS with Direct Connect, a VPN, or a network address translation (NAT) gateway

Before you start to forward queries, you create Resolver inbound and/or outbound endpoints in the connected VPC. These endpoints provide a path for inbound or outbound queries:

Inbound endpoint: DNS resolvers on your network can forward DNS queries to Route 53 VPC Resolver through this endpoint

There are two types of inbound endpoints, a **default inbound endpoint** that forwards to IP addresses, and a **delegation inbound endpoint** that delegates the authority for a subdomain hosted in Route 53 private hosted Zone to the Route 53 VPC Resolver. Inbound endpoints allow your DNS resolvers to easily resolve domain names for AWS resources such as EC2 instances or records in a Route 53 private hosted zone. For more information, see How DNS resolvers on your network forward DNS queries to Resolver endpoints.

Outbound endpoint: VPC Resolver conditionally forwards queries to resolvers on your network through this endpoint

To forward selected queries, you create Resolver rules that specify the domain names for the DNS queries that you want to forward (such as example.com), and the IP addresses of the DNS resolvers on your network that you want to forward the queries to. If a query matches multiple rules (example.com, acme.example.com), VPC Resolver chooses the rule with the most specific match (acme.example.com) and forwards the query to the IP addresses that you

specified in that rule. There are three types of rules, **forward**, **system**, and **delegation**. For more information, see How Resolver endpoints forward DNS queries from your VPCs to your network.

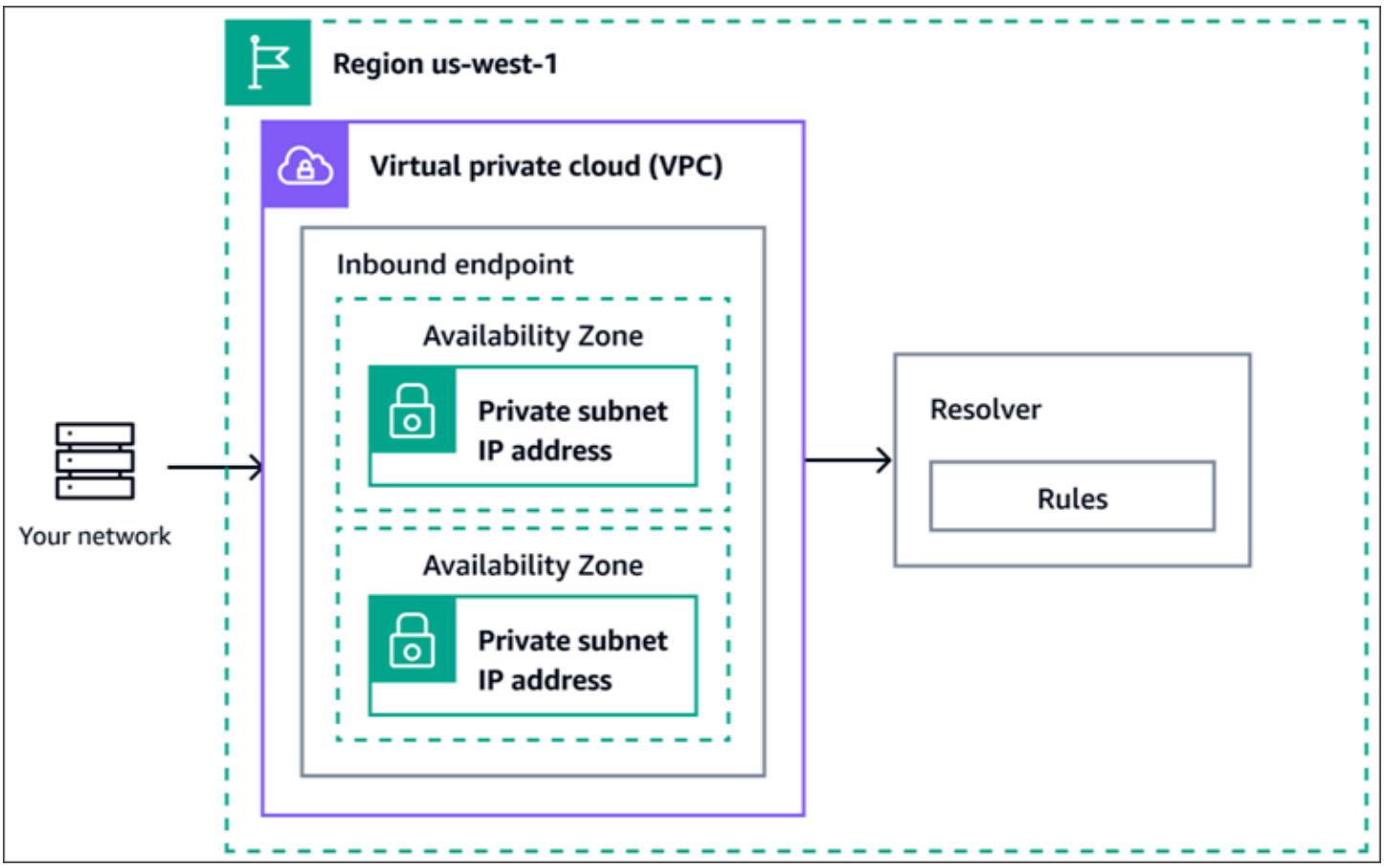
Like Amazon VPC, VPC Resolver is regional. In each Region where you have VPCs, you can choose whether to forward queries from your VPCs to your network (outbound queries), from your network to your VPCs (inbound queries), or both.

You can't create Resolver endpoints in a VPC that you don't own. Only the VPC owner can create VPC-level resources such as inbound endpoints.

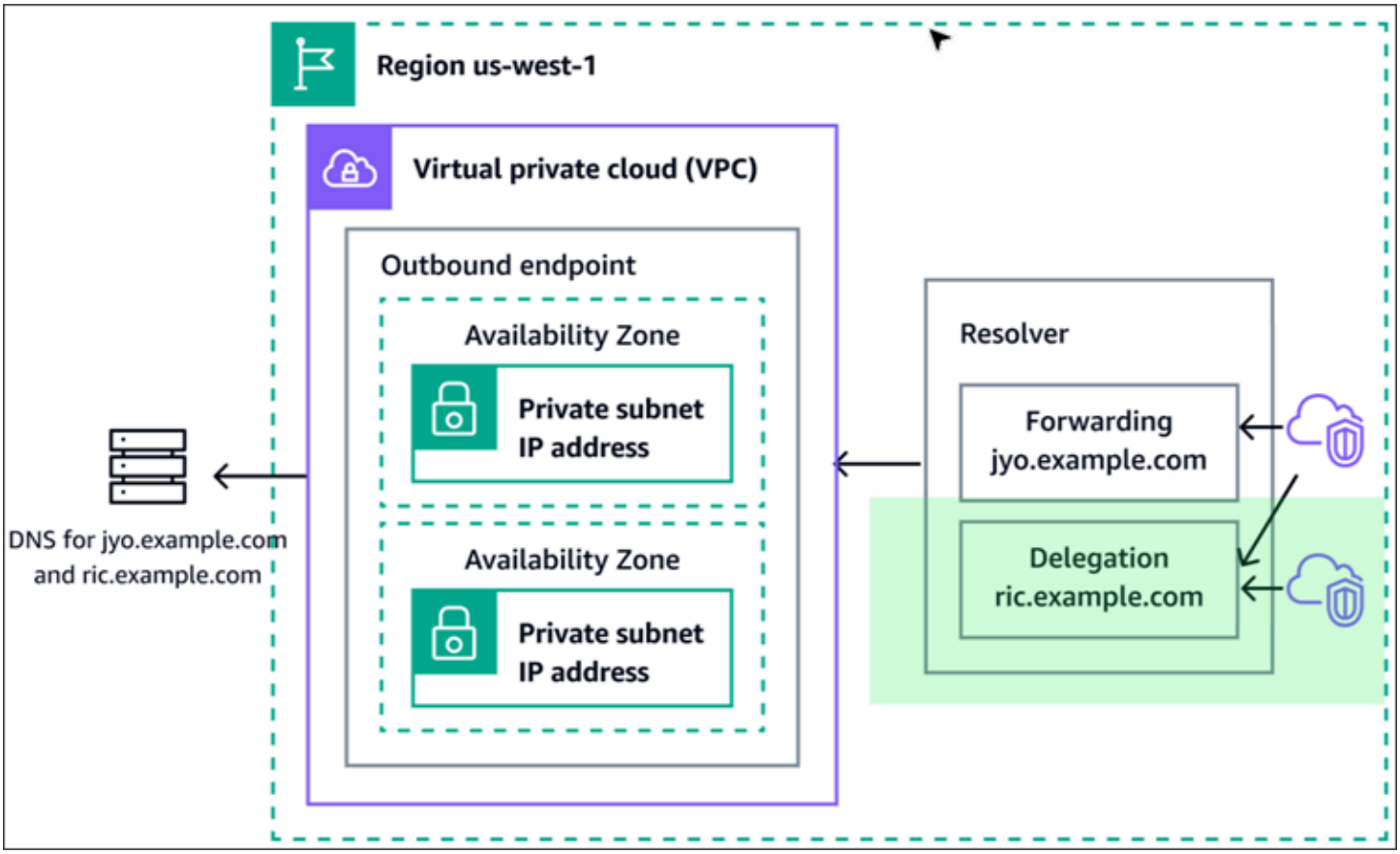
> **Note**
>
> When you create a Resolver endpoint, you can't specify a VPC that has the instance tenancy attribute set to `dedicated`. For more information, see .

To use inbound or outbound forwarding, you create a Resolver endpoint in your VPC. As part of the definition of an endpoint, you specify the IP addresses, or DNS delegation, that you want to forward inbound DNS queries to or the IP addresses that you want outbound queries to originate from. For each IP address and delegation that you specify, VPC Resolver automatically creates a VPC elastic network interface.

The following diagram shows the path of a DNS query from a DNS resolver on your network to Route 53 Resolver endpoints.



The following diagram shows the path of a DNS query from an EC2 instance in one of your VPCs to a DNS resolver on your network. The jyo.example.com domain uses a forwarding rule, whereas the ric.example.com subdomain has delegated the forwarding authority to VPC Resolver.



For an overview of VPC network interfaces, see Elastic network interfaces in the *Amazon VPC User Guide*.

Topics

- How DNS resolvers on your network forward DNS queries to Resolver endpoints
- How Resolver endpoints forward DNS queries from your VPCs to your network
- Considerations when creating inbound and outbound endpoints

How DNS resolvers on your network forward DNS queries to Resolver endpoints

To forward DNS queries from your network to Route 53 VPC Resolver, you create inbound endpoints in an AWS Region. There are two categories of inbound endpoints, **default** and **delegation**.

Steps for creating default inbound endpoints

1. You create a default Resolver inbound endpoint in a VPC and specify the IP addresses that the resolvers on your network forward DNS queries to, along with a VPC security group that includes inbound rules allowing TCP and UDP access on port 53. For instructions see Configuring inbound forwarding.

   For each IP address that you specify for the inbound endpoint, VPC Resolver creates a VPC elastic network interface in the VPC where you created the inbound endpoint.
2. You configure resolvers on your network to forward DNS queries for the applicable domain names to the IP addresses that you specified in the inbound endpoint. For more information, see Considerations when creating inbound and outbound endpoints.

Here's how VPC Resolver resolves DNS queries that originate on your network through a default inbound endpoint:

1. A web browser or another application on your network submits a DNS query for a domain name that you forwarded to VPC Resolver.
2. A resolver on your network forwards the query to the IP addresses in your inbound endpoint.
3. The inbound endpoint forwards the query to VPC Resolver.
4. VPC Resolver gets the applicable value for the domain name in the DNS query, either internally or by performing a recursive lookup against public name servers.
5. VPC Resolver returns the value to the inbound endpoint.
6. The inbound endpoint returns the value to the resolver on your network.
7. The resolver on your network returns the value to the application.
8. Using the value that was returned by VPC Resolver, the application submits a request, for example, a request for an object in an Amazon S3 bucket.

Steps for creating delegation inbound endpoints

1. You create a delegation Resolver inbound endpoint in a VPC. For instructions see Configuring inbound forwarding.

   For each IP address that you specify for the inbound endpoint, VPC Resolver creates a VPC elastic network interface in the VPC where you created the inbound endpoint.
2. You configure resolvers on your network to delegate DNS queries for the applicable domain names to VPC Resolver. For the glue records you must enter the IP addresses for the inbound

endpoints. For more information, see Considerations when creating inbound and outbound endpoints.

Here's how VPC Resolver resolves DNS queries that originate on your network through a delegation inbound endpoint:

1. As a prerequisite, you must delegate the subdomain that is hosted in the private hosted zone from on-premises. Because you are delegating the subdomain through the inbound delegation endpoint, you use the inbound endpoint IP addresses as the glue records for the subdomain that's being delegated.

   > **Note**
   >
   > You might also need to include the glue records to make sure the DNS query is resolvable. If you delegate a subdomain to name servers that are in the same zone as the parent domain, glue records are needed.

2. A web browser or another application on your network submits a DNS query for a domain name that you delegated to the VPC Resolver.
3. A resolver on your network forwards the query to the IP addresses in your inbound endpoint.
4. The inbound endpoint delegates the query to VPC Resolver.
5. VPC Resolver returns the address to the AWS resource from the private hosted zone to the inbound endpoint.
6. The inbound endpoint returns the value to the resolver on your network.
7. The resolver on your network returns the value to the application.
8. Using the value that was returned by VPC Resolver, the application submits a request, for example, a request for an object in an Amazon S3 bucket.

Creating an inbound endpoint doesn't change the behavior of VPC Resolver, it just provides a path from a location outside the AWS network to VPC Resolver.

How Resolver endpoints forward DNS queries from your VPCs to your network

When you want to forward DNS queries from the EC2 instances in one or more VPCs in an AWS Region to your network, you perform the following steps.

1. You create a Resolver outbound endpoint in a VPC, and you specify several values:
   - The VPC that you want DNS queries to pass through on the way to the resolvers on your network.
   - A VPC security group that includes outbound rules allowing TCP and UDP access on port 53 (or the port you're using for DNS queries on your network)

   For each IP address that you specify for the outbound endpoint, VPC Resolver creates an Amazon VPC elastic network interface in the VPC that you specify. For more information, see Considerations when creating inbound and outbound endpoints.
2. You create one or more rules, which specify the domain names of the DNS queries that you want to delegate to VPC Resolver to forward, or want VPC Resolver to forward to resolvers on your network. For forwarding rules, you also specify the IP addresses of the resolvers. For more information, see Using rules to control which queries are forwarded to your network.
3. You associate each rule with the VPCs for which you want to forward DNS queries to your network.

Using rules to control which queries are forwarded to your network

Rules control which DNS queries Resolver endpoint forwards to DNS resolvers on your network and which queries VPC Resolver answers itself.

You can categorize rules in a couple of ways. One way is by who creates the rules:

- **Autodefined rules** – VPC Resolver automatically creates autodefined rules and associates the rules with your VPCs. Most of these rules apply to the AWS-specific domain names that VPC Resolver answers queries for. For more information, see Domain names that VPC Resolver creates autodefined system rules for.
- **Custom rules** – You create custom rules and associate the rules with VPCs. Currently, you can create two types of custom rules, **conditional forwarding rules**, also known as forwarding rules, and **delegation rules**. **Forwarding** rules cause VPC Resolver to forward DNS queries from your VPCs to the IP addresses for DNS resolvers on your network.

  If you create a forwarding rule for the same domain as an autodefined rule, VPC Resolver forwards queries for that domain name to DNS resolvers on your network based on the settings in the forwarding rule.

Delegation rules forward DNS queries with the delegation records in the delegation rule that match the NS records in response to the resolvers on your network.

Another way to categorize rules is by what they do:

- **Conditional forwarding rules** – You create conditional forwarding rules (also known as forwarding rules) when you want to forward DNS queries for specified domain names to DNS resolvers on your network.
- **System rules** – System rules cause VPC Resolver to selectively override the behavior that is defined in a forwarding rule. When you create a system rule, VPC Resolver resolves DNS queries for specified subdomains that would otherwise be resolved by DNS resolvers on your network.

  By default, forwarding rules apply to a domain name and all its subdomains. If you want to forward queries for a domain to a resolver on your network but you don't want to forward queries for some subdomains, you create a system rule for the subdomains. For example, if you create a forwarding rule for example.com but you don't want to forward queries for acme.example.com, you create a system rule and specify acme.example.com for the domain name.
- **Recursive rule** – VPC Resolver automatically creates a recursive rule named **Internet Resolver**. This rule causes Route 53 VPC Resolver to act as a recursive resolver for any domain names that you didn't create custom rules for and that VPC Resolver didn't create autodefined rules for. For information about how to override this behavior, see "Forwarding All Queries to Your Network" later in this topic.

You can create custom rules that apply to specific domain names (yours or most AWS domain names), to public AWS domains names, or to all domain names.

Forwarding queries for specific domain names to your network

To forward queries for a specific domain name, such as example.com, to your network, you create a rule and specify that domain name. For **forwarding** rules you also specify the IP addresses of the DNS resolvers on your network that you want to forward the queries to, or, for **delegation** rules, create the delegation record for which you would like to delegate the authority to on-prem resolvers. You then associate each rule with the VPCs for which you want to forward DNS queries to your network. For example, you can create separate rules for

example.com, example.org, and example.net. Then you can associate the rules with the VPCs in an AWS Region in any combination.

Forwarding queries for amazonaws.com to your network

The domain name amazonaws.com is the public domain name for AWS resources such as EC2 instances and S3 buckets. If you want to forward queries for amazonaws.com to your network, create a rule, specify amazonaws.com for the domain name, and specify **Forward** or **Delegation** for the rule type depending on which method you want to use.

> **Note**
>
> VPC Resolver doesn't automatically forward DNS queries for some amazonaws.com subdomains even if you create a forwarding rule for amazonaws.com. For more information, see Domain names that VPC Resolver creates autodefined system rules for. For information about how to override this behavior, see "Forwarding All Queries to Your Network," immediately following.

Forwarding all queries to your network

If you want to forward all queries to your network, you create a rule, specify "." (dot) for the domain name, and associate the rule with the VPCs for which you want to forward all DNS queries to your network. VPC Resolver still doesn't forward all DNS queries to your network because using a DNS resolver outside of AWS would break some functionality. For example, some internal AWS domain names have internal IP address ranges that aren't accessible from outside of AWS. For a list of the domain names for which queries aren't forwarded to your network when you create a rule for ".", see Domain names that VPC Resolver creates autodefined system rules for.

However, autodefined system rules for reverse DNS can be disabled, allowing the "." rule to forward all reverse DNS queries to your network. For more information on how to turn off the autodefined rules, see Forwarding rules for reverse DNS queries in VPC Resolver.

If you want to try forwarding DNS queries for all domain names to your network, including the domain names that are excluded from forwarding by default, you can create a "." rule and do one of the following:

- Set the `enableDnsHostnames` flag for the VPC to `false`

- Create rules for the domain names that are listed in Domain names that VPC Resolver creates autodefined system rules for

> **Important**
>
> If you forward all domain names to your network, including the domain names that VPC Resolver excludes when you create a "." rule, some features might stop working.

How VPC Resolver determines whether the domain name in a query matches any rules

Route 53 VPC Resolver compares the domain name in the DNS query with the domain name in the rules that are associated with the VPC that the query originated from. VPC Resolver considers the domain names to match in the following cases:

- The domain names match exactly
- The domain name in the query is a subdomain of the domain name in the rule

For example, if the domain name in the rule is acme.example.com, VPC Resolver considers the following domain names in a DNS query to be a match:

- acme.example.com
- zenith.acme.example.com

The following domain names are not a match:

- example.com
- nadir.example.com

If the domain name in a query matches the domain name in more than one rule (such as example.com and www.example.com), VPC Resolver routes outbound DNS queries using the rule that contains the most specific domain name (www.example.com).

How VPC Resolver determines where to forward DNS queries

When an application that runs on an EC2 instance in a VPC submits a DNS query, Route 53 VPC Resolver performs the following steps:

1. Resolver checks for domain names in rules.

   If the domain name in a query matches the domain name in a default forward rule, VPC Resolver forwards the query to the IP address that you specified when you created the outbound endpoint. The outbound endpoint then forwards the query to the IP addresses of resolvers on your network, which you specified when you created the rule.

   If the delegation record in response matches the delegation rule, then the Resolver delegate the authority to on-prem resolvers through the outbound endpoint associated with the delegation rule.

   For more information, see How VPC Resolver determines whether the domain name in a query matches any rules.
2. Resolver endpoint forwards DNS queries based on the settings in the "." rule.

   If the domain name in a query doesn't match the domain name in any other rules, VPC Resolver forwards the query based on the settings in the autodefined "." (dot) rule. The dot rule applies to all domain names except some AWS internal domain names and record names in private hosted zones. This rule causes VPC Resolver to forward DNS queries to public name servers if the domain names in queries don't match any names in your custom forwarding rules. If you want to forward all queries to the DNS resolvers on your network, you can create a custom forwarding rule, specify "." for the domain name, specify **Forwarding** for **Type**, and specify the IP addresses of those resolvers.
3. VPC Resolver returns the response to the application that submitted the query.

Using rules in multiple Regions

Route 53 VPC Resolver is a Regional service, so objects that you create in one AWS Region are available only in that Region. To use the same rule in more than one Region, you must create the rule in each Region.

The AWS account that created a rule can share the rule with other AWS accounts. For more information, see Sharing Resolver rules with other AWS accounts and using shared rules.

Domain names that VPC Resolver creates autodefined system rules for

Resolver automatically creates autodefined system rules that define how queries for selected domains are resolved by default:

- For private hosted zones and for Amazon EC2–specific domain names (such as compute.amazonaws.com and compute.internal), autodefined rules make sure that your private hosted zones and EC2 instances continue to resolve if you create conditional forwarding rules for less specific domain names such as "." (dot) or "com".
- For publicly reserved domain names (such as localhost and 10.in-addr.arpa), DNS best practices recommend that queries are answered locally instead of being forwarded to public name servers. See RFC 6303, Locally Served DNS Zones.

> **Note**
>
> If you create a conditional forwarding rule for "." (dot) or "com", we recommend that you also create a system rule for amazonaws.com. (System rules cause VPC Resolver to locally resolve DNS queries for specific domains and subdomains.) Creating this system rule improves performance, reduces the number of queries that are forwarded to your network, and reduces VPC Resolver charges.

If you want to override an autodefined rule, you can create a conditional forwarding rule for the same domain name.

You can also disable some of the autodefined rules. For more information, see Forwarding rules for reverse DNS queries in VPC Resolver.

VPC Resolver creates the following autodefined rules.

Rules for private hosted zones

For each private hosted zone that you associate with a VPC, VPC Resolver creates a rule and associates it with the VPC. If you associate the private hosted zone with multiple VPCs, VPC Resolver associates the rule with the same VPCs.

The rule has a type of **Forward**.

Rules for various AWS internal domain names

All rules for the internal domain names in this section have a type of **Forward**. VPC Resolver forwards DNS queries for these domain names to the authoritative name servers for the VPC.

> **Note**
>
> VPC Resolver creates most of these rules when you set the `enableDnsHostnames` flag for a VPC to `true`. VPC Resolver creates the rules even if you aren't using Resolver endpoints.

VPC Resolver creates the following autodefined rules and associates them with a VPC when you set the `enableDnsHostnames` flag for the VPC to `true`:

- *Region-name*.compute.internal, for example, eu-west-1.compute.internal. The us-east-1 Region doesn't use this domain name.
- *Region-name*.compute.*amazon-domain-name*, for example, eu-west-1.compute.amazonaws.com or cn-north-1.compute.amazonaws.com.cn. The us-east-1 Region doesn't use this domain name.
- ec2.internal. Only the us-east-1 Region uses this domain name.
- compute-1.internal. Only the us-east-1 Region uses this domain name.
- compute-1.amazonaws.com. Only the us-east-1 Region uses this domain name.

The following autodefined rules are for the reverse DNS lookup for the rules that VPC Resolver creates when you set the `enableDnsHostnames` flag for the VPC to `true`.

- 10.in-addr.arpa
- 16.172.in-addr.arpa through 31.172.in-addr.arpa
- 168.192.in-addr.arpa
- 254.169.254.169.in-addr.arpa
- Rules for each of the CIDR ranges for the VPC. For example, if a VPC that has a CIDR range of 10.0.0.0/23, VPC Resolver creates the following rules:
  - 0.0.10.in-addr.arpa
  - 1.0.10.in-addr.arpa

The following autodefined rules, for localhost-related domains, also are created and associated with a VPC when you set the `enableDnsHostnames` flag for the VPC to `true`:

- localhost
- localdomain
- 127.in-addr.arpa
- 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.arpa
- 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.arpa

VPC Resolver creates the following autodefined rules and associates them with your VPC when you connect the VPC with another VPC through transit gateway or VPC peering, and with DNS support enabled:

- The reverse DNS lookup for the peer VPC's IP address ranges, for example, 0.192.in-addr.arpa

  If you add an IPv4 CIDR block to a VPC, VPC Resolver adds an autodefined rule for the new IP address range.
- If the other VPC is in another Region, the following domain names:
  - *Region-name*.compute.internal. The us-east-1 Region doesn't use this domain name.
  - *Region-name*.compute.*amazon-domain-name*. The us-east-1 Region doesn't use this domain name.
  - ec2.internal. Only the us-east-1 Region uses this domain name.
  - compute-1.amazonaws.com. Only the us-east-1 Region uses this domain name.

A rule for all other domains

VPC Resolver creates a "." (dot) rule that applies to all domain names that aren't specified earlier in this topic. The "." rule has a type of **Recursive**, which means that the rule causes VPC Resolver to act as a recursive resolver.

Considerations when creating inbound and outbound endpoints

Before you create inbound and outbound Resolver endpoints in an AWS Region, consider the following issues.

Topics

- Number of inbound and outbound endpoints in each Region
- Using the same VPC for inbound and outbound endpoints
- Inbound endpoints and private hosted zones

- VPC peering
- IP addresses in shared subnets
- Connection between your network and the VPCs that you create endpoints in
- When you share rules, you also share outbound endpoints
- Choosing protocols for the endpoints
- Using VPC Resolver in VPCs that are configured for dedicated instance tenancy

Number of inbound and outbound endpoints in each Region

When you want to integrate DNS for the VPCs in an AWS Region with DNS for your network, you typically need one Resolver inbound endpoint (for DNS queries that you're forwarding to your VPCs) and one outbound endpoint (for queries that you're forwarding from your VPCs to your network). You can create multiple inbound endpoints and multiple outbound endpoints, but one inbound or outbound endpoint is sufficient to handle the DNS queries for each respective direction. Note the following:

- For each Resolver endpoint, you specify two or more IP addresses in different Availability Zones. Each IP address in an endpoint can handle a large number of DNS queries per second. (For the current maximum number of queries per second per IP address in an endpoint, see Quotas on Route 53 VPC Resolver.) If you need VPC Resolver to handle more queries, you can add more IP addresses to your existing endpoint instead of adding another endpoint.
- VPC Resolver pricing is based on the number of IP addresses in your endpoints and on the number of DNS queries that the endpoint processes. Each endpoint includes a minimum of two IP addresses. For more information about VPC Resolver pricing, see Amazon Route 53 Pricing.
- Each rule specifies the outbound endpoint that DNS queries are forwarded from. If you create multiple outbound endpoints in an AWS Region and you want to associate some or all Resolver rules with every VPC, you need to create multiple copies of those rules.

Using the same VPC for inbound and outbound endpoints

You can create inbound and outbound endpoints in the same VPC or in different VPCs in the same Region.

For more information, see Best practices for Amazon Route 53.

Inbound endpoints and private hosted zones

If you want VPC Resolver to resolve inbound DNS queries using records in a private hosted zone, associate the private hosted zone with the VPC that you created the inbound endpoint in. For information about associating private hosted zones with VPCs, see Working with private hosted zones.

VPC peering

You can use any VPC in an AWS Region for an inbound or an outbound endpoint regardless of whether the VPC that you choose is peered with other VPCs. For more information, see Amazon Virtual Private Cloud VPC peering.

IP addresses in shared subnets

When you create an inbound or outbound endpoint, you can specify an IP address in a shared subnet only if the current account created the VPC. If another account creates a VPC and shares a subnet in the VPC with your account, you can't specify an IP address in that subnet. For more information about shared subnets, see Working with shared VPCs in the *Amazon VPC User Guide*.

Connection between your network and the VPCs that you create endpoints in

You must have one of the following connections between your network and the VPCs that you create endpoints in:

- **Inbound endpoints** – You must set up either an Direct Connect connection or a VPN connection between your network and each VPC that you create an inbound endpoint for.
- **Outbound endpoints** – You must set up an Direct Connect connection, a VPN connection, or a network address translation (NAT) gateway between your network and each VPC that you create an outbound endpoint for.

When you share rules, you also share outbound endpoints

When you create a rule, you specify the outbound endpoint that you want VPC Resolver to use to forward DNS queries to your network. If you share the rule with another AWS account, you also indirectly share the outbound endpoint that you specify in the rule. If you used more than one AWS account to create VPCs in an AWS Region, you can do the following:

- Create one outbound endpoint in the Region.

- Create rules using one AWS account.
- Share the rules with all the AWS accounts that created VPCs in the Region.

This allows you to use one outbound endpoint in a Region to forward DNS queries to your network from multiple VPCs even if the VPCs were created using different AWS accounts.

Choosing protocols for the endpoints

Endpoint protocols determine how data is transmitted to an inbound endpoint and from an outbound endpoint. Encrypting DNS queries for VPC traffic is not needed because every packet flow on the network is individually authorized against a rule to validate the correct source and destination before it is transmitted and delivered. It is highly improbable for information to arbitrarily pass between entities without specifically being authorized by both the transmitting and receiving entity. If a packet is being routed to a destination without a rule that matches it, the packet is dropped. For more information, see VPC features.

The available protocols are:

- **Do53:** DNS over port 53. The data is relayed by using the VPC Resolver without additional encryption. While the data cannot be read by external parties, it can be viewed within the AWS networks. Uses either UDP or TCP to send the packets. Do53 is primarily used for traffic within and between Amazon VPCs. Currently, this is the only protocol available for delegation inbound endpoints.
- **DoH:** The data is transmitted over an encrypted HTTPS session. DoH adds an added level of security where data can't be decrypted by unauthorized users, and cannot be read by anyone except the intended recipient. Not available for delegation inbound endpoints.
- **DoH-FIPS:** The data is transmitted over an encrypted HTTPS session that is compliant with the FIPS 140-2 cryptographic standard. Supported for inbound endpoints only. For more information, see FIPS PUB 140-2. Not available for delegation inbound endpoints.

For an inbound endpoint of type **Forward**, you can apply the protocols as follows:

- Do53 and DoH in combination.
- Do53 and DoH-FIPS in combination.
- Do53 alone.
- DoH alone.

- DoH-FIPS alone.
- None, which is treated as Do53.

For an outbound endpoint you can apply the protocols as follows:

- Do53 and DoH in combination.
- Do53 alone.
- DoH alone.
- None, which is treated as Do53.

See also Values that you specify when you create or edit inbound endpoints and Values that you specify when you create or edit outbound endpoints.

Using VPC Resolver in VPCs that are configured for dedicated instance tenancy

When you create a Resolver endpoint, you can't specify a VPC that has the instance tenancy attribute set to `dedicated`. VPC Resolver doesn't run on single-tenant hardware.

You can still use VPC Resolver to resolve DNS queries that originate in a VPC. Create at least one VPC that has the instance tenancy attribute set to `default`, and specify that VPC when you create inbound and outbound endpoints.

When you create a forwarding rule, you can associate it with any VPC, regardless of the setting for the instance tenancy attribute.

Route 53 VPC Resolver availability and scaling

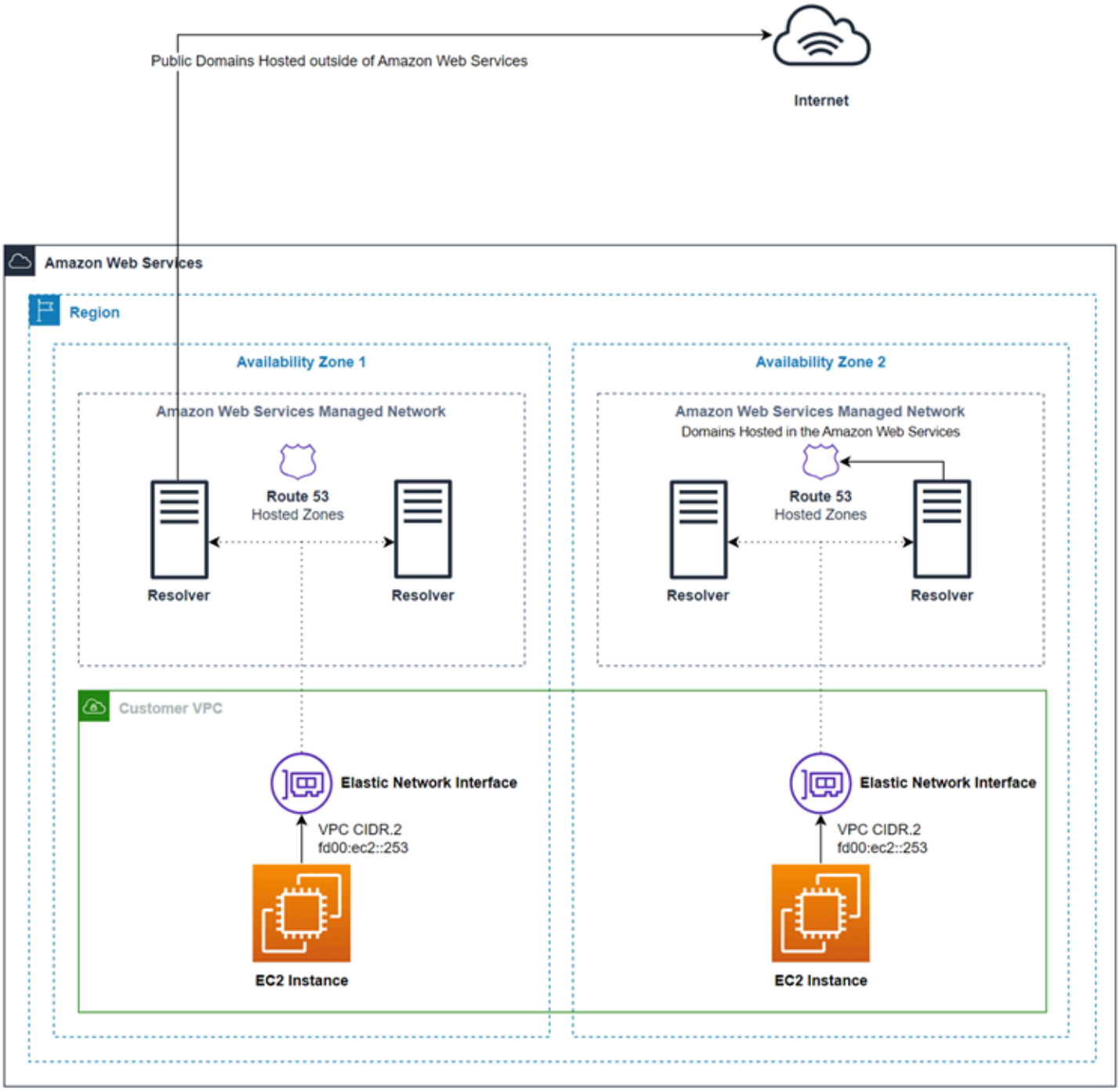
Route 53 VPC Resolver, running on the Amazon VPC CIDR + 2 address and fd00:ec2::253, is available by default in all VPCs, and responds recursively to DNS queries for public records, Amazon VPC-specific DNS names, and Route 53 private hosted zones. There are two highly available components, transparent to users, that make up the Route 53 VPC Resolver: the Nitro Resolver service and the Zonal Resolver fleet. The Nitro Resolver Service is a service that runs in the Nitro card on Nitro instances and in Dom0 in older generation instances, and consumes packets addressed to the Route 53 VPC Resolver locally on the host server. For more information, see The Security Design of the AWS Nitro System.

The Nitro Resolver service carries a local cache which can help reduce latency by responding to repeat queries which are made over a short period of time by an instance. When the Nitro Resolver

service receives a query which it does not have a cached answer for, it forwards the query to the Zonal Resolver fleet, a highly available fleet of resolvers typically in the same Availability Zone as the instance. When there are failures handling queries by upstream name servers or other components in the path, the Nitro Resolver service is frequently able to handle these failures transparently without impact to the workloads running on the instance. Furthermore, if the Resolver encounters query timeouts, refused connections, or SERVFAILS from the domain's name servers, it might respond with a cached answer beyond the Time-To-Live (TTL) value to improve availability. Queries between the Nitro Resolver service and Zonal Resolver fleet are restricted to a tightly controlled network outside of the customer VPC, which is inaccessible to customers and subject to rigorous security controls. By handling queries between the Nitro Resolver service and Zonal Resolver fleet outside of the VPC, customers are prevented from intercepting DNS queries inside of their VPC. Queries destined to name servers outside of AWS will traverse the public internet, originating from public IP addresses belonging to the Zonal Resolver fleet. VPC Resolver does not support the EDNS0-Client Subnet attribute, which means all queries destined to public DNS name servers do not include information about the originating customer IP address.

The Nitro Resolver service is part of the Link-Local services on the instance. Link-Local services include Route 53 VPC Resolver, Amazon Time Service (NTP), Instance Metadata Service (IMDS) and Windows Licensing Service (for Windows instances). These services scale with each elastic network interface you create in your VPC, and each network interface allows 1024 packets per second (PPS) destined to Link-Local services. Packets exceeding this limit are rejected. You can determine if you have exceeded this limit from the `linklocal_allowance_exceeded` value returned by ethtool. For more information about the ethtool, see Monitor network performance for your Amazon EC2 instance in the *Amazon EC2 User Guide*. This metric can also be reported to CloudWatch metrics by the CloudWatch Agent. Since the Route 53 VPC Resolver is implemented per network interface, it scales and becomes more reliable as you add more instances in more Availability Zones. There is no per-VPC aggregate limit on the number of queries, thus the Route 53 VPC Resolver can scale within the bounds of a VPC, which is inherently based on network address usage (NAU). For more information, see Network Address Usage for your VPC in the *Amazon Virtual Private Cloud User Guide*.

The following diagram shows an overview of how Route 53 VPC Resolver resolves DNS queries within Availability Zones.



Getting started with Route 53 VPC Resolver

The Route 53 VPC Resolver console includes a wizard that guides you through the following steps for getting started with VPC Resolver:

- Create endpoints: inbound, outbound, or both.

- For outbound endpoints, create one or more forwarding rules, which specify the domain names for which you want to route DNS queries to your network.
- If you created an outbound endpoint, choose the VPC that you want to associate the rules with.

To configure Route 53 VPC Resolver using the wizard

1. Sign in to the AWS Management Console and open the VPC Resolver console at https://console.aws.amazon.com/route53resolver/.
2. On the **Welcome to Route 53 VPC Resolver** page, choose **Configure endpoints**.
3. On the navigation bar, choose the Region where you want to create a Resolver endpoint.
4. Under **Basic configuration**, choose the direction that you want to forward DNS queries:
   - **Inbound and outbound**: The wizard guides you through settings that let you both forward DNS queries from resolvers on your network to VPC Resolver in a VPC, and forward specified queries (such as example.com or example.net) from a VPC to resolvers on your network.
   - **Inbound only**: The wizard guides you through settings that let you forward DNS queries from resolvers on your network to VPC Resolver in a VPC.
   - **Outbound only**: The wizard guides you through settings that let you forward specified queries from a VPC to resolvers on your network.
5. Choose **Next**.
6. If you chose **Inbound and outbound** or **Inbound only**, enter the applicable values for configuring an inbound endpoint. Then continue with step 7. For more information, see Values that you specify when you create or edit inbound endpoints.

   If you choose **Outbound only**, skip to step 7.
7. Enter the applicable values for configuring an outbound endpoint. For more information, see Values that you specify when you create or edit outbound endpoints.
8. If you chose **Inbound and outbound** or **Outbound only**, enter the applicable values for creating a rule. For more information, see Values that you specify when you create or edit rules.
9. On the **Review and create** page, confirm that the settings that you specified on previous pages are correct. If necessary, choose **Edit** for the applicable section, and update settings. When you're satisfied with the settings, choose **Submit**.

> **Note**
>
> Creating an outbound endpoint takes a minute or two. You can't create another outbound endpoint until the first one is created.

10. If you want to create more rules, see Managing forwarding rules.
11. If you created an inbound endpoint, configure DNS resolvers on your network to forward the applicable DNS queries to the IP addresses for your inbound endpoint. For more information, refer to the documentation for your DNS application.

Forwarding inbound DNS queries to your VPCs

To forward DNS queries from your network to VPC Resolver, you create an inbound endpoint. An inbound endpoint specifies the IP addresses (from the range of IP addresses available to your VPC) that you want DNS resolvers on your network to forward DNS queries to. Those IP addresses aren't public IP addresses, so for each inbound endpoint, you need to connect your VPC to your network using either an Direct Connect connection or a VPN connection.

When implementing inbound delegation, you're delegating DNS authority for a subdomain from your on-premises DNS infrastructure to VPC Resolver. To properly configure this delegation, you must use the inbound endpoint's IP addresses as glue records (NS records) on your on-premises name server for the subdomain being delegated. For example, if you're delegating the subdomain "aws.example.com" to VPC Resolver through an inbound delegation endpoint with IP addresses 10.0.1.100 and 10.0.1.101, you would create NS records on your on-premises DNS server pointing "aws.example.com" to these IP addresses. This makes sure that DNS queries for the delegated subdomain are properly routed to the VPC Resolver through the inbound endpoint, allowing the VPC Resolver to respond with records from the associated private hosted zone.

Configuring inbound forwarding

To create an inbound endpoint, perform the following procedure.

To create an inbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Inbound endpoints**.

3. On the navigation bar, choose the Region where you want to create an inbound endpoint.
4. Choose **Create inbound endpoint**.
5. Enter the applicable values. For more information, see Values that you specify when you create or edit inbound endpoints.
6. Choose **Create**.
7. Configure DNS resolvers on your network to forward the applicable DNS queries to the IP addresses for your inbound endpoint. For more information, refer to the documentation for your DNS application.

Values that you specify when you create or edit inbound endpoints

When you create or edit an inbound endpoint, you specify the following values:

Outpost ID

If you are creating the endpoint for a VPC Resolver on an AWS Outposts VPC, this is the AWS Outposts ID.

Endpoint name

A friendly name that lets you easily find an inbound endpoint on the dashboard.

Endpoint category

Choose either **Default** or **Delegation**. When the category is **Default**, the resolver on your network forwards the DNS requests to the IP address of the inbound endpoint. When the category is **Delegation**, the authority for a domain is delegated to the VPC Resolver.

VPC in the *region-name* Region

All inbound DNS queries from your network pass through this VPC on the way to VPC Resolver.

Security group for this endpoint

The ID of one or more security groups that you want to use to control access to this VPC. The security group that you specify must include one or more inbound rules. Inbound rules must allow TCP and UDP access on port 53. If you are using the DoH protocol, you will also have to allow port 443 in security group.You can't change this value after you create the endpoint.

Some security group rules will cause your connection to be tracked and the overall maximum queries per second per IP address for an inbound endpoint can be as low as 1500. To avoid connection tracking caused by a security group, see Untracked connections.

> **Note**
>
> In order to add multiple security groups, use the AWS CLI command `create-resolver-endpoint`. For more information, see create-resolver-endpoint

For more information, see Security groups for your VPC in the *Amazon VPC User Guide*.

Endpoint type

The endpoint type can be either IPv4, IPv6, or dual-stack IP addresses. For a dual-stack endpoint, the endpoint has both IPv4 and IPv6 addresses that your DNS resolver on your network can forward DNS queries to.

> **Note**
>
> For security reasons, VPC Resolver denies direct IPv6 traffic access from the public internet for all dual-stack and IPv6 IP addresses.

IP addresses

The IP addresses that you want DNS resolvers on your network to forward DNS queries to. You must specify a minimum of two IP addresses for redundancy. If you created a delegation inbound endpoint, use these IP addresses as the glue NS records for the subdomain for which you want to delegate the authority to VPC Resolver. Note the following:

Multiple Availability Zones

We recommend that you specify IP addresses in at least two Availability Zones. You can optionally specify additional IP addresses in those or other Availability Zones.

IP addresses and Amazon VPC elastic network interfaces

For each combination of Availability Zone, Subnet, and IP address that you specify, VPC Resolver creates an Amazon VPC elastic network interface. For the current maximum number of DNS queries per second per IP address in an endpoint, see Quotas on Route 53 VPC Resolver. For information about pricing for each elastic network interface, see "Amazon Route 53" on the Amazon Route 53 pricing page.

> **Note**
>
> Resolver endpoint has a private IP address. These IP addresses will not change through the course of an endpoint's life.

For each IP address, specify the following values. Each IP address must be in an Availability Zone in the VPC that you specified in **VPC in the *region-name* Region**.

Availability Zone

The Availability Zone that you want DNS queries to pass through on the way to your VPC. The Availability Zone that you specify must be configured with a subnet.

Subnet

The subnet that contains the IP addresses you want assigned to your Resolver endpoint ENIs. These are the addresses you will send DNS queries to. The subnet must have an available IP address.

The subnet IP address must match the **Endpoint type**.

We recommend using VPC Resolver on AWS Outposts to create endpoints on AWS Outposts Racks.

> **Important**
>
> Outposts subnets with Local Network Interface (LNI) enabled are not compatible with VPC Resolver endpoints. If you enable LNI on a subnet that contains VPC Resolver endpoint elastic network interfaces (ENIs), those ENIs will stop functioning. For more information, see Subnet compatibility for Resolver endpoints.

IP address

The IP address that you want to assign to the inbound endpoints.

Choose whether you want VPC Resolver to choose an IP address for you from among the available IP addresses in the specified subnet, or you want to specify the IP address yourself.

If you choose to specify the IP address yourself, enter either an IPv4 or IPv6 address, or both.

Protocols

Endpoint protocol determines how data is transmitted to the inbound endpoint. Choose a protocol, or protocols, depending on the level of security needed.

- **Do53:** (Default) The data is relayed using the Route 53 VPC Resolver without additional encryption. While the data cannot be read by external parties, it can be viewed within the AWS networks. This is the only protocol currently available for the **Delegation** inbound endpoint category.
- **DoH:** The data is transmitted over an encrypted HTTPS session. DoH adds an added level of security where data can't be decrypted by unauthorized users, and can't be read by anyone except the intended recipient.
- **DoH-FIPS:** The data is transmitted over an encrypted HTTPS session that is compliant with the FIPS 140-2 cryptographic standard. Supported for inbound endpoints only. For more information, see FIPS PUB 140-2.

> **Note**
>
> For DoH/DoH-FIPS inbound endpoints, there is a known issue with incorrect source IP being published in the VPC Resolver query logging.

For an inbound endpoint you can apply the protocols as follows:

- Do53 and DoH in combination.
- Do53 and DoH-FIPS in combination.
- Do53 alone.
- DoH alone.
- DoH-FIPS alone.
- None, which is treated as Do53.

> **Important**
>
> You can't change the protocol of an inbound endpoint directly from only Do53 to only DoH, or DoH-FIPS. This is to prevent a sudden disruption to incoming traffic that relies on Do53. To change the protocol from Do53 to DoH, or DoH-FIPS, you must first enable both Do53 and DoH, or Do53 and DoH-FIPS, to make sure that all incoming traffic has transferred to using the DoH protocol, or DoH-FIPS, and then remove the Do53.

DNS64

Enable DNS64 to allow VPC Resolver to synthesize AAAA (IPv6) records for IPv4-only services. When an IPv6-only client queries for a service that has no IPv6 address, VPC Resolver synthesizes an IPv6 address by prepending the well-known `64:ff9b::/96` prefix to the IPv4 address. This enables IPv6-only clients that send queries through the inbound endpoint to reach IPv4-only services. DNS64 works with NAT64 to provide complete IPv6-to-IPv4 translation.

For more information, see DNS64 and NAT64 in the *Amazon VPC User Guide*.

Tags

Specify one or more keys and the corresponding values. For example, you might specify **Cost center** for **Key** and specify **456** for **Value**.

Managing inbound endpoints

To manage inbound endpoints, perform the applicable procedure.

Topics

- Viewing and editing inbound endpoints
- Viewing the status for inbound endpoints
- Deleting inbound endpoints

Viewing and editing inbound endpoints

To view and edit settings for an inbound endpoint, perform the following procedure.

To view and edit settings for an inbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Inbound endpoints**.
3. On the navigation bar, choose the Region where you created the inbound endpoint.
4. Choose the option for the endpoint that you want to view settings for or want to edit.
5. Choose **View details** or **Edit**.

For information about the values for inbound endpoints, see Values that you specify when you create or edit inbound endpoints.

6. If you chose **Edit**, enter the applicable values, and choose **Save**.

Viewing the status for inbound endpoints

To view the status for an inbound endpoint, perform the following procedure.

To view the status for an inbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Inbound endpoints**.
3. On the navigation bar, choose the Region where you created the inbound endpoint. The **Status** column contains one of the following values:

   Creating

   VPC Resolver is creating and configuring one or more Amazon VPC network interfaces for this endpoint.

   Operational

   The Amazon VPC network interfaces for this endpoint are correctly configured and able to pass inbound or outbound DNS queries between your network and VPC Resolver.

   Updating

   VPC Resolver is associating or disassociating one or more network interfaces with this endpoint.

   Auto recovering

   VPC Resolver is trying to recover one or more of the network interfaces that are associated with this endpoint. During the recovery process, the endpoint functions with limited capacity because of the limit on the number of DNS queries per IP address (per network interface). For the current limit, see Quotas on Route 53 VPC Resolver.

Action needed

This endpoint is unhealthy, and VPC Resolver can't automatically recover it. To resolve the problem, we recommend that you check each IP address that you associated with the endpoint. For each IP address that isn't available, add another IP address and then delete the IP address that isn't available. (An endpoint must always include at least two IP addresses.) A status of **Action needed** can have a variety of causes. Here are two common causes:

- One or more of the network interfaces that are associated with the endpoint were deleted using Amazon VPC.
- The network interface couldn't be created for some reason that's outside the control of VPC Resolver.

Deleting

VPC Resolver is deleting this endpoint and the associated network interfaces.

Deleting inbound endpoints

To delete an inbound endpoint, perform the following procedure.

> **Important**
>
> If you delete an inbound endpoint, DNS queries from your network are no longer forwarded to VPC Resolver in the VPC that you specified in the endpoint.

To delete an inbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Inbound endpoints**.
3. On the navigation bar, choose the Region where you created the inbound endpoint.
4. Choose the option for the endpoint that you want to delete.
5. Choose **Delete**.
6. To confirm that you want to delete the endpoint, enter the name of the endpoint and choose **Submit**.

Forwarding outbound DNS queries to your network

To forward DNS queries that originate on Amazon EC2 instances in one or more VPCs to your network, you create an outbound endpoint and one or more rules:

Outbound endpoint

To forward DNS queries from your VPCs to your network, you create an outbound endpoint. An outbound endpoint specifies the IP addresses that queries originate from. Those IP addresses, which you choose from the range of IP addresses available to your VPC, aren't public IP addresses. This means that, for each outbound endpoint, you need to connect your VPC to your network using Direct Connect connection, a VPN connection, or a network address translation (NAT) gateway. Note that you can use the same outbound endpoint for multiple VPCs in the same Region, or you can create multiple outbound endpoints. If you want your outbound endpoint to use DNS64, you can enable DNS64 using Amazon Virtual Private Cloud. For more information, see DNS64 and NAT64 in the *Amazon VPC User Guide*.

The target IP from the VPC Resolver rule is chosen at random by VPC Resolver and there is no preference on choosing a particular target IP over the other. If a target IP does not respond to the DNS request forwarded, VPC Resolver retries to a random IP address among the target IPs.

Make sure that all the target IP addresses are reachable from the Resolver endpoints. If VPC Resolver can't forward outbound DNS queries to any of the target IPs, it can lead to extended DNS resolution times.

Rules

To specify the domain names of the queries that you want to forward to DNS resolvers on your network, you create one or more rules. Each forwarding rule specifies one domain name. You then associate rules with the VPCs for which you want to forward queries to your network.

Outbound delegation rules follow specific delegation principles that differ from standard forwarding rules. When you create a delegation rule, VPC Resolver evaluates the delegation records in the rule against the NS records in DNS responses to determine if delegation should occur. The VPC Resolver will delegate authority to your on-premises resolvers only when there's a match between the delegation rule configuration and the actual NS records returned in the DNS response. Unlike forwarding rules that redirect queries based on domain name matching, delegation rules respect the DNS delegation chain and only activate when the authoritative name servers in the response match the delegation configuration.

For more information, see the following topics:

- Private hosted zones that have overlapping namespaces
- Private hosted zones and Route 53 VPC Resolver rules

Configuring outbound forwarding

To configure VPC Resolver to forward DNS queries that originate in your VPC to your network, perform the following procedures.

> **Important**
>
> After you create an outbound endpoint, you must create one or more rules and associate them with one or more VPCs. Rules specify the domain names of the DNS queries that you want to forward to your network.

To create an outbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Outbound endpoints**.
3. On the navigation bar, choose the Region where you want to create an outbound endpoint.
4. Choose **Create outbound endpoint**.
5. Enter the applicable values. For more information, see Values that you specify when you create or edit outbound endpoints.
6. Choose **Create**.

   > **Note**
   >
   > Creating an outbound endpoint takes a minute or two. You can't create another outbound endpoint until the first one is created.

7. Create one or more rules to specify the domain names of the DNS queries that you want to forward to your network. For more information, see the next procedure.

To create one or more forwarding rules, perform the following procedure.

To create forwarding rules and associate the rules with one or more VPCs

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Rules**.
3. On the navigation bar, choose the Region where you want to create the rule.
4. Choose **Create rule**.
5. Enter the applicable values. For more information, see Values that you specify when you create or edit rules.
6. Choose **Save**.
7. To add another rule, repeat steps 4 through 6.

Values that you specify when you create or edit outbound endpoints

When you create or edit an outbound endpoint, you specify the following values:

Outpost ID

If you are creating the endpoint for a VPC Resolver on an AWS Outposts VPC, this is the AWS Outposts ID.

Endpoint name

A friendly name that lets you easily find an outbound endpoint on the dashboard.

VPC in the *region-name* Region

All outbound DNS queries flow through this VPC on the way to your network.

Security group for this endpoint

The ID of one or more security groups that you want to use to control access to this VPC. The security group that you specify must include one or more outbound rules. Outbound rules must allow TCP and UDP access on the port that you're using for DNS queries on your network. You can't change this value after you create an endpoint.

Some security group rules will cause your connection to be tracked and potentially impact the maximum queries per second from outbound endpoint to your target name server. To avoid connection tracking caused by a security group, see Untracked connections.

For more information, see Security groups for your VPC in the *Amazon VPC User Guide*.

Endpoint type

The endpoint type can be either IPv4, IPv6, or dual-stack IP addresses. For a dual-stack endpoint, the endpoint has both IPv4 and IPv6 addresses that your DNS resolver on your network can forward DNS queries to.

> **Note**
>
> For security reasons, VPC Resolver denies direct IPv6 traffic access to the public internet for all dual-stack and IPv6 IP addresses by default. To enable this access, turn on **IPv6 internet access** when you create or edit the endpoint.

IP addresses

The IP addresses in your VPC that you want VPC Resolver to forward DNS queries to on the way to resolvers on your network. These are not the IP addresses of the DNS resolvers on your network; you specify resolver IP addresses when you create the rules that you associate with one or more VPCs. You must specify a minimum of two IP addresses for redundancy.

> **Note**
>
> Resolver endpoint has a private IP address. These IP addresses will not change through the course of an endpoint's life.

Note the following:

Multiple Availability Zones

We recommend that you specify IP addresses in at least two Availability Zones. You can optionally specify additional IP addresses in those or other Availability Zones.

IP addresses and Amazon VPC elastic network interfaces

For each combination of Availability Zone, Subnet, and IP address that you specify, VPC Resolver creates an Amazon VPC elastic network interface. For the current maximum number of DNS queries per second per IP address in an endpoint, see Quotas on Route 53 VPC Resolver. For information about pricing for each elastic network interface, see "Amazon Route 53" on the Amazon Route 53 pricing page.

Order of IP addresses

You can specify IP addresses in any order. When forwarding DNS queries, VPC Resolver doesn't choose IP addresses based on the order that the IP addresses are listed in.

For each IP address, specify the following values. Each IP address must be in an Availability Zone in the VPC that you specified in **VPC in the *region-name* Region**.

Availability Zone

The Availability Zone that you want DNS queries to pass through on the way to your network. The Availability Zone that you specify must be configured with a subnet.

Subnet

The subnet that contains the IP address that you want DNS queries to originate from on the way to your network. The subnet must have an available IP address.

The subnet IP address must match the **Endpoint type**.

We recommend using VPC Resolver on AWS Outposts to create endpoints on AWS Outposts Racks.

> **Important**
>
> Outposts subnets with Local Network Interface (LNI) enabled are not compatible with VPC Resolver endpoints. If you enable LNI on a subnet that contains VPC Resolver endpoint elastic network interfaces (ENIs), those ENIs will stop functioning. For more information, see Subnet compatibility for Resolver endpoints.

IP address

The IP address that you want DNS queries to originate from on the way to your network.

Choose whether you want VPC Resolver to choose an IP address for you from among the available IP addresses in the specified subnet, or you want to specify the IP address yourself.

If you choose to specify the IP address yourself, enter an IPv4 or IPv6 address, or both.

Protocols

Endpoint protocol determines how data is transmitted from the outbound endpoint. Choose a protocol, or protocols, depending on the level of security needed.

- **Do53:** (Default) The data is relayed using the Route 53 VPC Resolver without additional encryption. While the data cannot be read by external parties, it can be viewed within the AWS networks.
- **DoH:** The data is transmitted over an encrypted HTTPS session. DoH adds an added level of security where data can't be decrypted by unauthorized users, and can't be read by anyone except the intended recipient.

For an outbound endpoint you can apply the protocols as follows:

- Do53 and DoH in combination.
- Do53 alone.
- DoH alone.
- None, which is treated as Do53.

IPv6 internet access

Enable IPv6 internet access to allow the outbound endpoint to forward DNS queries to public IPv6 targets through an internet gateway. When enabled, the endpoint elastic network interfaces (ENIs) can send DNS queries to public IPv6 resolvers.

> **Important**
>
> When you enable IPv6 internet access, use network controls like security groups, NACLs, or egress-only internet gateways to protect the endpoint ENIs from unsolicited ingress traffic. Be aware that some network controls can affect DNS query throughput due to connection tracking. For more information, see Amazon EC2 security group connection tracking and Resolver endpoint scaling.

Tags

Specify one or more keys and the corresponding values. For example, you might specify **Cost center** for **Key** and specify **456** for **Value**.

Values that you specify when you create or edit rules

When you create or edit a forwarding rule, you specify the following values:

Rule name

A friendly name that lets you easily find a rule on the dashboard.

Rule type

Choose the applicable value:

- **Forward** – Choose this option when you want to forward DNS queries for a specified domain name to resolvers on your network.
- **Delegate** – Choose this option when you want to delegate authority for a subdomain, hosted in a private hosted zone, to your on-premises resolver (or to a VPC Resolver on another VPC).
- **System** – Choose this option when you want VPC Resolver to selectively override the behavior that is defined in a forwarding rule. When you create a system rule, VPC Resolver resolves DNS queries for specified subdomains that would otherwise be resolved by DNS resolvers on your network.

By default, forwarding rules apply to a domain name and all its subdomains. If you want to forward queries for a domain to a resolver on your network but you don't want to forward queries for some subdomains, you create a system rule for the subdomains. For example, if you create a forwarding rule for example.com but you don't want to forward queries for acme.example.com, you create a system rule and specify acme.example.com for the domain name.

Delegation record – for delegate rule only

DNS queries that mach to this domain are forwarded to the resolvers on your network.

> **Note**
>
> As a prerequisite for a delegate rule you must create NS records in the private hosted zone, when using private hosted zone to outbound delegation The record is: NS - Nameservers to delegate through the Resolver outbound endpoint with delegate rule. For more information, see NS record type.

VPCs that use this rule

The VPCs that use this rule to forward DNS queries for the specified domain name or names. You can apply a rule to as many VPCs as you want.

Domain name – for forward rule only

DNS queries for this domain name are forwarded to the IP addresses that you specify in **Target IP addresses**. For example, you can specify a specific domain (example.com), a top-level domain (com), or a dot (.) to forward all DNS queries. For more information, see How VPC Resolver determines whether the domain name in a query matches any rules.

Outbound endpoint

VPC Resolver forwards DNS queries through the outbound endpoint that you specify here to the IP addresses that you specify in **Target IP addresses**.

Target IP addresses

When a DNS query matches the name that you specify in **Domain name**, the outbound endpoint forwards the query to the IP addresses that you specify here. These are typically the IP addresses for DNS resolvers on your network.

Target IP addresses is available only when the value of **Rule type** is **Forward**.

Specify IPv4 or IPv6 addresses, the protocols, and ServerNameIndication you want to use for the endpoint. ServerNameIndication is applicable only when selected protocol is DoH.

Resolving the target IP address of the FQDN of a DoH resolver on your network over the outbound endpoint is not supported. Outbound endpoints need the target IP address of DoH resolver on your network to forward the DoH queries to. If the DoH resolver on your network needs the FQDN in the TLS SNI and in the HTTP Host header, ServerNameIndication must be provided.

ServerNameIndication

The Server Name Indication of the DoH server that you want to forward queries to. This is only used if the Protocol is DoH.

Tags

Specify one or more keys and the corresponding values. For example, you might specify **Cost center** for **Key** and specify **456** for **Value**.

These are the tags that AWS Billing and Cost Management provides for organizing your AWS bill. For more information about using tags for cost allocation, see Using cost allocation tags in the *AWS Billing User Guide*.

Managing outbound endpoints

To manage outbound endpoints, perform the applicable procedure.

Topics

- Viewing and editing outbound endpoints
- Viewing the status for outbound endpoints
- Deleting outbound endpoints

Viewing and editing outbound endpoints

To view and edit settings for an outbound endpoint, perform the following procedure.

To view and edit settings for an outbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Outbound endpoints**.
3. On the navigation bar, choose the Region where you created the outbound endpoint.
4. Choose the option for the endpoint that you want to view settings for or want to edit.
5. Choose **View details** or **Edit**.

   For information about the values for outbound endpoints, see Values that you specify when you create or edit outbound endpoints.
6. If you chose **Edit**, enter the applicable values, and then choose **Save**.

Viewing the status for outbound endpoints

To view the status for an outbound endpoint, perform the following procedure.

To view the status for an outbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Outbound endpoints**.
3. On the navigation bar, choose the Region where you created the outbound endpoint. The **Status** column contains one of the following values:

Creating

VPC Resolver is creating and configuring one or more Amazon VPC network interfaces for this endpoint.

Operational

The Amazon VPC network interfaces for this endpoint are correctly configured and able to pass inbound or outbound DNS queries between your network and VPC Resolver.

Updating

VPC Resolver is associating or disassociating one or more network interfaces with this endpoint.

Auto recovering

VPC Resolver is trying to recover one or more of the network interfaces that are associated with this endpoint. During the recovery process, the endpoint functions with limited capacity because of the limit on the number of DNS queries per IP address (per network interface). For the current limit, see Quotas on Route 53 VPC Resolver.

Action needed

This endpoint is unhealthy, and VPC Resolver can't automatically recover it. To resolve the problem, we recommend that you check each IP address that you associated with the endpoint. For each IP address that isn't available, add another IP address and then delete the IP address that isn't available. (An endpoint must always include at least two IP addresses.) A status of **Action needed** can have a variety of causes. Here are two common causes:

- One or more of the network interfaces that are associated with the endpoint were deleted using Amazon VPC.
- The network interface couldn't be created for some reason that's outside the control of VPC Resolver.

Deleting

VPC Resolver is deleting this endpoint and the associated network interfaces.

Deleting outbound endpoints

Before you can delete an endpoint, you must first delete any rules that are associated to a VPC.

To delete an outbound endpoint, perform the following procedure.

> **Important**
>
> If you delete an outbound endpoint, VPC Resolver stops forwarding DNS queries from your VPC to your network for rules that specify the deleted outbound endpoint.

To delete an outbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Outbound endpoints**.
3. On the navigation bar, choose the Region where you created the outbound endpoint.
4. Choose the option for the endpoint that you want to delete.
5. Choose **Delete**.
6. To confirm that you want to delete the endpoint, enter the name of the endpoint, and then choose **Submit**.

Managing forwarding rules

If you want VPC Resolver to forward queries for specified domain names to your network, you create one forwarding rule for each domain name and specify the name of the domain for which you want to forward queries.

Topics

- Viewing and editing forwarding rules
- Creating forwarding rules
- Adding rules for reverse lookup
- Associating forwarding rules with a VPC
- Disassociating forwarding rules from a VPC

- Sharing Resolver rules with other AWS accounts and using shared rules
- Deleting forwarding rules
- Forwarding rules for reverse DNS queries in VPC Resolver

Viewing and editing forwarding rules

To view and edit settings for a forwarding rule, perform the following procedure.

To view and edit settings for a forwarding rule

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Rules**.
3. On the navigation bar, choose the Region where you created the rule.
4. Choose the option for the rule that you want to view settings for or want to edit.
5. Choose **View details** or **Edit**.

   For information about the values for forwarding rules, see Values that you specify when you create or edit rules.
6. If you chose **Edit**, enter the applicable values, and then choose **Save**.

Creating forwarding rules

To create one or more forwarding rules, perform the following procedure.

To create forwarding rules and associate the rules with one or more VPCs

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Rules**.
3. On the navigation bar, choose the Region where you want to create the rule.
4. Choose **Create rule**.
5. Enter the applicable values. For more information, see Values that you specify when you create or edit rules.
6. Choose **Save**.

7. To add another rule, repeat steps 4 through 6.

Adding rules for reverse lookup

If you need to control reverse lookups in your VPC, you can add rules to your outbound resolver endpoint.

To create the reverse lookup rule

1. Follow the steps in the previous procedure, up to step 5.
2. When you specify your rule, enter the PTR record for the IP address or addresses that you want a reverse lookup forwarding rule for.

   For example, if you need to forward lookups for addresses in the 10.0.0.0/23 range, enter two rules:

   - 0.0.10.in-addr.arpa
   - 1.0.10.in-addr.arpa

   Any IP address in those subnets will be referenced as a subdomain of those PTR records—for example, 10.0.1.161 will have a reverse lookup address of 161.1.0.10.in-addr.arpa, which is a subdomain of 1.0.10.in-addra.arpa.
3. Specify the server to forward these lookups to.
4. Add these rules to your outbound resolver endpoint.

Note that turning on `enableDNSHostNames` for your VPC automatically adds PTR records. See What is Route 53 VPC Resolver?. The previous procedure is required only if you want to explicitly specify a resolver for given IP ranges—for example, when forwarding queries to an Active Directory server.

Associating forwarding rules with a VPC

After you create a forwarding rule, you must associate the rule with one or more VPCs. The rules will only work after they are associated with a VPC. When you associate a rule with a VPC, VPC Resolver starts to forward DNS queries for the domain name that's specified in the rule to the DNS resolvers that you specified in the rule. The queries pass through the outbound endpoint that you specified when you created the rule.

To associate a forwarding rule with one or more VPCs

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Rules**.
3. On the navigation bar, choose the Region where you created the rule.
4. Choose the name of the rule that you want to associate with one or more VPCs.
5. Choose **Associate VPC**.
6. Under **VPCs that use this rule**, choose the VPCs that you want to associate the rule with.
7. Choose **Add**.

Disassociating forwarding rules from a VPC

You disassociate a forwarding rule from a VPC in the following circumstances:

- For DNS queries that originate in this VPC, you want VPC Resolver to stop forwarding queries for the domain name specified in the rule to your network.
- You want to delete the forwarding rule. If a rule is currently associated with one or more VPCs, you must disassociate the rule from all VPCs before you can delete it.

If you want to disassociate a forwarding rule from one or more VPCs, perform the following procedure.

To disassociate a forwarding rule from a VPC

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Rules**.
3. On the navigation bar, choose the Region where you created the rule.
4. Choose the name of the rule that you want to disassociate from one or more VPCs.
5. Choose the option for the VPC that you want to disassociate the rule from.
6. Choose **Disassociate**.
7. Type **disassociate** to confirm.
8. Choose **Submit**.

Sharing Resolver rules with other AWS accounts and using shared rules

You can share the Resolver rules that you created using one AWS account with other AWS accounts. To share rules, the Route 53 VPC Resolver console integrates with AWS Resource Access Manager. For more information about Resource Access Manager, see the Resource Access Manager User Guide.

Note the following:

Associating shared rules with VPCs

If another AWS account has shared one or more rules with your account, you can associate the rules with your VPCs the same way that you associate rules that you created with your VPCs. For more information, see Associating forwarding rules with a VPC.

Deleting or unsharing a rule

If you share a rule with other accounts and then either delete the rule or stop sharing it, and if the rule was associated with one or more VPCs, Route 53 VPC Resolver starts to process DNS queries for those VPCs based on the remaining rules. The behavior is the same as if you disassociate the rule from the VPC.

If a rule is shared to an Organizational Unit (OU) and an account in the OU is moved to a different OU, all associations with the shared rule to any VPC in the account will be deleted. However, if the VPC Resolver rule was already shared with destination OU, then the VPC association will stay intact and will not be dissociated.

Maximum number of rules and associations

When an account creates a rule and shares it with one or more other accounts, the maximum number of rules per AWS Region applies to the account that created the rule.

When an account that a rule is shared with associates the rule with one or more VPCs, the maximum number of associations between rules and VPCs per Region applies to the account that the rule is shared with.

For current VPC Resolver quotas, see Quotas on Route 53 VPC Resolver.

Permissions

To share a rule with another AWS account, you must have permission to use the PutResolverRulePolicy action.

Restrictions on the AWS account that a rule is shared with

The account that a rule is shared with can't change or delete the rule.

Tagging

Only the account that created a rule can add, delete, or see tags on the rule.

To view the current sharing status of a rule (including the account that shared the rule or the account that a rule is shared with), and to share rules with another account, perform the following procedure.

To view sharing status and share rules with another AWS account

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Rules**.
3. On the navigation bar, choose the Region where you created the rule.

   The **Sharing status** column shows the current sharing status of rules that were created by the current account or that are shared with the current account:

   - **Not shared**: The current AWS account created the rule, and the rule is not shared with any other accounts.
   - **Shared by me**: The current account created the rule and shared it with one or more accounts.
   - **Shared with me**: Another account created the rule and shared it with the current account.
4. Choose the name of the rule that you want to display sharing information for or that you want to share with another account.

   On the **Rule: *`rule name`*** page, the value under **Owner** displays ID of the account that created the rule. That's the current account unless the value of **Sharing status** is **Shared with me**. In that case, **Owner** is the account that created the rule and shared it with the current account.
5. Choose **Share** to view additional information or to share the rule with another account. A page in the Resource Access Manager console appears, depending on the value of **Sharing status**:

   - **Not shared**: The **Create resource share** page appears. For information about how to share the rule with another account, OU, or organization, skip to step 6.

- **Shared by me**: The **Shared resources** page shows the rules and other resources that are owned by the current account and shared with other accounts.
- **Shared with me**: The **Shared resources** page shows the rules and other resources that are owned by other accounts and shared with the current account.

6. To share a rule with another AWS account, OU, or organization, specify the following values.

> **Note**
>
> You can't update sharing settings. If you want to change any of the following settings, you must reshare a rule with the new settings and then remove the old sharing settings.

Description

Enter a short description that helps you remember why you shared the rule.

Resources

Choose the check box for the rule that you want to share.

Principals

Enter the AWS account number, OU name, or organization name.

Tags

Specify one or more keys and the corresponding values. For example, you might specify **Cost center** for **Key** and specify **456** for **Value**.

These are the tags that AWS Billing and Cost Management provides for organizing your AWS bill; you can use also tags for other purposes. For more information about using tags for cost allocation, see Using cost allocation tags in the *AWS Billing User Guide*.

Deleting forwarding rules

To delete a forwarding rule, perform the following procedure.

Note the following:

- If the forwarding rule is associated with any VPCs, you must disassociate the rule from the VPCs before you can delete the rule. For more information, see Disassociating forwarding rules from a VPC.
- You can't delete the default **Internet Resolver** rule, which has a value of **Recursive** for **Type**. This rule causes Route 53 VPC Resolver to act as a recursive resolver for any domain names that you didn't create custom rules for and that VPC Resolver didn't create autodefined rules for. For more information about how rules are categorized, see Using rules to control which queries are forwarded to your network.

To delete a forwarding rule

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Rules**.
3. On the navigation bar, choose the Region where you created the rule.
4. Choose the option for the rule that you want to delete.
5. Choose **Delete**.
6. To confirm that you want to delete the rule, enter the name of the rule and choose **Submit**.

Forwarding rules for reverse DNS queries in VPC Resolver

When the `enableDnsHostnames` and `enableDnsSupport` are set to `true` for a virtual private cloud (VPC) from Amazon VPC, VPC Resolver automatically creates auto-defined system rules for reverse DNS queries. For more information about these settings, see DNS attributes in your VPC in the *Amazon VPC Developer Guide*.

Forwarding rules for reverse DNS queries are particularly useful for services like SSH or Active Directory, which have an option to authenticate users by performing a reverse DNS lookup for the IP address from which a customer is attempting to connect to a resource. For more information about auto-defined system rules, see Domain names that VPC Resolver creates autodefined system rules for.

You can turn off these rules and modify all reverse DNS queries so that they are, for example, forwarded to your on-premises name servers for resolution.

After you turn off the automatic rules, create rules to forward the queries as needed to your on-premises resources. For more information about how to manage forwarding rules, see Managing forwarding rules.

To turn off auto-defined rules

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, under **VPC Resolver** choose **VPCs**, and then choose a VPC ID.
3. Under **Autodefined rules for reverse DNS resolution**, deselect the check box. If the check box is already deselected, you can select it to turn on auto-defined reverse DNS resolution.

For the related APIs, see VPC Resolver configuration APIs.

Resolver delegation rules tutorial

The delegation rule allows the VPC Resolver to reach the name servers that host the delegated zone through the specified outbound endpoint. While the forward rule informs the VPC Resolver to forward the DNS queries to the name servers that match the specified domain through outbound endpoint, the delegation rule informs VPC Resolver to reach the delegated name servers through the outbound endpoint specified when the delegated NS records are returned. VPC Resolver sends a query to the delegated name servers when the NS records in the DNS response match the domain name specified in the delegation record.

Steps to Use the Resolver endpoint outbound delegation

1. Create a Resolver outbound endpoint in the VPC that you want DNS queries to originate from to the resolvers on your network.

   You can use either the following API or the CLI command:

   - `CreateResolverEndpoint` API
   - `create-resolver-endpoint` CLI
2. Create one or more delegation rules, which specify the domain names for which the queries should be delegated to your network through the specified outbound endpoint.

   Example delegation rule creation with CLI:

```
aws route53resolver create-resolver-rule \
--region REGION \
--creator-request-id delegateruletest \
--delegation-record example.com \
--name delegateruletest \
--rule-type DELEGATE \
--resolver-endpoint-id outbound endpoint ID
```

3. Associate the delegation rule with the VPCs for which you want the queries to be delegated.

   You can use either the following API or the CLI command:

   - AssociateResolverRule API.
   - associate-resolver-rules CLI command.

Types of delegation supported by Resolver outbound endpoint

VPC Resolver supports two types of outbound delegation:

- Route 53 private hosted zone to VPC Resolver outbound delegation:

  Delegates a sub-domain from a private hosted zone to either to on-premises DNS server or a public hosted zone on the internet using outbound delegation. This outbound delegation allows you to split management of your DNS records between the private hosted zone and the delegated zone. The delegation can be done with or without glue record in private hosted zone based on your DNS setup.For more information see Private hosted zone to outbound.

- VPC Resolver outbound to outbound delegation:

  Delegates a subdomain from your on-premises DNS server to another on-premises server in a same or different location using outbound to outbound delegation. This is similar to delegating from a private hosted zone to an outbound endpoint, where you can delegate to zone hosted on your on-premises name server. For more information see Outbound to outbound.

Route 53 private hosted zone to VPC Resolver outbound delegation example configuration

Let's assume a DNS setup where the parent hosted zone is hosted in an Route 53 private hosted zone in an Amazon VPC and subdomains are being delegated to name servers hosted in Europe, Asia and North America. All the DNS queries are passed thorough the VPC Resolver.

Follow the example steps to configure your private hosted zone and VPC Resolver.

Configure private hosted zone for outbound delegation

1. For the private hosted zone setup:

   Parent hosted zone: `hr.example.com`

   ```
   $TTL    86400 ; 24 hours
   $ORIGIN hr.example.com
   @  1D  IN     SOA @    root (20200322001 3H 15 1w 3h)
   @  1D  IN  NS @
   eu.hr.example.com IN NS ns1.eu.hr.example.com.
   apac.hr.example.com IN NS ns2.apac.hr.example.com.
   na.hr.example.com IN NS ns3.na.hr.example.net. # Out of Zone Delegation

   ns1.eu.hr.example.com IN A 10.0.0.30 # Glue Record
   ns2.apac.hr.example.com IN A 10.0.0.40 # Glue Record
   ```

2. For the on-premises name server in the Europe on-premises region:

   - Hosted Zone: `eu.hr.example.com` NS IP: `10.0.0.30`

   ```
   $TTL    86400 ; 24 hours
   $ORIGIN eu.hr.example.com
   @  1D  IN     SOA @    root (20200322001 3H 15 1w 3h)
   @  1D  IN  NS @

   test.eu.hr.example.com IN A 1.2.3.4
   ```

3. For the on-premises name server in the Asia on-premises region:

   Hosted Zone: `apac.hr.example.com`, `10.0.0.40`

   The apac name server can delegate the subdomain to other name servers.

```
$TTL    86400 ; 24 hours
$ORIGIN apac.hr.example.com
@  1D  IN     SOA @    root (20200322001 3H 15 1w 3h)
@  1D  IN  NS @

test.apac.hr.example.com IN A 5.6.7.8
engineering.apac.hr.example.com IN NS ns1.engineering.apac.hr.example.com
sales.apac.hr.example.com IN NS ns2.sales.apac.hr.example.com
ns1.engineering.apac.hr.example.com IN A 10.0.0.80
ns2.sales.apac.hr.example.com IN A 10.0.0.90
```

Hosted Zone: `engineering.apac.hr.example.com`, `10.0.0.80`

```
$TTL 86400 ; 24 hours
$ORIGIN engineering.apac.hr.example.com
@ 1D IN SOA @ root (20200322001 3H 15 1w 3h)
@ 1D IN NS @
test.engineering.apac.hr.example.com IN A 1.1.1.1
```

4. For the on-premises name server in the North America on-premises region:

   Hosted Zone: `na.hr.example.net` NS IP: `10.0.0.50`

```
$TTL    86400 ; 24 hours
$ORIGIN na.hr.example.net
@  1D  IN     SOA @    root (20200322001 3H 15 1w 3h)
@  1D  IN  NS @
ns3.na.hr.example.net. IN A 10.0.0.50
test.na.hr.example.com  IN A 9.10.11.12
```

VPC Resolver setup

- For the VPC Resolver you need to set up one forward rule and two delegation rules:

  Forward rule

  1. For forwarding the out-of-zone delegation record, so the Route 53 VPC Resolver knows the IP of the delegated NS to forward the initial request.

     domain-name: `hr.example.net` target-ips: `10.0.0.50`

Delegation rules

1. Delegation rule for in-zone delegation:

   delegation-record: `hr.example.com`

2. Delegation rule for out-of zone delegation:

   delegation-record: `hr.example.net`

Outbound to outbound delegation example configuration

Instead of having the parent hosted zone in an Amazon VPC, let's assume a DNS setup where the parent hosted zone is in the central on-premises location and subdomains are being delegated to name servers hosted in Europe, Asia, and North America. All the DNS queries are passed thorough the VPC Resolver.

Follow the example steps to configure your on-premises DNS and the VPC Resolver.

Configure on-premise DNS

1. For the on-premises name server in the central on-premises region:

   - **Parent hosted zone:** `hr.example.com`

     Hosted Zone `hr.example.com`, NS IP: `10.0.0.20`

```
$TTL    86400 ; 24 hours
$ORIGIN hr.example.com
@  1D  IN     SOA @    root (20200322001 3H 15 1w 3h)
@  1D  IN  NS @
eu.hr.example.com IN NS ns1.eu.hr.example.com.
apac.hr.example.com IN NS ns2.apac.hr.example.com.
na.hr.example.com IN NS ns3.na.hr.example.net. # Out of zone delegation

ns1.eu.hr.example.com IN A 10.0.0.30 # Glue record
ns2.apac.hr.example.com IN A 10.0.0.40 # Glue record
```

2. For the on-premises name server in the Europe on-premises region (the configuration for the Europe, Asia, and North America name servers is the same as for the private hosted zone to outbound delegation):

   - Hosted Zone: `eu.hr.example.com` NS IP: `10.0.0.30`

```
$TTL    86400 ; 24 hours
$ORIGIN eu.hr.example.com
@  1D  IN     SOA @    root (20200322001 3H 15 1w 3h)
@  1D  IN  NS @

test.eu.hr.example.com IN A 1.2.3.4
```

3. For the on-premises name server in the Asia on-premises region:

   Hosted Zone: `apac.hr.example.com`, `10.0.0.40`

   The apac name server can delegate the subdomain to other name servers.

```
$TTL    86400 ; 24 hours
$ORIGIN apac.hr.example.com
@  1D  IN     SOA @    root (20200322001 3H 15 1w 3h)
@  1D  IN  NS @

test.apac.hr.example.com IN A 5.6.7.8
engineering.apac.hr.example.com IN NS ns1.engineering.apac.hr.example.com
sales.apac.hr.example.com IN NS ns2.sales.apac.hr.example.com
ns1.engineering.apac.hr.example.com IN A 10.0.0.80
ns2.sales.apac.hr.example.com IN A 10.0.0.90
```

   Hosted Zone: `engineering.apac.hr.example.com`, `10.0.0.80`

```
$TTL 86400 ; 24 hours
$ORIGIN engineering.apac.hr.example.com
@ 1D IN SOA @ root (20200322001 3H 15 1w 3h)
@ 1D IN NS @
test.engineering.apac.hr.example.com IN A 1.1.1.1
```

4. For the on-premises name server in the North America on-premises region:

   Hosted Zone: `na.hr.example.net` NS IP: `10.0.0.50`

```
$TTL    86400 ; 24 hours
$ORIGIN na.hr.example.net
@  1D  IN     SOA @    root (20200322001 3H 15 1w 3h)
@  1D  IN  NS @
ns3.na.hr.example.net. IN A 10.0.0.50
test.na.hr.example.com  IN A 9.10.11.12
```

VPC Resolver setup

- For the VPC Resolver you need to set up forward rules and delegation rules:

  Forward rules

  1. For forwarding the initial request so the queries will be forwarded to the parent hosted zone `hr.example.com` in the central location:

     domain-name: `hr.example.com` target-ips: `10.0.0.20`
  2. For forwarding the out-of-zone delegation record, so the VPC Resolver knows the IP address of the delegated name server to forward the initial request:

     domain-name: `hr.example.net` target-ips: `10.0.0.50`

  Delegation rules

  1. Delegation rule for in-zone delegation:

     delegation record: `hr.example.com`
  2. Delegation Rule for out-of zone delegation:

     delegation-record: `hr.example.net`

Enabling DNSSEC validation in Amazon Route 53

When you enable DNSSEC validation for a virtual private cloud (VPC) in Amazon Route 53, DNSSEC signatures are cryptographically checked to make sure that the response was not tampered with. You enable DNSSEC validation on your VPC detail page.

DNSSEC validation is applied by VPC Resolver to public signed names when it is performing recursive DNS resolution.

However, if the VPC Resolver is forwarding to another DNS resolver, that resolver is performing recursive DNS resolution and, therefore, must also apply the DNSSEC validation.

Important

Enabling DNSSEC validation can impact DNS resolution for public DNS records from AWS resources in a VPC, which could result in an outage. Be aware that enabling or disabling DNSSEC validation can take several minutes.

Note

At this time, the Route 53 VPC Resolver in your VPC (aka AmazonProvidedDNS) ignores the DO (DNSSEC OK) EDNS header bit and the CD (Checking Disabled) bit in the DNS query. If you have configured DNSSEC, this means that while the VPC Resolver does perform DNSSEC validation, it doesn't return DNSSEC records nor set the AD bit in the response. Therefore, performing your own DNSSEC validation is not currently supported by the VPC Resolver. If you need to do this you will have to perform your own recursive DNS resolution.

To enable DNSSEC validation for a VPC

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, under **VPC Resolver**, choose **VPCs**.
3. Under **DNSSEC validation**, select the check box. If the check box is already selected, you can clear it to disable DNSSEC validation.

   Be aware that enabling or disabling DNSSEC validation can take several minutes.

Routing internet traffic to your AWS resources

You can use Amazon Route 53 to route traffic to a variety of AWS resources.

- Routing traffic to an Amazon API Gateway API by using your domain name
- Routing traffic to an Amazon CloudFront distribution by using your domain name
- Routing traffic to an Amazon EC2 instance
- Routing traffic to an AWS App Runner service
- Routing traffic to an AWS Global Accelerator
- Routing traffic to an AWS Elastic Beanstalk environment
- Routing traffic to an ELB load balancer
- Routing traffic to a website that is hosted in an Amazon S3 bucket
- Routing traffic to an Amazon Virtual Private Cloud interface endpoint by using your domain name
- Routing traffic to Amazon WorkMail
- Routing traffic to Amazon OpenSearch Service domain endpoint
- Routing traffic to Amazon VPC Lattice service domain endpoint
- Routing traffic to other AWS resources
- Creating Amazon Route 53 and Route 53 VPC Resolver resources with AWS CloudFormation

Routing traffic to an Amazon API Gateway API by using your domain name

You can use Amazon API Gateway to create, publish, maintain, monitor, and secure APIs. You can create APIs that access AWS services or other web services in addition to data stored in the AWS Cloud.

The method that you use to route domain traffic to an API Gateway API is the same regardless of whether you created a Regional API Gateway endpoint or an edge-optimized API Gateway endpoint. If you create a private API Gateway endpoint, the process is slightly different.

- **Regional API endpoint**: You create a Route 53 alias record that routes traffic to the Regional API endpoint.

- **Edge-optimized API endpoint**: You create a Route 53 alias record that routes traffic to the edge-optimized API. This causes traffic to be routed to the CloudFront distribution that's associated with the edge-optimized API.
- **Private API endpoint**: You create a Route 53 alias record that routes traffic to your private API endpoint using an interface VPC endpoint for API Gateway in a private hosted zone.

An alias record is a Route 53 extension to DNS that's similar to a CNAME record. For a comparison of alias and CNAME records, see Choosing between alias and non-alias records.

> **Note**
>
> Route 53 doesn't charge for alias queries to API Gateway APIs or other AWS resources.

Topics

- Prerequisites
- Configuring Route 53 to route traffic to an API Gateway endpoint

Prerequisites

To get started, you need the following:

- An API Gateway API that has a custom domain name, such as api.example.com that matches the name of the Route 53 record that you want to create.

  For more information, see the following topics:

  - Setting up custom domain names for HTTP APIs in the *Amazon API Gateway Developer Guide*.
  - Setting up custom domain names for REST APIs in the *Amazon API Gateway Developer Guide*.
  - Setting up custom domain names for WebSocket APIs in the *Amazon API Gateway Developer Guide*.
  - Custom domain names for private APIs in API Gateway in the *Amazon API Gateway Developer Guide*.
- A registered domain name. You can use Amazon Route 53 as your domain registrar or you can use a different registrar.

- Route 53 as the DNS service for the domain. If you register your domain name by using Route 53, we automatically configure Route 53 as the DNS service for the domain.

  For information about using Route 53 as the DNS service provider for your domain, see Making Amazon Route 53 the DNS service for an existing domain.

Configuring Route 53 to route traffic to an API Gateway endpoint

To configure Route 53 to route traffic to an API Gateway endpoint, perform the following procedure.

Custom domain names for public APIs

The following procedure describes how to route traffic to an API Gateway endpoint for a custom domain name for public APIs.

To route traffic to an API Gateway endpoint

1. If you created the Route 53 hosted zone and the endpoint using the same account, skip to step 2.

   If you created the hosted zone and the endpoint using different accounts, get the target domain name for the custom domain name that you want to use:

   a. Sign in to the AWS Management Console and open the API Gateway console at https://console.aws.amazon.com/apigateway/.

   b. In the navigation pane, choose **Custom domain names**.

   c. Select the custom domain name that you want to use and get the value of **API Gateway domain name**.

2. Open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Hosted zones**.
4. Choose the name of the hosted zone that has the domain name that you want to use to route traffic to your API.
5. Choose **Create record**.
6. Specify the following values:

> **Important**
>
> We recommend that you turn on Alias. For domain names that don't use a Route 53 Alias record, you might encounter issues if you use a VPC with private DNS enabled to invoke a private API. Private DNS overrides the default DNS resolution behavior within the VPC, which might cause conflicts with external DNS records.

Routing policy

Choose the applicable routing policy. For more information, see Choosing a routing policy.

Record name

Enter the domain name that you want to use to route traffic to your API.

The API that you want to route traffic to must include a custom domain name, such as api.example.com that matches the name of the Route 53 record.

Alias

If you are using the **Quick create** record creation method, turn on **Alias**.

Value/Route traffic to

Choose **Alias to API Gateway API**, then choose the Region that the endpoint is from.

How you specify the value for **Endpoint** depends on whether you created the hosted zone and the API using the same AWS account or different accounts:

- **Same account** – The list of target domain names includes only APIs that have a custom domain name that matches the value that you specified for **Record name**. Choose the applicable value.
- **Different accounts** – Enter the value that you got in step 1 of this procedure.

Record type

Choose **A – IPv4 address**.

Evaluate target health

For control over DNS failover, configure custom health checks. For an example, see Configure custom health checks for DNS failover in the *API Gateway user guide*.

7. Choose **Create records**.

   Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your API by using the name of the alias record that you created in this procedure.

Custom domain names for private APIs

The following procedure describes how to route traffic to an API Gateway endpoint for a custom domain name for private APIs.

To route traffic to an API Gateway endpoint

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the private hosted zone that has the domain name that you want to use to route traffic to your API.
4. Choose **Create record**.
5. Specify the following values:

   Routing policy

   Choose the applicable routing policy. For more information, see Choosing a routing policy.

   Record name

   Enter the domain name that you want to use to route traffic to your API.

   The API that you want to route traffic to must include a custom domain name, such as api.example.com that matches the name of the Route 53 record.

   Alias

   Turn on **Alias**.

Value/Route traffic to

Choose **Alias to VPC Endpoint**. Choose the Region that the endpoint is from, and then select your VPC endpoint.

Record type

If you are using IPv6 for your VPC endpoint, create an AAAA record type. If you are using dualstack for your VPC endpoint, create both an AAAA and an A record type.

Evaluate target health

For control over DNS failover, configure custom health checks. For an example, see Configure custom health checks for DNS failover in the *API Gateway user guide*.

6. Choose **Create records**.

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your API by using the name of the alias record that you created in this procedure.

Routing traffic to an Amazon CloudFront distribution by using your domain name

This topic provides comprehensive procedures for routing DNS traffic to any Amazon CloudFront distribution. If you're setting up a static website with Amazon CloudFront and Amazon Simple Storage Service, see Use an Amazon CloudFront distribution to serve a static website for a complete tutorial.

You can use Amazon CloudFront, the AWS content delivery network (CDN), as one way to speed up delivery of your web content. CloudFront can deliver your entire website—including dynamic, static, streaming, and interactive content—by using a global network of edge locations. Users who request your content are automatically routed to the edge location that gives them the lowest latency.

> **Note**
>
> You can route traffic to a CloudFront distribution only for public hosted zones.

To use CloudFront to distribute your website content, create a distribution and specify settings for it. For example, specify the Amazon S3 bucket or HTTP server that you want CloudFront to get your content from, whether you want only selected users to have access to your content, and whether you want users to use HTTPS.

When you create a distribution, CloudFront assigns a domain name to the distribution, such as d111111abcdef8.cloudfront.net. You can use this domain name in the URLs for your content, for example:

`http://d111111abcdef8.cloudfront.net/logo.jpg`

Alternatively, you can use your own domain name in URLs, for example:

`http://example.com/logo.jpg`

Follow the steps in the Amazon CloudFront Developer Guide to use your own domain name in your files' URLs in a CloudFront distribution, instead of the domain name that CloudFront assigns to your distribution. For more information, about using your own domain name with a CloudFront distribution, see Using custom URLs by adding alternate domain names (CNAMEs).

When you use a Route 53 domain name with a CloudFront distribution, use Amazon Route 53 to create an alias record that points to your CloudFront distribution. An alias record is a Route 53 extension to DNS. It's similar to a CNAME record, but you can create an alias record both for the root domain, such as example.com, and for subdomains, such as www.example.com. (You can create CNAME records only for subdomains.) When Route 53 receives a DNS query that matches the name and type of an alias record, Route 53 responds with the domain name that is associated with your distribution.

> **Note**
>
> Route 53 doesn't charge for alias queries to CloudFront distributions or other AWS resources.

Prerequisites

To get started, you need the following:

1. A registered domain name. You can use Amazon Route 53 as your domain registrar or you can use a different registrar.

2. Route 53 as the DNS service for the domain. If you register your domain name by using Route 53, we automatically configure Route 53 as the DNS service for the domain.

   For information about using Route 53 as the DNS service provider for your domain, see Making Amazon Route 53 the DNS service for an existing domain.

3. A CloudFront distribution or a CloudFront distribution tenant. The distribution must include an alternate domain name that matches the domain name that you want to use for your URLs instead of the domain name that CloudFront assigned to your distribution. For a CloudFront distribution tenant, it must contain the domain name that you want to use for your URLs.

   For example, if you want the URLs for your content to contain the domain name **example.com**, the **Alternate Domain Name** field for the distribution must include **example.com**.

   For more information, see the following documentation in the *Amazon CloudFront Developer Guide*:

   - Task list for creating a distribution
   - Creating or updating a distribution using the CloudFront console

   > **Note**
   >
   > If you're creating a static website, see Get started with a secure static website in the *Amazon CloudFront Developer Guide* for complete setup instructions.

4. (Optional) Request a public certificate so that Amazon CloudFront distributions require HTTPS. For more information, see DNS validation in the AWS Certificate Manager in the *AWS Certificate Manager User Guide.*

Configuring Amazon Route 53 to route traffic to a CloudFront distribution

To configure Amazon Route 53 to route traffic to a CloudFront distribution, follow these steps. For more information about using your own domain name with a CloudFront distribution, see Using custom URLs by adding alternate domain names (CNAMEs) in the Amazon CloudFront Developer Guide.

> **Note**
>
> Changes generally propagate to all Route 53 servers within 60 seconds. When the changes propagate, you'll be able to route traffic to your CloudFront distribution by using the name of the alias record that you create in this procedure.

To route traffic to a CloudFront distribution

1. Get the domain name that CloudFront assigned to your distribution and determine whether IPv6 is enabled:

   a. Sign in to the AWS Management Console and open the CloudFront console at https://console.aws.amazon.com/cloudfront/v4/home.

   b. in the **ID** column, select the linked name of the distribution that you want to route traffic to (not the check box).

   c. On the **General** tab, get the value of the **Distribution domain name** field.

   d. On the **General** tab, in the **Settings** section, choose edit and scroll to check the **IPv6** field to see whether IPv6 is enabled for the distribution. If IPv6 is enabled, you'll need to create two alias records for the distribution, one to route IPv4 traffic to the distribution, and one to route IPv6 traffic. Choose **Cancel**.

      For more information, see Enable IPv6 in the topic Values that you specify when you create or update a distribution in the *Amazon CloudFront Developer Guide*.

2. For a CloudFront distribution tenant,

   a. Choose **SaaS** in the left nav, then **Distribution tenants**, and choose the distribution tenant with the domain name that you want to route traffic to

   b. in the **General details** section, copy the value of the **Endpoint**.

3. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
4. In the navigation pane, choose **Hosted zones**.
5. Choose the linked name of the hosted zone for the domain that you want to use to route traffic to your CloudFront distribution.
6. Choose **Create record**.
7. Specify the following values:

Record name

Enter the domain name that you want to use to route traffic to your CloudFront distribution. The default value is the name of the hosted zone.

For example, if the name of the hosted zone is example.com and you want to use **acme.example.com** to route traffic to your distribution, enter **acme**.

Record type

Choose **A – IPv4 address**.

If IPv6 is enabled for the distribution and you're creating a second record, choose **AAAA – IPv6 address**.

Alias

Turn on **Alias**.

> **Important**
>
> You must create an Alias record for the CloudFront distribution to work.

Route traffic to

Choose **Alias to CloudFront distribution**. Choose the domain name that CloudFront assigned to the distribution when you created it. This is the value that you got in step 1.

For a CloudFront distribution tenant, choose the endpoint from step 2.

Evaluate target health

Accept the default value of **No**.

Routing policy

Choose the applicable routing policy. For more information, see Choosing a routing policy.

8. Choose **Create records**.
9. If IPv6 is enabled for the distribution, repeat steps 5 through 7. Specify the same settings except for the **Record type** field, as explained in step 6.

Routing traffic to an Amazon EC2 instance

Amazon EC2 provides scalable computing capacity in the AWS Cloud. You can launch an EC2 virtual computing environment (an instance) using a preconfigured template (an Amazon Machine Image or AMI). When you launch an EC2 instance, EC2 automatically installs the operating system (Linux or Microsoft Windows) and additional software included in the AMI, such as web server or database software.

You can route traffic for your domain, such as example.com, to your server by using Amazon Route 53, if you're hosting a website or running a web application on an EC2 instance.

Prerequisites

To get started, you need the following:

- An Amazon EC2 instance. For information about launching an EC2 instance, see the following documentation:
  - **Linux** – See Getting started with Amazon EC2 Linux instances in the *Amazon EC2 User Guide*
  - **Microsoft Windows** – See Getting started with Amazon EC2 Windows instances in the *Amazon EC2 User Guide*

  > **Important**
  >
  > We recommend that you also create an Elastic IP address and associate it with your EC2 instance. An Elastic IP address ensures that the IP address of your Amazon EC2 instance will never change. For information related to pricing, see Pricing for Elastic IP addresses.

- A registered domain name. You can use Amazon Route 53 as your domain registrar or you can use a different registrar.
- Route 53 as the DNS service for the domain. If you register your domain name by using Route 53, we automatically configure Route 53 as the DNS service for the domain.

  For information about using Route 53 as the DNS service provider for your domain, see Making Amazon Route 53 the DNS service for an existing domain.

Configuring Amazon Route 53 to route traffic to an Amazon EC2 instance

To configure Amazon Route 53 to route traffic to an EC2 instance, perform the following procedure.

To route traffic to an Amazon EC2 instance

1. Get the IP address for the Amazon EC2 instance:
   a. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
   b. In the Regions list in the upper right corner of the console, choose the Region that you launched the instance in.
   c. In the navigation pane, choose **Instances**.
   d. In the table, choose the instance that you want to route traffic to.
   e. In the bottom pane, on the **Description** tab, get the value of **Elastic IPs**.

      If you didn't associate an Elastic IP with the instance, get the value of **IPv4 Public IP**.
2. Open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Hosted zones**.
4. Choose the name of the hosted zone that matches the name of the domain that you want to route traffic for.
5. Choose **Create record**.
6. Specify the following values:

   Routing policy

   Choose the applicable routing policy. For more information, see Choosing a routing policy.

   Record name

   Enter the domain name that you want to use to route traffic to your EC2 instance. The default value is the name of the hosted zone.

   For example, if the name of the hosted zone is example.com and you want to use acme.example.com to route traffic to your EC2 instance, enter **acme**.

Value/Route traffic to

Choose **IP address or another value depending on the record type**. Enter the IP address that you got in step 1.

Record type

Choose **A – IPv4 address**.

TTL (seconds)

Accept the default value of **300**.

7. Choose **Create records**.

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your EC2 instance by using the name of the record that you created in this procedure.

> **Important**
>
> If you release the elastic IP, make sure you also delete the DNS record pointing to it. If you don't, you will have a dangling DNS record that can be taken over by an unauthorized user.

Routing traffic to an AWS App Runner service

AWS App Runner is a fully managed service that makes it easy for developers to deploy containerized web applications and APIs at scale and with no prior infrastructure experience required. Start with your source code or a container image. App Runner builds and deploys the web application automatically, load balances traffic with encryption, scales to meet your traffic needs, and makes it easy for your services to communicate with other AWS services and applications that run in a private Amazon VPC. With App Runner, rather than thinking about servers or scaling, you have more time to focus on your applications. For more information, see What is AWS App Runner in the *AWS App Runner Developer Guide.*

> **Important**
>
> Amazon Route 53 currently supports alias records for AWS App Runner services that are created after August 1, 2022.

To route domain traffic to an App Runner Service, use Amazon Route 53 to create an alias record that points to your App Runner service. An alias record is a Route 53 extension to DNS. It's similar to a CNAME record, except you can create an alias record both for the root domain, such as example.com, and for subdomains, such as www.example.com (http://www.example.com/). You can create only CNAME records for subdomains.

> **Note**
>
> Route 53 doesn't charge for alias queries to App Runner service or other AWS resources.

Prerequisites

To get started, you need the following:

- An App Runner service. For information about creating an App Runner service, see Getting started with App Runner.
- A registered domain name. You can use Amazon Route 53 as your domain registrar, or you can use a different registrar.
- Route 53 as the DNS service for the domain. If you register your domain name by using Route 53, we automatically configure Route 53 as the DNS service for the domain.

  For information about using Route 53 as the DNS service provider for your domain, see Making Amazon Route 53 the DNS service for an existing domain.
- Associated the custom domain to your App Runner service. For more information, see Managing custom domain names for App Runner.
- Configure the certificate validation record returned by App Runner to you Route 53 hosted zone to start the domain validation process. For more information, see DNS validation in the AWS Certificate Manager in the *AWS Certificate Manager User Guide.*

Configuring Amazon Route 53 to route traffic to an App Runner service

To configure Amazon Route 53 to route traffic to an App Runner service, perform the following procedure.

To route traffic to an App Runner service

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.

2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the hosted zone that matches the name of the domain that you want to route traffic for.
4. Choose **Create record**.
5. Specify the following values:

   Routing policy

   Choose the applicable routing policy. For more information, see Choosing a routing policy.

   Record name

   Enter the domain name that you want to use to route traffic to your App Runner service. The default value is the name of the hosted zone.

   For example, if the name of the hosted zone is example.com and you want to use acme.example.com to route traffic to your App Runner service, enter **acme**.

   Value/Route traffic to

   Choose **Alias to App Runner Service**, then choose the AWS Region. Choose the domain name of the environment that you want to route traffic to.

   Record type

   Accept the default value, **A – IPv4 address**.

   Evaluate target health

   Choose **No**. For information about evaluating target health, see Evaluate target health.
6. Choose **Create records**.

   Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your App Runner service by using the name of the alias record that you created in this procedure.

Routing traffic to an AWS Global Accelerator

AWS Global Accelerator is a service in which you create accelerators to improve the performance of your applications for local and global users. The service reacts instantly to changes in health or configuration to make sure that internet traffic from clients is always directed to healthy endpoints.

Global Accelerator includes a fault tolerant architecture, and incorporates AWS Shield Standard, for automated in-line mitigation from DDoS attacks. For more information, see What is Global Accelerator in the AWS Global Accelerator Developer Guide.

By default, Global Accelerator provides you with static IP addresses that you associate with your accelerator. The static IP addresses are anycast from the AWS edge network. Accelerators include a deafult DNS name, but in most scenarios, you can configure DNS to use your custom domain name (such as www.example.com) with your accelerator, instead of using the assigned static IP addresses or the default DNS name.

> **Note**
>
> Route 53 doesn't charge for alias queries to Global Accelerator or other AWS resources.

Prerequisites

To get started, you need the following:

- An accelerator. You can create either a standard accelerator, or a custom routing accelerator. For more information, see Create a standard accelerator and Create a custom routing accelerator.
- A registered domain name. You can use Amazon Route 53 as your domain registrar, or you can use a different registrar.
- Route 53 as the DNS service for the domain. If you register your domain name by using Route 53, we automatically configure Route 53 as the DNS service for the domain.

  For information about using Route 53 as the DNS service provider for your domain, see Making Amazon Route 53 the DNS service for an existing domain.

Configuring Amazon Route 53 to route traffic to an accelerator

To configure Amazon Route 53 to route traffic to an accelerator, perform the following procedure.

To route traffic to an accelerator

1. Open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.

3. Choose the name of the hosted zone that matches the name of the domain that you want to route traffic for.
4. Choose **Create record**.
5. Specify the following values:

   Routing policy

   Choose the applicable routing policy. For more information, see Choosing a routing policy.

   Record name

   Enter the domain name that you want to use to route traffic to your accelerator. The default value is the name of the hosted zone.

   For example, if the name of the hosted zone is example.com and you want to use acme.example.com to route traffic to your Global Accelerator, enter **acme**.

   Value/Route traffic to

   Choose **Alias to Global Accelerator**, then choose the AWS Region. Choose the DNS name for the accelerator.

   You can enter the DNS name of an accelerator that you created using the current AWS account or using a different AWS account.

   Record type

   Accept the default value, **A – IPv4 address**.

   Evaluate target health

   Accept the default value of **Yes**.
6. Choose **Create records**.

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your accelerator by using the name of the alias record that you created in this procedure.

Routing traffic to an AWS Elastic Beanstalk environment

If you're using AWS Elastic Beanstalk to deploy and manage applications in the AWS Cloud, you can use Amazon Route 53 to route DNS traffic for your domain, such as example.com, to a new or an existing Elastic Beanstalk environment.

To route DNS traffic to an Elastic Beanstalk environment, see the procedures in the following topics.

> **Note**
>
> These procedures assume that you're already using Route 53 as the DNS service for your domain. If you're using another DNS service, see Making Amazon Route 53 the DNS service for an existing domain for information about using Route 53 as the DNS service provider for your domain.

Topics

- Deploying an application into an Elastic Beanstalk environment
- Getting the domain name for your Elastic Beanstalk environment
- Creating an Amazon Route 53 record that routes traffic to your Elastic Beanstalk environment

Deploying an application into an Elastic Beanstalk environment

If you already have an Elastic Beanstalk environment that you want to route traffic to, skip to Getting the domain name for your Elastic Beanstalk environment.

To create an application and deploy it into an Elastic Beanstalk environment

- For information about creating an application and deploying it to an Elastic Beanstalk environment, see Getting started using Elastic Beanstalk in the *AWS Elastic Beanstalk Developer Guide*.

Getting the domain name for your Elastic Beanstalk environment

If you already know the domain name for your Elastic Beanstalk environment, skip to Creating an Amazon Route 53 record that routes traffic to your Elastic Beanstalk environment.

To get the domain name for your Elastic Beanstalk environment

1. Sign in to the AWS Management Console and open the Elastic Beanstalk console at https://console.aws.amazon.com/elasticbeanstalk/.
2. In the list of applications, find the application that you want to route traffic to, and get the value of **URL**. If you don't see the list of applications, choose **Applications** in the navigation pane.

   For more information about the URL, see Elastic Beanstalk environment's domain name in the *Elastic Beanstalk Developer Guide*.

Creating an Amazon Route 53 record that routes traffic to your Elastic Beanstalk environment

An Amazon Route 53 record contains the settings that control how traffic is routed to your Elastic Beanstalk environment. You create either a *CNAME record* or an *alias record*, depending on whether the domain name for the environment includes the Region, such as **us-east-2**, in which you deployed the environment. New environments include the Region in the domain name; environments that were created before early 2016 do not. For a comparison of CNAME and alias records, see Choosing between alias and non-alias records.

If the domain name does not include the Region

You must create a *CNAME record*. However, because of limitations imposed by DNS, you can create CNAME records only for subdomains, not for the root domain name. For example, if your domain name is example.com, you can create a record that routes traffic for acme.example.com to your Elastic Beanstalk environment, but you can't create a record that routes traffic for example.com to your Elastic Beanstalk environment.

See the procedure To create a CNAME record to route traffic to an Elastic Beanstalk environment.

If the domain name includes the Region

You can create an alias record. An alias record is specific to Route 53 and has two significant advantages over CNAME records:

- You can create alias records for the root domain name or for subdomains. For example, if your domain name is example.com, you can create a record that routes requests for example.com or for acme.example.com to your Elastic Beanstalk environment.

- Route 53 doesn't charge for requests that use an alias record to route traffic.

See the procedure To create an Amazon Route 53 alias record to route traffic to an Elastic Beanstalk environment.

To create a CNAME record to route traffic to an Elastic Beanstalk environment

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the hosted zone that you want to use to route traffic to your Elastic Beanstalk environment.
4. Choose **Create record**.
5. Choose **Switch to quick create**
6. Specify the following values:

   Routing policy

   Choose the applicable routing policy. For more information, see Choosing a routing policy.

   Record name

   Enter the domain name that you want to use to route traffic to your Elastic Beanstalk environment. The default value is the name of the hosted zone.

   For example, if the name of the hosted zone is example.com and you want to use acme.example.com to route traffic to your environment, enter **acme**.

   > **Important**
   >
   > You can't create a CNAME record that has the same name as the hosted zone.

   Alias

   If you are using the **Quick create** record creation method, turn on **Alias**.

Value/Route traffic to

Choose **IP address or another value depending on the record type** and enter the value that you get when you perform the procedure in the topic Getting the domain name for your Elastic Beanstalk environment. If you used different accounts to create your Route 53 hosted zone and your Elastic Beanstalk environment, enter the CNAME attributes for the Elastic Beanstalk environment.

Record type

Choose **CNAME**.

TTL (seconds)

Accept the default value of **300**.

7. Choose **Create records**.

Changes generally propagate to all Route 53 servers within 60 seconds.

To create an Amazon Route 53 alias record to route traffic to an Elastic Beanstalk environment

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the hosted zone that you want to use to route traffic to your Elastic Beanstalk environment.
4. Choose **Create record**.
5. Specify the following values:

Routing policy

Choose the applicable routing policy. For more information, see Choosing a routing policy.

Record name

Enter the domain name that you want to use to route traffic to your Elastic Beanstalk environment. The default value is the name of the hosted zone.

For example, if the name of the hosted zone is example.com and you want to use acme.example.com to route traffic to your environment, enter **acme**.

Value/Route traffic to

Choose **Alias to Elastic Beanstalk environment**, then choose the Region that the endpoint is from. Choose the domain name of the environment that you want to route traffic to. This is the value that you get when you perform the procedure in the topic Getting the domain name for your Elastic Beanstalk environment.

If you used different accounts to create your Route 53 hosted zone and your Elastic Beanstalk environment, enter the CNAME attribute for the Elastic Beanstalk environment.

Record type

Accept the default, **A – IPv4 address**.

Evaluate target health

Accept the default value of **Yes**.

6. Choose **Create records**.

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your Elastic Beanstalk environment by using the name of the alias record that you create in this procedure.

Routing traffic to an ELB load balancer

If you host a website on multiple Amazon EC2 instances, you can distribute traffic to your website across the instances by using an Elastic Load Balancing (ELB) load balancer. The ELB service automatically scales the load balancer as traffic to your website changes over time. The load balancer also can monitor the health of its registered instances and route domain traffic only to healthy instances.

To route domain traffic to an ELB load balancer, use Amazon Route 53 to create an alias record that points to your load balancer. An alias record is a Route 53 extension to DNS. It's similar to a CNAME record, but you can create an alias record both for the root domain, such as example.com, and for subdomains, such as www.example.com. (You can create CNAME records only for subdomains.)

> **Note**
>
> Route 53 doesn't charge for alias queries to ELB load balancers or other AWS resources.

Prerequisites

To get started, you need the following:

- An ELB load balancer. You can use an ELB Classic, Application, or Network Load Balancer. For information about creating a load balancer, see Getting started with Elastic Load Balancing in the *Elastic Load Balancing User Guide*.

  Give the load balancer a name that will help you remember what it's for later. The name that you specify when you create a load balancer is the name that you'll choose when you create an alias record in the Route 53 console.
- A registered domain name. You can use Route 53 as your domain registrar, or you can use a different registrar.
- Route 53 as the DNS service for the domain. If you register your domain name by using Route 53, we automatically configure Route 53 as the DNS service for the domain.

  For information about using Route 53 as the DNS service provider for your domain, see Making Amazon Route 53 the DNS service for an existing domain.

Configuring Amazon Route 53 to route traffic to an ELB load balancer

To configure Amazon Route 53 to route traffic to an ELB load balancer, perform the following procedure.

To route traffic to an ELB load balancer

1. If you created the Route 53 hosted zone and ELB load balancer using the same account, skip to step 2.

   If you created the hosted zone and the ELB load balancer using different accounts, perform the procedure Getting the DNS name for an Elastic Load Balancing load balancer to get the DNS name for the load balancer.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Hosted zones**.
4. Choose the name of the hosted zone that has the domain name that you want to use to route traffic to your load balancer.

5. Choose **Create record**.
6. Specify the following values:

Routing policy

Choose the applicable routing policy. For more information, see Choosing a routing policy.

Record name

Enter the domain or subdomain name that you want to use to route traffic to your ELB load balancer. The default value is the name of the hosted zone.

For example, if the name of the hosted zone is example.com and you want to use acme.example.com to route traffic to your load balancer, enter **acme**.

Alias

If you are using the **Quick create** record creation method, turn on **Alias**.

Value/Route traffic to

Choose **Alias to Application and Classic Load Balancer** or **Alias to Network Load Balancer**, then choose the Region that the endpoint is from.

If you created the hosted zone and the ELB load balancer using the same AWS account, choose the DNS name that you assigned to the load balancer when you created it.

If you created the hosted zone and the ELB load balancer using different accounts, enter the value that you got in step 1 of this procedure.

> **Note**
>
> The console prepends **dualstack.** to the DNS name of the application and Classic Load Balancer from the same AWS account only. When a client, such as a web browser, requests the IP address for your domain name (example.com) or subdomain name (www.example.com), the client can request an IPv4 address (an A record), an IPv6 address (an AAAA record), or both IPv4 and IPv6 addresses (in separate requests with IPv4 first). The **dualstack.** designation allows Route 53 to respond with the appropriate IP address for your load balancer based on which IP address format the client requested. You will need to prepend **dualstack.** for Application and Classic Load Balancer from the different account.

Record type

Choose **A – IPv4 address**.

Evaluate target health

If you want Route 53 to route traffic based on the health of your resources, choose **Yes**. For more information about checking the health of your resources, see Creating Amazon Route 53 health checks.

7. Choose **Create records**.

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your load balancer by using the name of the alias record that you created in this procedure.

Routing traffic to a website that is hosted in an Amazon S3 bucket

This topic provides comprehensive procedures for routing DNS traffic to any Amazon Simple Storage Service bucket configured for static website hosting. If you're setting up a static website with Amazon Simple Storage Service, see Use your domain for a static website in an Amazon S3 bucket for a complete tutorial.

Amazon Simple Storage Service (Amazon S3) provides secure, durable, highly scalable cloud storage. You can configure an S3 bucket to host a static website that can include webpages and client-side scripts. (S3 doesn't support server-side scripting.)

To route domain traffic to an S3 bucket, use Amazon Route 53 to create an alias record that points to your bucket. An alias record is a Route 53 extension to DNS. It's similar to a CNAME record, except you can create an alias record both for the root domain, such as example.com, and for subdomains, such as www.example.com. You can create CNAME records only for subdomains.

> **Note**
>
> Route 53 doesn't charge for alias queries to S3 buckets or other AWS resources.

Prerequisites

To get started, you need the following:

- An S3 bucket that's configured to host a static website.

  For more information, see Tutorial: Configuring a static website using a custom domain registered with Route 53 in the *Amazon Simple Storage Service User Guide*.

  > **Important**
  >
  > The bucket must have the same name as your domain or subdomain. For example, if you want to use the subdomain acme.example.com, the name of the bucket must be acme.example.com.

  You can route traffic for a domain and its subdomains, such as example.com and www.example.com, to a single bucket. Create a bucket for the domain and each subdomain, and configure all but one of the buckets to redirect traffic to the remaining bucket.

  > **Note**
  >
  > An S3 bucket that's configured as a website endpoint doesn't support SSL/TLS, so you need to route traffic to the CloudFront distribution and use the S3 bucket as the origin for the distribution.
  > For instructions on how to create a CloudFront distribution, see Routing traffic to an Amazon CloudFront distribution by using your domain name.

- A registered domain name. You can use Route 53 as your domain registrar, or you can use a different registrar.
- Route 53 as the DNS service for the domain. If you register your domain name by using Route 53, we automatically configure Route 53 as the DNS service for the domain.

  For information about using Route 53 as the DNS service provider for your domain, see Making Amazon Route 53 the DNS service for an existing domain.

Configuring Amazon Route 53 to route traffic to an S3 Bucket

To configure Amazon Route 53 to route traffic to an S3 bucket that is configured to host a static website, perform the following procedure.

To route traffic to an S3 bucket

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name of the hosted zone that has the domain name that you want to use to route traffic to your S3 bucket.
4. Choose **Create record**.
5. Specify the following values:

   Record name

   Enter the domain name that you want to use to route traffic to your S3 bucket. The default value is the name of the hosted zone.

   For example, if the name of the hosted zone is example.com and you want to use acme.example.com to route traffic to your bucket, enter **acme**.

   Record type

   Choose **A – IPv4 address**.

   Alias

   Turn on **Alias**.

   Route traffic to

   Choose **Alias to S3 website endpoint**, then choose the Region that the endpoint is from.

   Choose the bucket that has the same name that you specified for **Record name**.

   The list includes a bucket only if the bucket meets the following requirements:

   - The name of the bucket is the same as the name of the record that you're creating.
   - The bucket is configured as a website endpoint.
   - The bucket was created by the current AWS account.

If you created the bucket using a different AWS account, enter the name of the Region that you created your S3 bucket in. For the correct format for the Region name, see the **Website endpoint** column in the table Amazon S3 website endpoints in the *Amazon Web Services General Reference*.

Evaluate target health

Choose **No**. For information about evaluating target health, see Evaluate target health.

Routing policy

Choose the applicable routing policy. For more information, see Choosing a routing policy.

6. Choose **Create records**.

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your S3 bucket by using the name of the alias record that you created in this procedure.

Routing traffic to an Amazon Virtual Private Cloud interface endpoint by using your domain name

You can use AWS PrivateLink to access selected services with an Amazon Virtual Private Cloud (Amazon VPC) interface endpoint. These services include some AWS services, services that are hosted by other AWS customers and partners in their own VPCs, and supported AWS Marketplace partner services.

To route domain traffic to an interface endpoint, use Amazon Route 53 to create an alias record. An alias record is a Route 53 extension to DNS. It's similar to a CNAME record, but you can create an alias record both for the root domain, such as example.com, and for subdomains, such as www.example.com. You can create CNAME records only for subdomains.

> **Note**
>
> Route 53 doesn't charge for alias queries to interface endpoints or other AWS resources.

Topics

- Prerequisites

- Configuring Amazon Route 53 to route traffic to an Amazon VPC interface endpoint

Prerequisites

To get started, you need the following:

- An Amazon VPC interface endpoint. For more information, see Interface VPC endpoints (AWS PrivateLink) in the *Amazon VPC User Guide*.
- A registered domain name. You can use Amazon Route 53 as your domain registrar, or you can use a different registrar.
- Route 53 as the DNS service for the domain. If you register your domain name by using Route 53, we automatically configure Route 53 as the DNS service for the domain.

  For information about using Route 53 as the DNS service provider for your domain, see Making Amazon Route 53 the DNS service for an existing domain.

Configuring Amazon Route 53 to route traffic to an Amazon VPC interface endpoint

To configure Amazon Route 53 to route traffic to an Amazon VPC interface endpoint, perform the following procedure.

To route traffic to an Amazon VPC interface endpoint

1. If you created the Route 53 hosted zone and the Amazon VPC interface endpoint using the same account, skip to step 2.

   If you created the hosted zone and the interface endpoint using different accounts, get the service name for the interface endpoint:

   a. Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
   b. In the navigation pane, choose **Endpoints**.
   c. In the right pane, choose the endpoint that you want to route internet traffic to.
   d. In the bottom pane, get the value of DNS name, for example, **vpce-0fd00dd593example-dexample.cloudtrail.us-west-2.vpce.amazonaws.com**.
2. Open the Route 53 console at https://console.aws.amazon.com/route53/.

3. In the navigation pane, choose **Hosted zones**.
4. Choose the name of the hosted zone that has the domain name that you want to use to route traffic to your interface endpoint.
5. Choose **Create record**.
6. Specify the following values:

   Routing policy

   Choose the applicable routing policy. For more information, see Choosing a routing policy.

   Record name

   Enter the domain name that you want to use to route traffic to your Amazon VPC interface endpoint.

   Alias

   If you are using the **Quick create** record creation method, turn on **Alias**.

   Value/Route traffic to

   Choose **Alias to VPC endpoint**, then choose the Region that the endpoint is from.

   How you specify the value for **Endpoints** depends on whether you created the hosted zone and the interface endpoint using the same AWS account or different accounts:

   - **Same account** – Choose the list, and find the category **Amazon VPC endpoints**. Then choose the DNS name of the interface endpoint that you want to route internet traffic to.
   - **Different accounts** – Enter the value that you got in step 1 of this procedure.

   Record type

   Choose **A – IPv4 address**.

   Evaluate target health

   Choose **No**. For information about evaluating target health, see Evaluate target health.
7. Choose **Create records**.

Changes generally propagate to all Route 53 servers within 60 seconds. When propagation is done, you'll be able to route traffic to your interface endpoint by using the name of the alias record that you created in this procedure.

Routing traffic to Amazon WorkMail

You can use Route 53 to route traffic to your Amazon WorkMail email domain. The name of your Route 53 hosted zone (such as example.com) must match the name of an Amazon WorkMail domain.

> **Note**
>
> You can route traffic to an Amazon WorkMail domain only for public hosted zones.

To route traffic to Amazon WorkMail, perform the following four procedures.

To configure Amazon Route 53 as your DNS service and add an Amazon WorkMail organization and email domain

1. If you haven't registered the domain name that you want to use in your email addresses (such as john@example.com), register the domain now so you know that the domain is available. For more information, see Registering a new domain.

   If Amazon Route 53 is not the DNS service for the email domain that you added to Amazon WorkMail, migrate DNS service for the domain to Route 53. For more information, see Making Amazon Route 53 the DNS service for an existing domain.
2. Add an Amazon WorkMail organization and email domain. For more information, see Getting started for new users in the *Amazon WorkMail Administrator Guide*.

To create a Route 53 TXT record for Amazon WorkMail

1. In the navigation pane of the Amazon WorkMail console, choose **Domains**.
2. Choose the name of the email domain, such as example.com that you want to use to route traffic to Amazon WorkMail.
3. Open another browser tab, and open the Route 53 console.
4. In the Route 53 console, do the following:

   a. In the navigation pane, choose **Hosted zones**.

   b. Choose the name of the hosted zone that you want to use for your Amazon WorkMail email domain.

5. In the Amazon WorkMail console, in the section **Step 1: Verify domain ownership**, go to the **Hostname** column, and copy the part of the value that precedes your email domain name.

   For example, if your Amazon WorkMail email domain is **example.com** and the value of **Hostname** is **_amazonses.example.com**, copy **_amazonses**.

6. In the Route 53 console, do the following:

   a. Choose **Create record**, and choose **Simple routing**.

   b. For **Record name**, paste the value that you copied in step 5.

   c. For **Record type**, choose **TXT – Text**.

7. In the Amazon WorkMail console, for the TXT record, copy the value of the **Value** column, including the quotation marks.

8. In the Route 53 console, do the following:

   a. For **Value/Route traffic to**, choose **IP address or another value depending on the record type**, and paste the value that you copied in step 7.

      Don't change any other settings.

   b. Choose **Create**.

To create a Route 53 MX record for Amazon WorkMail

1. In the Amazon WorkMail console, in the section **Step 2: Finalize domain setup**, go to the row that has a **Record type** of **MX**, and copy the value of the **Value** column.

2. In the Route 53 console, do the following:

   a. Choose **Create record**.

   b. For **Value/Route traffic to**, choose **IP address or another value depending on the record type**, and paste the value that you copied in step 1.

   c. For **Record type**, choose **MX – Mail Exchange**.

      Don't change any other settings.

   d. Choose **Create records**.

To create four Route 53 CNAME records for Amazon WorkMail

1. In the Amazon WorkMail console, in the section **Step 2: Finalize domain setup**, go to the first row that has a **Record type** of **CNAME**. In the **Hostname** column, copy the part of the value that precedes your email domain name.

   For example, if your Amazon WorkMail email domain is **example.com** and the value of **Hostname** is **autodiscover.example.com**, copy **autodiscover**.
2. In the Route 53 console, do the following:

   a. Choose **Create record**.

   b. For **Record name**, paste the value that you copied in step 1.

   c. For **Record type**, choose **CNAME – Canonical Name**.
3. In the Amazon WorkMail console, in the first row that has a **Record type** of **CNAME**, copy the value of the **Value** column.
4. In the Route 53 console, do the following:

   a. For **Value/Route traffic to**, choose **IP address or another value depending on the record type**, and paste the value that you copied in step 3.

   Don't change any other settings.

   b. Choose **Create records**.
5. Repeat steps 1 through 4 for the remaining CNAME records that are listed in the Amazon WorkMail console.

Routing traffic to Amazon OpenSearch Service domain endpoint

Amazon OpenSearch Service is is a managed service that makes it easy to deploy, operate, and scale OpenSearch clusters in the AWS Cloud. An OpenSearch Service Service domain is synonymous with an OpenSearch Service cluster. Domains are clusters with the settings, instance types, instance counts, and storage resources that you specify. For more information, see What is Amazon OpenSearch Service in the *Amazon OpenSearch Service Developer Guide.*

Prerequisites

To get started, you need the following:

An OpenSearch Service domain that has a custom domain name, such as example.com that matches the name of the Route 53 record that you want to create.

For more information, see the following topics:

- Getting started in the *Amazon OpenSearch Service Developer Guide*.
- Creating a custom endpoint in the *Amazon OpenSearch Service Developer Guide*.

Configuring Amazon Route 53 to route traffic to an Amazon OpenSearch Service domain endpoint

To use Route 53 to route traffic to OpenSearch Service you first get the domain endpoint provided by OpenSearch Service. This dual stack endpoint is provided only if custom endpoint is enabled on an OpenSearch Service domain with dual-stack network mode. For more information, see Create a custom endpoint in the *Amazon OpenSearch Service Developer Guide*.

To route traffic to an OpenSearch Service endpoint

1. Go to https://aws.amazon.com and choose **Sign In to the Console**.
2. Under **Analytics**, choose **Amazon OpenSearch Service**.
3. Under **Managed clusters** choose **Domains**.
4. On the **Domains** page choose the name of the domain that you want to route traffic to.
5. On the domain detail page copy the value for the **Domain endpoint v2 (dual stack)**.
6. Open the Route 53 console at https://console.aws.amazon.com/route53/.
7. In the navigation pane, choose **Hosted zones**.
8. Choose the linked name of the hosted zone for the domain that you want to use to route traffic to your OpenSearch Service endpoint. The domain name must match the custom endpoint defined in OpenSearch Service.
9. Choose **Create record**.

   You can use the wizard to create the records or choose **Switch to quick create**.
10. Specify the following values:

Routing policy

Choose the applicable routing policy. For more information, see Choosing a routing policy.

Record name

Enter the domain name that you want to use to route traffic to your OpenSearch Service domain endpoint. The default value is the name of the hosted zone.

For example, if the name of the hosted zone is example.com and you want to use **acme.example.com** to route traffic to your distribution, enter **acme**.

Alias

If you are using the **Quick create** record creation method, turn on **Alias**.

Value/Route traffic to

Choose **Alias to OpenSearch Service domain endpoint**. Choose the Region that the OpenSearch Service domain was created in, and choose the value that you got in step 5.

Record type

Choose **A – IPv4 address**, **AAAA – IPv6 address** or both for dual-stack.

Evaluate target health

Choose **No**. For information about evaluating target health, see Evaluate target health.

11. Choose **Create records**.

Routing traffic to Amazon VPC Lattice service domain endpoint

Amazon VPC Lattice is a fully managed application networking service that you use to connect, secure, and monitor the services and resources for your application. You can use VPC Lattice with a single virtual private cloud (VPC) or across multiple VPCs from one or more accounts. For more information, see What is Amazon VPC Lattice in the *Amazon VPC Lattice User Guide*.

Prerequisites

To get started, you need the following:

A VPC Lattice service domain that has a custom domain name, such as example.com that matches the name of the Route 53 record that you want to create.

For more information, see Associate a custom domain name with your service in the *Amazon VPC Lattice User Guide*.

Configuring Amazon Route 53 to route traffic to a VPC Lattice service domain endpoint

To use Route 53 to route traffic to Amazon VPC Lattice service domain, you first get the domain service endpoint provided by VPC Lattice. For more information, see Associate a custom domain name with your service in the *Amazon VPC Lattice User Guide*.

To route traffic to VPC Lattice service domain endpoint

1. Go to https://aws.amazon.com and choose **Sign In to the Console**.
2. Under **Networking & Content Delivery**, choose **VPC**.
3. Under **PrivateLink and Lattice** choose **Lattice Services**.
4. Create a VPC Lattice service or select an existing VPC Lattice service.

   > **Note**
   >
   > When creating a VPC Lattice service, you must specify a custom domain configuration and supply a custom domain name. If you choose an existing service, it must also have a custom domain.

5. Under **Domain configuration**, copy the value for the custom domain name.
6. Open the Route 53 console at https://console.aws.amazon.com/route53/.
7. In the navigation pane, choose **Hosted zones**.
8. Choose the linked name of the hosted zone for the domain that you want to use to route traffic to your VPC Lattice service domain endpoint. The domain name must match the custom domain endpoint defined in VPC Lattice.
9. Choose **Create record**.

   You can use the wizard to create the records or choose **Switch to quick create**.
10. Specify the following values:

    Routing policy

    Choose the applicable routing policy. For more information, see Choosing a routing policy.

Record name

Enter the domain name that you want to use to route traffic to your VPC Lattice service domain endpoint. The default value is the name of the hosted zone.

For example, if the name of the hosted zone is example.com and you want to use **acme.example.com** to route traffic to your distribution, enter **acme**.

Alias

If you are using the **Quick create** record creation method, turn on **Alias**.

Value/Route traffic to

Choose **Alias to VPC Lattice service**. Choose the Region that the VPC Lattice service domain was created in, and choose the value that you got in step 5.

Record type

Choose **A – IPv4 address**, **AAAA – IPv6 address**, or both for dual-stack.

Evaluate target health

Choose **No**. For information about evaluating target health, see Evaluate target health.

11. Choose **Create records**.

Routing traffic to other AWS resources

The following is the list of topics in other guides on how to use Route 53 to route traffic to those services.

- Using AWS Cloud Map in the *AWS Cloud Map User Guide*.
- Manage custom domains in the *AWS App Runner Developer Guide*.
- Using Route 53 as a DNS provider in the *AWS Transfer Family User Guide*.
- Using Route 53 to point a domain to an Amazon Lightsail instance.

Creating Amazon Route 53 health checks

Amazon Route 53 health checks monitor the health and performance of your web applications, web servers, and other resources. Each health check that you create can monitor one of the following:

- The health of a specified resource, such as a web server.
- The status of other health checks.
- The status of an Amazon CloudWatch alarm.
- Additionally, with Amazon Application Recovery Controller (ARC), you can set up routing control health checks with DNS failover records to manage traffic failover for your application. To learn more, see Amazon Application Recovery Controller (ARC) Developer Guide.

For an overview of the types of health checks, see Types of Amazon Route 53 health checks. For information about creating health checks, see Creating and updating health checks.

After you create a health check, you can get the status of the health check, get notifications when the status changes, and configure DNS failover:

Getting health check status and notifications

You can view the current and recent status of your health checks on the Route 53 console. You can also work with health checks programmatically through one of the AWS SDKs, the AWS Command Line Interface, AWS Tools for Windows PowerShell, or the Route 53 API.

If you want to receive a notification when the status of a health check changes, you can configure an Amazon CloudWatch alarm for each health check.

For information about viewing health check status and receiving notifications, see Monitoring health check status and getting notifications.

Configuring DNS failover

If you have multiple resources that perform the same function, you can configure DNS failover so that Route 53 routes your traffic from an unhealthy resource to a healthy resource. For example, if you have two web servers and one web server becomes unhealthy, Route 53 can route traffic to the other web server. For more information, see Configuring DNS failover.

Topics

- Types of Amazon Route 53 health checks
- Private endpoint health checks
- Health check sources
- How Amazon Route 53 determines whether a health check is healthy
- Creating, updating, and deleting health checks
- Configuring DNS failover
- Naming and tagging health checks
- Using health checks with Amazon Route 53 API versions earlier than 2012-12-12

Types of Amazon Route 53 health checks

You can create the following types of Amazon Route 53 health checks:

Health checks that monitor a public endpoint

You can configure a health check that monitors a public endpoint that you specify either by IP address or by domain name. At regular intervals that you specify, Route 53 submits automated requests over the internet to your application, server, or other resource to verify that it's reachable, available, and functional. Optionally, you can configure the health check to make requests similar to those that your users make, such as requesting a web page from a specific URL.

Health checks that monitor a private endpoint

You can create a health check that monitors a private endpoint. A private endpoint is reachable from your VPC. Examples include a resource in a private subnet or an on-premises resource that you reach over AWS Direct Connect or VPN. Private endpoint health checks run from a health check source that you create in your VPC, and you don't expose anything to the internet. For more information, see Private endpoint health checks.

Health checks that monitor other health checks (calculated health checks)

You can create a health check that monitors whether Route 53 considers other health checks healthy or unhealthy. One situation where this might be useful is when you have multiple resources that perform the same function, such as multiple web servers, and your chief concern is whether some minimum number of your resources are healthy. You can create a health check for each resource without configuring notification for those health checks. Then you can create

a health check that monitors the status of the other health checks and that notifies you only when the number of available web resources drops below a specified threshold.

Health checks that monitor CloudWatch alarms

You can create CloudWatch alarms that monitor the status of CloudWatch metrics, such as the number of throttled read events for an Amazon DynamoDB database or the number of Elastic Load Balancing hosts that are considered healthy. After you create an alarm, you can create a health check that monitors the same data stream that CloudWatch monitors for the alarm.

To improve resiliency and availability, Route 53 doesn't wait for the CloudWatch alarm to go into the `ALARM` state. The status of a health check changes from healthy to unhealthy based on the data stream and on the criteria in the CloudWatch alarm.

Route 53 supports CloudWatch alarms with the following features:

- Standard-resolution metrics. High-resolution metrics aren't supported. For more information, see High-resolution metrics in the *Amazon CloudWatch User Guide*.
- Statistics: Average, Minimum, Maximum, Sum, and SampleCount. Extended statistics aren't supported.
- Route 53 does not support "M out of N" alarms. For more information, see Evaluating an alarm in the *Amazon CloudWatch guide*.
- A health check can only monitor a CloudWatch alarm that exists in the same AWS account as the health check.
- Route 53 does not support alarms that use metric math to query multiple CloudWatch metrics.

Amazon Application Recovery Controller (ARC) routing controller

Health checks in ARC are associated with routing controls, which are simple on/off switches. You configure each routing control health check with a failover DNS record. Then you can simply update your routing controls in ARC to reroute traffic and fail over your applications, for example, across Availability Zones or AWS-Regions. For more information, see Routing control in ARC in the ARC developer guide.

Private endpoint health checks

A private endpoint health check monitors an endpoint that is reachable from your VPC, such as a specific IP address, domain name, or port. Amazon Route 53 checks the endpoint from a health check source that you create in your VPC. You don't need to expose any endpoint to the internet.

When an endpoint becomes unhealthy, the private endpoint health check marks any associated DNS records as unhealthy. Route 53 can then redirect traffic to healthy endpoints. This extends Route 53 health checks and DNS failover from public endpoints to your private resources.

Private endpoint health checks are fully managed. You set them up through the Route 53 console, the AWS CLI, or CloudFormation, the same way that you set up public endpoint health checks.

How private endpoint health checks work

Private endpoint health checks work in three stages.

1. **Create a health check source** – A health check source is the location inside your VPC that health-check traffic comes from. You set up a health check source one time and use it for up to 200 private endpoint health checks. Each health check source has an IP address in each of three subnets, in three Availability Zones in the same Region as the VPC. For more information, see Health check sources.
2. **Create a health check that uses the source** – You create a private endpoint health check that uses the health check source. The health check monitors a specific private IP address, domain name, or port over HTTP, HTTPS, or TCP. You can also turn on string matching and 10-second fast checks.
3. **Associate the health check with your DNS records** – You can use any routing policy. When the endpoint becomes unhealthy, the health check marks the associated records as unhealthy, so that Route 53 routes traffic to healthy endpoints.

Note the following about private endpoint health checks:

- A private endpoint health check runs only from the AWS Region of its health check source.
- A private endpoint health check uses a single health check source.
- A private endpoint health check and its health check source must be in the same AWS account.
- You can't reference a private endpoint health check from a calculated health check.

Availability at launch

At launch, private endpoint health checks support the following:

Private endpoint health check support at launch

Supported at launch	Not supported at launch
• Private VPC endpoints. • On-premises and hybrid endpoints that are reachable from the health check source VPC.	• Calculated health checks that reference private endpoint health checks. • A combination of private and public health checks.

Setup considerations

Review the following considerations before you set up private endpoint health checks:

- **Network access** — Security groups and network ACLs must allow traffic from the health check source to your target endpoints.
- **Availability Zones** — A health check source covers three Availability Zones. If your VPC spans more than three Availability Zones, you can create additional health check sources to cover target endpoints in the other Availability Zones.
- **Account scope** — You can associate a private endpoint health check only with DNS records in the same AWS account. Cross-account association isn't supported.
- **FQDN endpoints** — Route 53 resolves names only through the Amazon VPC's Route 53 Resolver, not custom DNS servers in your DHCP options set. The Amazon VPC must have DNS resolution enabled and a primary IPv4 CIDR block in a private IP address range.

  For names that only your DNS servers resolve, you have three options. You can associate the private hosted zone with the Amazon VPC, add a Resolver forwarding rule, or use an IP address health check.

Regional considerations

A private endpoint health check is a Regional resource. Route 53 performs the health check from the AWS Region of its health check source. The health check's status, CloudWatch metrics, and alarms are in that Region.

A private endpoint health check runs within a single Region. A service event in that Region can affect your ability to health-check your endpoints from that Region.

As a best practice, create health check sources in more than one Region. This reduces the risk that a single Region's outage affects your health checks. Alternatively, use a health check source whose Amazon VPC has connectivity to another Region (for example, through Amazon VPC peering or Transit Gateway). Either approach lets Route 53 continue to health-check your endpoints if one Region is unavailable.

Health check sources

A health check source is a resource in your VPC that Route 53 health checkers use to reach private endpoints in that VPC.

To create a source, you choose an AWS Region, a VPC in that Region, and three subnets in three different Availability Zones. For each subnet, you choose one or more security groups (up to 10). Private endpoint health checks that use the source run from that Region.

> **Note**
>
> In US West (N. California), which has two Availability Zones, choose two subnets.

Keep the following characteristics in mind when you work with health check sources:

- Each health check source supports up to 200 health checks. To run more health checks, create additional sources.
- You can create up to 5 health check sources per AWS account.
- Health checks that use a health check source count toward the quota of 200 health checks per AWS account. For more information, see Quotas on health checks.
- You can associate health checks that use a source only with records in the same AWS account as the source.
- After you create a health check source, you can't change its network configuration (VPC, subnets, or security groups). You can still change its name and tags, and edit the health checks that use it. To change the network configuration, create a new source, reassociate your health checks to the new source, and then delete the old source.

- Creating a source can take a few minutes. A source must be in the **Available** status before you can use it in a health check.
- A health check source covers three Availability Zones. If your VPC spans more than three Availability Zones, you can create additional health check sources to cover target endpoints in the other Availability Zones.
- To create or delete a health check source, you need the `iam:CreateServiceLinkedRole` permission for the service-linked role `AWSServiceRoleForRoute53PrivateHealthChecks` (service principal `route53healthchecks.amazonaws.com`). Route 53 makes this call with your credentials on every `CreateHealthCheckSource` and `DeleteHealthCheckSource` request, even after the role exists, so keep the permission in place.

  The AWS managed policy `AmazonRoute53FullAccess` already includes it. For a custom policy, scope it with the condition `iam:AWSServiceName = route53healthchecks.amazonaws.com`. If the permission is missing, the request fails immediately with `AccessDenied` (HTTP 403), and Route 53 doesn't create a source.

Creating a health check source

The following procedure describes how to create a health check source by using the Route 53 console.

To create a health check source

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**. Then choose the **Health check sources** tab.
3. Choose **Create health check source**.
4. (Optional) For **Name**, enter a name.
5. For **Region**, choose the AWS Region of the VPC. Private endpoint health checks that use this source run from this Region. The status, CloudWatch metrics, and alarms for these health checks also appear in this Region, even when you set your console to a different Region.
6. For **VPC**, choose the VPC that can reach your private endpoints. Route 53 filters the list by the Region that you selected.
7. Under **Subnet configurations**, choose three Availability Zones. For each one, choose a subnet and one or more security groups (up to 10).

> **Note**
>
> In US West (N. California), which has two Availability Zones, choose two subnets.

8. Choose **Create health check source**.

> **Note**
>
> You can't change the network configuration of a health check source after you create it. To change it, create a new source.

A health check source can take a few minutes to provision. Wait until its status is **Available** before you use it in a health check.

You can also start creating a health check source while you create a private endpoint health check. In the **Health check source** field, choose **Create health check source** to open the create page in a new tab. For more information, see the section called “Values that you specify when you create or update a private endpoint health check”.

Viewing health check sources

The following procedure describes how to view your health check sources by using the Route 53 console.

To view your health check sources

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**. Then choose the **Health check sources** tab.

The page lists your health check sources with their status. A health check source can have one of the following statuses:

Pending

Route 53 accepts the request to create the source and starts creating it.

Creating

Route 53 is creating the health check source. Creating a source can take a few minutes.

Available

The health check source is ready. You can use a source in a health check only when it is in the **Available** status.

Create failed

Route 53 couldn't create the health check source.

Delete pending

Route 53 accepts the request to delete the source and starts deleting it.

Deleting

Route 53 is deleting the health check source.

Deleted

Route 53 deleted the health check source.

Delete failed

Route 53 couldn't delete the health check source.

Deleting a health check source

The following procedure describes how to delete a health check source by using the Route 53 console.

> **Note**
>
> Before you can delete a health check source, disassociate all health checks that use it. You can't delete a source while health checks remain associated with it.

To delete a health check source

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**. Then choose the **Health check sources** tab.

3. Select the health check source that you want to delete.
4. Choose **Delete**, and then confirm the deletion.

Deletion runs in the background. The source remains listed with a deleting status until the deletion completes.

How Amazon Route 53 determines whether a health check is healthy

The method that Amazon Route 53 uses to determine whether a health check is healthy depends on the type of health check.

How Route 53 determines the status of health checks that monitor a public endpoint

Route 53 has health checkers in locations around the world. When you create a health check that monitors a public endpoint, health checkers start to send requests to the endpoint that you specify. These requests determine whether the endpoint is healthy. You can choose which locations you want Route 53 to use, and you can specify the interval between checks: every 10 seconds or every 30 seconds. Note that Route 53 health checkers in different data centers don't coordinate with one another, so you'll sometimes see several requests per second regardless of the interval you chose, followed by a few seconds with no health checks at all.

Each health checker evaluates the health of the endpoint based on two values:

- Response time. A resource can be slow to respond or can fail to respond to a health check request for a variety of reasons. For example, the resource is shut down for maintenance, it's under a distributed denial of service (DDoS) attack, or the network is down.
- Whether the endpoint responds to a number of consecutive health checks that you specify (the failure threshold)

Route 53 aggregates the data from the health checkers and determines whether the endpoint is healthy:

- If more than 18% of health checkers report that an endpoint is healthy, Route 53 considers it healthy.

- If 18% of health checkers or fewer report that an endpoint is healthy, Route 53 considers it unhealthy.

The 18% value was chosen to make sure that health checkers in multiple regions consider the endpoint healthy. This prevents an endpoint from being considered unhealthy only because network conditions have isolated the endpoint from some health-checking locations. This value might change in a future release.

The response time that an individual health checker uses to determine whether an endpoint is healthy depends on the type of health check:

- **HTTP and HTTPS health checks** – Route 53 must be able to establish a TCP connection with the endpoint within four seconds. In addition, the endpoint must respond with an HTTP status code of 2xx or 3xx within two seconds after connecting.

  > **Note**
  >
  > HTTPS health checks don't validate SSL/TLS certificates, so checks don't fail if a certificate is invalid or expired.

- **TCP health checks** – Route 53 must be able to establish a TCP connection with the endpoint within ten seconds.
- **HTTP and HTTPS health checks with string matching** – As with HTTP and HTTPS health checks, Route 53 must be able to establish a TCP connection with the endpoint within four seconds, and the endpoint must respond with an HTTP status code of 2xx or 3xx within two seconds after connecting.

  After a Route 53 health checker receives the HTTP status code, it must receive the response body from the endpoint within the next two seconds. Route 53 searches the response body for a string that you specify. The string must appear entirely in the first 5,120 bytes of the response body or the endpoint fails the health check. If you're using the Route 53 console, you specify the string in the **Search String** field. If you're using the Route 53 API, you specify the string in the `SearchString` element when you create the health check.

For health checks that monitor an endpoint (except TCP health checks), if the response from the endpoint includes any headers, the headers must be in the format that is defined in RFC7230, Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing, section 3.2, "Header Fields."

Route 53 considers a new health check to be healthy until there's enough data to determine the actual status, healthy or unhealthy. If you chose the option to invert the health check status, Route 53 considers a new health check to be *unhealthy* until there's enough data.

How Route 53 determines the status of health checks that monitor other health checks

A health check can monitor the status of other health checks; this type of health check is known as a *calculated health check*. The health check that does the monitoring is the *parent health check*, and the health checks that are monitored are *child health checks*. One parent health check can monitor the health of up to 255 child health checks. Here's how the monitoring works:

- Route 53 adds up the number of child health checks that are considered to be healthy.
- Route 53 compares that number with the number of child health checks that must be healthy for the status of the parent health check to be considered healthy.

For more information, see Monitoring other health checks (calculated health checks) in Values that you specify when you create or update health checks.

Route 53 considers a new health check to be healthy until there's enough data to determine the actual status, healthy or unhealthy. If you chose the option to invert the health check status, Route 53 considers a new health check to be *unhealthy* until there's enough data.

How Route 53 determines the status of health checks that monitor CloudWatch alarms

When you create a health check that is based on a CloudWatch alarm, Route 53 monitors the data stream for the corresponding alarm instead of monitoring the alarm state. If the data stream indicates that the state of the alarm is **OK**, the health check is considered healthy. If the data stream indicates that the state is **Alarm**, the health check is considered unhealthy. If the data stream doesn't provide enough information to determine the state of the alarm, the health check status depends on the setting for **Health check status**: healthy, unhealthy, or last known status. (In the Route 53 API, this setting is `InsufficientDataHealthStatus`.)

Route 53 doesn't support cross-account CloudWatch alarms.

> **Note**
>
> Because Route 53 health checks monitor CloudWatch data streams instead of the state of CloudWatch alarms, you can't force the status of a health check to change by using the CloudWatch SetAlarmState API operation.

Route 53 considers a new health check to be healthy until there's enough data to determine the actual status, healthy or unhealthy. If you chose the option to invert the health check status, Route 53 considers a new health check to be *unhealthy* until there's enough data.

How Route 53 determines the status of a private endpoint health check

For a private endpoint health check, Route 53 sends requests from the health check source in your Amazon VPC to the endpoint that you specify. You can specify the endpoint by IP address or domain name, and you can use HTTP, HTTPS, or TCP. Route 53 sends requests at the interval that you choose: every 30 seconds (standard) or every 10 seconds (fast checks).

For HTTP and HTTPS, the endpoint is healthy when it responds with an HTTP status code in the 2xx or 3xx range within the response time limit. For TCP, the endpoint is healthy when Route 53 can establish a connection. Optionally, for HTTP/HTTPS you can enable string matching so the body must contain a string you specify.

Route 53 changes the status to unhealthy after the number of consecutive failures set by the failure threshold (default 3). Route 53 then updates DNS so that failover routing directs traffic to a healthy endpoint.

The health check source must be in the **Available** status. For more information about sources, see Health check sources.

Creating, updating, and deleting health checks

> **Important**
>
> If you're updating or deleting health checks that are associated with records, review the tasks in Updating or deleting health checks when DNS failover is configured before you proceed.

This section covers the following topics related to managing Route 53 health checks:

1. **Creating and updating health checks:**
   - Learn how to create and update health checks using the Route 53 console.
   - Understand the values you need to specify when creating or updating health checks, such as endpoint monitoring, protocol, IP address, domain name, and advanced configuration options.
2. **Values displayed when creating a health check:**
   - Discover the values that the Route 53 console displays based on your input when creating a health check, such as the full URL or IP address and port.
3. **Updating health checks for CloudWatch alarm changes:**
   - Find out how to update a health check when you change the settings of the associated CloudWatch alarm.
4. **Deleting health checks:**
   - Follow the procedure to delete health checks by using the Route 53 console.
5. **Updating or deleting health checks when DNS failover is configured:**
   - Learn the recommended tasks to perform when updating or deleting health checks associated with DNS records to ensure proper routing and failover configuration.
6. **Configuring router and firewall rules:**
   - Understand how to configure your router and firewall rules to allow inbound traffic from Route 53 health checkers, ensuring successful health checks.

By following the information provided in this section, you can effectively create, update, and delete Route 53 health checks, manage their configuration, and ensure proper integration with DNS failover and routing policies.

Topics

- Creating and updating health checks
- Values that you specify when you create or update health checks
- Values that Amazon Route 53 displays when you create a health check
- Updating health checks when you change CloudWatch alarm settings (health checks that monitor a CloudWatch alarm only)
- Disabling or enabling health checks
- Inverting health checks

- Deleting health checks
- Updating or deleting health checks when DNS failover is configured
- Configuring router and firewall rules for Amazon Route 53 health checks

Creating and updating health checks

The following procedure describes how to create and update health checks using the Route 53 console.

To create or update a health check

1. If you're updating health checks that are already associated with records, perform the recommended tasks in Updating or deleting health checks when DNS failover is configured.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Health Checks**.
4. If you want to update an existing health check, choose the linked ID of the health check, and then choose **Edit**.

   If you want to create a health check, choose **Create health check**.
5. Enter the applicable values. Note that some values can't be changed after you create a health check. For more information, see Values that you specify when you create or update health checks.

   > **Note**
   >
   > To monitor a private endpoint, choose **Private endpoint** for **Resource**, and then choose an existing health check source or create one. For the values to specify, see the section called "Values that you specify when you create or update a private endpoint health check". For more information about health check sources, see Health check sources.
6. Choose **Create health check**.

> **Note**
>
> Route 53 considers a new health check to be healthy until there's enough data to determine the actual status, healthy or unhealthy.

7. Associate the health check with one or more Route 53 records. For information about creating and updating records, see Working with records.

Values that you specify when you create or update health checks

When you create or update health checks, you specify the applicable values. Note that you can't change some values after you create a health check.

Topics

- Monitoring an endpoint
- Values that you specify when you create or update a private endpoint health check
- Monitoring other health checks (calculated health checks)
- Monitoring a CloudWatch alarm
- Advanced configuration ("Monitor an Endpoint" only)
- Get notified when a health check fails

Name

Optional, but recommended: The name that you want to assign to the health check. If you specify a value for **Name**, Route 53 adds a tag to the health check, assigns the value **Name** to the tag key, and assigns the value that you specify to the tag value. The value of the **Name** tag appears in the list of health checks in the Route 53 console, so you can easily distinguish health checks from one another.

For more information about tagging and health checks, see Naming and tagging health checks.

What to monitor

Whether you want this health check to monitor an endpoint or the status of other health checks:

- **Endpoint** – Route 53 monitors the health of an endpoint that you specify. You can specify the endpoint by providing either a domain name or an IP address and a port.

> **Note**
>
> If you specify a non-AWS endpoint, an additional charge applies. For more information, including a definition of AWS endpoints, see "Health Checks" on the Route 53 Pricing page.

- **Status of other health checks (calculated health check)** – Route 53 determines whether this health check is healthy based on the status of other health checks that you specify. You also specify how many of the health checks need to be healthy for this health check to be considered healthy.
- **State of CloudWatch alarm data stream** – Route 53 determines whether this health check is healthy by monitoring the data stream for a CloudWatch alarm.

Monitoring an endpoint

If you want this health check to monitor an endpoint, specify the following values:

- Specify endpoints by
- IP address
- Domain name

Specify endpoint by

Whether you want to specify the endpoint using an IP address or using a domain name.

After you create a health check, you can't change the value of **Specify endpoint by**.

IP address ("Specify endpoint by IP address" Only)

Choose the protocol in the drop-down, enter the IP address, the port, and the path in the text box.

- The protocol can be one of the following:

  HTTP – Route 53 tries to establish a TCP connection. If successful, Route 53 submits an HTTP request and waits for an HTTP status code of 2xx or 3xx.
- **HTTPS** – Route 53 tries to establish a TCP connection. If successful, Route 53 submits an HTTPS request and waits for an HTTP status code of 2xx or 3xx.

> **Important**
>
> If you choose **HTTPS**, the endpoint must support TLS v1.0, v1.1, or v1.2.

If you choose **HTTPS** for the value of **Protocol**, an additional charge applies. For more information, see Route 53 Pricing.

- **TCP** – Route 53 tries to establish a TCP connection.

For more information, see How Amazon Route 53 determines whether a health check is healthy.

After you create a health check, you can't change the value of **Protocol**.

For the **IP address** you can enter an IPv4 or IPv6 address of the endpoint on which you want Route 53 to perform health checks, if you chose **Specify endpoint by IP address**.

Route 53 cannot check the health of endpoints for which the IP address is in local, private, nonroutable, or multicast ranges. For more information about IP addresses that you can't create health checks for, see the following documents:

- RFC 5735, Special Use IPv4 Addresses
- RFC 6598, IANA-Reserved IPv4 Prefix for Shared Address Space.
- RFC 5156, Special-Use IPv6 Addresses

If the endpoint is an Amazon EC2 instance, we recommend that you create an Elastic IP address, associate it with your EC2 instance, and specify the Elastic IP address. This ensures that the IP address of your instance will never change. For more information, see Elastic IP addresses (EIP) in the *Amazon EC2 User Guide*.

If you delete the Amazon EC2 instance, make sure you also delete the health check associated to the EIP. For more information, see Best practices for Amazon Route 53 health checks.

> **Note**
>
> If you specify a non-AWS endpoint, an additional charge applies. For more information, including a definition of AWS endpoints, see "Health Checks" on the Route 53 Pricing page.

For the **port** you enter the port on the endpoint on which you want Route 53 to perform health checks.

For the **path (HTTP and HTTPS Protocols Only)** you enter the path that you want Route 53 to request when performing health checks. The path can be any value for which your endpoint will return an H TTP status code of 2xx or 3xx when the endpoint is healthy, such as the file /docs/route53-health-check.html. You can also include query string parameters, for example, /welcome.html?language=jp&login=y. If you don't include a leading slash (/) character, Route 53 automatically adds one.

Domain name ("Specify endpoint by domain name" Only, All Protocols)

The domain name (example.com) or subdomain name (backend.example.com) of the endpoint that you want Route 53 to perform health checks on, if you choose **Specify endpoint by domain name**.

If you choose to specify the endpoint by domain name, Route 53 sends a DNS query to resolve the domain name that you specify in **Domain name** at the interval that you specify in **Request interval**. Using an IP address that DNS returns, Route 53 then checks the health of the endpoint.

> **Note**
>
> If you specify the endpoint by domain name, Route 53 uses only IPv4 to send health checks to the endpoint. If there's no record with a type of A for the name that you specify for **Domain name**, the health check fails with a "DNS resolution failed" error.

If you want to check the health of az-affinity, failover, geolocation, geoproximity, latency, multivalue, or weighted records, and you choose to specify the endpoint by domain name, we recommend that you create a separate health check for each endpoint. For example, create a health check for each HTTP server that is serving content for www.example.com. For the value of **Domain name**, specify the domain name of the server (such as us-east-2-www.example.com), not the name of the records (www.example.com).

> **Important**
>
> In this configuration, if you create a health check for which the value of **Domain name** matches the name of the records and then associate the health check with those records, health check results will be unpredictable.

In addition, if the value of **Protocol** is **HTTP** or **HTTPS**, Route 53 passes the value of **Domain name** in the `Host` header as described in **Host name**, earlier in this list. If the value of **Protocol** is **TCP**, Route 53 doesn't pass a `Host` header.

> **Note**
>
> If you specify a non-AWS endpoint, an additional charge applies. For more information, including a definition of AWS endpoints, see "Health Checks" on the Route 53 Pricing page.

Values that you specify when you create or update a private endpoint health check

If you want this health check to monitor a private endpoint in your Amazon VPC or on-premises infrastructure that is reachable from a Amazon VPC, specify the following values:

Name

(Optional) A friendly name that helps you find the health check in the Route 53 console. The name can contain 1-256 characters, including the following: A-Z, a-z, 0-9, hyphen (-), and underscore (_).

Resource

Choose **Private endpoint** to monitor a private endpoint in your Amazon Virtual Private Cloud or on-premises infrastructure that is reachable from a Amazon VPC.

Health check source

Required. Choose a health check source that can reach your private endpoint. You can use only sources in the **Available** status. To create a source, choose **Create health check source**. The create page opens in a new tab.

For more information about health check sources, see Health check sources.

Specify endpoint by

Choose **IP address** or **Domain name**, and then provide the endpoint, protocol (HTTP, HTTPS, or TCP), port, and path to check. For example, `10.0.1.42:443/health`. For HTTP and HTTPS, the path can be any value for which your endpoint returns a 2xx or 3xx status code when it is healthy.

For the request interval, failure threshold, string matching, and other advanced options, see Advanced configuration ("Monitor an Endpoint" only). To add tags, see Naming and tagging health checks.

Monitoring other health checks (calculated health checks)

If you want this health check to monitor the status of other health checks, specify the following values:

- Health checks to monitor
- Report healthy when

Health checks to monitor

The health checks that you want Route 53 to monitor to determine the health of this health check.

You can add up to 256 health checks to **Health checks to monitor**. To remove a health check from the list, choose the **x** at the right of the highlight for that health check.

> **Note**
>
> You can't configure a calculated health check to monitor the health of other calculated health checks.

If you disable a health check that a calculated health check is monitoring, Route 53 considers the disabled health check to be healthy as it calculates whether the calculated health check is healthy. If you want the disabled health check to be considered unhealthy, choose the **Invert health check status** check box.

Report healthy when

The calculation that you want Route 53 to perform to determine whether this health check is healthy:

- **Report healthy when at least x of y selected health checks are healthy** – Route 53 considers this health check to be healthy when the specified number of health checks that you added to **Health checks to monitor** are healthy. Note the following:
  - If you specify a number greater than the number of health checks in **Health checks to monitor**, Route 53 always considers this health check to be unhealthy.
  - If you specify **0**, Route 53 always considers this health check to be healthy.
- **Report healthy when all health checks are healthy (AND)** – Route 53 considers this health check to be healthy only when all the health checks that you added to **Health checks to monitor** are healthy.
- **Report healthy when one or more health checks are healthy (OR)** – Route 53 considers this health check to be healthy when at least one of the health checks that you added to **Health checks to monitor** is healthy.

Monitoring a CloudWatch alarm

If you want this health check to monitor the alarm state of a CloudWatch alarm, specify the following values:

- CloudWatch alarm
- Health check status

CloudWatch alarm

Choose the CloudWatch alarm that you want Route 53 to use to determine whether this health check is healthy. The CloudWatch alarm must be in the same AWS account as the health check.

> **Note**
>
> Route 53 supports CloudWatch alarms with the following features:
>
> - Standard-resolution metrics. High-resolution metrics aren't supported. For more information, see High-resolution metrics in the *Amazon CloudWatch User Guide*.

- Statistics: `Average`, `Minimum`, `Maximum`, `Sum`, and `SampleCount`. Extended statistics aren't supported.
- Route 53 does not support "M out of N" alarms. For more information, see Evaluating an alarm in the *Amazon CloudWatch guide*.

Route 53 does not support alarms that use metric math to query multiple CloudWatch metrics.

If you want to create an alarm, perform the following steps:

1. Choose **create**. The CloudWatch console appears in a new browser tab.
2. Enter the applicable values. For more information, see Create or edit a CloudWatch alarm in the *Amazon CloudWatch User Guide*.
3. Return to the browser tab that the Route 53 console appears in.
4. Choose the refresh button next to the **CloudWatch alarm** list.
5. Choose the new alarm from the list.

Important

If you change settings for the CloudWatch alarm after you create a health check, you must update the health check. For more information, see Updating health checks when you change CloudWatch alarm settings (health checks that monitor a CloudWatch alarm only).

Health check status

Choose the status of the health check (healthy, unhealthy, or last known status) when CloudWatch has insufficient data to determine the state of the alarm that you chose for **CloudWatch alarm**. If you choose to use the last known status, Route 53 uses the status of the health check from the last time that CloudWatch had sufficient data to determine the alarm state. For new health checks that have no last known status, the default status for the health check is healthy.

The value of **Health check status** provides a temporary status when the data stream for a CloudWatch metric is briefly unavailable. (Route 53 monitors data streams for CloudWatch

metrics, not the state of the corresponding alarm.) If the metric will be unavailable frequently or for long periods (longer than a few hours), we recommend that you not use the last known status.

Advanced configuration ("Monitor an Endpoint" only)

- Request Interval
- Failure threshold
- String matching
- Search strings
- Latency graphs
- Enable SNI
- Host name

Request interval

The number of seconds between the time that each Route 53 health checker gets a response from your endpoint and the time that it sends the next health check request. If you choose an interval of 30 seconds, each of the Route 53 health checkers in data centers around the world will send your endpoint a health check request every 30 seconds. On average, your endpoint will receive a health check request about every two seconds. If you choose an interval of 10 seconds, the endpoint will receive a request more than once per second.

Note that Route 53 health checkers in different data centers don't coordinate with one another, so you'll sometimes see several requests per second regardless of the interval you chose, followed by a few seconds with no health checks at all.

After you create a health check, you can't change the value of **Request interval**.

> **Note**
>
> If you choose **Fast (10 seconds)** for the value of **Request interval**, an additional charge applies. For more information, see Route 53 Pricing.

Failure threshold

The number of consecutive health checks that an endpoint must pass or fail for Route 53 to change the current status of the endpoint from unhealthy to healthy or vice versa. For more information, see How Amazon Route 53 determines whether a health check is healthy.

String matching (HTTP and HTTPS Only)

Whether you want Route 53 to determine the health of an endpoint by submitting an HTTP or HTTPS request to the endpoint and searching the response body for a specified string. If the response body contains the value that you specify in **Search string**, Route 53 considers the endpoint healthy. If not, or if the endpoint doesn't respond, Route 53 considers the endpoint unhealthy. The search string must appear entirely within the first 5,120 bytes of the response body.

After you create a health check, you can't change the value of **String matching**.

> **Note**
>
> If you choose **Yes** for the value of **String matching**, an additional charge applies. For more information, see Route 53 Pricing.

How health checkers handle a compressed response

If the endpoint is a web server that returns a response that is compressed, the Route 53 health checker will uncompress the response before checking for the specified search string only if the web server compressed the response using a compression algorithm that health checkers support. Health checkers support the following compression algorithms:

- Gzip
- Deflate

If the response is compressed using another algorithm, the health checker can't decompress the response before searching for the string. In this case, the search will almost always fail, and Route 53 will consider the endpoint unhealthy

Search string (Only When "String matching" Is Enabled)

The string that you want Route 53 to search for in the body of the response from your endpoint. The maximum length is 255 characters.

Route 53 considers case when searching for **Search string** in the response body.

Latency graphs

Choose whether you want Route 53 to measure the latency between health checkers in multiple AWS Regions and your endpoint. If you choose this option, CloudWatch latency graphs appear on the **Latency** tab on the **Health checks** page in the Route 53 console. If Route 53 health checkers can't connect to the endpoint, Route 53 can't display latency graphs for that endpoint.

After you create a health check, you can't change the value of **Latency measurements**.

> **Note**
>
> If you configure Route 53 to measure the latency between health checkers and your endpoint, an additional charge applies. For more information, see Route 53 Pricing.

Enable SNI (HTTPS Only)

Specify whether you want Route 53 to send the host name to the endpoint in the `client_hello` message during TLS negotiation. This allows the endpoint to respond to the HTTPS request with the applicable SSL/TLS certificate.

Some endpoints require that HTTPS requests include the host name in the client_hello message. If you don't enable SNI, the health check status might show failure. The error message will depend on how the server is configured to respond the request containing no SNI information. A health check can also have a failure status for other reasons. If SNI is enabled and you're still getting the error, check the SSL/TLS configuration on your endpoint and confirm that your certificate is valid.

Note the following requirements:

- The endpoint must support SNI.
- The SSL/TLS certificate on your endpoint includes a domain name in the `Common Name` field and possibly several more in the `Subject Alternative Names` field. One of the domain names in the certificate must match the value that you specify for **Host name**.

Health checker regions

Choose whether you want Route 53 to check the health of the endpoint by using health checkers in the recommended regions or by using health checkers in regions that you specify.

If you update a health check to remove a Region that has been performing health checks, Route 53 continues to perform checks from that Region for up to an hour. This ensures that

some health checkers are always checking the endpoint (for example, if you replace three Regions with four different Regions).

If you choose **Customize**, choose the **x** for a Region to remove it. Choose the space at the bottom of the list to add a Region back to the list. You must specify at least three Regions.

Host name ("Specify endpoint by IP address" Only, HTTP and HTTPS Protocols Only)

The value that you want Route 53 to pass in the `Host` header in HTTP and HTTPS health checks. This is typically the fully qualified DNS name of the website on which you want Route 53 to perform health checks. When Route 53 checks the health of an endpoint, here is how it constructs the `Host` header:

- If you specify a value of **80** for **Port** and **HTTP** for **Protocol**, Route 53 passes to the endpoint a `Host` header that contains the value of **Host name**.
- If you specify a value of **443** for **Port** and **HTdTPS** for **Protocol**, Route 53 passes to the endpoint a `Host` header that contains the value of **Host name**.
- If you specify another value for **Port** and either **HTTP** or **HTTPS** for **Protocol**, Route 53 passes to the endpoint a `Host` header that contains the value *`Host name`*`:`*`Port`*.

If you choose to specify the endpoint by IP address and you don't specify a value for **Host name**, Route 53 substitutes the value of **IP address** in the `Host` header in each of the preceding cases.

Get notified when a health check fails

Use the following options to configure email notification when a health check fails:

- Create alarm
- Send notification to
- Topic name
- Recipient email addresses

Create alarm (Only When Creating Health Checks)

Specify whether you want to create a default CloudWatch alarm. If you choose **Yes**, CloudWatch sends you an Amazon SNS notification when the status of this endpoint changes to unhealthy and Route 53 considers the endpoint unhealthy for one minute.

> **Note**
>
> If you want CloudWatch to send you another Amazon SNS notification when the status changes back to healthy, you can create another alarm after you create the health check. For more information, see Creating Amazon CloudWatch alarms in the *Amazon CloudWatch User Guide*.

If you want to create an alarm for an existing health check or you want to receive notifications when Route 53 considers the endpoint unhealthy for more or less than one minute (the default value), select **No**, and add an alarm after you create the health check. For more information, see Monitoring health checks using CloudWatch.

Send notification to (Only When Creating an Alarm)

Specify whether you want CloudWatch to send notifications to an existing Amazon SNS topic or to a new one:

- **Existing SNS topic** – Select the name of the topic from the list. The topic must be in the US East (N. Virginia) Region.
- **New SNS topic** – Enter a name for the topic in **Topic name**, and enter the email addresses that you want to send notifications to in **Recipients**. Separate multiple addresses with commas (,), semicolons (;), or spaces.

  Route 53 will create the topic in the US East (N. Virginia) Region.

Topic name (Only When Creating a New SNS Topic)

If you specified **New SNS Topic**, enter the name of the new topic.

Recipient email addresses (Only When Creating a New SNS Topic)

If you specified **New SNS topic**, enter the email addresses that you want to send notifications to. Separate multiple names with commas (,), semicolons (;), or spaces.

Values that Amazon Route 53 displays when you create a health check

The **Create Health Check** page displays the following values based on the values that you typed:

URL

Either the full URL (for HTTP or HTTPS health checks) or the IP address and port (for TCP health checks) to which Route 53 will send requests when performing health checks.

Health Check Type

Either **Basic** or **Basic + additional options** based on the settings that you specified for this health check. For information about pricing for the additional options, see Route 53 Pricing.

Updating health checks when you change CloudWatch alarm settings (health checks that monitor a CloudWatch alarm only)

If you create a Route 53 health check that monitors the data stream for a CloudWatch alarm and then you update the settings in the CloudWatch alarm, Route 53 doesn't automatically update the alarm settings in the health check. If you want the health check to start using the new alarm settings, you need to update the health check.

> **Note**
>
> To update a health check programmatically, you can use the `UpdateHealthCheck` API. Just specify the current values for `AlarmIdentifier` and `Region`, and Route 53 will get the latest settings from CloudWatch. For more information, see UpdateHealthCheck in the *Amazon Route 53 API Reference*.

To update a health check with new CloudWatch alarm settings

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**.
3. Select the linked ID for the health check that you want to update.
4. Choose **Edit**.

   A note explains that the CloudWatch alarm for the health check has changed. The **Details** field shows the new alarm settings.
5. Choose **Save**.

Disabling or enabling health checks

Disabling a health check stops Route 53 from performing health checks. When you disable a health check, Route 53 stops aggregating the status of the referenced health checks. After you disable a health check, Route 53 considers the status of the health check to always be healthy. If you configured DNS failover, Route 53 continues to route traffic to the corresponding resources. If you want to stop routing traffic to a resource, change the value of **Inverted**.

You can also disable or enable a health check when you create or edit it. For more information, see Values that you specify when you create or update health checks.

To disable or enable a health check, perform the following procedure.

To disable or enable a health check

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**.
3. In the **Actions** column select the three dots and then **Disable** or **Enable**.

   Or, select the linked ID of the health check that you want to disable or enable.
4. On the **Configuration** table, the **Status** field specifies whether the health check is enabled, or disabled.
5. Choose **Disable** or **Enable** to either disable or enable the health check.

Inverting health checks

If you invert a health check, Route 53 considers the health check to be unhealthy when the status is healthy and vice versa.

You can also invert a health check when you create or edit it. For more information, see Values that you specify when you create or update health checks.

To invert a health check, perform the following procedure.

To invert a health check

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

2. In the navigation pane, choose **Health checks**.
3. In the **Actions** column select the three dots and then **Invert**.

   Or- select the linked ID of the health check that you want to invert.
4. On the **Configuration** table, the **Inverted** filed specifies whether the health check is inverted (**Yes**) or not (**No**).
5. Choose **Invert** to invert the health check.

   If you want to undo the inverted status, and the **Inverted** field is **Yes**, choose **Invert** again.

Deleting health checks

To disable health checks, perform the following procedure.

> **Note**
>
> If you're using AWS Cloud Map and you configured AWS Cloud Map to create a Route 53 health check when you register an instance, you can't use the Route 53 console to delete the health check. The health check is deleted automatically when you deregister the instance. There can be a delay of several hours before the health check no longer appears in the Route 53 console.

To delete a health check

1. If you're deleting health checks that are associated with records, perform the recommended tasks in Updating or deleting health checks when DNS failover is configured.
2. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
3. In the navigation pane, choose **Health checks**.
4. Select the linked ID of the health check that you want to delete.
5. Choose **Delete**.
6. Enter **`confirm`** in the text box and then choose **Delete**.

Updating or deleting health checks when DNS failover is configured

When you want to update or delete health checks that are associated with records, or you want to change records that have associated health checks, you must consider how your changes affect routing of DNS queries and your DNS failover configuration.

> **Important**
>
> Route 53 doesn't prevent you from deleting a health check even if the health check is associated with one or more records. If you delete a health check and you don't update the associated records, Route 53 considers the health check healthy. This affects the routing of DNS queries for your DNS failover configuration.

To update or delete health checks that are already associated with records, we recommend that you perform the following tasks:

1. Identify the records that are associated with the health checks. To identify the records that are associated with a health check, you must do one of the following:
   - Review the records in each hosted zone using the Route 53 console. For more information, see Listing records.
   - Run the `ListResourceRecordSets` API action on each hosted zone and review the response. For more information, see ListResourceRecordSets in the *Amazon Route 53 API Reference*.
2. Assess the change in behavior that will result from updating or deleting health checks, or from updating records. Based on that assessment, determine which changes to make.

   For more information, see What happens when you omit health checks?
3. Change health checks and records as applicable. For more information, see the following topics:
   - Creating and updating health checks
   - Editing records
4. Delete the health checks that you're no longer using, if any. For more information, see Deleting health checks.

Configuring router and firewall rules for Amazon Route 53 health checks

When Route 53 checks the health of an endpoint, it sends an HTTP, HTTPS, or TCP request to the IP address and port that you specified when you created the health check. For a health check to succeed, your router and firewall rules must allow inbound traffic from the IP addresses that the Route 53 health checkers use.

For the current list of IP addresses for Route 53 health checkers, for Route 53 name servers, and for other AWS services, see IP address ranges of Amazon Route 53 servers.

In Amazon EC2, security groups act as firewalls. For more information, see Amazon EC2 security groups in the *Amazon EC2 User Guide*.To configure your security groups to allow Route 53 health checks, you can either allow inbound traffic from each IP address range, or you can use an AWS-managed prefix list.

To use the AWS-managed prefix list, modify your security group to allow inbound traffic from `com.amazonaws.<region>.route53-healthchecks`, where the `<region>` is the AWS Region of your Amazon EC2 instance or resource. If you are using Route 53 health checks to check IPv6 endpoints, you should also allow inbound traffic from `com.amazonaws.<region>.ipv6.route53-healthchecks`.

For more information about AWS-managed prefix lists, see Work with AWS-managed prefix lists in the *Amazon VPC User Guide*.

Important

When you add IP addresses to a list of allowed IP addresses, add all the IP addresses in the CIDR range for each AWS Region that you specified when you created health checks, as well as the Global CIDR range. You might see that health check requests come from just one IP address in a Region. However, that IP address can change at any time to another of the IP addresses for that Region.

If you want to make sure that you include both the current and older health checker IP addresses, add ALL /26 and /18 IP address ranges to the allow list. For a complete list, see AWS IP address ranges in the *AWS General Reference*.

When you add the AWS-managed prefix list to your inbound security group, it automatically adds all necessary ranges.

Configuring DNS failover

When you have more than one resource performing the same function—for example, more than one HTTP server or mail server—you can configure Amazon Route 53 to check the health of your resources and respond to DNS queries using only the healthy resources. For example, suppose your website, example.com, is hosted on six servers, two each in three data centers around the world. You can configure Route 53 to check the health of those servers and to respond to DNS queries for example.com using only the servers that are currently healthy.

Route 53 can check the health of your resources in both simple and complex configurations:

- In simple configurations, you create a group of records that all have the same name and type, such as a group of weighted records with a type of A for example.com. You then configure Route 53 to check the health of the corresponding resources. Route 53 responds to DNS queries based on the health of your resources. For more information, see How health checks work in simple Amazon Route 53 configurations.
- In more complex configurations, you create a tree of records that route traffic based on multiple criteria. For example, if latency for your users is your most important criterion, then you might use latency alias records to route traffic to the Region that provides the best latency. The latency alias records might have weighted records in each Region as the alias target. The weighted records might route traffic to EC2 instances based on the instance type. As with a simple configuration, you can configure Route 53 to route traffic based on the health of your resources. For more information, see How health checks work in complex Amazon Route 53 configurations.

Topics

- Task list for configuring DNS failover
- How health checks work in simple Amazon Route 53 configurations
- How health checks work in complex Amazon Route 53 configurations
- How Amazon Route 53 chooses records when health checking is configured
- Active-active and active-passive failover
- Configuring failover in a private hosted zone
- How Amazon Route 53 averts failover problems

Task list for configuring DNS failover

To use Route 53 to configure DNS failover, perform the following tasks:

1. Draw a complete tree diagram of your configuration, and indicate which type of record you're creating (weighted alias, failover, latency, and so on) for each node. At the top of the tree put the records for the domain name, such as example.com, that your users will use to access your website or web application.

   The kinds of records that appear in your tree diagram depend on the complexity of the configuration:

   - In a simple configuration, either your diagram won't include any alias records, or the alias records will route traffic directly to a resource, such as an ELB load balancer, instead of to another Route 53 record. For more information, see How health checks work in simple Amazon Route 53 configurations.
   - In a complex configuration, your diagram will include a combination of alias records (such as weighted alias and failover alias) and non-alias records in a multi-level tree like the examples in the topic How health checks work in complex Amazon Route 53 configurations.

   > **Note**
   >
   > To quickly and easily create records for complex routing configurations and associate the records with health checks, you can use the traffic flow visual editor and save the configuration as a traffic policy. You can then associate the traffic policy with one or more domain names (such as example.com) or subdomain names (such as www.example.com), in the same hosted zone or in multiple hosted zones. In addition, you can roll back the updates if the new configuration isn't performing as you expected it to. For more information, see Using Traffic Flow to route DNS traffic.

   For more information, see the following documentation:

   - Choosing a routing policy
   - Choosing between alias and non-alias records

2. Create health checks for the resources that you can't create alias records for, such as Amazon EC2 servers and email servers running in your data center. You'll associate these health checks with your non-alias records.

For more information, see Creating, updating, and deleting health checks.

3. If necessary, configure router and firewall rules so that Route 53 can send regular requests to the endpoints that you specified in your health checks. For more information, see Configuring router and firewall rules for Amazon Route 53 health checks.
4. Create all the non-alias records in your diagram, and associate the health checks that you created in step 2 with the applicable records.

   If you're configuring DNS failover in a configuration that doesn't include any alias records, skip the remaining tasks.
5. Create the alias records that route traffic to AWS resources, such as ELB load balancers and CloudFront distributions. If you want Route 53 to try another branch of the tree when a resource is unhealthy, set the value of **Evaluate Target Health** to **Yes** for each of your alias records. (**Evaluate Target Health** isn't supported for some AWS resources.)
6. Starting at the bottom of the tree diagram that you created in step 1, create the alias records that route traffic to the records that you created in steps 4 and 5. If you want Route 53 to try another branch of the tree when all the non-alias records are unhealthy in a branch of your tree, set the value of **Evaluate Target Health** to **Yes** for each of your alias records.

   Remember that you can't create an alias record that routes traffic to another record until you have created the other record.

How health checks work in simple Amazon Route 53 configurations

When you have two or more resources that perform the same function, such as two or more web servers for example.com, you can use the following health-checking features to route traffic only to the healthy resources:

Check the health of EC2 instances and other resources (non-alias records)

If you're routing traffic to resources that you can't create alias records for, such as EC2 instances, you create a record and a health check for each resource. Then you associate each health check with the applicable record. Health checks regularly check the health of the corresponding resources, and Route 53 routes traffic only to the resources that health checks report as healthy.

Evaluate the health of an AWS resource (alias records)

If you're using alias records to route traffic to selected AWS resources, such as ELB load balancers, you can configure Route 53 to evaluate the health of the resource and to route traffic

only to resources that are healthy. When you configure an alias record to evaluate the health of a resource, you don't need to create a health check for the resource.

Here's an overview of how you configure Route 53 to check the health of your resources in simple configurations:

1. You identify the resources that you want Route 53 to monitor. For example, you might want to monitor all the HTTP servers that respond to requests for example.com.
2. You create health checks for the resources that you can't create alias records for, such as EC2 instances or servers in your own data center. You specify how to send health-checking requests to the resource: which protocol to use (HTTP, HTTPS, or TCP), which IP address and port to use, and, for HTTP/HTTPS health checks, a domain name and path.

   > **Note**
   >
   > If you're using any resources that you can create alias records for, such as ELB load balancers, don't create health checks for those resources.

   A common configuration is to create one health check for each resource and to use the same IP address for the health check endpoint as for the resource. The health check sends requests to the specified IP address.

   > **Note**
   >
   > Route 53 can't check the health of resources that have an IP address in local, private, nonroutable, or multicast ranges. For more information about IP addresses that you can't create health checks for, see RFC 5735, Special Use IPv4 Addresses and RFC 6598, IANA-Reserved IPv4 Prefix for Shared Address Space.

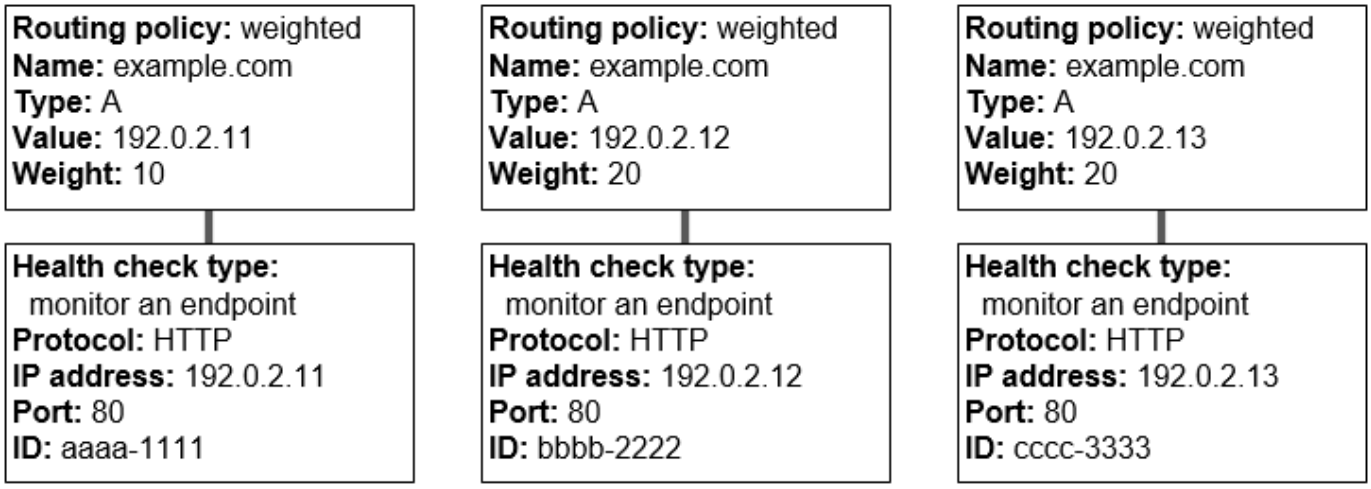
   For more information about creating health checks, see Creating, updating, and deleting health checks.
3. You might need to configure router and firewall rules so that Route 53 can send regular requests to the endpoints that you specified in your health checks. For more information, see Configuring router and firewall rules for Amazon Route 53 health checks.

4. You create a group of records for your resources, for example, a group of weighted records. You can mix alias and non-alias records, but they all must have the same value for **Name**, **Type**, and **Routing Policy**.

   How you configure Route 53 to check the health of your resources depends on whether you're creating alias records or non-alias records:

   - **Alias records** – Specify **Yes** for **Evaluate Target Health**.
   - **Non-alias records** – Associate the health checks that you created in step 2 with the corresponding records.

   When you're finished, your configuration looks similar to the following diagram, which includes only non-alias records.



   For more information about creating records by using the Route 53 console, see Creating records by using the Amazon Route 53 console.

5. If you created health checks, Route 53 periodically sends requests to the endpoint for each health check; it doesn't perform the health check when it receives a DNS query. Based on the responses, Route 53 decides whether the endpoints are healthy and uses that information to determine how to respond to queries. For more information, see How Amazon Route 53 determines whether a health check is healthy.

   Route 53 doesn't check the health of the resource specified in the record, such as the IP address that is specified in an A record for example.com. When you associate a health check with a record, Route 53 begins to check the health of the endpoint that you specified in the health check. You can also configure Route 53 to monitor the health of other health checks or

monitor the data streams for CloudWatch alarms. For more information, see Types of Amazon Route 53 health checks.

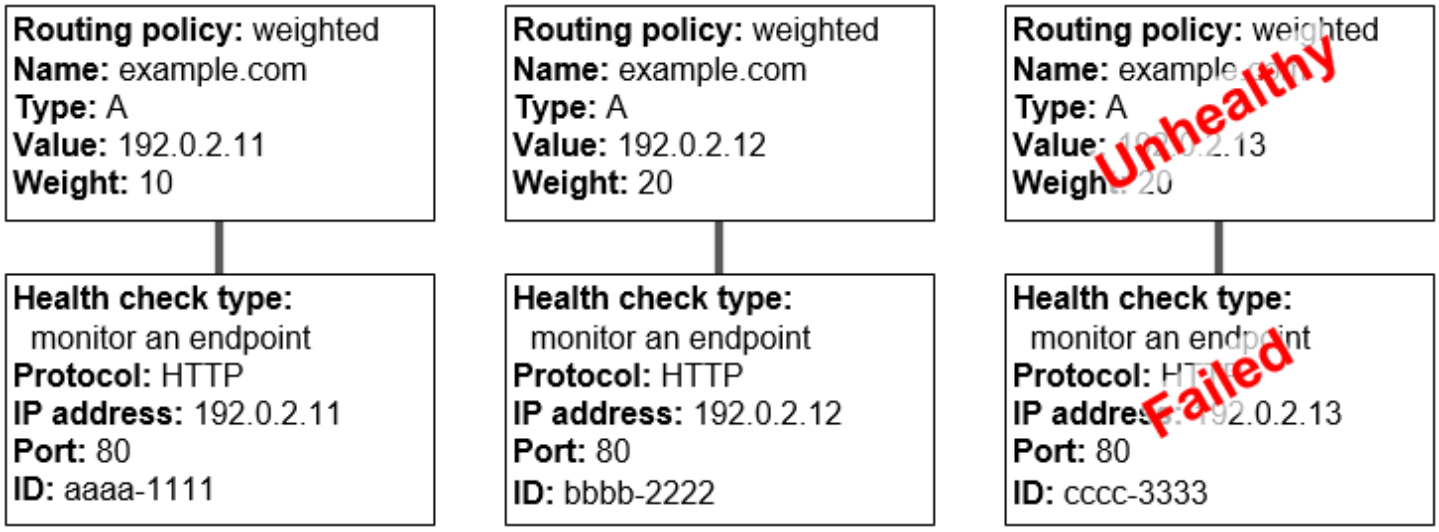
Here's what happens when Route 53 receives a query for example.com:

1. Route 53 chooses a record based on the routing policy. In this case, it chooses a record based on weight.
2. It determines the current health of the selected record by checking the status of the health check for that record.
3. If the selected record is unhealthy, Route 53 chooses a different record. This time, the unhealthy record isn't considered.

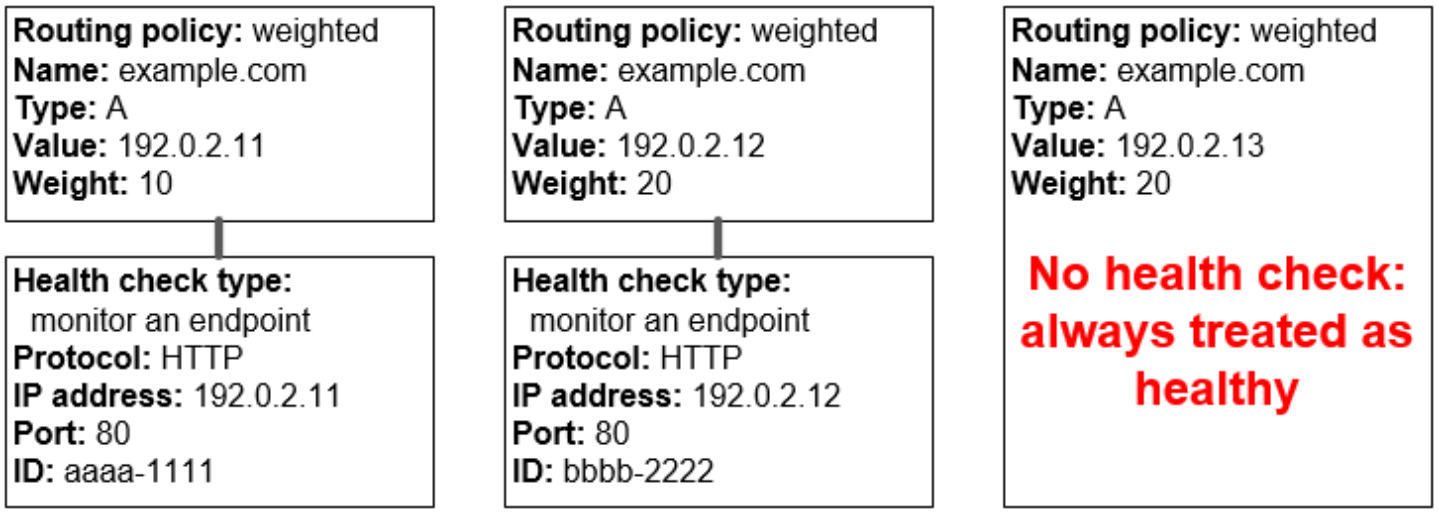
   For more information, see How Amazon Route 53 chooses records when health checking is configured.
4. When Route 53 finds a healthy record, it responds to the query with the applicable value, such as the IP address in an A record.

The following example shows a group of weighted records in which the third record is unhealthy. Initially, Route 53 selects a record based on the weights of all three records. If it happens to select the unhealthy record the first time, Route 53 selects another record, but this time it omits the weight of the third record from the calculation:

- When Route 53 initially selects from among all three records, it responds to requests using the first record about 20% of the time, 10/(10 + 20 + 20).
- When Route 53 determines that the third record is unhealthy, it responds to requests using the first record about 33% of the time, 10/(10 + 20).



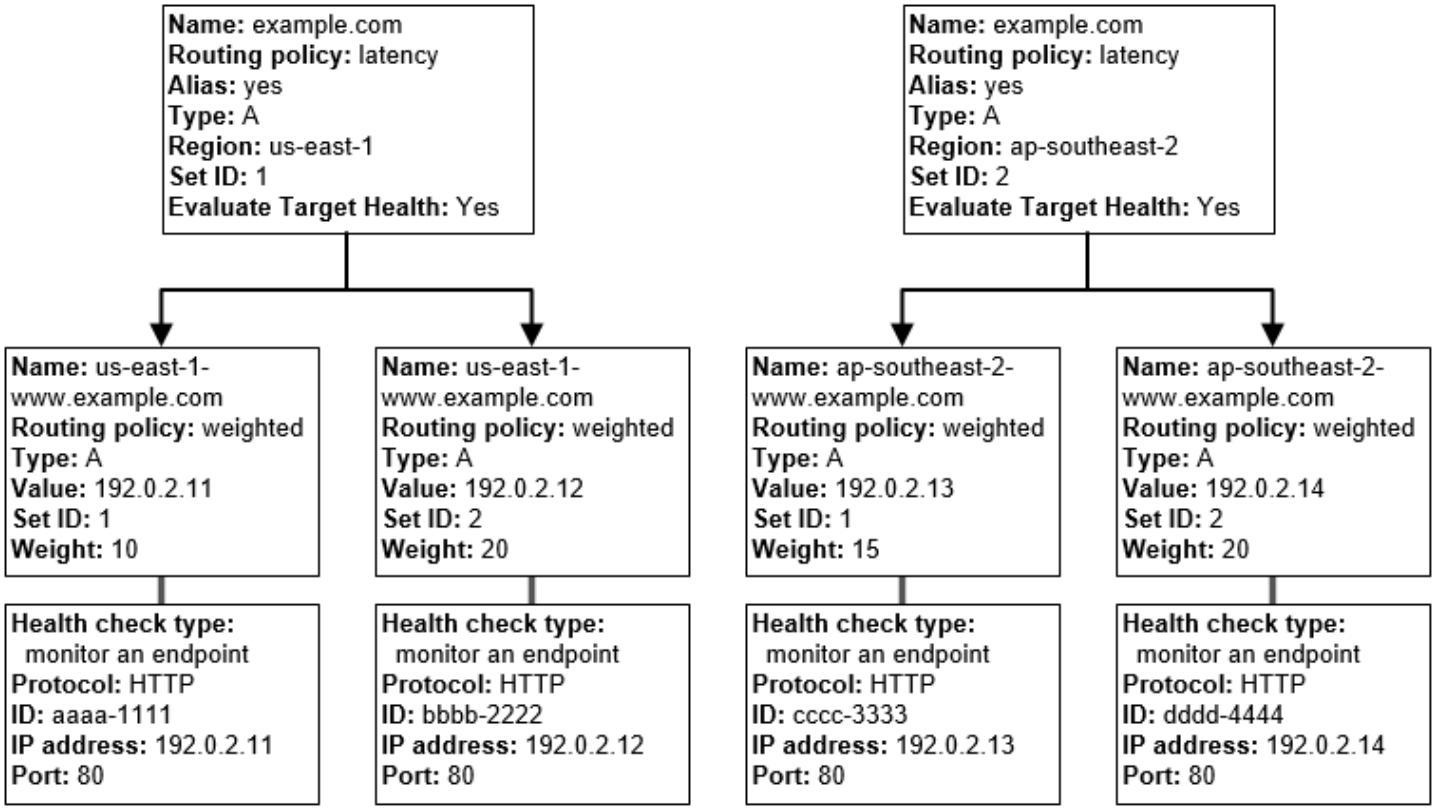
If you omit a health check from one or more records in a group of records, Route 53 has no way to determine the health of the corresponding resource. Route 53 treats those records as healthy.



How health checks work in complex Amazon Route 53 configurations

Checking the health of resources in complex configurations works much the same way as in simple configurations. However, in complex configurations, you use a combination of alias records (such as weighted alias and failover alias) and non-alias records to build a decision tree that gives you greater control over how Route 53 responds to requests.

For example, you might use latency alias records to select a Region close to a user and use weighted records for two or more resources within each Region to protect against the failure of a single endpoint or an Availability Zone. The following diagram shows this configuration.



Here's how Amazon EC2 and Route 53 are configured. Let's start at the bottom of the tree because that's the order that you'll create records in:

- You have two EC2 instances in each of two regions, us-east-1 and ap-southeast-2. You want Route 53 to route traffic to your EC2 instances based on whether they're healthy, so you create a health check for each instance. You configure each health check to send health-checking requests to the corresponding instance at the Elastic IP address for the instance.

  Route 53 is a global service, so you don't specify the Region that you want to create health checks in.

- You want to route traffic to the two instances in each Region based on the instance type, so you create a weighted record for each instance and give each record a weight. (You can change the weight later to route more or less traffic to an instance.) You also associate the applicable health check with each instance.

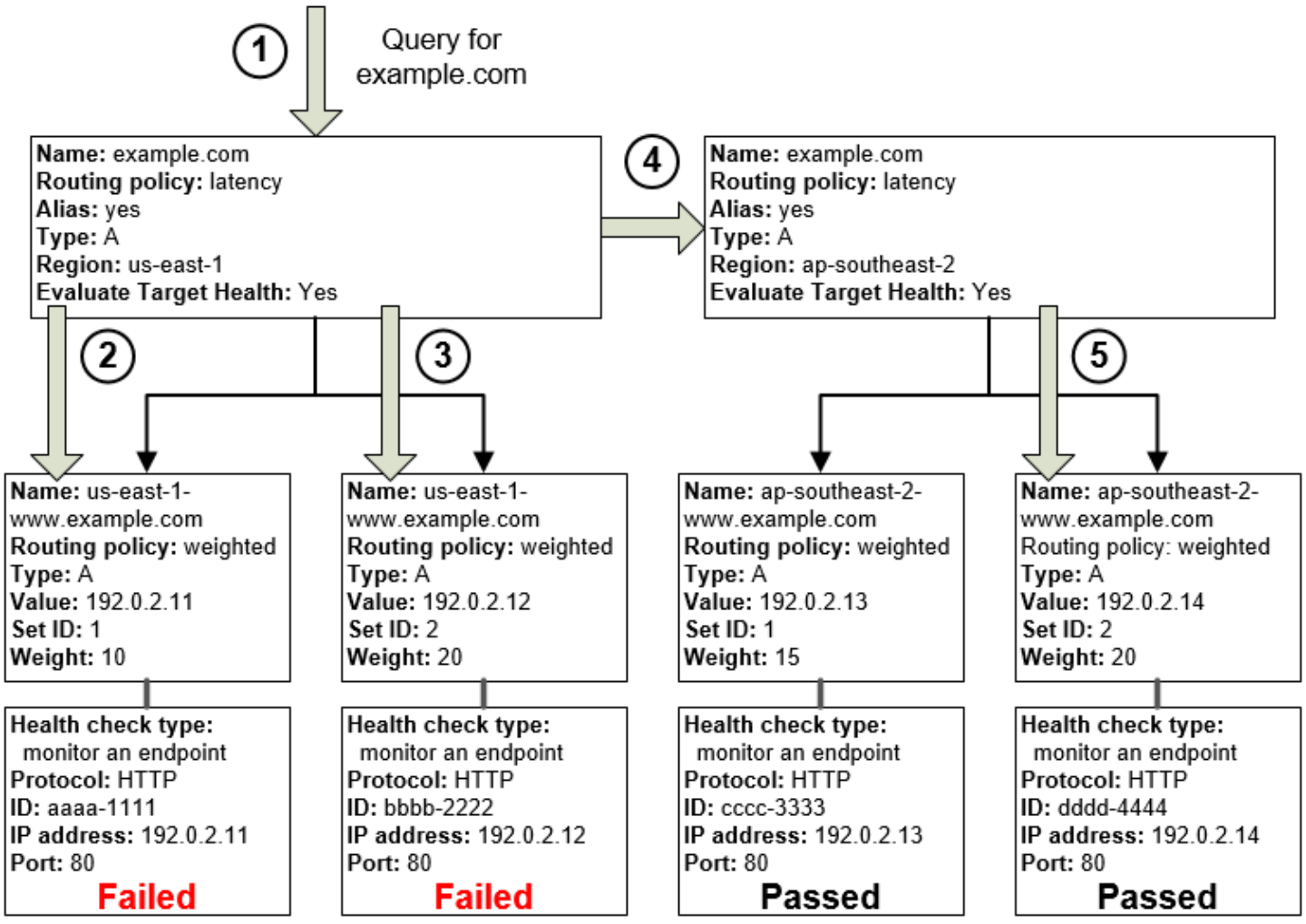
  When you create the records, you use names such as us-east-1-www.example.com. and ap-southeast-2-www.example.com. You'll wait until you get to the top of the tree to give records the names that your users will use to access your website or web application, such as example.com.

- You want to route traffic to the Region that provides the lowest latency for your users, so you choose the latency routing policy for the records at the top of the tree.

  You want to route traffic to the *records* in each Region, not directly to the *resources* in each Region (the weighted records already do that). As a result, you create latency alias records.

  When you create the alias records, you give them the name that you want your users to use to access your website or web application, such as example.com. The alias records route traffic for example.com to the us-east-1-www.example.com and ap-southeast-2-www.example.com records.

  For both latency alias records, you set the value of **Evaluate Target Health** to **Yes**. This causes Route 53 to determine whether there are any healthy resources in a Region before trying to route traffic there. If not, Route 53 chooses a healthy resource in the other Region.



The preceding diagram illustrates the following sequence of events:

1. Route 53 receives a query for example.com. Based on the latency for the user making the request, Route 53 selects the latency alias record for the us-east-1 region.
2. Route 53 selects a weighted record based on weight. **Evaluate Target Health** is **Yes** for the latency alias record, so Route 53 checks the health of the selected weighted record.
3. The health check failed, so Route 53 chooses another weighted record based on weight and checks its health. That record also is unhealthy.
4. Route 53 backs out of that branch of the tree, looks for the latency alias record with the next-best latency, and chooses the record for ap-southeast-2.
5. Route 53 again selects a record based on weight, and then checks the health of the selected resource. The resource is healthy, so Route 53 returns the applicable value in response to the query.

Topics

- What happens when you associate a health check with an alias record?
- What happens when you omit health checks?
- What happens when you set evaluate target health to No?

What happens when you associate a health check with an alias record?

You can associate a health check with an alias record instead of or in addition to setting the value of **Evaluate Target Health** to **Yes**. However, it's generally more useful if Route 53 responds to queries based on the health of the underlying resources—the HTTP servers, database servers, and other resources that your alias records refer to. For example, suppose the following configuration:

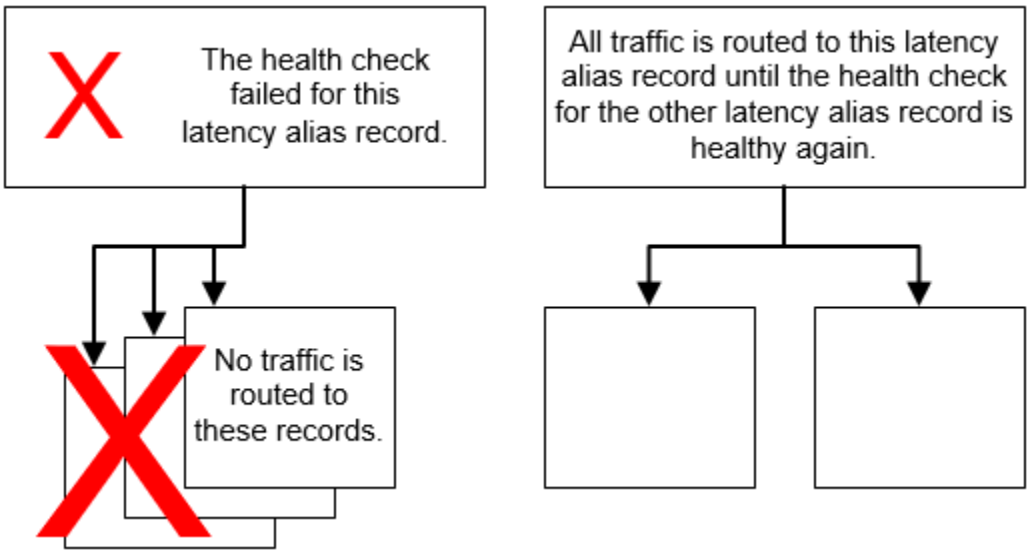
- You assign a health check to a latency alias record for which the alias target is a group of weighted records.
- You set the value of **Evaluate Target Health** to **Yes** for the latency alias record.

In this configuration, both of the following must be true before Route 53 will return the applicable value for a weighted record:

- The health check associated with the latency alias record must pass.

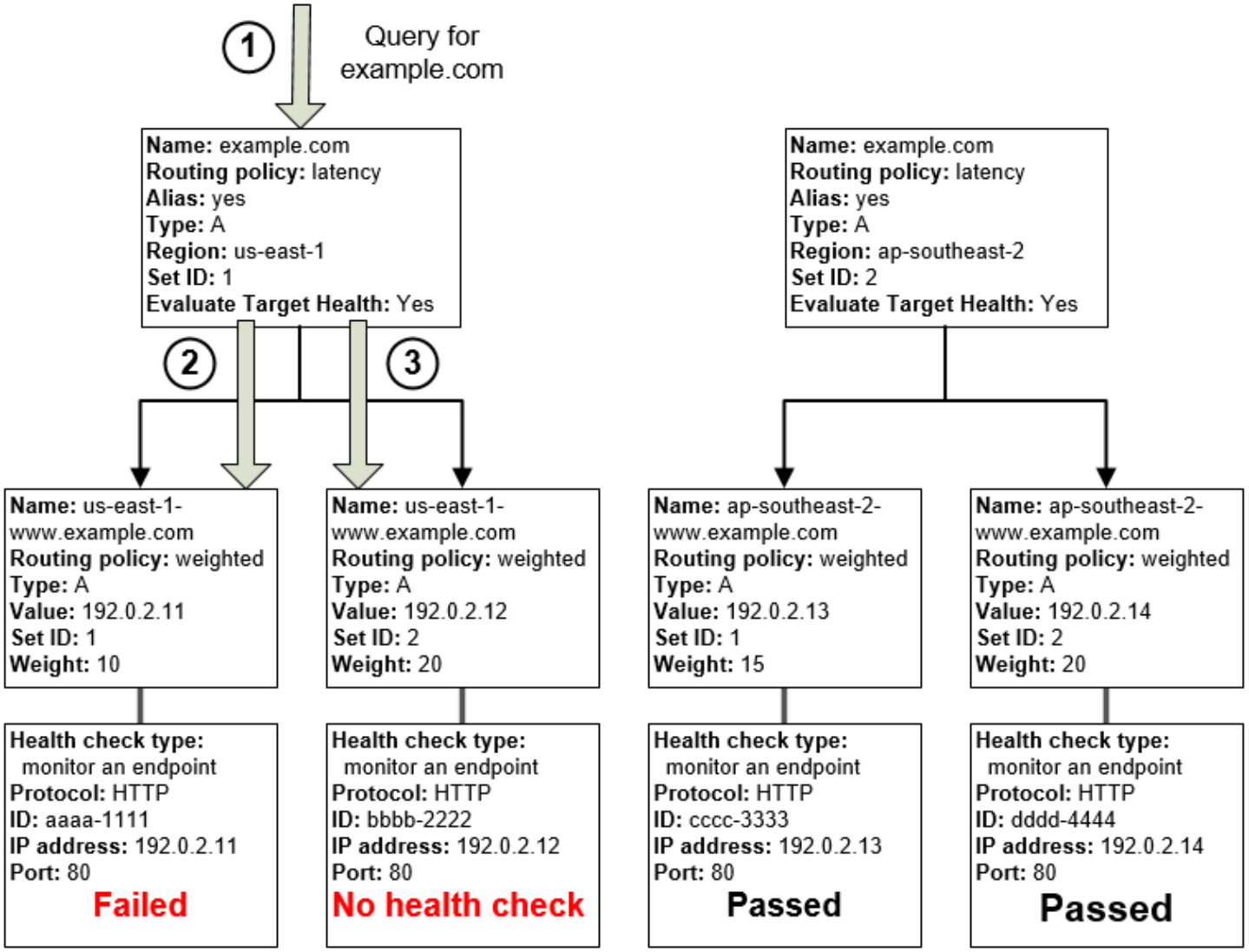
- At least one weighted record must be considered healthy, either because it's associated with a health check that passes or because it's not associated with a health check. In the latter case, Route 53 always considers the weighted record healthy.

In the following illustration, the health check for the latency alias record on the top left failed. As a result, Route 53 stops responding to queries using any of the weighted records that the latency alias record refers to even if they're all healthy. Route 53 begins to consider these weighted records again only when the health check for the latency alias record is healthy again. (For exceptions, see How Amazon Route 53 chooses records when health checking is configured.)



What happens when you omit health checks?

In a complex configuration, it's important to associate health checks with all the non-alias records. In the following example, a health check is missing on one of the weighted records in the us-east-1 region.



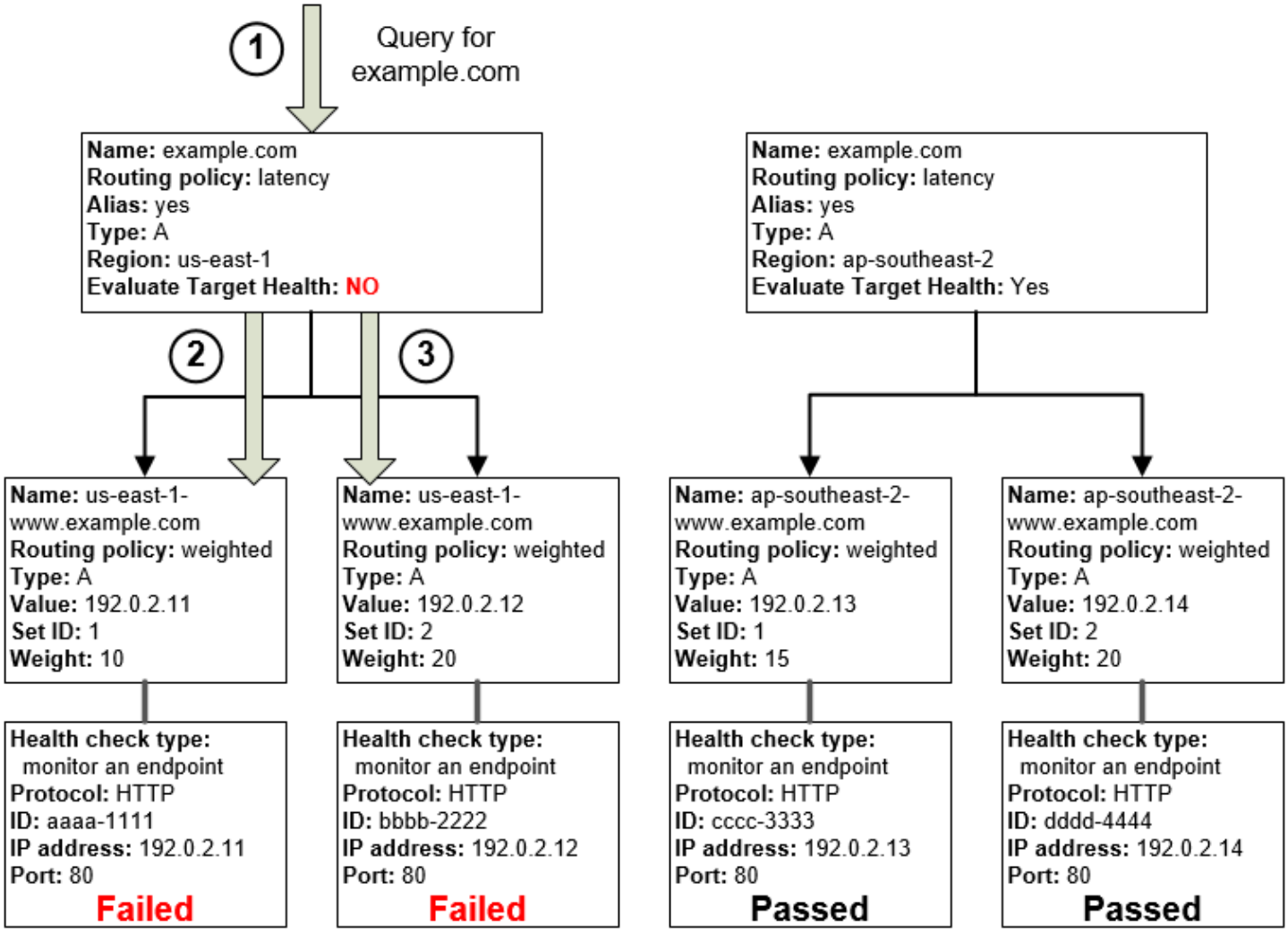
Here's what happens when you omit a health check on a non-alias record in this configuration:

1. Route 53 receives a query for example.com. Based on the latency for the user making the request, Route 53 selects the latency alias record for the us-east-1 region.
2. Route 53 looks up the alias target for the latency alias record, and checks the status of the corresponding health checks. The health check for one weighted record failed, so that record is omitted from consideration.
3. The other weighted record in the alias target for the us-east-1 region has no health check. The corresponding resource might or might not be healthy, but without a health check, Route 53 has no way to know. Route 53 assumes that the resource is healthy and returns the applicable value in response to the query.

What happens when you set evaluate target health to No?

In general, you should set **Evaluate Target Health** to **Yes** for all the alias records in a tree. If you set **Evaluate Target Health** to **No**, Route 53 continues to route traffic to the records that an alias record refers to even if health checks for those records are failing.

In the following example, all the weighted records have associated health checks, but **Evaluate Target Health** is set to **No** for the latency alias record for the us-east-1 region:



Here's what happens when you set **Evaluate Target Health** to **No** for an alias record in this configuration:

1. Route 53 receives a query for example.com. Based on the latency for the user making the request, Route 53 selects the latency alias record for the us-east-1 region.
2. Route 53 determines what the alias target is for the latency alias record, and checks the corresponding health checks. They're both failing.

3. Because the value of **Evaluate Target Health** is **No** for the latency alias record for the us-east-1 region, Route 53 must choose one record in this branch instead of backing out of the branch and looking for a healthy record in the ap-southeast-2 region.

How Amazon Route 53 chooses records when health checking is configured

If you configure health checking for all the records in a group of records that have the same name, the same type (such as A or AAAA), and the same routing policy (such as weighted or failover), Route 53 responds to DNS queries by choosing a healthy record and returning the applicable value from that record.

For example, suppose you create three weighted A records, and you assign health checks to all three. If the health check for one of the records is unhealthy, Route 53 responds to DNS queries with the IP addresses in one of the other two records.

Here's how Route 53 chooses a healthy record:

1. Route 53 initially chooses a record based on the routing policy and on the values that you specify for each record. For example, for weighted records, Route 53 chooses a record based on the weight that you specify for each record.
2. Route 53 determines whether the record is healthy:
   - **Non-alias record with an associated health check** – If you associated a health check with a non-alias record, Route 53 checks the current status of the health check.

     Route 53 periodically checks the health of the endpoint that is specified in a health check; it doesn't perform the health check when the DNS query arrives.

     You can associate health checks with alias records, but we recommend that you associate health checks only with non-alias records. For more information, see What happens when you associate a health check with an alias record?.
   - **Alias record with Evaluate Target Health set to Yes** – Route 53 checks the health status of the resource that the alias record references, for example, an ELB load balancer or another record in the same hosted zone.
3. If the record is healthy, Route 53 responds to the query with the applicable value, such as an IP address.

If the record is unhealthy, Route 53 chooses another record using the same criteria and repeats the process until it finds a healthy record.

Route 53 uses the following criteria when choosing a record:

Records without a health check are always healthy

If a record in a group of records that have the same name and type doesn't have an associated health check, Route 53 always considers it healthy and always includes it among possible responses to a query.

If no record is healthy, all records are healthy

If none of the records in a group of records are healthy, Route 53 needs to return something in response to DNS queries, but it has no basis for choosing one record over another. In this circumstance, Route 53 considers all the records in the group to be healthy and selects one based on the routing policy and on the values that you specify for each record.

Weighted records that have a weight of 0

If you add health checks to all the records in a group of weighted records, but you give nonzero weights to some records and zero weights to others, health checks work the same as when all records have nonzero weights with the following exceptions:

- Route 53 initially considers only the nonzero weighted records, if any.
- If all the records that have a weight greater than 0 are unhealthy, then Route 53 considers the zero-weighted records.

Because Route 53 will consider the zero-weighted records in some circumstances, it is important to make sure that the zero-weight target also has a viable answer to a DNS query.

For more information about weighted records, see Health checks and weighted routing.

Alias records

You can also configure health checking for alias records by setting **Evaluate Target Health** to **Yes** for each alias record. This causes Route 53 to evaluate the health of the resource that the record routes traffic to, for example, an ELB load balancer or another record in the same hosted zone.

For example, suppose the alias target for an alias record is a group of weighted records that all have nonzero weights:

- As long as at least one of the weighted records is healthy, Route 53 considers the alias record to be healthy.
- If none of the weighted records is healthy, Route 53 considers the alias record to be unhealthy.
- Route 53 stops considering records in that branch of the tree until at least one weighted record becomes healthy again.

For more information, see How health checks work in complex Amazon Route 53 configurations.

Failover records

Failover records generally work the same way as other routing types. You create health checks and associate them with non-alias records, and you set **Evaluate Target Health** to **Yes** for alias records. Note the following:

- Both the primary and secondary records can be a non-alias record or an alias record.
- If you associate health checks with both the primary and secondary failover records, here's how Route 53 responds to requests:
  - If Route 53 considers the primary record healthy (if the health check endpoint is healthy), Route 53 returns only the primary record in response to a DNS query.
  - If Route 53 considers the primary record unhealthy and the secondary record healthy, Route 53 returns the secondary record instead.
  - If Route 53 considers both the primary and secondary records unhealthy, Route 53 returns the primary record.
- When you're configuring the secondary record, adding a health check is optional. If you omit the health check for the secondary record, and if the health check endpoint for the primary record is unhealthy, Route 53 always responds to DNS queries by using the secondary record. This is true even if the secondary record is unhealthy.

For more information, see the following topics:

- Configuring active-passive failover with one primary and one secondary resource
- Configuring active-passive failover with multiple primary and secondary resources

AZ-affinity records

For AZ-affinity records, Route 53 first chooses the record for the most specific location that matches where the query comes from. The order is Availability Zone, Zone Group, AWS Region,

AWS partition, and then Global. If that record is unhealthy, Route 53 falls back to the record for the next less specific location that you created a record for. Note the following:

- A non-alias record that doesn't have a health check is always healthy, so Route 53 doesn't fall back past it. For an AZ-affinity alias record, set **Evaluate target health** to **Yes** if you want Route 53 to fall back when the alias target is unhealthy.
- If no record matches a query and you didn't create a Global record, Route 53 responds with no answer (NODATA).
- If all the records that match where the query comes from are unhealthy, Route 53 responds with the most specific of those records.

For more information, see Health checks and AZ-affinity routing.

Active-active and active-passive failover

You can use Route 53 health checking to configure active-active and active-passive failover configurations. You configure active-active failover using any routing policy (or combination of routing policies) other than failover, and you configure active-passive failover using the failover routing policy.

Topics

- Active-active failover
- Active-passive failover

Active-active failover

Use this failover configuration when you want all of your resources to be available the majority of the time. When a resource becomes unavailable, Route 53 can detect that it's unhealthy and stop including it when responding to queries.

In active-active failover, all the records that have the same name, the same type (such as A or AAAA), and the same routing policy (such as weighted or latency) are active unless Route 53 considers them unhealthy. Route 53 can respond to a DNS query using any healthy record.

Active-passive failover

Use an active-passive failover configuration when you want a primary resource or group of resources to be available the majority of the time and you want a secondary resource or group of

resources to be on standby in case all the primary resources become unavailable. When responding to queries, Route 53 includes only the healthy primary resources. If all the primary resources are unhealthy, Route 53 begins to include only the healthy secondary resources in response to DNS queries.

Topics

- Configuring active-passive failover with one primary and one secondary resource
- Configuring active-passive failover with multiple primary and secondary resources
- Configuring active-passive failover with weighted records

Configuring active-passive failover with one primary and one secondary resource

To create an active-passive failover configuration with one primary record and one secondary record, you just create the records and specify **Failover** for the routing policy. When the primary resource is healthy, Route 53 responds to DNS queries using the primary record. When the primary resource is unhealthy, Route 53 responds to DNS queries using the secondary record.

Configuring active-passive failover with multiple primary and secondary resources

You can also associate multiple resources with the primary record, the secondary record, or both. In this configuration, Route 53 considers the primary failover record to be healthy as long as at least one of the associated resources is healthy. For more information, see How Amazon Route 53 chooses records when health checking is configured.

To configure active-passive failover with multiple resources for the primary or secondary record, perform the following tasks.

1. Create a health check for each resource that you want to route traffic to, such as an EC2 instance or a web server in your data center.

   > **Note**
   >
   > If you're routing traffic to any AWS resources that you can create alias records for, don't create health checks for those resources. When you create the alias records, you set **Evaluate Target Health** to **Yes** instead.

   For more information, see Creating and updating health checks.

2. Create records for your primary resources, and specify the following values:
   - Give each record the same name, type, and routing policy. For example, you might create three weighted A records that are all named failover-primary.example.com.
   - If you're using AWS resources that you can create alias records for, specify **Yes** for **Evaluate Target Health.**

     If you're using resources that you can't create alias records for, associate the applicable health check from step 1 with each record.

   For more information, see Creating records by using the Amazon Route 53 console.
3. Create records for your secondary resources, if applicable, and specify the following values:
   - Give each record the same name, type, and routing policy. For example, you might create three weighted A records that are all named failover-secondary.example.com.
   - If you're using AWS resources that you can create alias records for, specify **Yes** for **Evaluate Target Health.**

     If you're using resources that you can't create alias records for, associate the applicable health check from step 1 with each record.

   > **Note**
   >
   > Some customers use a web server as their primary resource and an Amazon S3 bucket that is configured as a website endpoint as their secondary resource. The S3 bucket contains a simple "temporarily unavailable" message. If you're using that configuration, you can skip this step and just create a failover alias record for the secondary resource in step 4.

4. Create two failover alias records, one primary and one secondary, and specify the following values:

   Primary record

   - **Name** – Specify the domain name (example.com) or the subdomain name (www.example.com) that you want Route 53 to route traffic for.
   - **Alias** – Specify **Yes**.
   - **Alias Target** – Specify the name of the records that you created in step 2.
   - **Routing Policy** – Specify **Failover**.
   - **Failover Record Type** – Specify **Primary**.

- **Evaluate Target Health** – Specify **Yes**.
- **Associate with Health Check** – Specify **No**.

Secondary record

- **Name** – Specify the same name that you specified for the primary record.
- **Alias** – Specify **Yes**.
- **Alias Target** – If you created records for your secondary resource in step 3, specify the name of the records. If you're using an Amazon S3 bucket for the secondary resource, specify the DNS name of the website endpoint.
- **Routing Policy** – Specify **Failover**.
- **Failover Record Type** – Specify **Secondary**.
- **Evaluate Target Health** – Specify **Yes**.
- **Associate with Health Check** – Specify **No**.

Configuring active-passive failover with weighted records

You can also use weighted records for active-passive failover, with caveats. If you specify nonzero weights for some records and zero weights for other records, Route 53 responds to DNS queries using only healthy records that have nonzero weights. If all the records that have a weight greater than 0 are unhealthy, then Route 53 responds to queries using the zero-weighted records.

> **Note**
>
> All the records with nonzero weights must be unhealthy before Route 53 starts to respond to DNS queries using records that have weights of zero. This can make your web application or website unreliable if the last healthy resource, such as a web server, can't handle all the traffic when other resources are unavailable.

Configuring failover in a private hosted zone

If you're creating failover records in a private hosted zone, note the following:

- Route 53 health checkers are outside the VPC. To check the health of an endpoint within a VPC by IP address, you must assign a public IP address to the instance in the VPC.

- You can create a CloudWatch metric, associate an alarm with the metric, and then create a health check that is based on the data stream for the alarm. For example, you might create a CloudWatch metric that checks the status of the EC2 `StatusCheckFailed` metric, add an alarm to the metric, and then create a health check that is based on the data stream for the alarm to check instances within a Virtual Private Cloud (VPC) that only have private IP addresses. For information about creating CloudWatch metrics and alarms by using the CloudWatch console, see the Amazon CloudWatch User Guide.

For more information, see Working with private hosted zones and Monitoring health checks using CloudWatch.

How Amazon Route 53 averts failover problems

The failover algorithms implemented by Route 53 are designed not only to route traffic to endpoints that are healthy, but also to avoid making disaster scenarios worse due to misconfigured health checks and applications, endpoint overloads, and partition failures.

Topics

- How Amazon Route 53 averts cascading failures
- How Amazon Route 53 handles internet partitions

How Amazon Route 53 averts cascading failures

As a first defense against cascading failures, each request routing algorithm (such as weighted and failover) has a mode of last resort. In this special mode, when all records are considered unhealthy, the Route 53 algorithm reverts to considering all records healthy.

For example, if all instances of an application, on several hosts, are rejecting health check requests, Route 53 DNS servers will choose an answer anyway and return it rather than returning no DNS answer or returning an NXDOMAIN (non-existent domain) response. An application can respond to users but still fail health checks, so this provides some protection against misconfiguration.

Similarly, if an application is overloaded, and one out of three endpoints fails its health checks, so that it's excluded from Route 53 DNS responses, Route 53 distributes responses between the two remaining endpoints. If the remaining endpoints are unable to handle the additional load and they fail, Route 53 reverts to distributing requests to all three endpoints.

How Amazon Route 53 handles internet partitions

Although uncommon, there occasionally are significant internet partitions, meaning that large geographic regions can't communicate with one another over the internet. During these partitions, Route 53 locations might reach different conclusions about the health status of an endpoint and might differ from the status reported to CloudWatch. Route 53 health checkers in each AWS Region are constantly sending health check statuses to all Route 53 locations. During internet partitions, each Route 53 location might have access only to a partial set of these statuses, usually from its closest regions.

For example, during an internet partition that affects connectivity to and from South America, the Route 53 DNS servers in the Route 53 South America (São Paulo) location might have good access to the health check endpoints in the South America (São Paulo) AWS Region, but poor access to endpoints elsewhere. At the same time, Route 53 in US East (Ohio) might have poor access to health check endpoints in the South America (São Paulo) Region, and conclude that the corresponding records are unhealthy.

Partitions such as these can give rise to situations where Route 53 locations make different conclusions about the health status of endpoints, based on their local visibility of those endpoints. This is why each Route 53 location considers an endpoint healthy when only a portion of reachable health checkers consider it healthy.

Naming and tagging health checks

You can add tags to Amazon Route 53 health checks to give each health check a name that is more comprehensible than the health check ID. These are the same tags that AWS Billing and Cost Management provides for organizing your AWS bill. For more information about using tags for cost allocation, see Use cost allocation tags for custom billing reports in the *AWS Billing User Guide*.

Each tag consists of a key (the name of the tag) and a value, both of which you define. When you add tags to a health check, we recommend that you add one tag that has the following values for the key and value:

- **key** – **Name**
- **value** – The name that you want to give to the health check

The value of the **Name** tag appears in the list of health checks on the Route 53 console, so you can readily distinguish health checks from one another. To see other tags for a health check, you choose the health check and then choose the **Tags** tab.

For more information about tags, see the following topics:

- To add, edit, or delete the **Name** tag when you add or edit health checks in the Route 53 console, see Creating and updating health checks.
- For an overview of tagging Route 53 resources, see Tagging Amazon Route 53 resources.

Tag restrictions

The following basic restrictions apply to tags:

- Maximum number of tags per resource – 50.
- Maximum **Key** length – 128 Unicode characters
- Maximum **Value** length – 256 Unicode characters
- Valid values for **Key** and **Value** – uppercase and lowercase letters in the UTF-8 character set, numbers, space, and the following characters: _ . : / = + - and @
- Tag keys and values are case sensitive
- Don't use the `aws:` prefix for either keys or values; it's reserved for AWS use

Adding, editing, and deleting tags for health checks

The following procedures show you how to use tags for your health checks on the Route 53 console.

To add tags to health checks

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**.
3. Select the linked ID of the health check for which you want to add tags.
4. In the bottom page, choose the **Tags** tab, and then choose **Manage** and then **Add new tags**.
5. Enter a name for the tag in the **Key** field, and enter a value in the **Value** field.
6. Choose **Save**.

To edit tags for health checks

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**.
3. Select the linked ID of a health check.
4. In the bottom pane, choose the **Tags** tab, and then choose **Manage**.
5. You can now edit and add more tags.
6. Choose **Save**.

To delete tags for health checks

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**.
3. Select the linked ID of a health check.
4. In the bottom pane, choose the **Tags** tab, and then choose **Manage**.
5. Choose **Remove** next to the tag that you want to delete.
6. Choose **Save**.

Using health checks with Amazon Route 53 API versions earlier than 2012-12-12

Health checks are supported starting with the 2012-12-12 version of the Amazon Route 53 API. If a hosted zone contains records that health checks are configured for, we recommend that you use only the 2012-12-12 API or later. Note the following restrictions on using health checks with earlier API versions.

- The `ChangeResourceRecordSets` action cannot create or delete records that include the `EvaluateTargetHealth`, `Failover`, or `HealthCheckId` elements.
- The `ListResourceRecordSets` action can list records that include these elements, but the elements are not included in the output. Instead, the `Value` element of the response contains a message that says the record includes an unsupported attribute.

Monitoring health check status and getting notifications

You monitor the status of your health checks on the Amazon Route 53 console. You can also set CloudWatch alarms and get automated notifications when the status of your health check status changes.

Topics

- Viewing health check status and the reason for health check failures
- Monitoring the latency between health checkers and your endpoint
- Monitoring health checks using CloudWatch

Viewing health check status and the reason for health check failures

On the Route 53 console, you can view the status (healthy or unhealthy) of your health checks as reported by Route 53 health checkers. For all health checks except calculated health checks, you can also view the reason for the last health check failure, for example, health checkers were unable to establish a connection with the endpoint.

To view the status and last failure reason for a health check

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**.
3. For an overview of the status of all of your health checks—healthy or unhealthy—view the **Status** column. For more information, see How Amazon Route 53 determines whether a health check is healthy.
4. For all health checks except calculated health checks, you can view the status of the Route 53 health checkers that are checking the health of a specified endpoint.
5. Choose the linked ID of the health check you want to view details for.
6. In the bottom pane, choose the **Health checkers** tab.

> **Note**
>
> New health checks must propagate to Route 53 health checkers before the health check status and last failure reason appear in the **Status** column. Until propagation has finished, the message in that column explains that no status is available.

7. The table includes the following values:

   Health checker IP

   The IP address of the Route 53 health checker that performed the health check.

   Last checked

   The date and time of the health check or the date and time of the last failure.

   Status

   Either the current status of the health check or the reason for the last health check failure.

Monitoring the latency between health checkers and your endpoint

When you create a health check, if you choose to monitor the status of an endpoint (not the status of other health checks) and you choose the **Latency graphs** option, you can view the following values on CloudWatch graphs on the Route 53 console:

- The average time, in milliseconds, that it took Route 53 health checkers to establish a TCP connection with the endpoint
- The average time, in milliseconds, that it took Route 53 health checkers to receive the first byte of the response to an HTTP or HTTPS request
- The average time, in milliseconds, that it took Route 53 health checkers to complete the SSL/TLS handshake

> **Note**
>
> You can't enable latency monitoring for existing health checks.

> **Important**
>
> The health checkers are run on 16 redundant availability zones. Occasionally an availability zone can be unavailable because of deployments, updates, maintenance, and so on. The health check system is designed to account for this without any customer impact.

To view the latency between Route 53 health checkers and your endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**.
3. Select the linked ID for the health check for which you want to view metrics. You can view latency data only for health checks that monitor the status of an endpoint and for which the **Latency graphs** option is enabled.
4. In the bottom pane, choose the **Metrics** tab.
5. Choose the time range and the geographic region that you want to display latency graphs for.

   The graphs display the status for the specified time range:

   TCP connection time (HTTP and TCP only)

   The average time, in milliseconds, that it took Route 53 health checkers in the selected geographic region to establish a TCP connection with the endpoint.

   Time to first byte (HTTP and HTTPS only)

   The average time, in milliseconds, that it took Route 53 health checkers in the selected geographic region to receive the first byte of the response to an HTTP or HTTPS request.

   Time to complete SSL handshake (HTTPS only)

   The average time, in milliseconds, that it took Route 53 health checkers in the selected geographic region to complete the SSL/TLS handshake.
6. To view a larger graph and specify different settings, choose the three dots on the upper right of the graph. You can change the following settings:

   Statistic

   Changes the calculation that CloudWatch performs on the data.

Time range

Displays the status of a health check over a different period, for example, overnight or last week.

Period

Changes the interval between data points in the graph.

Note the following:

- If you just created a health check, you might need to wait for a few minutes for data to appear in the graph and for the health check metric to appear in the list of available metrics.
- The graph doesn't refresh itself automatically. To update the display, choose the refresh ( ) icon.
- If health checks are failing for some reason, such as a connection timeout, Route 53 can't measure latency, and latency data will be missing from the graph for the affected period.

Monitoring health checks using CloudWatch

Route 53 health checks integrate with CloudWatch metrics so that you can do the following:

- Verify that a health check is properly configured.
- Review the status of a health check over a specified period of time.
- Configure CloudWatch to send an Amazon SNS alert when the status of a health check is unhealthy. Note that several minutes might elapse between the time that a health check fails and the time that you receive the associated SNS notification.

For more information, see How Amazon Route 53 determines whether a health check is healthy.

Topics

- View the status of your health check
- View health check alarms
- View health check metrics on the CloudWatch console
- Create an alarm with an SNS notification

View the status of your health check

To view the status of a health check

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Health checks**.
3. Choose the linked ID of the health check for which you want to view metrics.
4. In the bottom pane, choose the **Metrics** tab.

   The two graphs display the status for the last hour in one-minute intervals:

   Health check status

   The graph shows the Route 53 assessment of endpoint health. **1** indicates healthy and **0** indicates unhealthy.

   Health checkers that report the endpoint healthy (%)

   For health checks that monitor an endpoint only, the graph shows the percentage of Route 53 health checkers that consider the selected endpoint to be healthy.

   When a health check is disabled, this metric isn't available.

   Number of healthy child health checks

   For calculated health checks only, the graph shows the number of child health checks that are healthy.
5. To view a larger graph and specify different settings, choose the three dots on the upper right, and then **Enlarge**. You can change the following settings:

   Statistic

   Changes the calculation that CloudWatch performs on the data.

   Time range

   Displays the status of a health check over a different period, for example, overnight or last week.

   Period

   Changes the interval between data points in the graph.

Note the following:

- If you just created a health check, you might need to wait for a few minutes for data to appear in the graph and for the health check metric to appear in the list of available metrics.
- The graph doesn't refresh itself automatically. To update the display, choose the refresh ( ) icon.

View health check alarms

To view CloudWatch alarm status and edit alarms for Amazon Route 53

1. In the navigation pane of the Route 53 console, choose **Health checks**.
2. Choose the linked ID of the health check for which you want to view the alarms.
3. In the details page, on the bottom of the page, choose the **Alarms** tab.

   The **Alarms** list contains all the Route 53 alarms that you have created for the selected health check.

   The **State** column shows the current status of each alarm:

   OK

   CloudWatch has accumulated enough statistics from Route 53 health checks to determine that the endpoint doesn't meet the alarm threshold.

   INSUFFICIENT DATA

   CloudWatch hasn't accumulated enough statistics to determine whether the endpoint meets the alarm threshold. This is the initial state of a new alarm. The alarm state also changes to **INSUFFICIENT DATA** if CloudWatch metrics become unavailable or if you delete the health check without deleting the associated alarm.

   ALARM

   CloudWatch has accumulated enough statistics from Route 53 health checks to determine that the endpoint meets the alarm threshold and to send notification to the specified email address.

4. To view an alarm in the CloudWatch console, which provides more detailed information about the alarm (for example, a history of updates to the alarm and changes in status), choose the linked name of the alarm. You can also edit the alarm on the CloudWatch console.
5. To create a new CloudWatch alarm on the CloudWatch console, choose **Create a CloudWatch alarm**. For more information, see Find and create recommended alarms in the *CloudWatch User guide*.

View health check metrics on the CloudWatch console

To view Route 53 metrics on the CloudWatch console

1. Sign in to the AWS Management Console and open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. Change the current region to **US East (N. Virginia)**. Route 53 metrics are not available if you select any other region as the current region.
3. In the navigation pane, choose **Metrics**.
4. On the **All metrics** tab, choose **Route 53**.
5. Choose **Health Check Metrics**.
6. You can also set up SNS notification on the CloudWatch console. For more information, see Create recommended alarms in the *CloudWatch User Guide*.

Create an alarm with an SNS notification

To create an alarm that sends an Amazon SNS notification when a health check is unhealthy, use the CloudWatch console. For more information, see Find and create recommended alarms in the *CloudWatch User Guide*.

Using DNS Firewall to filter outbound DNS traffic

With Resolver DNS Firewall, you can filter and regulate outbound DNS traffic for your virtual private cloud (VPC). To do this, you create reusable collections of filtering rules in DNS Firewall rule groups, associate the rule groups to your VPC, and then monitor activity in DNS Firewall logs and metrics. Based on the activity, you can adjust the behavior of DNS Firewall accordingly.

DNS Firewall provides protection for outbound DNS requests from your VPCs. These requests route through VPC Resolver for domain name resolution. A primary use of DNS Firewall protections is to help prevent DNS exfiltration of your data. DNS exfiltration can happen when a bad actor compromises an application instance in your VPC and then uses DNS lookup to send data out of the VPC to a domain that they control. With DNS Firewall, you can monitor and control the domains that your applications can query. You can deny access to the domains that you know to be bad and allow all other queries to pass through. Alternatively, you can deny access to all domains except for the ones that you explicitly trust.

You can also use DNS Firewall to block resolution requests to resources in private hosted zones (shared or local) including VPC endpoint names. It can also block requests for public or private Amazon EC2 instance names.

DNS Firewall is a feature of Route 53 VPC Resolver and doesn't require any additional VPC Resolver setup to use.

AWS Firewall Manager supports DNS Firewall

You can use Firewall Manager to centrally configure and manage your DNS Firewall rule group associations for your VPCs across your accounts in AWS Organizations. Firewall Manager automatically adds associations for VPCs that come into scope of your Firewall Manager DNS Firewall policy. For more information, see AWS Firewall Manager in the *AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide*.

How DNS Firewall works with AWS Network Firewall

DNS Firewall and Network Firewall both offer domain-based filtering, but they act at different points in a connection.

- DNS Firewall provides filtering for outbound DNS queries that pass through the Route 53 VPC Resolver from applications within your VPCs. You can also configure DNS Firewall to send custom responses for queries to blocked domain names.

- Network Firewall provides filtering for both network and application layer traffic, but does not have visibility into queries made by Route 53 VPC Resolver.

You can also use DNS Firewall and Network Firewall together for resolved IP validation. Network Firewall checks the DNS resolutions from VPCs that are associated with a DNS Firewall rule group to confirm that the VPC Resolver returned a connection's destination IP address. For more information, see Resolved IP validation with Network Firewall.

For more information about Network Firewall, see the Network Firewall Developer Guide.

How Resolver DNS Firewall works

With Resolver DNS Firewall, you can control access to sites and block DNS-level threats for DNS queries going out from your VPC through the Route 53 VPC Resolver. With DNS Firewall, you define domain name filtering rules in rule groups that you associate with your VPCs. You can specify lists of domain names to allow or block, or Resolver DNS Firewall Advanced rules that offer protection from DNS tunneling and Domain Generation Algorithm (DGA) based threats. You can customize the responses for the DNS queries that you block. For rules that contain a domain list, you can also fine-tune the rule to allow certain query types, such as MX-records, through.

DNS Firewall only filters on the domain name. It does not resolve that name to an IP address to be blocked. Additionally, DNS Firewall filters DNS traffic, but it doesn't filter other application layer protocols, such as HTTPS, SSH, TLS, FTP, and so on.

Resolver DNS Firewall components and settings

You manage DNS Firewall with the following central components and settings.

DNS Firewall rule group

Defines a named, reusable collection of DNS Firewall rules for filtering DNS queries. You populate the rule group with the filtering rules, then associate the rule group with one or more VPCs. When you associate a rule group with a VPC, you enable DNS Firewall filtering for the VPC. Then, when VPC Resolver receives a DNS query for a VPC that has a rule group associated with it, VPC Resolver passes the query to DNS Firewall for filtering.

If you associate multiple rule groups with a single VPC, you indicate their processing order through the priority setting in each association. DNS Firewall processes rule groups for a VPC from the lowest numeric priority setting on up.

For more information, see DNS Firewall rule groups and rules.

DNS Firewall rule

Defines a filtering rule for DNS queries in a DNS Firewall rule group. Each rule specifies one domain list, or DNS Firewall protection and an action to take on DNS queries whose domains match the domain specifications in the rule. You can allow (rules with domain lists only), block, or alert on matching queries. In rules with domain lists you can also specify query types for the domains in the list, for example, you can block or allow an MX query type for a specific domain or domains. You can also define custom responses for blocked queries.

For DNS Firewall rules you can only block or alert on matching queries.

Each rule in a rule group has a priority setting that's unique within the rule group. DNS Firewall processes the rules in a rule group from the lowest numeric priority setting on up.

DNS Firewall rules exist only in the context of the rule group in which they're defined. You can't reuse a rule or reference it independent of its rule group.

For more information, see DNS Firewall rule groups and rules.

Domain list

Defines a named, reusable collection of domain specifications for use in DNS filtering. Each rule in a rule group requires a single domain list. You might choose to specify the domains that you want to allow access to, the domains that you want to deny access to, or a combination of both. You can create your own domain lists and you can use domain lists that AWS manages for you.

For more information, see DNS Firewall Foundational Rules.

Domain redirection setting (Domain lists only)

The domain redirection setting allows you to configure a DNS Firewall rule to inspect all the domains in the DNS redirection chain (default), such as CNAME, DNAME, etc., or just the first domain and trust the rest. If you choose to inspect the entire DNS redirection chain, you must add the subsequent domains to a domain list set to ALLOW in the rule. If you choose to inspect the entire DNS redirection chain, you must add the subsequent domains to a domain list and set to the action you want the rule to take, either ALLOW, BLOCK, or ALERT.

> **Important**
>
> The trust behavior of the domain redirection setting only applies within a single DNS query transaction. If a DNS client on your host separately queries a domain that appears

in a DNS redirection chain (for example, querying the redirection target directly), DNS Firewall evaluates it as an independent query with no trust context from the original query. To allow such queries, add the redirection target domains to your domain list.

For more information, see Rule settings in DNS Firewall.

Query type (Domain lists only)

The query type setting allows you to configure a DNS Firewall rule to filter a particular DNS query type. If you don't select a query type, the rule is applied to all DNS query types. For example, you might want to block all the query types for a particular domain, but allow MX records.

For more information, see Rule settings in DNS Firewall.

DNS Firewall Advanced protection

Detects suspicious DNS queries based on known threat signatures in DNS queries. Each rule in a rule group requires a single DNS Firewall Advanced protection setting. You can choose protection from:

- Domain Generation Algorithms (DGAs)

  DGAs are used by attackers to generate a large number of domains to launch malware attacks.

- DNS tunneling

  DNS tunneling is used by attackers to exfiltrate data from the client by using the DNS tunnel without making a network connection to the client.

- Dictionary DGA

  Dictionary DGAs are used by attackers to generate domains using dictionary words to evade detection in malware command-and-control communications.

In a DNS Firewall Advanced rule you can choose to either block, or alert on a query that matches the threat. The threat protection algorithms are managed and updated by AWS.

For more information, see DNS Firewall Advanced Rules.

Confidence threshold(DNS Firewall Advanced protection only)

The confidence threshold for DNS threat protection. You must provide this value when you create a DNS Firewall Advanced rule. The confidence level values mean:

- High – Detects only the most well corroborated threats with a low rate of false positives.
- Medium – Provides a balance between detecting threats and false positives.
- Low – Provides the highest detection rate for threats, but also increases false positives.

For more information, see Rule settings in DNS Firewall.

Association between a DNS Firewall rule group and a VPC

Defines a protection for a VPC using a DNS Firewall rule group and enables the VPC Resolver DNS Firewall configuration for the VPC.

If you associate multiple rule groups with a single VPC, you indicate their processing order through the priority setting in the associations. DNS Firewall processes rule groups for a VPC from the lowest numeric priority setting on up.

For more information, see Enabling Resolver DNS Firewall protections for your VPC.

DNS Firewall configuration for a VPC

Specifies how VPC Resolver should handle DNS Firewall protections at the VPC level. This configuration is in effect whenever you have at least one DNS Firewall rule group associated with the VPC.

This configuration specifies how Route 53 VPC Resolver handles queries when DNS Firewall fails to filter them. By default, if VPC Resolver doesn't receive a response from DNS Firewall for a query, it fails closed and blocks the query.

For more information, see DNS Firewall VPC configuration.

Monitoring DNS Firewall actions

You can use Amazon CloudWatch to monitor the number of DNS queries that are filtered by DNS Firewall rule groups. CloudWatch collects and processes raw data into readable, near real-time metrics.

For more information, see Monitoring Resolver DNS Firewall rule groups with Amazon CloudWatch.

You can use Amazon EventBridge, a serverless service that uses events to connect application components together, to build scalable event-driven applications.

For more information, see Managing Resolver DNS Firewall events using Amazon EventBridge.

How Resolver DNS Firewall filters DNS queries

When a DNS Firewall rule group is associated with your VPC's Route 53 VPC Resolver, the following traffic is filtered by the firewall:

- DNS queries originating within that VPC and passing through VPC DNS.
- DNS queries that pass through Resolver endpoints from on-premises resources into that same VPC that has DNS Firewall associated to its resolver.

When DNS Firewall receives a DNS query, it filters the query using the rule groups, rules, and other settings that you've configured and sends the results back to VPC Resolver:

- DNS Firewall evaluates the DNS query using the rule groups that are associated with the VPC until it finds a match or exhausts all of the rule groups. DNS Firewall evaluates the rule groups in order of the priority that you set in the association, starting with the lowest numeric setting. For more information, see DNS Firewall rule groups and rules and Enabling Resolver DNS Firewall protections for your VPC.
- Within each rule group, DNS Firewall evaluates the DNS query against each rule's domain list or DNS Firewall Advanced protections until it finds a match or exhausts all rules. DNS Firewall evaluates the rules in order of priority, starting with the lowest numeric setting. For more information, see DNS Firewall rule groups and rules.
- When DNS Firewall finds a match with a rule's domain list, or anomalies identified by DNS Firewall Advanced rule protections, it terminates the query evaluation and responds to VPC Resolver with the result. If the action is `alert`, DNS Firewall also sends an alert to the configured VPC Resolver logs. For more information, see Rule actions in DNS Firewall, DNS Firewall Foundational Rules, and DNS Firewall Advanced Rules.
- If DNS Firewall evaluates all rule groups without finding a match, it responds to the query as normal.

VPC Resolver routes the query according to the response from DNS Firewall. In the unlikely event that DNS Firewall fails to respond, VPC Resolver applies the VPC's configured DNS Firewall fail mode. For more information, see DNS Firewall VPC configuration.

High-level steps for using Resolver DNS Firewall

To implement Resolver DNS Firewall filtering in your Amazon Virtual Private Cloud VPC, you perform the following high-level steps.

- **Define your filtering approach, your domain lists, or DNS Firewall protections** – Decide how you want to filter queries, identify the domain specifications that you'll need, and define the logic you'll use to evaluate queries. For example, you might want to allow all queries except for those that are in a list of known bad domains. Or you might want to do the opposite and block all but an approved list of domains, in what is known as a walled garden approach. You can create and manage your own lists of approved or blocked domain specifications and you can use domain lists that AWS manages for you. For DNS Firewall protections you can filter the queries by blocking them all, or you can alert on any suspicious query traffic to domains that might contain anomalies associated with threats (DGA, DNS tunneling, Dictionary DGA) to test your DNS Firewall settings. For more information, see DNS Firewall Foundational Rules and DNS Firewall Advanced Rules.
- **Create a firewall rule group** – In DNS Firewall, create a rule group to filter DNS queries for your VPC. You must create a rule group in each Region where you want to use it. You might also want to separate your filtering behavior into more than one rule group for reusability in multiple filtering scenarios for your different VPCs. For information about rule groups, see DNS Firewall rule groups and rules.
- **Add and configure your rules** – Add a rule to your rule group for each domain list and filtering behavior that you want the rule group to provide. Set the priority settings for your rules so they process in the correct order within the rule group, giving the lowest priority to the rule that you want to evaluate first. For information about rules, see DNS Firewall rule groups and rules.
- **Associate the rule group to your VPC** – To begin using your DNS Firewall rule group, associate it with your VPC. If you are using more than one rule group for your VPC, set the priority of each association so the rule groups are processed in the correct order, giving the lowest priority to the rule group that you want to evaluate first. For more information, see Managing associations between your VPC and Resolver DNS Firewall rule group.
- **(Optional) Change the firewall configuration for the VPC** – If you want Route 53 VPC Resolver to block queries when DNS Firewall fails to send a response back for them, in VPC Resolver,

change the VPC's DNS Firewall configuration. For more information, see DNS Firewall VPC configuration.

Using Resolver DNS Firewall rule groups in multiple Regions

Resolver DNS Firewall is a Regional service, so objects that you create in one AWS Region are available only in that Region. To use the same rule group in more than one Region, you must create it in each Region.

The AWS account that created a rule group can share it with other AWS accounts. For more information, see Sharing Resolver DNS Firewall rule groups between AWS accounts.

Region availability for Resolver DNS Firewall

The DNS Firewall is available in the following AWS Regions:

- Africa (Cape Town)
- Asia Pacific (Hong Kong)
- Asia Pacific (Hyderabad)
- Asia Pacific (Jakarta)
- Asia Pacific (Malaysia)
- Asia Pacific (Melbourne)
- Asia Pacific (Mumbai)
- Asia Pacific (Osaka) Region
- Asia Pacific (Seoul)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Thailand)
- Asia Pacific (Tokyo)
- Canada (Central) Region
- Canada West (Calgary)
- Europe (Frankfurt) Region
- Europe (Ireland) Region
- Europe (London) Region

- Europe (Milan)
- Europe (Paris) Region
- Europe (Spain)
- Europe (Stockholm)
- Europe (Zurich)
- Israel (Tel Aviv)
- Mexico (Central)
- Middle East (Bahrain)
- Middle East (UAE)
- South America (São Paulo)
- US East (N. Virginia)
- US East (Ohio)
- US West (N. California)
- US West (Oregon)
- China (Beijing)
- China (Ningxia)
- AWS GovCloud (US)

Getting started with Resolver DNS Firewall

The DNS Firewall console includes a wizard that guides you through the following steps for getting started with DNS Firewall:

- Create rule groups for each set of rules that you want to use.
- For each rule, populate the domain list that you want to inspect for. You can create your own domain lists and you can use AWS managed domain lists.
- Associate your rule groups with the VPCs where you want to use them.

Resolver DNS Firewall walled garden example

In this tutorial, you'll create a rule group that blocks all but a select group of domains that you trust. This is called a closed platform, or walled garden approach.

> **Tip**
>
> If you don't know in advance every domain that your workloads need, you can use an Automated Allowlist rule. DNS Firewall observes the DNS query traffic of your VPCs, and you review and add the domains to allow. For more information, see Automated Allowlists.

To configure a DNS Firewall rule group using the console wizard

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   Choose **DNS Firewall** in the navigation pane to open the DNS Firewall **Rule groups** page on the Amazon VPC console. Continue to step 3.

   - OR -

   Sign in to the AWS Management Console and open the

   the Amazon VPC console under https://console.aws.amazon.com/vpc/.
2. In the navigation pane, under **DNS Firewall**, choose **Rule groups**.
3. On the navigation bar, choose the Region for the rule group.
4. In the **Rule groups** page, choose **Add rule group**.
5. For the rule group name, enter **`WalledGardenExample`**.

   In the **Tags** section, you can optionally enter a key-value pair for a tag. Tags help you organize and manage your AWS resources. For more information, see Tagging Amazon Route 53 resources.
6. Choose **Add rule group**.
7. On the **WalledGardenExample** details page, choose the **Rules tab**, and then **Add rule**.
8. In the **Rule details** pane, enter the rule name **`BlockAll`**.
9. In the **Domain list** pane, select **Add my own domain list**.
10. Under **Choose or create a new domain list** select **Create new domain list**.
11. Enter a domain list name **`AllDomains`**, then in the **Enter one domain per line** text box, enter an asterisk: ***** .
12. For **Domain redirection setting** accept the default, and leave **Query type - optional** empty.

13. For the **Action**, select **BLOCK** and then leave the response to send at the default setting of **NODATA** .
14. Choose **Add rule**. Your rule **BlockAll** is displayed in the **Rules** tab on the **WalledGardenExample** page.
15. On the **WalledGardenExample** page, choose **Add rule** to add a second rule to your rule group.
16. In the **Rule details** pane, enter the rule name **`AllowSelectDomains`** .
17. In the **Domain list** pane, select **Add my own domain list**.
18. Under **Choose or create a new domain list**, select **Create new domain list**.
19. Enter a domain list name **`ExampleDomains`**.
20. In the **Enter one domain per line** text box, on the first line, enter **`example.com`** and on the second line, enter **`example.org`**.

    > **Note**
    >
    > If you want the rule to apply to subdomains as well, you need to add those domains to the list also. For example, to add all of the example.com's subdomains, add **`*.example.com`** to the list.

21. For **Domain redirection setting** accept the default, and leave **Query type - optional** empty.
22. For the **Action**, select **ALLOW**.
23. Choose **Add rule**. Your rules are both displayed in the **Rules** tab on the **WalledGardenExample** page.
24. In the **Rules** tab on the **WalledGardenExample** page, you can adjust the evaluation order of the rules in your rule group by selecting the number listed in the **Priority column** and typing in a new number. DNS Firewall evaluates rules starting with the lowest priority setting, so the rule with the lowest priority is the first one evaluated. For this example, we want DNS Firewall to first identify and allow DNS queries for the select list of domains, and then block any remaining queries.

    Adjust the rule priority so that **AllowSelectDomains** has a lower priority.

You now have a rule group that allows only specific domain queries through. To begin using it, you associate it with the VPCs where you want to use the filtering behavior. For more information, see Managing associations between your VPC and Resolver DNS Firewall rule group.

Resolver DNS Firewall block list example

In this tutorial, you'll create a rule group that blocks domains that you know to be malicious. You'll also add a DNS query type that is allowed for the domains in the blocked list. The rule group allows all other outbound DNS requests over the VPC Resolver.

To configure a DNS Firewall block list by using the console wizard

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   Choose **DNS Firewall** in the navigation pane to open the DNS Firewall **Rule groups** page on the Amazon VPC console. Continue to step 3.

   - OR -

   Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, under **DNS Firewall**, choose **Rule groups**.
3. On the navigation bar, choose the Region for the rule group.
4. In the **Rule groups** page, choose **Add rule group**.
5. For the rule group name, enter **`BlockListExample`**.

   In the **Tags** section, you can optionally enter a key-value pair for a tag. Tags help you organize and manage your AWS resources. For more information, see Tagging Amazon Route 53 resources.
6. On the **BlockListExample** details page, choose the **Rules** tab, and then **Add rule**.
7. In the **Rule details** pane, enter the rule name **`BlockList`**.
8. In the **Domain list** pane, select **Add my own domain list**.
9. Under **Choose or create a new domain list**, select **Create new domain list**.
10. Enter a domain list name **`MaliciousDomains`**, then in the text box, enter the domains you want to block. For example, **`example.org`**. Enter one domain per line.

> **Note**
>
> If you want the rule to apply to subdomains as well, you must add those domains to the list also. For example, to add all of the example.org's subdomains, add **`*.example.org`** to the list.

11. For **Domain redirection setting** accept the default, and leave **Query type - optional** empty.
12. For the action, select **BLOCK** and then leave the response to send at the default setting of **NODATA**.
13. Choose **Add rule**. Your rule is displayed in the **Rules** tab on the **BlockListExample** page
14. in the **Rules** tab on the **BlockedListExample** page, you can adjust the evaluation order of the rules in your rule group by selecting the number listed in the **Priority column** and typing in a new number. DNS Firewall evaluates rules starting with the lowest priority setting, so the rule with the lowest priority is the first one evaluated.

    Select and adjust the rule priority so that **BlockList** is evaluated either before or after any other rules you might have. Most of the time, known malicious domains should be blocked first. That is, the rules associated with them should have the lowest priority number.
15. To add a rule that allows MX records for the BlockList domains, on the **BlockedListExample** details page in the **Rules** tab, choose **Add rule**.
16. In the **Rule details** pane, enter the rule name **`BlockList-allowMX`**.
17. In the **Domain list** pane, select **Add my own domain list**.
18. Under **Choose or create a new domain list**, select **`MaliciousDomains`**.
19. For **Domain redirection setting** accept the default.
20. In the **DNS query type** list, select **MX: Specifies mail servers**.
21. For the action, select **ALLOW**.
22. Choose **Add rule**.
23. in the **Rules** tab on the **BlockedListExample** page, you can adjust the evaluation order of the rules in your rule group by selecting the number listed in the **Priority column** and typing in a new number. DNS Firewall evaluates rules starting with the lowest priority setting, so the rule with the lowest priority is the first one evaluated.

Select and adjust the rule priority so that **BlockList-allowMX** is evaluated either before or after any other rules you might have. Because you want to allow MX queries, make sure that the **BlockList-allowMX** rule has a lower priority than **BlockList**.

You now have a rule group that blocks specific malicious domain queries, but allows a specific DNS query type. To begin using it, you associate it with the VPCs where you want to use the filtering behavior. For more information, see Managing associations between your VPC and Resolver DNS Firewall rule group.

DNS Firewall rule groups and rules

This section describes the settings that you can configure for your DNS Firewall rule groups and rules, to define the DNS Firewall behavior for your VPCs. It also describes how to manage the settings for your rule groups and rules.

When you have your rule groups configured the way you want them, you use them directly and you can share and manage them between accounts and across your organization in AWS Organizations.

- You can associate a rule group with multiple VPCs, to provide consistent behavior across your organization. For information, see Managing associations between your VPC and Resolver DNS Firewall rule group.
- You can share rule groups between accounts, for consistent DNS query management across your organization. For information, see Sharing Resolver DNS Firewall rule groups between AWS accounts.
- You can use rule groups across your organization in AWS Organizations by managing them in AWS Firewall Manager policies. For information about Firewall Manager, see AWS Firewall Manager in the *AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide*.

Rule group settings in DNS Firewall

When you create or edit a DNS Firewall rule group, you specify the following values:

Name

A unique name that lets you easily find a rule group on the dashboard.

(Optional) Description

A short description that provides more context for the rule group.

Region

The AWS Region that you choose when you create the rule group. A rule group that you create in one Region is available only in that Region. To use the same rule group in more than one Region, you must create it in each Region.

Rules

The rule group filtering behavior is contained in its rules. For information, see the following section.

Tags

Specify one or more keys and the corresponding values. For example, you might specify **Cost center** for **Key** and specify **456** for **Value**.

These are the tags that AWS Billing and Cost Management provides for organizing your AWS bill. For more information about using tags for cost allocation, see Using cost allocation tags in the *AWS Billing User Guide* .

Rule settings in DNS Firewall

When you create or edit a rule in a DNS Firewall rule group, you specify the following values:

Name

A unique identifier for the rule in the rule group.

(Optional) Description

A short description that provides more information about the rule.

Domain list

The list of domains that the rule inspects for. You can create and manage your own domain list or you can subscribe to a domain list that AWS manages for you. For more information, see DNS Firewall Foundational Rules.

A rule can contain ether a domain list or a DNS Firewall Advanced protection, but not both.

Domain redirection setting (domain lists only)

You can choose for the DNS Firewall rule to inspect only the first domain or all (default) the domains in the DNS redirection chain, such as CNAME, DNAME, etc. If you choose to inspect all the domains, you must add the subsequent domains in the DNS redirection chain to the domain list and set to the action you want the rule to take, either ALLOW, BLOCK, or ALERT. For more information, see Resolver DNS Firewall components and settings.

> **Important**
>
> The trust behavior of the domain redirection setting only applies within a single DNS query transaction. If a DNS client on your host separately queries a domain that appears in a DNS redirection chain (for example, querying the redirection target directly), DNS Firewall evaluates it as an independent query with no trust context from the original query. To allow such queries, add the redirection target domains to your domain list.

Query type (domain lists only)

The list of DNS query types that the rule inspects for. The following are the valid values:

- A: Returns an IPv4 address.
- AAAA: Returns an Ipv6 address.
- CAA: Restricts CAs that can create SSL/TLS certifications for the domain.
- CNAME: Returns another domain name.
- DS: Record that identifies the DNSSEC signing key of a delegated zone.
- MX: Specifies mail servers.
- NAPTR: Regular-expression-based rewriting of domain names.
- NS: Authoritative name servers.
- PTR: Maps an IP address to a domain name.
- SOA: Start of authority record for the zone.
- SPF: Lists the servers authorized to send emails from a domain.
- SRV: Application specific values that identify servers.
- TXT: Verifies email senders and application-specific values.

- A query type you define by using the DNS type ID, for example 28 for AAAA. The values must be defined as TYPE *NUMBER*, where the *NUMBER* can be 1-65334, for example, TYPE28. For more information, see List of DNS record types.

  You can create one query type per rule.

  > **Note**
  >
  > If you set up a firewall BLOCK rule with action NXDOMAIN on query type equals AAAA, this action will not be applied to synthetic IPv6 addresses generated when DNS64 is enabled.

DNS Firewall Advanced protection

Detects suspicious DNS queries based on known threat signatures in DNS queries. You can choose protection from:

- Domain Generation Algorithms (DGAs)

  DGAs are used by attackers to generate a large number of domains to launch malware attacks.

- DNS tunneling

  DNS tunneling is used by attackers to exfiltrate data from the client by using the DNS tunnel without making a network connection to the client.

- Dictionary DGA

  Dictionary DGAs are used by attackers to generate domains using dictionary words to evade detection in malware command-and-control communications.

In a DNS Firewall Advanced rule you can choose to either block, or alert on a query that matches the threat.

For more information, see For more information, see DNS Firewall Advanced Rules.

A rule can contain ether a DNS Firewall Advanced protection or a domain list, but not both.

Confidence threshold (DNS Firewall Advanced only)

The confidence threshold for DNS Firewall Advanced. You must provide this value when you create a DNS Firewall Advanced rule. The confidence level values mean:

- High – Detects only the most well corroborated threats with a low rate of false positives.
- Medium – Provides a balance between detecting threats and false positives.
- Low – Provides the highest detection rate for threats, but also increases false positives.

For more information, see Rule settings in DNS Firewall.

Action

How you want DNS Firewall to handle a DNS query whose domain name matches the specifications in the rule's domain list. For more information, see Rule actions in DNS Firewall.

Priority

Unique positive integer setting for the rule within the rule group that determines processing order. DNS Firewall inspects DNS queries against the rules in a rule group starting with the lowest numeric priority setting and going up. You can change a rule's priority at any time, for example to change the order of processing or make space for other rules.

Rule actions in DNS Firewall

When DNS Firewall finds a match between a DNS query and a domain specification in a rule, it applies the action that's specified in the rule to the query.

You are required to specify one of the following options in each rule that you create:

- **Allow** – Stop inspecting the query and permit it to go through. Not available for DNS Firewall Advanced.
- **Alert** – Stop inspecting the query, permit it to go through, and log an alert for the query in the Route 53 VPC Resolver logs.
- **Block** – Discontinue inspection of the query, block it from going to its intended destination, and log the block action for the query in the Route 53 VPC Resolver logs.

  Reply with the configured block response, from the following:

  - **NODATA** – Respond indicating that the query was successful, but no response is available for it.
  - **NXDOMAIN** – Respond indicating that the query's domain name doesn't exist.
  - **OVERRIDE** – Provide a custom override in the response. This option requires the following additional settings:
    - **Record value** – The custom DNS record to send back in response to the query.

- **Record type** – The DNS record's type. This determines the format of the record value. This must be CNAME.
- **Time to live in seconds** – The recommended amount of time for the DNS resolver or web browser to cache the override record and use it in response to this query, if it is received again. By default, this is zero, and the record isn't cached.

For more information about the query logs configuration and the contents, see Resolver query logging and Values that appear in VPC Resolver query logs.

Use Alert to test blocking rules

When you first create a blocking rule, you can test it by configuring it with the action set to Alert. You can then look at the number of queries that the rule alerts on to see how many would be blocked if you set the action to Block.

Managing rule groups and rules in DNS Firewall

To manage rule groups and rules in the console, follow the guidance in this section.

When you make changes to DNS Firewall entities, like rules and domain lists, DNS Firewall propagates the changes everywhere that the entities are stored and used. Your changes are applied within seconds, but there might be a brief period of inconsistency when the changes have arrived in some places and not in others. So, for example, if you add a domain to a domain list that's referenced by a blocking rule, the new domain might briefly be blocked in one area of your VPC while still allowed in another. This temporary inconsistency can occur when you first configure your rule group and VPC associations and when you change existing settings. Generally, any inconsistencies of this type last only a few seconds.

Creating a rule group and rules

To create a rule group and add rules to it, follow the steps in this procedure.

To create a rule group and its rules

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   Choose **DNS Firewall** in the navigation pane to open the DNS Firewall **Rule groups** page on the Amazon VPC console. Continue to step 3.

- OR -

Sign in to the AWS Management Console and open the

the Amazon VPC console under https://console.aws.amazon.com/vpc/.

2. In the navigation pane, under **DNS Firewall**, choose **Rule groups**.
3. On the navigation bar, choose the Region for the rule group.
4. Choose **Add rule group**, then follow the wizard guidance to specify your rule group and rule settings.

   When you add rules to a rule group, first select the rule type: **Foundation** or **Advanced**.

   - **Foundation** – Select a rule protection to create rules with your own custom domain lists or AWS essential managed domain lists.

     For more information about custom domain lists, see Managing your own domain lists.

     For more information about managed domain lists, see Managed Domain Lists.

   - **Advanced** – Select one or more advanced managed domain lists (threat and content categories) or an advanced protection (Domain Generation Algorithms, Dictionary DGA, or DNS tunneling).

     For more information about advanced managed domain lists, see Advanced Managed Domain Lists.

     For more information about advanced protections, see Advanced DNS Protections.
5. With each rule, you can select one or more rule settings.

   For information about the values for rule groups, see Rule group settings in DNS Firewall.

   For information about the values for rules, see Rule settings in DNS Firewall.

Viewing and updating a rule group and rules

Use the following procedure to view the rule groups and the rules assigned to them. You can also update the rule group and rule settings.

To view and update a rule group

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   Choose **DNS Firewall** in the navigation pane to open the DNS Firewall **Rule groups** page on the Amazon VPC console. Continue to step 3.

   - OR -

   Sign in to the AWS Management Console and open the

   the Amazon VPC console under https://console.aws.amazon.com/vpc/.
2. In the navigation pane, under **DNS Firewall**, choose **Rule groups**.
3. On the navigation bar, choose the Region for the rule group.
4. Select the rule group that you want to view or edit, then choose **View details**.
5. In the rule group's page, you can view and edit settings.

   For information about the values for rule groups, see Rule group settings in DNS Firewall.

   For information about the values for rules, see Rule settings in DNS Firewall.

Deleting a rule group

To delete a rule group, perform the following procedure.

> **Important**
>
> If you delete a rule group that's associated with a VPC, DNS Firewall removes the association and stops the protections that the rule group was providing to the VPC.

Deleting DNS Firewall entities

When you delete an entity that you can use in DNS Firewall, like a domain list that might be in use in a rule group, or a rule group that might be associated with a VPC, DNS Firewall checks to see if the entity is currently being used. If it finds that it is in use, DNS Firewall warns you. DNS Firewall is almost always able to determine if an entity is in use. However, in rare cases it might not be able

to do so. If you need to be sure that nothing is currently using the entity, check for it in your DNS Firewall configurations before deleting it. If the entity is a referenced domain list, check that no rule groups are using it. If the entity is a rule group, check that it is not associated with any VPCs.

To delete a rule group

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   Choose **DNS Firewall** in the navigation pane to open the DNS Firewall **Rule groups** page on the Amazon VPC console. Continue to step 3.

   - OR -

   Sign in to the AWS Management Console and open the

   the Amazon VPC console under https://console.aws.amazon.com/vpc/.
2. In the navigation pane, under **DNS Firewall**, choose **Rule groups**.
3. On the navigation bar, choose the Region for the rule group.
4. Select the rule group that you want to delete, then choose **Delete**, and confirm the deletion.

DNS Firewall Foundational Rules

DNS Firewall Foundational rules provide you with the essential DNS protections to help you get started with DNS Firewall, with two main types of foundational rules:

- Managed domain lists, which AWS creates and maintains for you.
- Your own domain lists, which you create and maintain. You can use a single domain list in multiple rules and any updates that you do to the domain list automatically affect all rules that use it.

A *domain list* is a reusable set of domain specifications that you use in a DNS Firewall rule, inside a rule group. When you associate a rule group with a VPC, DNS Firewall compares your DNS queries against the domain lists that are used in the rules. If it finds a match, it handles the DNS query according to the matching rule's action. For more information about rule groups and rules, see DNS Firewall rule groups and rules.

Managed Domain Lists

Managed Domain Lists contain domain names that are associated with malicious activity or other potential threats. AWS maintains these lists to enable Route 53 VPC Resolver customers to check outbound DNS queries against them for free when using DNS Firewall.

Keeping up to date on the constantly changing threat landscape can be time consuming and expensive. Managed Domain Lists can save you time when you implement and use DNS Firewall. AWS automatically updates the lists when new vulnerabilities and threats emerge. AWS is often notified of new vulnerabilities before public disclosure, so DNS Firewall can deploy mitigations for you often before a new threat has become widely known.

Managed domain lists are designed to help protect you from common web threats and they add another layer of security for your applications. The AWS Managed Domain Lists source their data from both internal AWS sources as well as RecordedFuture, and are continually updated. However, AWS Managed Domain Lists aren't intended as a replacement for other security controls, such as Amazon GuardDuty, which are determined by the AWS resources that you select.

As a best practice, before using a Managed Domain List in production, test it in a non-production environment, with the rule action set to `Alert`. Evaluate the rule using Amazon CloudWatch metrics combined with Resolver DNS Firewall sampled requests or DNS Firewall logs. When you're satisfied that the rule does what you want, change the action setting as needed.

Available AWS Managed Domain Lists

This section describes the Managed Domain Lists that are currently available for DNS Firewall Foundational rules. When you're in a Region where these lists are supported, you see them on the console when you manage domain lists and when you specify the domain list for a rule. In the logs, the domain list is logged within the `firewall_domain_list_id` field.

AWS provides the following Managed Domain Lists under the Foundational rule type, in the Regions they are available, for all users of Resolver DNS Firewall.

Managed domain lists

Threat Type	Description
Malware	Domains associated with sending malware, hosting malware, or distributing malware.

Threat Type	Description
Botnet/Command and Control	Domains associated with controlling networks of computers that are infected with spamming malware.
Aggregate Threat List	Domains associated with multiple DNS threat categories including malware, ransomware, botnet, spyware, and DNS tunneling to help block multiple types of threats. Aggregate Threat List includes all the domains in the other AWS Managed Domain Lists listed here.
Amazon GuardDuty Threat List	Domains associated with Amazon GuardDuty DNS security findings. The domains are sourced from the GuardDuty's threat intelligence systems only, and do not contain domains sourced from external third-party sources. More specifically, currently this list will only block domains that are internally generated and used for following detections in GuardDuty: Impact:EC2/AbusedDomainRequest.Reputation, Impact:EC2/BitcoinDomainRequest.Reputation, Impact:EC2/MaliciousDomainRequest.Reputation, Impact:Runtime/AbusedDomainRequest.Reputation, Impact:Runtime/BitcoinDomainRequest.Reputation, and Impact:Runtime/MaliciousDomainRequest.Reputation. For more information see Finding types in the *Amazon GuardDuty User Guide*.

AWS Managed Domain Lists cannot be downloaded or browsed. To protect intellectual property, you can't view or edit the individual domain specifications within an AWS Managed Domain Lists. This restriction also helps to prevent malicious users from designing threats that specifically circumvent published lists.

To test the Managed Domain lists

We provide the following set of domains for testing the Managed Domain Lists:

AWSManagedDomainsBotnetCommandandControl

- controldomain1.botnetlist.firewall.route53resolver.us-east-1.amazonaws.com
- controldomain2.botnetlist.firewall.route53resolver.us-east-1.amazonaws.com
- controldomain3.botnetlist.firewall.route53resolver.us-east-1.amazonaws.com

AWSManagedDomainsMalwareDomainList

- controldomain1.malwarelist.firewall.route53resolver.us-east-1.amazonaws.com
- controldomain2.malwarelist.firewall.route53resolver.us-east-1.amazonaws.com
- controldomain3.malwarelist.firewall.route53resolver.us-east-1.amazonaws.com

AWSManagedDomainsAggregateThreatList and AWSManagedDomainsAmazonGuardDutyThreatList

- controldomain1.aggregatelist.firewall.route53resolver.us-east-1.amazonaws.com
- controldomain2.aggregatelist.firewall.route53resolver.us-east-1.amazonaws.com
- controldomain3.aggregatelist.firewall.route53resolver.us-east-1.amazonaws.com

These domains will resolve to 1.2.3.4 if they aren't blocked. If you're using the Managed Domain Lists in a VPC, querying for these domains will return the response that a block action in the rule is set to (for example NODATA).

For more information about Managed Domain Lists, contact the AWS Support Center.

The following table lists the Region availability for AWS Managed Domain Lists.

Managed Domain List Region availability

Region	Managed Domain Lists available?
Africa (Cape Town)	Yes
Asia Pacific (Hong Kong)	Yes
Asia Pacific (Hyderabad)	Yes

Region	Managed Domain Lists available?
Asia Pacific (Jakarta)	Yes
Asia Pacific (Malaysia)	Yes
Asia Pacific (Melbourne)	Yes
Asia Pacific (Mumbai)	Yes
Asia Pacific (Osaka) Region	Yes
Asia Pacific (Seoul)	Yes
Asia Pacific (Singapore)	Yes
Asia Pacific (Sydney)	Yes
Asia Pacific (Thailand)	Yes
Asia Pacific (Tokyo)	Yes
Canada (Central) Region	Yes
Canada West (Calgary)	Yes
Europe (Frankfurt) Region	Yes
Europe (Ireland) Region	Yes

Region	Managed Domain Lists available?
Europe (London) Region	Yes
Europe (Milan)	Yes
Europe (Paris) Region	Yes
Europe (Spain)	Yes
Europe (Stockholm)	Yes
Europe (Zurich)	Yes
Israel (Tel Aviv)	Yes
Middle East (Bahrain)	Yes
Middle East (UAE)	Yes
South America (São Paulo)	Yes
US East (N. Virginia)	Yes
US East (Ohio)	Yes
US West (N. California)	Yes
US West (Oregon)	Yes

Region	Managed Domain Lists available?
China (Beijing)	Yes
China (Ningxia)	Yes
AWS GovCloud (US)	Yes

Additional security considerations

AWS Managed Domain Lists are designed to help protect you from common web threats. When used in accordance with the documentation, these lists add another layer of security for your applications. However, the Managed Domain Lists aren't intended as a replacement for other security controls, which are determined by the AWS resources that you select. To make sure that your resources in AWS are properly protected, see the guidance at Shared Responsibility Model.

Mitigating false positive scenarios

If you are encountering false-positive scenarios in rules that use Managed Domain Lists to block queries, perform the following steps:

1. In the VPC Resolver logs, identify the rule group and managed domain list that are causing the false positive. You do this by finding the log for the query that DNS Firewall is blocking, but that you want to allow through. The log record lists the rule group, rule action, and the managed list. For information about the logs, see Values that appear in VPC Resolver query logs.
2. Create a new rule in the rule group that explicitly allows the blocked query through. When you create the rule, you can define your own domain list with just the domain specification that you want to allow. Follow the guidance for rule group and rule management at Creating a rule group and rules.
3. Prioritize the new rule inside the rule group so that it runs before the rule that's using the managed list. To do this, give the new rule a lower numeric priority setting.

When you have updated your rule group, the new rule will explicitly allow the domain name that you want to allow before the blocking rule runs.

Managing your own domain lists

You can create your own domain lists to specify domain categories that you either don't find in the managed domain list offerings or that you prefer to handle on your own.

In addition to the procedures described in this section, in the console, you can create a domain list in the context of Resolver DNS Firewall rule management, when you create or update a rule.

Each domain specification in your domain list must satisfy the following requirements:

- It can optionally start with * (asterisk).
- With the exception of the optional starting asterisk and a period, as a delimiter between labels, it must only contain the following characters: `A-Z` , `a-z`, `0-9`, `-` (hyphen).
- It must be from 1-255 characters in length.

When you make changes to DNS Firewall entities, like rules and domain lists, DNS Firewall propagates the changes everywhere that the entities are stored and used. Your changes are applied within seconds, but there might be a brief period of inconsistency when the changes have arrived in some places and not in others. So, for example, if you add a domain to a domain list that's referenced by a blocking rule, the new domain might briefly be blocked in one area of your VPC while still allowed in another. This temporary inconsistency can occur when you first configure your rule group and VPC associations and when you change existing settings. Generally, any inconsistencies of this type last only a few seconds.

Test your domain list before using it in production

As a best practice, before using a domain list in production, test it in a non-production environment, with the rule action set to `Alert`. Evaluate the rule using Amazon CloudWatch metrics and the VPC Resolver logs. The logs provide the domain list name for all alerts and blocking actions. When you're satisfied that the domain list is matching your DNS queries the way you want it to, change the rule action setting as needed. For information about CloudWatch metrics and the query logs, see Monitoring Resolver DNS Firewall rule groups with Amazon CloudWatch, Values that appear in VPC Resolver query logs, and Managing Resolver query logging configurations.

To add a domain list

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

Choose **DNS Firewall** in the navigation pane to open the DNS Firewall **Rule groups** page on the Amazon VPC console. Continue to step 2.

- OR -

Sign in to the AWS Management Console and open the

the Amazon VPC console under https://console.aws.amazon.com/vpc/.

2. In the navigation pane, under **DNS Firewall**, choose **Domain lists**. In the **Domain lists** page, you can select and edit existing domain lists and you can add your own.
3. To add a domain list, choose **Add domain list**.
4. Provide a name for your domain list, and then enter your domain specifications in the text box, one per line.

   If you slide **Switch to bulk upload** to **on**, enter the URI of the Amazon S3 bucket where you created a domain list. This domain list should have one domain name per line.

   > **Note**
   >
   > Duplicate domain names will cause the bulk import to fail.

5. Choose **Add domain list**. The **Domain lists** page lists your new domain list.

After you create the domain list, you can reference it by name from your DNS Firewall rules.

Deleting DNS Firewall entities

When you delete an entity that you can use in DNS Firewall, like a domain list that might be in use in a rule group, or a rule group that might be associated with a VPC, DNS Firewall checks to see if the entity is currently being used. If it finds that it is in use, DNS Firewall warns you. DNS Firewall is almost always able to determine if an entity is in use. However, in rare cases it might not be able to do so. If you need to be sure that nothing is currently using the entity, check for it in your DNS Firewall configurations before deleting it. If the entity is a referenced domain list, check that no rule groups are using it. If the entity is a rule group, check that it is not associated with any VPCs.

To delete a domain list

1. In the navigation pane, choose **Domain lists**.

2. On the navigation bar, choose the Region for the domain list.
3. Select the domain list that you want to delete, then choose **Delete**, and confirm the deletion.

DNS Firewall Advanced Rules

DNS Firewall Advanced Rule tier provides you with protections to help you detect and monitor for more advanced DNS threats (for example, Domain Generation Algorithms (DGA) or DNS Tunneling) or with more granular protections based on threat and web-content (for example, gambling, social networking).

There are four main types of advanced rules:

- Advanced Protections that help detect suspicious DNS queries based on known threat signatures (for example, DGA) in DNS queries.
- Advanced DNS threat and content categories that provide more granular control to block DNS queries based on the type of DNS threats (for example, spam, phishing) or web content (for example, adult content, social networking, gambling sites).
- Partner Managed DNS protections from Palo Alto Networks Advanced DNS Security and Infoblox (requires AWS Marketplace subscription).
- Automated Allowlists that help you build an allowlist of domains from the DNS query traffic of your VPCs, so that you can block all other domains (*) and move to a walled-garden setup. For more information, see Automated Allowlists.

Advanced DNS Protections

DNS Firewall Advanced detects suspicious DNS queries based on known threat signatures in DNS queries. You can specify a threat type in a rule that you use in a DNS Firewall rule, inside a rule group. When you associate a rule group with a VPC, DNS Firewall compares your DNS queries against the domains that are flagged in the rules. If it finds a match, it handles the DNS query according to the matching rule's action.

DNS Firewall Advanced works by identifying suspicious DNS threat signatures by inspecting a range of key identifiers in the DNS payload including the timestamp of requests, frequency of request and responses, the DNS query strings, and the length, type or size of both outbound and inbound DNS queries. Based on the type of threat signature, you can configure policies to block, or simply log and alert on the query. By using an expanded set of threat identifiers, you can protect

against DNS threats from domain sources that might yet be unclassified by threat intelligence feeds maintained by the broader security community.

Currently, DNS Firewall Advanced offers protections from:

- Domain Generation Algorithms (DGAs)

  DGAs are used by attackers to generate a large number of domains to launch malware attacks.
- DNS tunneling

  DNS tunneling is used by attackers to exfiltrate data from the client by using the DNS tunnel without making a network connection to the client.
- Dictionary DGA

  Dictionary DGAs are used by attackers to generate domains using dictionary words to evade detection in malware command-and-control communications.

To learn how to create rules, see Creating a rule group and rules and Rule settings in DNS Firewall.

Mitigating false positive scenarios

If you are encountering false-positive scenarios in rules that use DNS Firewall Advanced protections to block queries, perform the following steps:

1. In the VPC Resolver logs, identify the rule group and DNS Firewall Advanced protections that are causing the false positive. You do this by finding the log for the query that DNS Firewall is blocking, but that you want to allow through. The log record lists the rule group, rule action, and the DNS Firewall Advanced protection. For information about the logs, see Values that appear in VPC Resolver query logs.
2. Create a new rule in the rule group that explicitly allows the blocked query through. When you create the rule, you can define your own domain list with just the domain specification that you want to allow. Follow the guidance for rule group and rule management at Creating a rule group and rules.
3. Prioritize the new rule inside the rule group so that it runs before the rule that's using the managed list. To do this, give the new rule a lower numeric priority setting.

When you have updated your rule group, the new rule will explicitly allow the domain name that you want to allow before the blocking rule runs.

Advanced Managed Domain Lists

Advanced Managed Domain Lists contain domain names that are associated with malicious activity or other not-safe-for-work domains. Advanced Managed Domain Lists are categorized into Threat and Content categories, designed to help protect you from common web threats and also block query resolution to domains not safe-for-work.

As a best practice, before using a Managed Domain List in production, test it in a non-production environment, with the rule action set to `Alert`. Evaluate the rule using Amazon CloudWatch metrics combined with DNS Firewall logs. When you're satisfied that the rule does what you want, change the action setting as needed.

Available Advanced Managed Domain Lists

This section describes the Advanced Managed Domain Lists that are currently available for DNS Firewall under the Advanced rule tier. AWS provides the following Managed Domain Lists, classified by **Threat** or **Content** Type.

Threat categories

Threat
Spam
Phishing

Content categories

Content
Violence and Hate Speech
For Kids
Online Ads
Science
Family and Parenting
Pets

Content
Career and Job Search
Religion
Lifestyle
Home and Garden
Criminal and Illegal Activities
Sports and Recreation
Vehicles
Financial Services
Real Estate
Hobbies and Interests
Travel
Food and Dining
Government and Legal
Education
Fashion
Health
Shopping
Adult and Mature Content
Technology and Internet
Business and Economy

Content
News
Search Engines and Portals
Arts and Culture
Entertainment
Military
Social Networking
Proxy Avoidance
Redirect
Email
Translation
Child Abuse
Abortion
Gambling
Hacking
Marijuana
Cryptocurrency
Dating
Artificial Intelligence and Machine Learning
Parked Domains
Private IP Address

Using partner managed DNS threat protection with Palo Alto Networks in DNS Firewall

This topic covers how to subscribe to, configure, and manage Palo Alto Networks (PANW) DNS threat protection rules within Amazon Route 53 Resolver DNS Firewall rule groups.

Palo Alto Networks Advanced DNS Security for Route 53 delivers cloud-native DNS threat protection for your Amazon Virtual Private Cloud instances by integrating Palo Alto Networks threat intelligence directly into DNS Firewall. This integration eliminates the need to deploy separate firewalls, manage Amazon VPC routing configurations, or forward DNS traffic to external inspection points. You enforce Palo Alto Networks DNS protections through the same DNS Firewall rules and rule groups you already use to manage DNS security in AWS.

The integration helps you in two key ways:

1. It provides access to 30+ DNS threat detections from Palo Alto Networks Advanced DNS Security. Examples include fast flux detection, DNS hijacking, domain generation algorithm (DGA) detection, and identification of newly registered domains.
2. It eliminates the infrastructure overhead of deploying Palo Alto Networks firewalls across multiple VPCs and accounts solely for DNS inspection. This helps to reduce both cost and operational complexity while maintaining the same threat efficacy.

Availability

Partner managed DNS threat protection with Palo Alto Networks is generally available in all commercial AWS Regions where DNS Firewall is available.

Prerequisites

Before you can use partner managed DNS threat protection with Palo Alto Networks, you must have the following:

- An AWS account with appropriate IAM permissions for DNS Firewall and AWS Marketplace.
- A DNS Firewall rule group already created (or create one during the workflow).
- The Advanced plan pricing option selected for your rule group (required for partner managed DNS threat protection). You can select this during DNS Firewall rule creation.

Subscribe to Palo Alto Networks Advanced DNS Security

You can subscribe to Palo Alto Networks Advanced DNS Security directly from the DNS Firewall console during the rule creation flow, or from AWS Marketplace. When you subscribe, you authorize AWS to share DNS query metadata with Palo Alto Networks.

Subscribe through the DNS Firewall console (recommended)

Use the following procedure to subscribe through the console.

1. Sign in to the AWS Management Console and open the Amazon Virtual Private Cloud console.
2. In the navigation pane, under **DNS Firewall**, choose **Rule groups**.
3. Select the rule group where you want to add partner managed DNS protections.
4. Choose **Add rule**.
5. Under **Rule details**, enter a **Name** for the rule (for example, `PAN Threat Protection Rule`). Valid characters: A-Z, a-z, 0-9, hyphen (-), and underscore (_). Maximum 64 characters. Select the **Advanced** pricing option.
6. Under **Rule configurations** > **Rule type**, select **Partner managed DNS threat protection**.
7. Under **Vendor**, select **Palo Alto Networks**.

If you are not yet subscribed, a banner appears indicating that a subscription is required through AWS Marketplace for Palo Alto Networks.

1. Choose **View subscription options**. A subscription dialog appears showing available offers, pricing details, and legal terms.
2. Review the terms and pricing, then choose **Subscribe**.
3. Wait for the subscription to process. A status banner indicates that your AWS Marketplace subscription is being processed. The form updates automatically when your subscription is active.
4. After successful subscription, the DNS security categories dropdown becomes available, allowing you to select threat list feeds for rule creation.

Subscribe through AWS Marketplace

Use the following procedure to subscribe through Marketplace.

1. Open the AWS Marketplace console.

2. Search for the "Palo Alto Networks Advanced DNS Security for Amazon Route 53" product.
3. Review the product listing, pricing, and terms.
4. Choose **Subscribe**.
5. After subscribing, return to the DNS Firewall console to create rules using the partner managed rule type.

> **Note**
>
> When subscribing through the Marketplace console, you are redirected to the seller's home page. If you subscribe through the DNS Firewall console, you are not redirected. We recommend visiting the seller's home page separately to complete any required registration details.

Unsubscribe from Palo Alto Networks

> **Important**
>
> To stop subscription charges, you must remove all PANW rules from your DNS Firewall rule groups in addition to unsubscribing from AWS Marketplace. If you unsubscribe but leave the rules in your rule groups, charges continue until you remove the rules.

Remove PANW rules from all rule groups

Use the following procedure to remove PANW rules.

1. Open the Amazon Virtual Private Cloud console.
2. In the navigation pane, choose **DNS Firewall**, then choose **Rule groups**.
3. For each rule group containing Palo Alto Networks rules, select the partner managed rules and choose **Delete**.
4. Confirm deletion.

Cancel the Marketplace subscription

Use the following procedure to cancel the subscription.

1. Open the AWS Marketplace console.
2. Choose **Manage subscriptions**.
3. Open the **Delivery method** list and choose **SaaS**.
4. Under **Agreement**, open the **Actions** list and choose **Cancel subscription** next to the Palo Alto Networks product.
5. In the **Cancel subscription** dialog box, enter **`confirm`**, then choose **Yes, cancel subscription**.

> **Note**
>
> If you unsubscribe but leave the rules in your rule groups, charges continue until you remove the rules.

Adding Palo Alto Networks rules to DNS Firewall rule groups

After you subscribe, you can create one or more rules using Palo Alto Networks threat feeds within any DNS Firewall rule group.

To add a Palo Alto Networks rule

Use the following procedure to add a rule.

1. Open the Amazon Virtual Private Cloud console and in the navigation pane, choose **DNS Firewall**, then choose **Rule groups**.
2. Select the target rule group.
3. Choose **Add rule**.
4. Under **Rule details**, enter a **Name** and select the **Advanced** plan pricing option.
5. Under **Rule configurations** > **Rule type**, select **Partner managed DNS threat protection**.
6. Under **Vendor**, confirm **Palo Alto Networks** is selected.
7. Under **DNS security categories**, choose one or more threat list feeds from the dropdown. For more information about the available categories, see the following table.
8. Under **Action**, choose the action: **ALERT** (logs but allows) or **BLOCK** (blocks the query).
9. Choose **Add rule**. A success banner confirms rule creation.

Available DNS security categories

The following table describes the DNS security categories available from Palo Alto Networks.

DNS security categories

Category	Description
Palo Alto Networks - Dynamic DNS Hosted Domains	Domains using dynamic DNS providers that allow rapid IP address changes, frequently abused by threat actors to evade IP-based blocking.
Palo Alto Networks - Parked Domains	Domains displaying placeholder or "for sale" content. Attackers might acquire these to leverage residual reputation and redirect visitors to malicious content.
Palo Alto Networks - Proxy Avoidance and Anonymizers	Domains providing services to bypass network security controls, including web proxies, VPN services, and anonymizing networks.
Palo Alto Networks - Ad Tracking Domains	Domains used by advertising networks and third-party trackers to monitor user activity. These can serve as vectors for malvertising campaigns.
Palo Alto Networks - Newly Registered Domains	Domains registered within the past 32 days. Attackers use fresh domains because they lack reputation history, making them harder to detect.
Palo Alto Networks - Grayware Domains	Domains associated with applications that exhibit unwanted behavior such as excessive tracking, unsolicited advertising, or browser hijacking.
Palo Alto Networks - Phishing Domains	Domains impersonating legitimate organizations to steal credentials or personal information through typosquatting, homograph attacks, or domain shadowing.
Palo Alto Networks - Malware Domains	Domains that host or distribute malicious software, including exploit kits, ransomware droppers, and trojan delivery networks.

Category	Description
Palo Alto Networks - Command and Control Domains	Domains used by malware to communicate with attacker-controlled infrastructure for instructions, data exfiltration, or payload delivery.

Tip

Each selected security category creates a separate rule entry within the rule group. If you select 4 categories, 4 rules are created with sequential priorities.

View managed rules from Palo Alto Networks

You can view the available PANW managed rule groups and their current state using the console, API, or CLI.

To view managed rules using the console

Use the following procedure to view rules.

1. Open the Amazon Virtual Private Cloud console.
2. In the navigation pane, choose **DNS Firewall**, then choose **Rule groups**.
3. Select a rule group that contains PANW rules.
4. In the **Rules** tab, view rules with the following columns: Name, Action, Priority, Rule type, Resource value, Query type, Domain redirection setting, and Status.

Rule states for Palo Alto Networks rules

PANW rules in DNS Firewall can have different sync states that indicate their status.

Rule sync states

Sync state	Description	Recommended action
CREATED	The rule has been successfully created and is being enforced normally.	No action needed.

Sync state	Description	Recommended action
CREATING	The rule is currently being created, pending an entitlement check. If you are subscribed to the PANW product, the rule eventually transitions to CREATED.	No action needed.
CREATION_FAILED	The rule failed to create because you are not subscribed to the PANW product.	To retry rule creation after subscribing, you must delete the rule from the rule group and create the rule again.

To check rule status in the console, navigate to the DNS Firewall rule group and check the **Status** column in the rules table.

Best practices for Palo Alto Networks partner managed rules

We recommend the following best practices when using Palo Alto Networks partner managed DNS threat protection:

- **Testing before production:** Before deploying to production, use ALERT mode to perform a dry run. Review alert logs, then switch to BLOCK after it is validated.
- **Multiple rules per category:** Each security category creates a separate rule. You can assign different actions to different categories.
- **Amazon VPC association:** After adding rules, make sure the rule group is associated with a Amazon VPC for the rules to take effect.
- **AWS Firewall Manager integration:** Use the **Associate with an AWS Firewall Manager policy** option to apply rule groups across your organization.

Using partner managed DNS threat protection with Infoblox in DNS Firewall

This topic covers how to subscribe to, configure, and manage Infoblox DNS threat protection rules within Amazon Route 53 Resolver DNS Firewall rule groups.

Infoblox DNS threat protection for Route 53 brings Infoblox threat intelligence directly into DNS Firewall for your Amazon Virtual Private Cloud instances. You don't need to deploy separate DNS security appliances, change Amazon VPC routing, or forward DNS traffic to an external inspection point. You enforce Infoblox protections through the same DNS Firewall rules and rule groups that you already use to manage DNS security in AWS.

The integration helps you in two key ways:

1. It gives you Infoblox DNS threat intelligence feeds, grouped by threat level (informational through critical).
2. It removes the infrastructure overhead of running separate DNS security infrastructure across multiple VPCs and accounts. This helps reduce both cost and operational complexity.

Availability

Partner managed DNS threat protection with Infoblox is available in preview in all commercial AWS Regions where DNS Firewall is available.

Prerequisites

Before you can use partner managed DNS threat protection with Infoblox, you must have the following:

- An AWS account with appropriate IAM permissions for DNS Firewall and AWS Marketplace.
- A DNS Firewall rule group already created (or create one during the workflow).
- The Advanced plan pricing option selected for your rule (required for partner managed DNS threat protection). You can select this during DNS Firewall rule creation.

Subscribe to Infoblox DNS threat protection

You can subscribe to Infoblox directly from the DNS Firewall console during the rule creation flow, or from AWS Marketplace. When you subscribe, you authorize AWS to share DNS query metadata with Infoblox.

Subscribe through the DNS Firewall console (recommended)

Use the following procedure to subscribe through the console.

1. Sign in to the AWS Management Console and open the Amazon Virtual Private Cloud console.
2. In the navigation pane, under **DNS Firewall**, choose **Rule groups**.
3. Select the rule group where you want to add partner managed DNS protections.
4. Choose **Add rule**.
5. Under **Rule details**, enter a **Name** for the rule (for example, `Infoblox Threat Protection Rule`). Valid characters: A-Z, a-z, 0-9, hyphen (-), and underscore (_). Maximum 64 characters. Select the **Advanced** pricing option.
6. Under **Rule configurations** > **Rule type**, select **Partner managed DNS threat protection**.
7. Under **Vendor**, select **Infoblox**.

If you are not yet subscribed, a banner appears indicating that a subscription is required through AWS Marketplace for Infoblox.

1. Choose **View subscription options**. A subscription dialog appears showing available offers, pricing details, and legal terms.
2. Review the terms and pricing, then choose **Subscribe**.
3. Wait for the subscription to process. A status banner indicates that your AWS Marketplace subscription is being processed. The form updates automatically when your subscription is active.
4. After your subscription is active, the DNS security categories dropdown becomes available, so you can select threat list feeds for rule creation.

Subscribe through AWS Marketplace

Use the following procedure to subscribe through Marketplace.

1. Open the AWS Marketplace console.
2. Search for **`Infoblox`**, and choose the Infoblox DNS threat protection product for DNS Firewall.
3. Review the product listing, pricing, and terms.
4. Choose **Subscribe**.
5. After subscribing, return to the DNS Firewall console to create rules using the partner managed rule type.

> **Note**
>
> When you subscribe through the Marketplace console, you are redirected to the seller's home page. If you subscribe through the DNS Firewall console, you are not redirected. We recommend that you visit the seller's home page separately to complete any required registration details.

Unsubscribe from Infoblox

> **Important**
>
> To stop subscription charges, you must remove all Infoblox rules from your DNS Firewall rule groups in addition to unsubscribing from AWS Marketplace. If you unsubscribe but leave the rules in your rule groups, charges continue until you remove the rules.

Remove Infoblox rules from all rule groups

Use the following procedure to remove Infoblox rules.

1. Open the Amazon Virtual Private Cloud console.
2. In the navigation pane, choose **DNS Firewall**, then choose **Rule groups**.
3. For each rule group containing Infoblox rules, select the partner managed rules and choose **Delete**.
4. Confirm deletion.

Cancel the Marketplace subscription

Use the following procedure to cancel the subscription.

1. Open the AWS Marketplace console.
2. Choose **Manage subscriptions**.
3. Open the **Delivery method** list and choose **SaaS**.
4. Under **Agreement**, open the **Actions** list and choose **Cancel subscription** next to the Infoblox product.

5. In the **Cancel subscription** dialog box, enter **`confirm`**, then choose **Yes, cancel subscription**.

Adding Infoblox rules to DNS Firewall rule groups

After you subscribe, you can create one or more rules using Infoblox threat feeds within any DNS Firewall rule group.

To add an Infoblox rule

Use the following procedure to add a rule.

1. Open the Amazon Virtual Private Cloud console and in the navigation pane, choose **DNS Firewall**, then choose **Rule groups**.
2. Select the target rule group.
3. Choose **Add rule**.
4. Under **Rule details**, enter a **Name** and select the **Advanced** plan pricing option.
5. Under **Rule configurations** > **Rule type**, select **Partner managed DNS threat protection**.
6. Under **Vendor**, confirm that **Infoblox** is selected.
7. Under **DNS security categories**, choose one or more threat list feeds from the dropdown. For more information about the available categories, see the following table.
8. Under **Action**, choose the action: **ALERT** (logs but allows) or **BLOCK** (blocks the query).
9. Choose **Add rule**. A success banner confirms rule creation.

Available DNS security categories

The following table describes the DNS security categories available from Infoblox.

Infoblox DNS security categories

Category	Description
Infoblox - Informational	Domains that Infoblox lists at the informational threat level. These domains carry little known risk. Use them with the ALERT action to monitor DNS traffic.
Infoblox - Low Risk	Domains that Infoblox rates at the low threat level.

Category	Description
Infoblox - Medium Risk	Domains that Infoblox rates at the medium threat level.
Infoblox - High Risk	Domains that Infoblox rates at the high threat level.
Infoblox - Critical Risk	Domains that Infoblox rates at the critical threat level, the most severe level.

Tip

Each selected security category creates a separate rule entry within the rule group. If you select 4 categories, 4 rules are created with sequential priorities.

Rule states for Infoblox rules

Infoblox rules in DNS Firewall can have different sync states that indicate their status.

Rule sync states

Sync state	Description	Recommended action
`CREATED`	The rule has been successfully created and is being enforced normally.	No action needed.
`CREATING`	The rule is currently being created, pending an entitlement check. If you are subscribed to the Infoblox product, the rule eventually transitions to `CREATED`.	No action needed.
`CREATION_FAILED`	The rule failed to create because you are not subscribed to the Infoblox product.	To retry rule creation after subscribing, you must delete the rule from the rule group and create the rule again.

To check rule status in the console, navigate to the DNS Firewall rule group and check the **Status** column in the rules table.

Best practices for Infoblox partner managed rules

We recommend the following best practices when using Infoblox partner managed DNS threat protection:

- **Testing before production:** Before deploying to production, use ALERT mode to perform a dry run. Review alert logs, then switch to BLOCK after it is validated. This matters most for the lower threat levels, which can match more legitimate domains.
- **Multiple rules per category:** Each security category creates a separate rule. You can assign different actions to different categories, for example BLOCK for critical and high risk domains and ALERT for the rest.
- **Amazon VPC association:** After adding rules, make sure the rule group is associated with a Amazon VPC for the rules to take effect.
- **AWS Firewall Manager integration:** Use the **Associate with an AWS Firewall Manager policy** option to apply rule groups across your organization.

Automated Allowlists

With Automated Allowlists, DNS Firewall helps you build an allowlist of domains from the DNS query traffic of your VPCs. An Automated Allowlist rule is a rule type with the ALLOW action. DNS Firewall creates a domain list for the rule and shows you the domains that it observes in your DNS query traffic. You review the observed domains and add the domains that you want to allow.

Use Automated Allowlists when you want to allow only known domains and block all others (a *walled garden*), but you don't know in advance every domain that your workloads need.

How Automated Allowlists work

DNS Firewall observes DNS queries from every VPC that the rule group is associated with. This includes VPCs that use the rule group through Route 53 Profiles, and VPCs in other accounts that you share the rule group with through AWS Firewall Manager or AWS Resource Access Manager (AWS RAM).

When you create the rule, DNS Firewall lists the domains that your VPC clients queried during the look-back period that you choose, including queries that were allowed. After that, DNS Firewall adds a domain to the list of observed domains when any DNS Firewall rule blocks a query for it from an associated VPC. This includes rules in other rule groups, such as rules that block known threats. Review these domains before you allow them. You can then add an observed domain to

the rule's domain list so that DNS Firewall allows queries for it. If the answer for a domain includes a redirection chain, such as a CNAME record, you also need to allow the redirection targets. For more information, see Considerations for Automated Allowlists.

A typical walled-garden rule group contains the following rules:

1. (Optional) Rules that block known threats, with the highest priority (the lowest numeric priority settings), so that adding an observed domain to the allowlist can't override a threat block.
2. An Automated Allowlist rule with a higher priority (a lower numeric priority setting) than the rule that blocks all other domains.
3. A rule with a lower priority that blocks all other domains, using a domain list that contains the wildcard `*`.

Each domain that DNS Firewall tracks for an Automated Allowlist rule has one of the following statuses:

- `OBSERVED` – Domains that DNS Firewall observed in your DNS query traffic. When you create the rule, DNS Firewall adds the domains that your VPCs queried during the look-back period that you specify, including queries that were allowed. After that, DNS Firewall adds domains when a DNS Firewall rule blocks a query for them. Review these domains to decide which ones to allow. Domains from the look-back period that your VPCs keep querying, and that no rule blocks, move to `ALLOWED` within about a day after you create the rule, so review the observed domains soon after the rule is created.
- `ALLOWED` – Domains that DNS Firewall allowed queries for, whether or not they're in the rule's domain list.
- `DENIED` (**Rejected** in the console) – Domains that you chose not to add to the allowlist. If your VPCs still query a rejected domain and no rule blocks it, the domain can move to `ALLOWED`. If a rule later blocks it, it can appear in the observed domains again.

The `ALLOWED` status isn't the allowlist itself. In the console, the **Analysis** tab shows the domains that DNS Firewall allowed queries for, with their request counts, and DNS Firewall updates this information every hour. The **Allowed domains** view on the **Domain list** tab shows the rule's domain list, which is the allowlist that DNS Firewall enforces.

Considerations for Automated Allowlists

Note the following when you use Automated Allowlists:

- A rule group can contain only one Automated Allowlist rule.
- The rule action is always `ALLOW`, and the rule applies to all DNS query types.
- Automated Allowlist rules are Regional.
- DNS Firewall keeps up to 10,000 domains in each status for a rule. When a status is full, DNS Firewall doesn't add new domains to it, even domains with a lot of traffic, until existing domains in that status stop receiving queries.
- Domains in the `OBSERVED`, `ALLOWED`, and `DENIED` statuses are always fully qualified domain names. In the rule's domain list, you can also add wildcard domains, such as `*.example.com`, to allow all subdomains of a domain. DNS Firewall doesn't expand a wildcard domain into separate entries. Traffic for each matching subdomain still appears on the **Analysis** tab in the console. An observed domain that matches a wildcard domain that you just added can remain in the `OBSERVED` status for up to an hour.
- The rule's domain list counts toward your domain list and domain quotas. You can't delete the list directly, use it in another rule, replace its contents, or import domains into it.
- Only the owner of the rule group can view the observed domains. For shared rule groups, DNS Firewall combines the traffic from all VPCs and doesn't show which VPC or account queried a domain.
- If you associate a VPC with the rule group after you create the rule, DNS Firewall doesn't analyze that VPC's past DNS query traffic. It adds only domains from queries that the VPC makes after the association.
- When you delete the rule, DNS Firewall also deletes the rule's domain list. If you create a new Automated Allowlist rule, it gets a new domain list and doesn't keep the previous rule's observed or rejected domains.
- Observed domains can include domains that other rules blocked, including rules in other rule groups that block known threats. When a rule matches a query, DNS Firewall stops evaluating, so if the Automated Allowlist rule runs before a threat rule, adding such a domain lets queries for it resolve. To prevent this, put rules that block known threats in the same rule group as the Automated Allowlist rule, with a higher priority (a lower numeric priority setting). For threat rules in another rule group, associate that rule group with your VPCs at a lower priority number than the Automated Allowlist rule's rule group. You can't move a rule group that Firewall Manager associates as a last rule group ahead of your other rule groups. For more information, see Managing associations between your VPC and Resolver DNS Firewall rule group.

  If the threat rules reach your VPCs through a different Route 53 Profile than the Automated Allowlist rule, a rule group's priority in one Profile doesn't order it against a rule group in the

other Profile. Instead, put the threat rules in the same Profile at a lower priority number than the Automated Allowlist rule's rule group, associate them directly with your VPCs at a lower priority number, or use a Profile with a lower priority number. Changing a Profile's priority also changes which Profile's Resolver rules and private hosted zones apply to your VPCs. For more information, see How Route 53 Profile settings are prioritized.

- The Automated Allowlist rule always inspects the domains in a DNS redirection chain, such as the targets of CNAME records. Observed domains and the **Analysis** tab show only the domain names that your VPCs queried, not the targets in the answers. Before you add a rule that blocks all other domains, check the answers in the Resolver query logs for the VPCs that use the rule group, and add the redirection targets, or a wildcard domain that covers them, to the allowlist. Otherwise, queries for those domains are blocked. For VPCs in other accounts, the answers are in the query logs of the query logging configurations that those VPCs use, for example a configuration that you share with those accounts through AWS RAM. Query logs contain only queries that VPCs make after you configure logging. For more information, see Resolver query logging and Rule settings in DNS Firewall.

Creating an Automated Allowlist rule

When you create the rule, you choose the look-back period for the DNS query traffic that DNS Firewall uses to build the first set of observed domains, from 1 hour to 1 week. You can't change this setting after you create the rule.

If you already use a walled-garden setup with your own allowlist, you can optionally start the Automated Allowlist rule with the domains from that domain list. This is a one-time copy, and afterward the two lists are independent. The source must be a completed, Regional domain list that you own, not an AWS managed domain list or another Automated Allowlist domain list. Imported domains are added to the rule's domain list. While the import runs, the status of the rule's domain list is `IMPORTING`, and then it changes to `COMPLETE`, or to `COMPLETE_IMPORT_FAILED` if the import fails. You can't add, remove, or reject domains until the import finishes.

DNS Firewall creates a domain list for the rule, named `AutoAllowlist - rule-name`. The rule status is `CREATING` while DNS Firewall builds the first set of observed domains, and changes to `COMPLETE` (**Created** in the console) when they're ready.

For general information about rules, see Creating a rule group and rules and Rule settings in DNS Firewall.

Console

Before you start, you need a rule group that's associated with one or more VPCs. The rule group can be empty or can already contain rules. For more information, see Creating a rule group and rules and Managing associations between your VPC and Resolver DNS Firewall rule group.

To add an Automated Allowlist rule

1. Open the Amazon VPC console. In the navigation pane, under **DNS Firewall**, choose **Rule groups**. Choose the name of your rule group, and then on the **Rules** tab, choose **Add rule**.
2. Under **Rule details**, enter a **Name** for the rule. For **Tier option**, choose **Advanced**.
3. Under **Rule configurations**, for **Rule type**, choose **Automated allow list**.
4. For **Analysis timeframe**, choose the look-back period for the DNS query traffic that DNS Firewall analyzes to find observed domains, from 1 hour to 1 week. You can't change the timeframe after you create the rule.
5. (Optional) For **Import a domain list**, choose an existing domain list to copy into the allowlist. This is a one-time copy. Afterward, the two lists are independent.
6. Under **Action**, the action is always **ALLOW**. Choose **Add rule**.

   The rule appears on the **Rules** tab with the rule type **Automated allow list** and the status **Creating**. The console gives the rule the next available priority, after any existing rules in the rule group. If the rule group already has a rule that blocks all other domains, change the priority of that rule to a higher number than the Automated Allowlist rule's priority, so that the Automated Allowlist rule runs first. You can't change the Automated Allowlist rule until its status is **Created**.
7. (Optional) After you review the observed domains and add the ones that your workloads need to the allowlist, add a rule that blocks all other domains. Before you add it, compare the **Analysis** tab with **Allowed domains** to find domains that your workloads need that aren't in the allowlist yet. Also add the redirection targets from the Resolver query logs for the VPCs that use the rule group. The **Analysis** tab might not show every domain if a status is full, so also check the query names in those query logs for domains that the allowlist and its wildcard domains don't cover yet. Use a domain list that contains the wildcard *, set the action to **BLOCK**, and give the rule a higher priority number than the Automated Allowlist rule. For an example, see Resolver DNS Firewall walled garden example.

To review the observed domains, choose **View domain list** in the success message, or choose the rule's **Resource value**.

CLI

You can create an Automated Allowlist rule by running a AWS CLI command like the following and using your own values. Use `--priority` to set the rule's priority. To start the allowlist with the domains from an existing domain list, add `"ImportFromFirewallDomainListId":"`*`domain-list-id`*`"` to `AutoAllowlist`.

```
aws route53resolver create-firewall-rule \
    --creator-request-id my-request-id \
    --firewall-rule-group-id rslvr-frg-0123456789example \
    --name my-automated-allowlist \
    --priority 100 \
    --action ALLOW \
    --firewall-rule-type '{"AutoAllowlist":{"InitialDataTimeframeHours":24}}'
```

The response includes the rule's `FirewallThreatProtectionId` and, in `FirewallRuleType`, the `AutoAllowlistFirewallDomainListId` of the rule's domain list. You use these IDs to review and update domains. For more information, see create-firewall-rule in the *AWS CLI Command Reference*.

Reviewing and approving domains

You can view the domains for a rule by status. You can filter them by part of a domain name, and sort them by the time the domain was added, by requests in the last 24 hours, or by total requests.

DNS Firewall keeps up to 10,000 domains in each status for a rule. When a status is full, DNS Firewall doesn't add new domains to it until existing domains stop receiving queries, so the list and the **Analysis** tab might not include every domain that your VPCs query. For more information, see Considerations for Automated Allowlists.

Console

To review and approve domains

1. On the rule group's **Rules** tab, find the Automated Allowlist rule. When its **Status** is **Created**, choose the domain list ID in the **Resource value** column.
2. On the domain list page, on the **Domain list** tab, choose **Observed domains**.

3. Review the observed domains. For each domain, the table shows **Total requests**, **Daily frequency**, and **Date first observed**. To find specific domains, use **Filter resources** or sort by a column.
4. To allow domains, select them, and then choose **Add to allow list**. Allowing a domain can take up to 1 hour to take effect. While DNS Firewall applies the change, the **Domain list status** is **Updating**.
5. To reject domains, select them, and then choose **Reject**. Rejected domains appear under **Rejected domains**. To allow a rejected domain later, select it there, and then choose **Add to allow list**.
6. To see the domains that DNS Firewall allowed queries for, choose the **Analysis** tab. For each allowed domain, the table shows **Total requests**, **Daily frequency**, and **Date first observed**.

CLI

You can list the observed domains for a rule, with the most requested domains first, by running a AWS CLI command like the following and using your own values. To list allowed or denied domains, set `DomainStatus` to `ALLOWED` or `DENIED`. `ALLOWED` lists the domains that DNS Firewall allowed queries for, whether or not they're in the rule's domain list.

```
aws route53resolver list-firewall-rule-domains \
    --firewall-rule-group-id rslvr-frg-0123456789example \
    --firewall-threat-protection-id 0123456789example \
    --query-options DomainStatus=OBSERVED,SortBy=TOTAL_REQUESTS,SortOrder=DESC
```

The following is an example output after you run the command:

```
{
    "Domains": [
        {
            "DomainName": "example.com",
            "DomainStatus": "OBSERVED",
            "Last24HourRequests": 2000,
            "TotalRequests": 15000,
            "AddedTime": "2026-11-16T10:00:00Z"
        }
    ]
}
```

For more information, see list-firewall-rule-domains in the *AWS CLI Command Reference*.

To list the domains in the allowlist itself, specify the rule's domain list instead:

```
aws route53resolver list-firewall-rule-domains \
    --firewall-rule-group-id rslvr-frg-0123456789example \
    --firewall-domain-list-id rslvr-fdl-0123456789example
```

Updating domains in an allowlist

You can make the following updates to the rule's domain list:

- ADD – Approves a domain. You can also add domains that DNS Firewall hasn't observed, including wildcard domains.
- REMOVE – Removes a domain from the allowlist.
- DENY – Rejects an observed domain. If the domain is in the allowlist, DNS Firewall also removes it. DENY accepts exact domain names only, not wildcards.

Console

To add domains to an allowlist

1. On the domain list page, on the **Domain list** tab, choose **Allowed domains**, and then choose **Add domains**.
2. Under **Enter one domain per line**, enter up to 1,000 domains. You can add domains that DNS Firewall hasn't observed. To allow all subdomains of a domain, start the name with an asterisk, for example `*.example.com`.
3. Choose **Add**. The domains appear under **Allowed domains**. Allowing a domain can take up to 1 hour to take effect.
4. (Optional) After you add the domains that your workloads need, add a rule that blocks all other domains, if the rule group doesn't already have one. Before you add it, compare the **Analysis** tab with **Allowed domains** to find domains that your workloads need that aren't in the allowlist yet. Also add the redirection targets from the Resolver query logs for the VPCs that use the rule group. The **Analysis** tab might not show every domain if a status is full, so also check the query names in those query logs for domains that the allowlist and its wildcard domains don't cover yet. Use a domain list that contains the wildcard *, set the action to **BLOCK**, and give the rule a higher priority number than the Automated Allowlist rule. For an example, see Resolver DNS Firewall walled garden example.

To change or remove a domain, select it under **Allowed domains**, and then choose **Edit** or **Delete**.

CLI

You can approve observed domains by running a AWS CLI command like the following and using your own values. To remove or reject domains, set `--operation` to `REMOVE` or `DENY`.

```
aws route53resolver update-firewall-domains \
    --firewall-domain-list-id rslvr-fdl-0123456789example \
    --operation ADD \
    --domains example.com "*.example.org"
```

For more information, see update-firewall-domains in the *AWS CLI Command Reference*.

Pricing

Automated Allowlist rules are billed as DNS Firewall Advanced rules. For more information, see Amazon Route 53 Pricing.

Configuring logging for DNS Firewall

You can evaluate your DNS Firewall rules by using Amazon CloudWatch metrics and the Resolver query logs. The logs provide the domain list name for all alerts and blocking actions. For more information about Amazon CloudWatch, see Monitoring Resolver DNS Firewall rule groups with Amazon CloudWatch.

When you enable DNS Firewall, associate it to a VPC, and you have logging enabled, `firewall_rule_group_id`, `firewall_rule_action`, and `firewall_domain_list_id` are the DNS Firewall specific fields provided within your logs.

> **Note**
>
> The query logs will show the additional DNS Firewall fields for only the queries that are blocked by DNS Firewall rules.

To start logging the DNS queries that are filtered by DNS Firewall rules that originate in your VPCs, you perform the following tasks in the Amazon Route 53 console:

To configure Resolver query logging for DNS Firewall

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. Expand the Route 53 console menu. In the upper left corner of the console, choose the three horizontal bars

   ( )

   icon.
3. Within the Resolver menu, choose **Query logging**.
4. In the Region selector, choose the AWS Region where you want to create the query logging configuration.

   This must be the same Region where you created the VPCs that are associated with DNS Firewall that you want to log queries for. If you have VPCs in multiple Regions, you must create at least one query logging configuration for each Region.
5. Choose **Configure query logging**.
6. Specify the following values:

   Query logging configuration name

   Enter a name for your query logging configuration. The name appears in the console in the list of query logging configurations. Enter a name that will help you find this configuration later.

   Query logs destination

   Choose the type of AWS resource that you want VPC Resolver to send query logs to. For information about how to choose among the options (CloudWatch Logs log group, S3 bucket, and Firehose delivery stream), see AWS resources that you can send VPC Resolver query logs to.

   After you choose the type of resource, you can either create another resource of that type or choose an existing resource that was created by the current AWS account.

> **Note**
>
> You can choose only resources that were created in the AWS Region that you chose in step 4, the Region where you're creating the query logging configuration. If you choose to create a new resource, that resource will be created in the same Region.

VPCs to log queries for

This query logging configuration will log DNS queries that originate in the VPCs that you choose. Check the check box for each VPC in the current Region that you want VPC Resolver to log queries for, then choose **Choose**.

> **Note**
>
> VPC log delivery can be enabled only once for a specific destination type. The logs can't be delivered to multiple destinations of the same type. For example, VPC logs can't be delivered to two Amazon S3 destinations.

7. Choose **Configure query logging**.

> **Note**
>
> You should start to see DNS queries made by resources in your VPC in the logs within a few minutes of successfully creating the query logging configuration.

Sharing Resolver DNS Firewall rule groups between AWS accounts

You can share DNS Firewall rule groups between AWS accounts. To share rule groups, you use AWS Resource Access Manager (AWS RAM). The DNS Firewall console integrates with the AWS RAM console. For more information about AWS RAM, see the Resource Access Manager User Guide.

Note the following:

Associating shared rule groups with VPCs

If another AWS account has shared a rule group with your account, you can associate it with your VPCs the same way that you associate rule groups that you've created. For more information, see Managing associations between your VPC and Resolver DNS Firewall rule group.

Deleting or unsharing a shared rule group

If you share a rule group with other accounts and then either delete the rule group or stop sharing it, DNS Firewall removes all associations that the other accounts created between the rule group and their VPCs.

Maximum settings for rule groups and associations

Shared rule groups and their associations with VPCs are included in the counts for the accounts with which the rule groups are shared.

For current DNS Firewall quotas, see Quotas on Resolver DNS Firewall.

Permissions

To share a rule group with another AWS account, you must have permission to use the PutFirewallRuleGroupPolicy action.

Restrictions on the AWS account that a rule group is shared with

The account that a rule group is shared with can't change or delete the rule group.

Tagging

Only the account that created a rule group can add, delete, or see tags on the rule group.

To view the current sharing status of a rule group (including the account that shared the rule group or the account that a rule group is shared with), and to share rule groups with another account, perform the following procedure.

To view sharing status and share rule groups with another AWS account

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Rule groups**.
3. On the navigation bar, choose the Region where you created the rule group.

The **Sharing status** column shows the current sharing status of rule groups that were created by the current account or that are shared with the current account:

- **Not shared**: The current AWS account created the rule group, and the rule group is not shared with any other accounts.
- **Shared by me**: The current account created the rule group and shared it with one or more accounts.
- **Shared with me**: Another account created the rule group and shared it with the current account.

4. Choose the name of the rule group that you want to display sharing information for or that you want to share with another account.

   On the **Rule group: *`rule group name`*** page, the value under **Owner** displays the ID of the account that created the rule group. That's the current account unless the value of **Sharing status** is **Shared with me**. In that case, **Owner** is the account that created the rule group and shared it with the current account.

5. Choose **Share** to view additional information or to share the rule group with another account. A page in the AWS RAM console appears, depending on the value of **Sharing status**:

   - **Not shared**: The **Create resource share** page appears. For information about how to share the rule group with another account, organizational unit (OU), or organization, go to the step that follows this one.
   - **Shared by me**: The **Shared resources** page shows the rule groups and other resources that are owned by the current account and shared with other accounts.
   - **Shared with me**: The **Shared resources** page shows the rule groups and other resources that are owned by other accounts and shared with the current account.

6. To share a rule group with another AWS account, OU, or organization, specify the following values.

> **Note**
>
> You can't update sharing settings. To change any of the following settings, you must reshare a rule group with the new settings and then remove the old sharing settings.

Description

Enter a short description that helps you remember why you shared the rule group.

Resources

Choose the check box for the rule group that you want to share.

Principals

Enter the AWS account number, OU name, or organization name.

Tags

Specify one or more keys and the corresponding values. For example, you might specify **Cost center** for **Key** and specify **456** for **Value**.

These are the tags that AWS Billing and Cost Management provides for organizing your AWS bill; you can use also tags for other purposes. For more information about using tags for cost allocation, see Using cost allocation tags in the *AWS Billing User Guide*.

Enabling Resolver DNS Firewall protections for your VPC

You enable DNS Firewall protections for your VPC by associating one or more rule groups with the VPC. Whenever a VPC is associated with a DNS Firewall rule group, Route 53 VPC Resolver provides the following DNS Firewall protections:

- VPC Resolver routes the VPC's outbound DNS queries through DNS Firewall, and DNS Firewall filters the queries using the associated rule groups.
- VPC Resolver enforces the settings in the VPC's DNS Firewall configuration.

To provide DNS Firewall protections to your VPC, you do the following:

- Create and manage associations between your DNS Firewall rule groups and your VPC. For information about rule groups, see DNS Firewall rule groups and rules.
- Configure how you want VPC Resolver to handle DNS queries for the VPC during a failure, for example if DNS Firewall doesn't provide a response for a DNS query.

Managing associations between your VPC and Resolver DNS Firewall rule group

To view a rule group's VPC associations

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   Choose **DNS Firewall** in the navigation pane to open the DNS Firewall **Rule groups** page on the Amazon VPC console.

   - OR -

   Sign in to the AWS Management Console and open the

   the Amazon VPC console under https://console.aws.amazon.com/vpc/.
2. In the navigation pane, under **DNS Firewall**, choose **Rule groups**.
3. On the navigation bar, choose the Region for the rule group.
4. Select the rule group that you want to associate.
5. Choose **View details**. The rule group page displays.
6. Toward the bottom, you can see a tabbed details area that includes rules and associated VPCs. Choose the tab **Associated VPCs**.

To associate a rule group with a VPC

1. Locate the rule group's VPC associations by following the instructions in the preceding procedure **To view a rule group's VPC associations**.
2. In the **Associated VPCs** tab, choose **Associate VPC**.
3. Locate the VPC that you want to associate with the rule group in the dropdown. Select it, then choose **Associate**.

In the rule group page, your VPC is listed in the **Associated VPCs** tab. At first, the association's **Status** reports **Updating**. When the association is complete, the status changes to **Complete**.

To remove an association between a rule group and a VPC

1. Locate the rule group's VPC associations by following the instructions in the preceding procedure **To view a rule group's VPC associations**.
2. Select the VPC that you want to remove from the list, then choose **Disassociate**. Verify, and then confirm the action.

On the rule group page, your VPC is listed in the **Associated VPCs** tab with the status of **Disassociating**. When the operation completes, DNS Firewall updates the list to remove the VPC.

DNS Firewall VPC configuration

The DNS Firewall configuration for your VPC determines whether Route 53 VPC Resolver allows queries through or blocks them during failures, for example when DNS Firewall is impaired, unresponsive, or not available in the zone. VPC Resolver enforces a VPC's firewall configuration whenever you have one or more DNS Firewall rule groups associated with the VPC.

You can configure a VPC to fail open or fail closed.

- By default, the failure mode is closed, which means that VPC Resolver blocks any queries for which it doesn't receive a reply from DNS Firewall and sends a `SERVFAIL` DNS response. This approach favors security over availability.
- If you enable fail open, VPC Resolver allows queries through if it doesn't receive a reply from DNS Firewall. This approach favors availability over security.

To change the DNS Firewall configuration for a VPC (console)

1. Sign in to the AWS Management Console and open the VPC Resolver console at https://console.aws.amazon.com/route53resolver/.
2. In the navigation pane under **Resolvers**, choose **VPCs**.
3. In the **VPCs** page, locate and edit the VPC. Change the DNS Firewall configuration to fail open or fail closed as needed.

To change the DNS Firewall behavior for a VPC (API)

- Update your VPC firewall configuration by calling UpdateFirewallConfig and enabling or disabling `FirewallFailOpen`.

You can retrieve a list of your VPC firewall configurations through the API by calling ListFirewallConfigs.

Resolved IP validation with Network Firewall

With AWS Network Firewall resolved IP validation, you can detect and block connections that try to bypass domain-based filtering by connecting directly to an IP address. A standard Network Firewall domain rule matches the host name that a client asserts. Examples include the TLS Server Name Indication (SNI) and the HTTP `Host` header. If a client sets a `Host` header for an allowed domain but connects to an unrelated IP address, resolved IP validation detects the mismatch.

When a connection reaches the firewall, Network Firewall checks whether the Route 53 VPC Resolver returned the connection's destination IP address for any domain. You can also add Network Firewall rules that check whether the VPC Resolver returned the destination IP address for the domain that the connection targets. Network Firewall checks resolutions only for the VPCs that are associated with the Resolver DNS Firewall rule group that you specify.

Because resolved IP validation doesn't rely on an application-layer header, you can also apply domain-based filtering to traffic that has no domain name, such as protocols other than HTTP and HTTPS.

To get started, see Resolved IP validation in AWS Network Firewall in the *Network Firewall Developer Guide*.

Using resolved IP validation with and without DNS Firewall rules

DNS Firewall rules filter outbound DNS queries, whereas resolved IP validation acts on the connections that your VPCs make after those queries resolve. You can use resolved IP validation with or without DNS Firewall rules:

- **Without DNS Firewall rules** – To use resolved IP validation without DNS Firewall domain filtering, associate a rule group that contains no rules with the VPCs whose traffic the firewall inspects. DNS Firewall doesn't filter any queries, but resolved IP validation still works.
- **With DNS Firewall rules** – When you add rules to the rule group, DNS Firewall filtering and resolved IP validation work as two independent layers of protection. If a rule blocks a DNS query, the VPC Resolver never returns an IP address for that query. As a result, Network Firewall doesn't allow connections based on it. If a rule only alerts on a query, the query still resolves and the connection can pass validation.

A rule group with no rules that you use only for resolved IP validation doesn't incur DNS Firewall charges. Standard DNS Firewall pricing applies after you add rules to the rule group.

What are Amazon Route 53 Profiles?

With Route 53 Profiles, you can apply and manage DNS-related Route 53 configurations across many VPCs and in different AWS accounts. Profiles make managing the DNS settings for many VPCs as easy as managing them for a single VPC and when you update a Profile, its settings are propagated to all the VPCs associated to the Profile. You can also share a Profile with AWS accounts in the same Regions by using AWS RAM. The currently Route 53 supported resources you can associate to a Profile are:

- Private hosted zones and the settings specified in them. For more information about working with private hosted zones, see Working with private hosted zones.
- Resolver rules, both forwarding and system. For more information about Resolver rules, see Managing forwarding rules.
- DNS Firewall rule groups. For more information about DNS Firewall rule groups, see DNS Firewall rule groups and rules.
- Interface VPC endpoints. For more information about interface VPC endpoints, see interface VPC endpoints in the *Amazon VPC User Guide*.
- VPC Resolver query logging configurations. For more information about VPC Resolver query logging, see Resolver query logging.

Some of the VPC configurations are directly managed on the Profile. The configurations are:

- Reverse DNS lookup configuration for Resolver Rules.
- DNS Firewall failure mode configuration.
- DNSSEC validation configuration.

For example, you can enable the DNS Firewall failure mode configuration for all the VPCs the Profile is associated to, but keep the VPC's existing DNSSEC validation configuration.

> **Important**
>
> After you enable the Profile settings for the preceding configurations, and associate the Profile to a VPC, the Profile settings take effect immediately.

You can also use CloudFormation to set up consistent DNS settings for newly provisioned VPCs.

You can associate up to two Profiles to a VPC. Each association between a Profile and a VPC has a *priority*, a value from 100 to 9900. A lower value means a higher priority. VPC Resolver evaluates the Profiles that are associated to a VPC in priority order. For more information, see How Route 53 Profile settings are prioritized. The number of resources you can associate per Profile varies. For more information, see Quotas on Route 53 Profiles.

How Route 53 Profile settings are prioritized

You can have the local DNS settings and associations set for Profiles for migration, or other testing purposes. When a DNS query matches both the Resolver rule for a private hosted zone that is directly associated with the VPC and a Resolver rule for a private hosted zone that is associated to the Profile, the local DNS settings take precedence. When DNS query is made for a conflicting domain name, the most specific one wins. The following table includes examples of the evaluation order:

DNS evaluation order examples

DNS query	Profile rule	VPC rule	Evaluated rule
example.com	example.com	example.com	Local VPC
test.example.com	test.example.com	example.com	Profile
marketing.example.com	None	marketing.example.com	Local VPC

When more than one Profile is associated to a VPC, each association has a *priority*, a value from 100 to 9900. A lower value means a higher priority. VPC Resolver evaluates the Profiles in priority order, starting with the association that has the lowest priority value:

- **Resolver rules and private hosted zones** – For each DNS query, VPC Resolver uses the first Profile, in priority order, that has a Resolver rule or a private hosted zone that matches the domain name in the query. VPC Resolver doesn't evaluate the remaining Profiles for that query, even if one of them has a more specific match. VPC Resolver then compares the settings in

that Profile with the local VPC settings: the most specific domain name wins, and the local VPC settings take precedence when both are equally specific.

- **Resolver DNS Firewall rule groups** – DNS Firewall evaluates the rule groups that are associated directly to the VPC together with the rule groups in the highest priority Profile association that has a rule that matches the query. The rule group priority determines the evaluation order, and a rule group that is associated directly to the VPC takes precedence over a Profile rule group that has the same rule group priority. DNS Firewall doesn't evaluate rule groups in the remaining Profiles for that query.
- **VPC Resolver query logging configurations** – VPC Resolver logs queries to the query logging configurations in every Profile that is associated to the VPC, regardless of priority.
- **Profile configurations** – For reverse DNS lookup configuration for Resolver rules, DNS Firewall failure mode configuration, and DNSSEC validation configuration, the setting in the highest priority Profile that sets the configuration applies. VPC Resolver skips Profiles that use the VPC's own setting. If no associated Profile sets the configuration, the VPC's own setting applies.

For example, Profile A is associated to a VPC with priority 100, Profile B is associated to the same VPC with priority 200, and the VPC has a local Resolver rule for `corp.example.com`. The following table shows which rule VPC Resolver uses for queries from this VPC.

DNS query	Profile A rule (priority 100)	Profile B rule (priority 200)	VPC rule	Evaluated rule
test.example.com	example.com	test.example.com	None	Profile A (example.com)
app.example.net	None	example.net	None	Profile B (example.net)
hr.corp.example.com	example.com	hr.corp.example.com	corp.example.com	Local VPC (corp.example.com)

If a VPC has no other Profile associations, you can omit the priority when you associate a Profile, and the association gets priority 100. When the VPC already has a Profile association, you must

specify a priority, and each Profile association for the VPC must have a unique priority. Associations that were created before you could associate multiple Profiles to a VPC have priority 100. You can change the priority later. For more information, see Updating the priority of a Route 53 Profile association with a VPC.

Every Profile association for a VPC that isn't deleted keeps its priority and counts toward the quota for Profiles per VPC. This includes associations whose status is `FAILED` or `DELETING`. When you disassociate a Profile from a VPC, the other Profile associations for the VPC keep their priorities, and VPC Resolver evaluates the remaining Profiles in priority order.

Route 53 Profiles Region availability

To view the Region availability and the endpoints, see Service endpoints for Route 53 in the *AWS General Reference* guide.

High-level steps for using Route 53 Profiles

To implement Amazon Route 53 Profiles in your Amazon Virtual Private Cloud VPCs, you perform the following high-level steps.

1. **Create an empty Profile** – The first step is to create an empty Profile to which you can associate DNS resources. For more information, see Creating Route 53 Profiles.
2. **Associate DNS resources to the Profile** – The resources you can currently associate to a Profile are private hosted zones, Resolver rules, both forwarding and system, DNS Firewall rule groups, VPC Resolver query logging configurations, and interface VPC endpoints. For more information, see Associate DNS Firewall rule groups to a Route 53 Profile, Associate private hosted zones to a Route 53 Profile, Associate Resolver rules to a Route 53 Profile, Associate VPC Resolver query logging configurations to a Route 53 Profile, and Associate interface VPC endpoints to a Route 53 Profile.
3. **Configure some of the VPC settings for the Profile** – Some of the DNS settings, like hosted zones associated to the Profile, are applied to the VPCs immediately. For DNSSEC validation, Resolver reverse DNS lookup, and DNS Firewall failure mode configurations you can choose one of the following options:
   - For DNSSEC validation, you can choose to use the local VPC configuration (default), enable the validation, or disable the validation for all the VPCs associated to the Profile.
   - For Resolver reverse DNS lookup configuration you can enable it, disable it, or use the auto defined rules defined for the VPC locally (default).

- For DNS Firewall failure mode configuration you can enable it, disable it, or use the failure mode configuration defined for the VPC locally (default).

For more information, see Edit Route 53 Profile configurations.

4. **Associate the Profile to one or more VPCs** – To begin using your Profile, associate it with one or more VPCs. For more information, see Associate a Route 53 Profile to VPCs.

Creating Route 53 Profiles

To create Route 53 Profiles, follow the guidance in this topic. Choose a tab to create a Route 53 Profile by using the Route 53 console, or AWS CLI.

- Console
- CLI

Console

To create a Route 53 Profile

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. On the navigation bar, choose the Region where you want to create the Profile.
4. Enter a name for the Profile, optionally add tags, and choose **Create Profile**.

   This creates an empty Profile with default configurations to which you can associate resources. After you associate resources to the Profile, you can associate it to a number of VPCs and edit the how some of the Resolver configurations apply to the VPCs.

CLI

You can create a Profile by running a AWS CLI command like the following and using your own value for `name`.

```
aws route53profiles create-profile --name test
```

The following is an example output after you run the command:

```
{
    "Profile": {
        "Arn": "arn:aws:route53profiles:us-east-1:123456789012:profile/
rp-6ffe47d5example",
        "ClientToken": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
        "CreationTime": 1710850903.578,
        "Id": "rp-6ffe47d5example",
        "ModificationTime": 1710850903.578,
        "Name": "test",
        "OwnerId": "123456789012",
        "ShareStatus": "NOT_SHARED",
        "Status": "COMPLETE",
        "StatusMessage": "Created Profile"
    }
}
```


To associate your Profiles with different resources and edit the VPC configurations for the Profile, see the following procedures:

Topics

- Associate DNS Firewall rule groups to a Route 53 Profile
- Associate private hosted zones to a Route 53 Profile
- Associate Resolver rules to a Route 53 Profile
- Associate interface VPC endpoints to a Route 53 Profile
- Associate VPC Resolver query logging configurations to a Route 53 Profile
- Edit Route 53 Profile configurations
- Associate a Route 53 Profile to VPCs

Associate DNS Firewall rule groups to a Route 53 Profile

For instructions for creating a rule group, see Creating a rule group and rules, and then choose a tab to associate DNS Firewall rule groups to a Route 53 Profile by using the Route 53 console, or AWS CLI.

- Console
- CLI

Console

To associate DNS Firewall rule groups

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the navigation bar, choose the Region where you created the Profile.
3. In the navigation pane, choose **Profiles** and on the **Profiles** table, choose the linked name of the Profile you want to work with.
4. On the **<Profile name>** page, choose the **DNS Firewall rule groups** tab and then **Associate**.
5. In the **DNS Firewall rule groups** section you can select up to 10 rule groups you have previously created. If you want to associate more than 10 rule groups, use the APIs. For more information, see AssociateResourceToProfile.

   To create new rule groups, see Creating a rule group and rules.
6. Choose **Next**.
7. On the **Define priority** page you can set the order in which the rule groups are processed by clicking the pre-assigned priority number and typing in a new one. The allowed values for the priority are between 100 and 9900.

   The rule groups are evaluated starting with the lowest numeric priority setting and going up. You can change a rule group's priority at any time, for example to change the order of processing or make space for other rule groups.

   Choose **Submit**.
8. The association progress is displayed in the **Status** column in the **DNS Firewall** rule groups dialog box.

CLI

You can associate rule group to a Profile by running a AWS CLI command like the following and using your own values for `name` `profile-id`, `resource-arn`, and `priority`:

```
aws route53profiles associate-resource-to-profile --name test-
resource-association --profile-id rp-4987774726example --resource-arn
arn:aws:route53resolver:us-east-1:123456789012:firewall-rule-group/
rslvr-frg-cfe7f72example --resource-properties "{\"priority\": 102}"
```

The following is an example output after you run the command:

```
{
    "ProfileResourceAssociation": {
        "CreationTime": 1710851216.613,
        "Id": "rpr-001913120a7example",
        "ModificationTime": 1710851216.613,
        "Name": "test-resource-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceArn": "arn:aws:route53resolver:us-east-1:123456789012:firewall-
rule-group/rslvr-frg-cfe7f72example",
        "ResourceProperties": "{\"priority\":102}",
        "ResourceType": "FIREWALL_RULE_GROUP",
        "Status": "UPDATING",
        "StatusMessage": "Updating the Profile to DNS Firewall rule group
 association"
    }
}
```

Associate private hosted zones to a Route 53 Profile

For intructions for how to create a private hosted zone, see Creating a private hosted zone, and then follow the steps in this procedure to associate a private hosted zone to a Profile.

To associate private hosted zones

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the navigation bar, choose the Region where you created the Profile.
3. In the navigation pane, choose **Profiles** and on the **Profiles** table, choose the linked name of the Profile you want to work with.
4. On the **<Profile name>** page, choose the **Private hosted zones** tab, and then **Associate**.
5. On the **Associate private hosted zones** page you can select up to 10 private hosted zones you have previously created. If you want to associate more than 10 private hosted zones, use the APIs. For more information, see AssociateResourceToProfile.

   To create private hosted zones, see Creating a private hosted zone.
6. Choose **Associate**

7. The association progress is displayed in the **Status** column on the **Private hosted zones** tab.

Associate Resolver rules to a Route 53 Profile

For instructions for how to create a Resolver rule, see Creating forwarding rules, and then follow the steps in this procedure to associate Resolver rules to a Profile.

To associate VPC Resolver rules

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the navigation bar, choose the Region where you created the Profile.
3. On the **<Profile name>** page, choose the **Resolver rules** tab, and then **Associate**.
4. On the **Associate Resolver rules** page, in the **Resolver rules** table you can select up to 10 Resolver rules you have previously created. If you want to associate more than 10 resolver rules, use the APIs. For more information, see AssociateResourceToProfile.

   To create Resolver rules, see Creating forwarding rules.
5. Choose **Associate**
6. The association progress is displayed in the **Status** column on the **Resolver rules** tab.

Associate interface VPC endpoints to a Route 53 Profile

For instructions on how to create a interface VPC endpoint, see Create a VPC endpoint in the *VPC User Guide.* and then follow the steps in this procedure to associate a VPC endpoint to a Profile.

To associate VPC endpoints

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the navigation bar, choose the Region where you created the Profile.
3. On the **<Profile name>** page, choose the **VPC endpoints** tab, and then **Associate**.
4. On the **Associate VPC endpoints** page, in the **VPC endpoints** table you can select up to 10 endpoints you have previously created. If you want to associate more than 10 endpoints, use the APIs. For more information, see AssociateResourceToProfile.

To create Resolver rules, see Creating forwarding rules.

5. Choose **Associate**
6. The association progress is displayed in the **Status** column on the **VPC endpoints** tab.

Associate VPC Resolver query logging configurations to a Route 53 Profile

For instructions on how to create a VPC Resolver query logging configuration, see Configuring (VPC Resolver query logging), and then choose a tab to associate VPC Resolver configuration to a Route 53 Profile by using the Route 53 console, or AWS CLI.

- Console
- CLI

Console

To associate query logging configurations

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the navigation bar, choose the Region where you created the Profile.
3. On the **Profile name** page, choose the **Resolver query log configurations** tab, and then **Associate**.
4. On the **Associate query logging configurations** page, in the **Resolver query log configurations** table, you can select up to three configurations that you previously created. If you want to associate more than three query logging configurations, use the API. For more information, see AssociateResourceToProfile.
5. To create new VPC Resolver query logging configurations, see Configuring (VPC Resolver query logging).
6. Choose **Associate**
7. The association progress is displayed in the **Status** column on the **Resolver query log configurations** tab.

CLI

You can associate a query log configuration to a Profile by running a AWS CLI command like the following and using your own values for `name` `profile-id` and `resource-arn`.

```
aws route53profiles associate-resource-to-profile --name test-resource-association --profile-id rp-4987774726example --resource-arn arn:aws:route53resolver:us-east-1:123456789012:resolver-query-log-config/rqlc-cfe7f72example
```

The following is an example output after you run the command:

```
{
    "ProfileResourceAssociation": {
        "CreationTime": 1710851226.613,
        "Id": "rpr-001913120b8example",
        "ModificationTime": 1710851226.613,
        "Name": "test-resource-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceArn": "arn:aws:route53resolver:us-east-1:123456789012:resolver-query-log-config/rqlc-cfe7f72example",
        "ResourceType": "RESOLVER_QUERY_LOG_CONFIG",
        "Status": "CREATING",
        "StatusMessage": "Creating rp-4987774726example to rqlc-cfe7f72example association"
    }
}
```

Edit Route 53 Profile configurations

After you associate resources to a Profile, you can edit the default VPC configurations to decide how they are applied to the VPCs.

To edit Profile configurations

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the navigation bar, choose the Region where you created the Profile.

3. In the navigation pane, choose **Profiles** and on the **Profiles** table, choose the linked name of the Profile you want to work with.
4. On the **<Profile name>** page, choose the **Configuration** tab and then **Edit**.
5. On the **Edit Configuration** page, choose one of the values for the VPC DNSSEC configuration, Resolver reverse DNS lookup configuration, and DNS Firewall failure mode configuration.

   For more information about the values, see Configuration settings for Route 53 Profile.
6. Choose **Update**.

Configuration settings for Route 53 Profile

When you edit a Route 53 Profile configuration, you specify the following values:

DNSSEC configuration

Choose one of the following values:

- **Use local VPC DNSSEC configuration - default**

  Choose this option to have all the VPCs associated to this Profile keep their local DNSSEC validation configuration.
- **Enable DNSSEC validation**

  Choose this option to enable DNSSEC validation in all the VPCs associated to this Profile.
- **Disable DNSSEC validation**

  Choose this option to disable DNSSEC validation in all VPCs that are associated to this Profile.

Resolver reverse DNS lookup configuration

Choose one of the following values:

- **Enable**

  Choose this option to create auto defined rules for reverse DNS look up in all the associated VPCs.
- **Not enabled**

  Choose this option to not create auto defined rules for reverse DNS look up in all the associated VPCs.

- **Use local auto defined rules - default**

  Choose this option to use the local VPC settings for reverse DNS lookup for the associated VPCs.

DNS Firewall failure mode configuration

Choose one of the following values:

- **Disable**

  Choose this option to close the DNS Firewall failure mode for the associated VPCs. With this option, DNS Firewall will block all queries it can't properly evaluate.

- **Enabled**

  Choose this option to keep the DNS Firewall failure mode open for all the associated VPCs. With this option, DNS Firewall will allow queries to proceed if it's unable to properly evaluate them.

- **Use local failure mode settings - default**

  Choose this option to use the local VPC DNS Firewall failure mode settings.

For more information about the configurations, see

- Enabling DNSSEC validation in Amazon Route 53
- Forwarding rules for reverse DNS queries in VPC Resolver
- DNS Firewall VPC configuration

Associate a Route 53 Profile to VPCs

To associate a Route 53 Profile to a VPC, follow the guidance in this topic. Choose a tab to associate a Route 53 Profile to a VPC by using the Route 53 console, or AWS CLI.

- Console
- CLI

Console

To associate VPCs

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the navigation bar, choose the Region where you created the Profile.
3. On the **<Profile name>** page, choose the **VPCs** tab, and then **Associate**.
4. On the **Associate VPCs** page you can select up to 10 VPCs you have previously created. If you want to associate more than 10 VPCs, use the APIs. For more information, see AssociateProfile.
5. (Optional) In the **Review profile prioritization** table, set the **Priority** for the new association, and change the priority of the Profiles that are already associated to the VPC if needed. To change a priority, choose the **Edit priority** (pencil) icon in the **Priority** column, enter a value from 100 to 9900, and then choose **Submit edit**. A lower value means a higher priority, and each Profile associated to a VPC must have a unique priority. For more information, see How Route 53 Profile settings are prioritized.
6. Choose **Associate**
7. The association progress is displayed in the **Status** column on the **VPCs** page.

CLI

To associate a Profile to a VPC, use the associate-profile AWS CLI command. For example, the following command associates Profile `rp-4987774726example` to VPC `vpc-0af3b96b3example`. Use your own values for `name`, `profile-id`, and `resource-id`.

```
aws route53profiles associate-profile --name test-association --profile-id rp-4987774726example --resource-id vpc-0af3b96b3example
```

The following is an example output after you run the command:

```
{
    "ProfileAssociation": {
        "CreationTime": 1709338817.148,
        "Id": "rpassoc-489ce212fexample",
        "ModificationTime": 1709338817.148,
        "Name": "test-association",
```

```
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceId": "vpc-0af3b96b3example",
        "Priority": 100,
        "Status": "CREATING",
        "StatusMessage": "Creating Profile Association"
    }
}
```

To set the order in which VPC Resolver evaluates the Profiles associated to the VPC, add the `--priority` parameter with a value from 100 to 9900. A lower value means a higher priority. If the VPC has no other Profile associations, you can omit `--priority`, and the association gets priority 100. If the VPC already has a Profile association, `--priority` is required and must be different from the priority of every other Profile association for the VPC.

Viewing and updating Amazon Route 53 Profiles

Choose the console tab to view and edit Route 53 Profile. Choose the CLI tab to use AWS CLI to list Profiles you own, are shared by you, or shared to you.

- Console
- CLI

Console

Viewing and updating Route 53 Profiles

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. Select the button next to the name of the Profile you want to view or edit.
4. On the **<Profile name>** page you can view the currently associated DNS resources, associate new ones, and edit the tags and VPC configurations.

CLI

You can list the Profiles by running a AWS CLI command like the following:

```
aws route53profiles list-profiles
```

The following is an example output after you run the command:

```
{
    "ProfileSummaries": [
        {
            "Arn": "arn:aws:route53profiles:us-east-1:123456789012:profile/
rp-4987774726example",
            "Id": "rp-4987774726example",
            "Name": "test",
            "ShareStatus": "NOT_SHARED"
        }
    ]
}
```

You can get information about a particular VPS the Profile is associated to by running an AWS CLI command like the following and using your own value for `profile-association-id`:

```
aws route53profiles get-profile-association --profile-association-id
rpassoc-489ce212fexample
```

The following is an example output after you run the command:

```
{
    "ProfileAssociation": {
        "CreationTime": 1709338817.148,
        "Id": "rpassoc-489ce212fexample",
        "ModificationTime": 1709338974.772,
        "Name": "test-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceId": "vpc-0af3b96b3example",
        "Priority": 100,
        "Status": "COMPLETE",
        "StatusMessage": "Created Profile Association"
    }
}
```

Deleting a Amazon Route 53 Profile

Choose a tab to delete a Route 53 Profile by using the Route 53 console, or AWS CLI.

- Console
- CLI

Console

To delete a Route 53 Profile

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. Select the button next to the name of the Profile you want to delete, and then choose **Delete**.

> **Important**
>
> You can't delete a Profile if it is associated to VPCs. Additionally, if the Profile is shared to another AWS account, any VPCs that the Profile configurations are associated to, will lose those configurations.

4. On the **Delete <Profile name>** dialog, type in **`confirm`**, and then choose **Delete**.

CLI

> **Important**
>
> You can't delete a Profile if it is associated to VPCs. Additionally, if the Profile is shared to another AWS account, any VPCs that the Profile configurations are associated to, will lose those configurations.

You can delete a Profile by running an AWS CLI command like the following and using your own value for `profile-id`:

```
aws route53profiles delete-profile --profile-id rp-6ffe47d5example
```

The following is an example output after you run the command:

```
{
```

```
    "Profile": {
        "Arn": "arn:aws:route53profiles:us-east-1:123456789012:profile/
rp-6ffe47d5example",
        "ClientToken": "0a15fec0-05d9-4f78-bec0-EXAMPLE11111",
        "CreationTime": 1710850903.578,
        "Id": "rp-6ffe47d5example",
        "ModificationTime": 1710850903.578,
        "Name": "test",
        "OwnerId": "123456789012",
        "ShareStatus": "NOT_SHARED",
        "Status": "DELETED",
        "StatusMessage": "Deleted Profile"
    }
}
```

Viewing and updating Route 53 resources associated to an Amazon Route 53 Profile

Choose the console tab to view the Route 53 Profile resource associations, and optionally edit the DNS Firewall rule group priority. Choose the CLI tab to use AWS CLI to list the resource associations and to see an example update to a priority of a DNS Firewall rule group.

- Console
- CLI

Console

To view and update resources associated to a Profile

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. On the navigation bar, choose the Region where you created the Profile.
4. Select the button next to the name of the Profile for which you want to view or edit the resource associations.
5. On the **<Profile name>** page choose the tab for the resource you want to view or edit, either , **DNS Firewall rule groups**, **Private hosted zones**, **Resolver rules**, or **VPC endpoints**.

6. On the tab page for a resource you can view the names, ARN and status for the associated resources. You can also choose the gear icon to adjust what is displayed in the resource table.

   On the **DNS Firewall rule groups** tab page you can also choose the rule group priority entry, and edit it to a smaller or a bigger number. The rule groups are evaluated in order starting from the lowest priority number in order to the highest priority number.

CLI

You can list resources associated to a Profile by running an AWS CLI command like the following and using your own value for `profile-id`:

`aws route53profiles list-profile-resource-associations --profile-id` **`rp-4987774726example`**

The following is an example output after you run the command:

```
{
    "ProfileResourceAssociations": [
        {
            "CreationTime": 1710851216.613,
            "Id": "rpr-001913120a7example",
            "ModificationTime": 1710851216.613,
            "Name": "test-resource-association",
            "OwnerId": "123456789012",
            "ProfileId": "rp-4987774726example",
            "ResourceArn": "arn:aws:route53resolver:us-east-1:123456789012:firewall-
rule-group/rslvr-frg-cfe7f72example",
            "ResourceProperties": "{\"priority\":102}",
            "ResourceType": "FIREWALL_RULE_GROUP",
            "Status": "COMPLETE",
            "StatusMessage": "Completed creation of Profile to DNS Firewall rule
 group association"
        }
    ]
}
```

You can update the priority of a DNS Firewall rule group associated to a Profile by running an AWS CLI command like the following and using your own value for and using your own values for `profile-resource-association-id` and `--resource-properties`:

```
aws route53profiles update-profile-resource-association --profile-
resource-association-id rpr-001913120a7example --resource-properties
"{\"priority\": 105}"
```

The following is an example output after you run the command:

```
{
    "ProfileResourceAssociation": {
        "CreationTime": 1710851216.613,
        "Id": "rpr-001913120a7example",
        "ModificationTime": 1710852303.798,
        "Name": "test-resource-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceArn": "arn:aws:route53resolver:us-east-1:123456789012:firewall-
rule-group/rslvr-frg-cfe7f72example",
        "ResourceProperties": "{\"priority\":105}",
        "ResourceType": "FIREWALL_RULE_GROUP",
        "Status": "UPDATING",
        "StatusMessage": "Updating the Profile to DNS Firewall rule group
 association"
    }
}
```

Disassociating a resource from an Amazon Route 53 Profile

Before you delete a Profile, you must dissociate all resources from it.

To disassociate a resource associated to a Route 53 Profile

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. On the navigation bar, choose the Region where the Profile from which you want to disassociate a resource was created.
4. Select the button next to the name of the Profile from which you want to disassociate a resource.

5. On the **Profile name** page choose the tab for the resource you want to delete, either , **DNS Firewall rule groups**, **Private hosted zones**, **Resolver query logging**, **Resolver rules** or **VPC endpoints**.
6. On the tab page for the resource, choose the resource you want to disassociate and then **Disassociate**.
7. In the **Disassociate resources** dialog, type in **`confirm`**, and then choose **Disassociate**.

Viewing VPCs associated to an Amazon Route 53 Profile

Choose the console tab to view and edit Route 53 Profile to VPC associations. Choose the CLI tab to use AWS CLI to list Profile to VPC associations, or to get information about a specific association

- Console
- CLI

Console

To view VPCs associated to a Profile

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. On the navigation bar, choose the Region where you created the Profile.
4. Select the button next to the name of the Profile for which you want to view the associated VPCs.
5. On the **<Profile name>** page choose the **VPCs** tab.
6. On the tab page for VPCs you can view the names, ARN and status for the associated VPCs.

CLI

You can list the VPCs the Profile is associated to by running an AWS CLI command like the following:

```
aws route53profiles list-profile-associations
```

The following is an example output after you run the command:

```
{
    "ProfileAssociations": [
        {
            "CreationTime": 1709338817.148,
            "Id": "rpassoc-489ce212fexample",
            "ModificationTime": 1709338974.772,
            "Name": "test-association",
            "OwnerId": "123456789012",
            "ProfileId": "rp-4987774726example",
            "ResourceId": "vpc-0af3b96b3example",
            "Priority": 100,
            "Status": "COMPLETE",
            "StatusMessage": "Created Profile Association"
        }
    ]
}
```


You can get information about a particular VPS the Profile is associated to by running an AWS CLI command like the following and using your own value for `profile-association-id`:

`aws route53profiles get-profile-association --profile-association-id` **`rpassoc-489ce212fexample`**

The following is an example output after you run the command:


```
{
    "ProfileAssociation": {
        "CreationTime": 1709338817.148,
        "Id": "rpassoc-489ce212fexample",
        "ModificationTime": 1709338974.772,
        "Name": "test-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceId": "vpc-0af3b96b3example",
        "Priority": 100,
        "Status": "COMPLETE",
        "StatusMessage": "Created Profile Association"
    }
}
```

Disassociating a VPC from an Amazon Route 53 Profile

Choose a tab to dissociate a Route 53 Profile from a VPC by using the Route 53 console, or AWS CLI.

- Console
- CLI

Console

To disassociate a VPC associated to a Route 53 Profile

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. On the navigation bar, choose the Region where the Profile from which you want to disassociate a VPC was created.
4. Select the button next to the name of the Profile from which you want to disassociate a VPC.
5. On the **<Profile name>** page choose the **VPCs** tab.
6. On the VPCs tab page for the resource, choose the VPC you want to disassociate and then **Disassociate**.
7. In the **Disassociate resources** dialog, type in **`confirm`**, and then choose **Disassociate**.

CLI

To disassociate a Profile from a VPC, use the disassociate-profile AWS CLI command. For example, the following command disassociates Profile `rp-4987774726example` from VPC `vpc-0af3b96b3example`. Use your own values for `profile-id` and `resource-id`.

```
aws route53profiles disassociate-profile --profile-id
rp-4987774726example --resource-id vpc-0af3b96b3example
```

The following is an example output after you run the command:

```
{
    "ProfileAssociation": {
```

```
        "CreationTime": 1710851336.527,
        "Id": "rpassoc-489ce212fexample",
        "ModificationTime": 1710851401.362,
        "Name": "test-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceId": "vpc-0af3b96b3example",
        "Priority": 100,
        "Status": "DELETING",
        "StatusMessage": "Deleting Profile Association"
    }
}
```

Updating the priority of a Route 53 Profile association with a VPC

When more than one Profile is associated to a VPC, the priority of each association determines the order in which VPC Resolver evaluates the Profiles. The priority is a value from 100 to 9900, a lower value means a higher priority, and each Profile association for a VPC must have a unique priority. For more information, see How Route 53 Profile settings are prioritized.

Only the account that associated the Profile to the VPC can change the priority of the association. While the change propagates, the association status is `UPDATING`. When the new priority is in effect, the status changes to `COMPLETE`.

If you associate a Profile that another account shared with you to one of your VPCs, your account sets the priority of that association, and only your account can change it. The account that owns the Profile can't change the priority.

Because each Profile association for a VPC must have a unique priority, you can't swap the priorities of two associations in a single request. To swap them, make these three changes:

1. Change the priority of the first association to a temporary value that no other association for the VPC uses.
2. Change the priority of the second association to the original priority of the first association.
3. Change the priority of the first association to the original priority of the second association.

You can make these changes in the console, the AWS CLI, or the API. On the **Associated VPC: <VPC ID>** page in the console, you change one priority at a time.

When you associate a Profile to a VPC in the console, you can also change the priorities of the Profiles that are already associated to the VPC, and the console makes the changes in an order that avoids duplicate priorities. If one of the changes fails, a Profile might be left at a temporary priority, so check the priorities of the VPC's Profiles and set them again. For more information, see Associate a Route 53 Profile to VPCs.

Choose a tab to update the priority by using the Route 53 console or the AWS CLI.

Console

To update the priority of a VPC association

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. On the navigation bar, choose the Region where you created the Profile.
4. Choose the **Associated VPCs** tab, and then choose the ID of the VPC you want to work with. You can also choose the linked name of a Profile, choose the **VPCs** tab, and then choose the name of the VPC.
5. On the **Associated VPC: <VPC ID>** page, in the **Associated profiles** table, find the Profile whose priority you want to change. If the association status is **Complete** or **Updating**, choose the **Edit priority** (pencil) icon in the **Priority** column.
6. Enter a new **Priority** value from 100 to 9900, and then choose **Submit edit**.

CLI

To view the Profiles that are associated to a VPC and the priority of each association, use the list-profile-associations AWS CLI command. For example, the following command lists the Profiles that are associated to VPC `vpc-0af3b96b3example`.

```
aws route53profiles list-profile-associations --resource-id
vpc-0af3b96b3example
```

The following is an example output after you run the command:

```
{
```

```
    "ProfileAssociations": [
        {
            "CreationTime": 1709338526.311,
            "Id": "rpassoc-7c1a5e93dexample",
            "ModificationTime": 1709338702.645,
            "Name": "shared-services-association",
            "OwnerId": "123456789012",
            "ProfileId": "rp-6ffe47d5example",
            "ResourceId": "vpc-0af3b96b3example",
            "Priority": 100,
            "Status": "COMPLETE",
            "StatusMessage": "Created Profile Association"
        },
        {
            "CreationTime": 1709338817.148,
            "Id": "rpassoc-489ce212fexample",
            "ModificationTime": 1709338974.772,
            "Name": "test-association",
            "OwnerId": "123456789012",
            "ProfileId": "rp-4987774726example",
            "ResourceId": "vpc-0af3b96b3example",
            "Priority": 200,
            "Status": "COMPLETE",
            "StatusMessage": "Created Profile Association"
        }
    ]
}
```


To update the priority of a Profile association with a VPC, use the update-profile-association AWS CLI command. For example, the following command changes the priority of association `rpassoc-489ce212fexample` to 150.

```
aws route53profiles update-profile-association --profile-association-id
rpassoc-489ce212fexample --priority 150
```

The following is an example output after you run the command:


```
{
    "ProfileAssociation": {
        "CreationTime": 1709338817.148,
        "Id": "rpassoc-489ce212fexample",
        "ModificationTime": 1709339188.402,
        "Name": "test-association",
```

```
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceId": "vpc-0af3b96b3example",
        "Priority": 150,
        "Status": "UPDATING",
        "StatusMessage": "Updating Profile Association"
    }
}
```

Working with shared Route 53 Profiles

You can share a Profile with other accounts by:

- Granting read-only permissions, which means the other account can associate the Profile to their VPCs. In this case all the DNS resources and configurations will be in effect on the associated VPCs.
- Granting admin permissions. In this case the accounts with the shared Profile can modify the Profile and then associate it with their VPCs. An owner can also create customer managed permissions that can be used to specify which actions can be performed by the consumer account. For more information, see Customer managed permissions in the *AWS RAM User Guide*.

Amazon Route 53 Profile integrates with AWS Resource Access Manager (AWS RAM) to enable resource sharing. AWS RAM is a service that enables you to share some Route 53 resources with other AWS accounts or through AWS Organizations. With AWS RAM, you share resources that you own by creating a *resource share*. A resource share specifies the resources to share, and the consumers with whom to share them. Consumers can include:

- Specific AWS accounts
- An organizational unit inside its organization in AWS Organizations
- Its entire organization in AWS Organizations

For more information about AWS RAM, see the *AWS RAM User Guide*.

This topic explains how to share resources that you own, and how to use resources that are shared with you.

Contents

- Granting permissions for sharing Route 53 Profiles
- Prerequisites for sharing Route 53 Profiles
- Sharing a Route 53 Profile
- Unsharing a shared Route 53 Profile
- Identifying a shared Route 53 Profile
- Responsibilities and permissions for shared Route 53 Profiles
- Billing and metering
- Instance quotas

Granting permissions for sharing Route 53 Profiles

A minimum set of permissions is required for an IAM principal to share a Profile. We recommend using the `AmazonRoute53ProfilesFullAccess` managed IAM policy to make sure your IAM principals have the required permissions to share and use shared Profiles.

If you use a custom IAM policy, the `route53profiles:GetProfilePolicy` and `route53profiles:PutProfilePolicy` actions are required. These are permission-only IAM actions. If an IAM principal doesn't have these permissions granted, an error will occur when attempting to share the Profile using the AWS RAM service.

Prerequisites for sharing Route 53 Profiles

- To share a Route 53 Profile, you must own it in your AWS account. This means that the resource must be allocated or provisioned in your account. You cannot share a Route 53 Profile that has been shared with you.
- To share a Route 53 Profile with your organization or an organizational unit in AWS Organizations, you must enable sharing with AWS Organizations. For more information, see Enable Sharing with AWS Organizations in the *AWS RAM User Guide*.

Sharing a Route 53 Profile

When you share a Profile that you own with another AWS account, you enable them to apply the DNS-related settings of the Profile to their VPCs. This makes it easier to apply uniform DNS configurations across thousands of VPCs with minimal management overhead.

To share a Route 53 Profile, you must add it to a resource share. A resource share is an AWS RAM resource that lets you share your resources across AWS accounts. A resource share specifies the resources to share, and the consumers with whom they are shared. When you share a Route 53 Profile using the Route 53 console, you add it to an existing resource share. To add the Route 53 Profile to a new resource share, you must first create the resource share using the AWS RAM console.

If you are part of an organization in AWS Organizations and sharing within your organization is enabled, consumers in your organization are automatically granted access to the shared Route 53 Profile. Otherwise, consumers receive an invitation to join the resource share and are granted access to the shared Route 53 Profile after accepting the invitation.

You can get started sharing a Route 53 Profile that you own on the Route 53 console and continue on the AWS RAM console.

To share a Route 53 Profile that you own using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. Select the Profile you want to share, and on the **Profile details** page, choose **Share profile**.
4. You're taken to the AWS RAM console where you can follow these steps: Creating a Resource Share in the *AWS RAM User Guide*.
5. If a Profile is shared to you, the **Profiles** table includes the text **Shared with me**.

   When you have shared a Profile, it is listed as **Shared** in the **Profiles** table.

To share a Route 53 Profile that you own using the AWS RAM console

See Creating a Resource Share in the *AWS RAM User Guide*.

To share a Route 53 Profile that you own using the AWS CLI

Use the create-resource-share command.

Unsharing a shared Route 53 Profile

When you unshare a Profile, and VPCs that have that Profile's configurations associated to them, will lose them, and default to the VPC-specific configurations.

To unshare a shared Route 53 Profile that you own, you must remove it from the resource share. You can do this using the Route 53 console, AWS RAM console, or the AWS CLI.

To unshare a shared Route 53 Profile that you own using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. Select the linked name of the Profile you want to unshare, and on the **<Profile name>** page, choose **Manage sharing**.
4. You're taken to the AWS RAM console where you can follow these steps: Updating a Resource Share in the *AWS RAM User Guide*.

To unshare a shared Route 53 Profile that you own using the AWS RAM console

See Updating a Resource Share in the *AWS RAM User Guide*.

To unshare a shared Route 53 Profile that you own using the AWS CLI

Use the disassociate-resource-share command.

Identifying a shared Route 53 Profile

Owners and consumers can identify shared Route 53 Profiles using the Route 53 console and AWS CLI.

To identify a shared Route 53 Profile using the Route 53 console

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Profiles**.
3. If a Profile is shared to you, the **Profiles** table includes the text **Shared with me**.

   When you have shared a Profile, it is listed as **Shared** in the **Profiles** table.

To identify a shared Route 53 Profile using the AWS CLI

Use the get-profile or the list-profile command. The commands returns information about the Route 53 Profiles that you own and the Route 53 Profiles sharing status.

Responsibilities and permissions for shared Route 53 Profiles

Permissions for owners

A Profile owner can view, manage, and delete Profile resource associations, including resource associations made by the consumer accounts. The owner is able to view and delete the VPC associations they own. Additionally, only a Profile owner can delete a Profile they own, and this also automatically removes all resource associations of the Profile.

> **Note**
>
> You must create a custom managed permission which includes the `route53profiles:AssociateResourceToProfile` action in addition to the default ones to associate any resources from the accounts the Profile is shared to, because the default policy `AWSRAMPermissionRoute53ProfileAllowAssociation` does not include it.

Permissions for consumers

Default permission for consumers of a shared Profile is read-only. With read-only permission they can see the associated resources and associate it to VPCs, but can't manage the resource associations.

When a consumer associates a shared Profile to one of its VPCs, the consumer account sets the priority of that association, and only the consumer account can change it. The Profile owner can't change the priority of the association. For more information, see Updating the priority of a Route 53 Profile association with a VPC.

An owner can also create customer managed permissions on the AWS RAM console. For more information, see Creating and using customer managed permissions in the *AWS RAM User Guide*.

Billing and metering

Route 53 Profiles are billed based on the number of VPC associations. The Profile owner is responsible for the bill for the VPC associations by the customer.

Each association between a Profile and a VPC counts separately. For example, a VPC that is associated to two Profiles counts as two associations.

Instance quotas

The Profile owners and consumers share the same quota, except for the number of Route 53 Profiles per account in a Region. For more information, see Quotas on Route 53 Profiles

What is Amazon Route 53 on Outposts?

AWS Outposts is a fully managed service that extends AWS infrastructure, services, APIs, and tools to customer premises. With AWS Outposts, you can run AWS services with on-premises workloads by using the same programming interfaces as in AWS Regions. For more information, see What is AWS Outposts? in the *AWS Outposts User Guide*.

Route 53 on Outposts offers two capabilities:

- A VPC Resolver that caches all DNS queries that originate from the AWS Outposts.
- Hybrid connectivity between an Outpost and an on-premises DNS resolver when you deploy inbound and outbound endpoints.

For more information, see What is Route 53 VPC Resolver?.

Additionally, Route 53 "VPC resolver" on Outposts reduces network latency by allowing queries to be resolved within the Outpost instead of making the round-trip to the nearest AWS Region.

> **Note**
>
> If you have a version of AWS Outposts racks that aren't compatible with Route 53 on Outposts, an AWS account team is notified and will contact you to help you upgrade AWS Outposts.

Architecture overview

Route 53 on Outposts implements a distributed DNS architecture:

- **DNS records and hosted zones** remain managed in Amazon Route 53 in the AWS Region
- **Resolver functionality** extends to your AWS Outposts rack for local query processing

This design optimizes query performance and availability while maintaining centralized DNS record management. DNS records continue to be stored in the AWS Region, not locally on the AWS Outposts rack.

Amazon Route 53 on Outposts features

The following table describes how Route 53 on Outposts features differ from Amazon Route 53 features.

Route 53 on Outposts compared to Route 53

Route 53 feature	Route 53 on first-generation AWS Outposts	Route 53 on second-generation AWS Outposts
VPC Resolver	Supported - VPC Resolver maintains a local cache of records for applications hosted on the Outpost rack, the peered VPC in the AWS Region, and any publicly accessible host names.	Supported - VPC Resolver maintains a local cache of records for applications hosted on the Outpost rack, the peered VPC in the AWS Region, and any publicly accessible host names.
Health checks	Not supported - Health checks are calculated and reported from the AWS Region. If an Outpost disconnects from the cloud, the endpoints fail open and can't fail over to a backup.	Not supported - Health checks are calculated and reported from the AWS Region. If an Outpost disconnects from the cloud, the endpoints fail open and can't fail over to a backup.
Resolver endpoints	Supported - Resolver endpoints on the Outpost rack allow DNS queries to be forwarded and received from DNS servers on-premises. Only the IPv4 endpoint type is available for endpoints.	Not supported.
Resolver DNS Firewall	Not supported.	Not supported.
Traffic flow	Not supported.	Not supported.

VPC Resolver behavior when AWS Outposts is disconnected from the VPC

If the AWS Outposts is disconnected from the AWS Region, the Resolver on Outpost behaves as follows:

- Control plane changes are not available.
- Health checks and DNS failover capability are not available.
- DNS queries for resources that are hosted locally on the Outposts are resolved but in some cases the response might be stale if the IP address for the resource was updated while the Outpost was in a disconnected state.
- DNS queries for resources hosted on the in-Region VPC are resolvable. However, the resources will not be accessible until the Outpost connection to the AWS Region is restored.
- DNS queries for public DNS resources can be resolved if they are available in the VPC Resolver cache on Outpost.

Getting started with VPC Resolver on AWS Outposts

After you have ordered your AWS Outposts racks and they have been delivered, as described here: Create an AWS Outposts in the *AWS Outposts guide*, you can set up Resolver on Outpost.

Second-generation AWS Outposts

For second-generation AWS Outposts, VPC Resolver is enabled by default, and you don't need to take any action to use it. To opt in to or opt out of VPC Resolver, contact AWS Support.

First-generation AWS Outposts

> **Important**
>
> Resolver on Outpost can only be created by the AWS account that owns the AWS Outposts rack. If the AWS Outposts rack is shared with other accounts, those accounts cannot create Resolver on Outpost on the shared rack.

You can also use APIs to manage Route 53 on Outposts. For more information, see Resolver on Outpost actions.

> **Important**
>
> It can take up to 30-150 minutes to create a VPC Resolver cache on an AWS Outposts.

After you have your AWS Outposts racks delivered, you can opt in to Route 53 on Outposts.

To configure Resolver on Outpost

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. On the **Resolver on Outpost** page, choose **Create VPC Resolver**.
5. On the **Create VPC Resolver** page:
   - Under **AWS Outposts** select an AWS Outposts you want to create the VPC Resolver on.
   - Type in a name for the VPC Resolver in the **VPC Resolver name** text box.
   - After the **Recommended instance types for VPC Resolver** populates with Amazon EC2 instances, choose one.

     For more information about the instance types, see Quotas for Resolver on first-generation AWS Outposts.
   - For **Number of instances**, choose the number of elastic interface instances for the VPC VPC Resolver. The default value is 4.

     If your AWS Outposts doesn't have an instance type that supports VPC Resolver, you won't be able to create a VPC Resolver.
6. Choose **Create VPC Resolver**.

   You can monitor the VPC Resolver creation on the **Resolver on Outpost** page.

Creating inbound endpoints (first-generation AWS Outposts only)

> **Note**
>
> This content applies to first-generation AWS Outposts only. It doesn't apply to second-generation AWS Outposts.

After you have created a Resolver on Outpost, you can add both inbound and outbound endpoints to resolve DNS queries to and from your on-premises network.

To configure inbound endpoints for Resolver on Outpost

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the check box next to the VPC Resolver that is in operational state and choose **View details**.
5. On the **Inbound endpoints** table, choose **Create inbound endpoint**.
6. On the **Create inbound endpoint** page, enter the applicable values. For more information, see Values that you specify when you create or edit inbound endpoints on an Outpost.
7. Choose **Create endpoint**.

Values that you specify when you create or edit inbound endpoints on an Outpost

When you create or edit an inbound endpoint, you specify the following values:

Outpost ID

If you are creating the endpoint for a VPC Resolver on an AWS Outposts VPC, this is the AWS Outposts ID.

Endpoint name

A friendly name that lets you easily find an inbound endpoint on the dashboard.

VPC in the *region-name* Region

All inbound DNS queries from your network pass through this VPC on the way to VPC Resolver.

Security group for this endpoint

The ID of one or more security groups that you want to use to control access to this inbound endpoint. The security group that you specify must include one or more inbound rules. Inbound rules must allow TCP and UDP access on port 53. You can't change this value after you create the endpoint.

For more information, see Security groups for your VPC in the *Amazon VPC User Guide*.

IP addresses

The IP addresses that you want to assign to the inbound endpoints. We require you to specify a minimum of two IP addresses for redundancy. Note the following:

IP addresses and Amazon VPC elastic network interfaces

For each combination of Availability Zone, Subnet, and IP address that you specify, VPC Resolver creates an Amazon VPC elastic network interface. For the current maximum number of DNS queries per second per IP address in an endpoint, see Quotas on Route 53 VPC Resolver. For information about pricing for each elastic network interface, see Amazon Route 53 on the Amazon Route 53 pricing page.

> **Note**
>
> Resolver endpoint has a private IP address. These IP addresses will not change through the course of an endpoint's life.

For each IP address, specify the following values. Each IP address must be in an Availability Zone in the VPC that you specified in **VPC in the *region-name* Region**.

Availability Zone

The Availability Zone that you want DNS queries to pass through on the way to your VPC. The Availability Zone that you specify must be configured with a subnet.

Subnet

The subnet that contains the IP address that you want to forward DNS queries to. The subnet must have an available IP address.

Specify the subnet for an IPv4 address. IPv6 is not supported.

IP address

The IP address that you want to forward DNS queries to.

Choose whether you want VPC Resolver to choose an IP address for you from among the available IP addresses in the specified subnet, or you want to specify the IP address yourself.

If you choose to specify the IP address yourself, enter an IPv4 address. IPv6 is not supported.

Tags

Specify one or more keys and the corresponding values. For example, you might specify **Cost center** for **Key** and specify **456** for **Value**.

These are the tags that AWS Billing and Cost Management provides for organizing your AWS bill; you can use tags for other purposes as well. For more information about using tags for cost allocation, see Using cost allocation tags in the *AWS Billing User Guide*.

Creating outbound endpoints (first-generation AWS Outposts only)

> **Note**
>
> This content applies to first-generation AWS Outposts only. It doesn't apply to second-generation AWS Outposts.

After you have opted in and configured a VPC Resolver, you can also add both inbound and outbound endpoints to resolve DNS queries to your on-premises network.

> **Note**
>
> When you configure outbound endpoints, VPC Resolver caches DNS responses so that queries can still be resolved if your Outpost becomes disconnected from the Region. Maintaining this cache might increase DNS requests to your on-premises resolvers.

To configure outbound endpoints for Resolver on Outpost

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the checkmark next to the VPC Resolver that is in operational state and choose **View details**.
5. On the **Outbound endpoints** table, choose **Create outbound endpoint**.
6. On the **Create outbound endpoint** page, enter the applicable values. For more information, see Values that you specify when you create or edit outbound endpoints in an AWS Outposts.
7. Choose **Create endpoint**.

Values that you specify when you create or edit outbound endpoints in an AWS Outposts

When you create or edit an outbound endpoint, you specify the following values:

Outpost ID

If you are creating the endpoint for a VPC Resolver on an AWS Outposts VPC, this is the AWS Outposts ID.

Endpoint name

A friendly name that lets you easily find an outbound endpoint on the dashboard.

VPC in the *region-name* Region

All outbound DNS queries from your VPC pass through this VPC on the way to your network.

Security group for this endpoint

The ID of one or more security groups that you want to use to control access to this VPC. The security group that you specify must include one or more outbound rules. Outbound rules must allow TCP and UDP access on the port that you're using for DNS queries on your network. You can't change this value after you create the endpoint.

For more information, see Security groups for your VPC in the *Amazon VPC User Guide*.

IP addresses

The IP addresses that you want to assign to the outbound endpoints. We require you to specify a minimum of two IP addresses for redundancy. Note the following:

IP addresses and Amazon VPC elastic network interfaces

For each combination of Availability Zone, Subnet, and IP address that you specify, VPC Resolver creates an Amazon VPC elastic network interface. For the current maximum number of DNS queries per second per IP address in an endpoint, see Quotas on Route 53 VPC Resolver. For information about pricing for each elastic network interface, see "Amazon Route 53" on the Amazon Route 53 pricing page.

> **Note**
>
> Resolver endpoint has a private IP address. These IP addresses will not change through the course of an endpoint's life.

For each IP address, specify the following values. Each IP address must be in an Availability Zone in the VPC that you specified in **VPC in the *region-name* Region**.

Availability Zone

The Availability Zone that you want DNS queries to pass through on the way from your VPC. The Availability Zone that you specify must be configured with a subnet.

Subnet

The subnet that contains the IP address that you want to forward DNS queries from. The subnet must have an available IP address.

Specify the subnet for an IPv4 address. IPv6 is not supported.

IP address

The IP address that you want to assign to the outbound endpoints.

Choose whether you want VPC Resolver to choose an IP address for you from among the available IP addresses in the specified subnet, or you want to specify the IP address yourself.

If you choose to specify the IP address yourself, enter an IPv4 address. IPv6 is not supported.

Tags

Specify one or more keys and the corresponding values. For example, you might specify **Cost center** for **Key** and specify **456** for **Value**.

These are the tags that AWS Billing and Cost Management provides for organizing your AWS bill; you can use also tags for other purposes. For more information about using tags for cost allocation, see Using cost allocation tags in the *AWS Billing User Guide*.

Creating forwarding rules for outbound endpoints

You can also create forwarding rules for outbound endpoints. For more information, see To create forwarding rules and associate the rules with one or more VPCs

Managing Resolver on Outpost

How you manage Resolver on Outpost depends on your AWS Outposts generation:

- For first-generation AWS Outposts, you can edit, view, and delete Resolver.
- For second-generation AWS Outposts, you can't edit or delete Resolver. You can only view it.
- If you don't want to use Resolver with second-generation AWS Outposts, you must contact AWS Support to opt out.
- If Resolver fails on second-generation AWS Outposts, AWS contacts you to help resolve the issue.

Topics

- Editing Resolver on Outpost (first-generation AWS Outposts only)
- Viewing Resolver on Outpost status

- Confirming that Resolver on Outpost is working
- Deleting Resolver on Outpost (first-generation AWS Outposts only)

Editing Resolver on Outpost (first-generation AWS Outposts only)

To edit a Resolver on Outpost, perform the following procedure.

To edit a Resolver on Outpost

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the checkmark next to the VPC Resolver that is in operational state and choose **Edit**.
5. You can edit the following information:
   - The VPC Resolver name
   - The instance type
   - The number of instances
6. After you are done editing, choose **Save changes**.

Viewing Resolver on Outpost status

For second-generation AWS Outposts, viewing the resolver status is the primary way to confirm that Resolver on Outpost is present and healthy. You can't edit or delete it. AWS monitors Resolver on Outpost on second-generation AWS Outposts and contacts you if it fails.

To view the status for Resolver on Outpost, perform the following procedure.

To view the status of your Resolver on Outpost

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.

4. Select the checkmark next to the VPC Resolver that is in operational state and choose **View details**.
5. The **Status** column in the **Resolver on Outpost** page, contains one of the following values:

   Creating

   The Resolver on Outpost is in the process of being created.

   Operational

   The Resolver on Outpost is running and can answer DNS queries for workloads in the Outpost's associated VPCs. This is the healthy, working state.

   Updating

   The Resolver on Outpost is updating instance types.

   Action needed

   This VPC Resolver is unhealthy and can't be automatically recovered. To resolve the problem, we recommend that you make sure the instance AWS Outposts can support Resolver on Outpost. In this state, the resolver isn't serving DNS queries until you resolve the problem.

   Deleting

   The Resolver on Outpost is in the process of being deleted.

   Failed creation

   The creation of Resolver on Outpost failed.

   Failed deletion

   The deletion of Resolver on Outpost failed. To fix this issue, try again in a few minutes.

Confirming that Resolver on Outpost is working

Resolver on Outpost works transparently for your workloads. Instances in your AWS Outposts subnets continue to send DNS queries to the same VPC+2 address for your VPC. This is the same Route 53 Resolver address that they used before. You don't need to change any client configuration.

You can confirm that Resolver on Outpost is running and healthy in the following ways:

- **Console** — Confirm that the resolver **Status** is **Operational**. For more information, see the section called "Viewing Resolver on Outpost status".
- **AWS CLI** — List the resolvers on your AWS Outposts and confirm that the status is `OPERATIONAL`.

```
aws route53resolver list-outpost-resolvers --outpost-arn outpost-arn
```

  To view a specific resolver, use the `get-outpost-resolver` command with the resolver ID.

- **Resolution performance** — Because Resolver on Outpost caches responses locally on your AWS Outposts, you can also observe that it's working from resolution times. While Resolver on Outpost is enabled, queries for names that are already cached on the AWS Outposts resolve nearly immediately (typically 0–10 ms). The first query for a name that isn't cached takes the usual amount of time. This query traverses the AWS Outposts service link to the VPC Resolver instances in the parent AWS Availability Zone. The AWS Outposts cache then serves subsequent queries for the same name, which resolve nearly immediately.

Deleting Resolver on Outpost (first-generation AWS Outposts only)

> **Note**
>
> Before you can delete a Resolver on Outpost, you must first delete any endpoints associated with it.

To delete a Resolver on Outpost, perform the following procedure.

To delete a Resolver on Outpost

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the check box next to the VPC Resolver that is in operational state and choose **Delete**.
5. In the **Delete VPC Resolver** dialog box, enter **`delete`** in the text box, and choose **Delete**.

Managing inbound endpoints on Resolver on Outpost (first-generation AWS Outposts only)

> **Note**
>
> This content applies to first-generation AWS Outposts only. It doesn't apply to second-generation AWS Outposts.

To manage inbound endpoints on Resolver on Outpost, perform the applicable procedure.

Topics

- Viewing and editing inbound endpoints
- Viewing the status for inbound endpoints
- Deleting inbound endpoints

Viewing and editing inbound endpoints

To view and edit settings for an inbound endpoint, perform the following procedure.

To view and edit settings for an inbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the check box next to the VPC Resolver that is in operational state and choose **View details**.
5. In the **Inbound endpoints** list, choose the option for the endpoint that you want to view settings for or want to edit.
6. Choose **View details** or **Edit**.

   For information about the values for inbound endpoints, see Values that you specify when you create or edit outbound endpoints in an AWS Outposts.
7. If you chose **Edit**, enter the applicable values, and choose **Save**.

Viewing the status for inbound endpoints

To view the status for an inbound endpoint, perform the following procedure.

To view the status for an inbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the check box next to the VPC Resolver that is in operation state and choose **View details**.
5. The **Status** column of the **inbound endpoints** list contains one of the following values:

   Creating

   VPC Resolver is creating and configuring one or more Amazon VPC network interfaces for this endpoint.

   Operational

   The Amazon VPC network interfaces for this endpoint are correctly configured and able to pass inbound or outbound DNS queries between your network and VPC Resolver.

   Updating

   Resolver is associating or disassociating one or more network interfaces with this endpoint.

   Auto recovering

   VPC Resolver is trying to recover one or more of the network interfaces that are associated with this endpoint. During the recovery process, the endpoint functions with limited capacity because of the limit on the number of DNS queries per IP address (per network interface). For the current limit, see Quotas on Route 53 VPC Resolver.

   Action needed

   This endpoint is unhealthy, and VPC Resolver can't automatically recover it. To resolve the problem, we recommend that you check each IP address that you associated with the endpoint. For each IP address that isn't available, add another IP address and then delete the IP address that isn't available. An endpoint must always include at least two IP

addresses. A status of **Action needed** can have a variety of causes. Here are two common causes:

- One or more of the network interfaces that are associated with the endpoint were deleted using Amazon VPC.
- The network interface couldn't be created for some reason that's outside the control of VPC Resolver.

Deleting

Resolver is deleting this endpoint and the associated network interfaces.

Deleting inbound endpoints

To delete an inbound endpoint, perform the following procedure.

> **Important**
>
> If you delete an inbound endpoint, DNS queries from your network are no longer forwarded to VPC Resolver in the VPC that you specified in the endpoint.

To delete an inbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the check box next to the VPC Resolver that is in operation state and choose **View details**.
5. Choose the check box next to the endpoint that you want to delete.
6. Choose **Delete**.
7. To confirm that you want to delete the endpoint, enter the name of the endpoint and choose **Submit**.

Managing outbound endpoints on Resolver on Outpost (first-generation AWS Outposts only)

> **Note**
>
> This content applies to first-generation AWS Outposts only. It doesn't apply to second-generation AWS Outposts.

To manage outbound endpoints on Resolver on Outpost, perform the applicable procedure.

Topics

- Viewing and editing outbound endpoints
- Viewing the status for outbound endpoints
- Deleting outbound endpoints

Viewing and editing outbound endpoints

To view and edit settings for an outbound endpoint, perform the following procedure.

To view and edit settings for an outbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the check box next to the VPC Resolver that is in operation state and choose **View details**.
5. In the **Outbound endpoints** list, choose the check box next to the endpoint that you want to view settings for or want to edit.
6. Choose **View details** or **Edit**.

   For information about the values for outbound endpoints, see Values that you specify when you create or edit outbound endpoints in an AWS Outposts.
7. If you chose **Edit**, enter the applicable values, and then choose **Save**.

Viewing the status for outbound endpoints

To view the status for an outbound endpoint, perform the following procedure.

To view the status for an outbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. On the navigation bar, choose the Region where your AWS Outposts is located.
4. Select the check box next to the VPC Resolver that is in operation state and choose **View details**.
5. In the **Outbound endpoints** list, the **Status** column contains one of the following values:

   Creating

   VPC Resolver is creating and configuring one or more Amazon VPC network interfaces for this endpoint.

   Operational

   The Amazon VPC network interfaces for this endpoint are correctly configured and able to pass inbound or outbound DNS queries between your network and VPC Resolver.

   Updating

   Resolver is associating or disassociating one or more network interfaces with this endpoint.

   Auto recovering

   VPC Resolver is trying to recover one or more of the network interfaces that are associated with this endpoint. During the recovery process, the endpoint functions with limited capacity because of the limit on the number of DNS queries per IP address (per network interface). For the current limit, see Quotas on Route 53 VPC Resolver.

   Action needed

   This endpoint is unhealthy, and VPC Resolver can't automatically recover it. To resolve the problem, we recommend that you check each IP address that you associated with the endpoint. For each IP address that isn't available, add another IP address and then delete the IP address that isn't available. (An endpoint must always include at least two IP

addresses.) A status of **Action needed** can have a variety of causes. Here are two common causes:

- One or more of the network interfaces that are associated with the endpoint were deleted using Amazon VPC.
- The network interface couldn't be created for some reason that's outside the control of VPC Resolver.

Deleting

Resolver is deleting this endpoint and the associated network interfaces.

Deleting outbound endpoints

Before you can delete an endpoint, you must first delete any rules that are associated to a VPC.

To delete an outbound endpoint, perform the following procedure.

> **Important**
>
> If you delete an outbound endpoint, VPC Resolver stops forwarding DNS queries from your VPC to your network for rules that specify the deleted outbound endpoint.

To delete an outbound endpoint

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the left navigation pane, expand **Resolver**, and then navigate to **Outposts**.
3. Select the check box next to the VPC Resolver that is in operation state and choose **View details**.
4. In the **Outbound endpoints** list choose the option for the endpoint that you want to delete.
5. Choose **Delete**.
6. To confirm that you want to delete the endpoint, enter the name of the endpoint, and then choose **Submit**.

What is Route 53 Global Resolver?

Route 53 Global Resolver is a DNS resolver reachable over the internet and over your private connections (managed through AWS Cloud WAN) that provides easy, secure, and reliable DNS resolution for authorized clients in your organization across remote locations, branch offices, and on-premises environments. You can resolve queries for domains hosted on Amazon Route 53 private hosted zones or public domains on the internet, while maintaining high availability through global anycast architecture. Route 53 Global Resolver helps you protect your clients from DNS-based data exfiltration attacks, by filtering queries to potentially malicious domains.

Route 53 Global Resolver uses anycast IP addresses that automatically route DNS queries to the AWS Region closest to your source query location for optimal latency and availability. You can select the Regions where your global resolver should provide DNS resolution from for your authorized clients, configure different views for the clients for the resolution of private and public domains, filter malicious domains, and monitor DNS activity across your organization.

Topics

- Why Route 53 Global Resolver?
- Key concepts and components for Route 53 Global Resolver
- How Route 53 Global Resolver works
- DNS security and split-horizon use cases for Route 53 Global Resolver
- AWS services that integrate with Route 53 Global Resolver for enhanced DNS security
- How to access and manage Route 53 Global Resolver through AWS interfaces
- Supported AWS Regions
- Opt-in Region considerations for Route 53 Global Resolver
- Setting up account access for Route 53 Global Resolver
- Tutorial: Create your first Route 53 Global Resolver
- Tutorial: Create your first Route 53 Global Resolver with AWS Cloud WAN
- Managing DNS views with Route 53 Global Resolver
- Managing access controls with access sources and tokens in Route 53 Global Resolver
- Securing DNS for clients with Route 53 Global Resolver
- Configuring private hosted zone associations with Route 53 Global Resolver
- Sharing Route 53 Global Resolver DNS views between AWS accounts

- Connecting a global resolver to a core network
- Monitoring DNS activity and performance with Route 53 Global Resolver
- Troubleshooting DNS issues with Route 53 Global Resolver
- Quotas for Route 53 Global Resolver

Why Route 53 Global Resolver?

Route 53 Global Resolver provides centralized DNS security and filtering for remote and hybrid clients and workloads across multiple locations. Regardless of the location of your workloads, applications, or users, Route 53 Global Resolver ensures consistent DNS protection without requiring complex on-premises infrastructure.

Key benefits for remote and hybrid environments:

- **Simplified management** - Configure private and public domain resolution using a single solution instead of managing multiple on-premises DNS servers
- **Unified DNS security** - Apply consistent filtering policies across all remote clients and hybrid workloads
- **Scalable protection** - Automatically scales to handle DNS queries from growing remote workforces and cloud workloads
- **Reduced infrastructure** - Minimizes the need for DNS security appliances at each remote location

To get started, see:

- the section called “Concepts and terminology” - Core concepts for DNS security deployment
- the section called “How it works” - How Route 53 Global Resolver protects remote and hybrid environments
- the section called “Tutorial: Create your first Route 53 Global Resolver” - Tutorial for your first DNS filtering setup

Key concepts and components for Route 53 Global Resolver

Route 53 Global Resolver uses several key components that work together to provide split-traffic DNS resolution, high availability through global anycast architecture, and comprehensive DNS

security for your organization. Understanding these Route 53 Global Resolver concepts helps you design and deploy solutions that enable seamless access to both private and public resources, help ensure service continuity across multiple Regions, and protect against DNS-based threats.

DNS resolver for clients at on-premises and remote locations

To deploy Route 53 Global Resolver for your distributed workloads, customer locations, and users, configure these key components:

Global resolver

The main service instance that provides DNS resolution and filtering for your organization across multiple AWS Regions. Your *global resolver* uses anycast technology to automatically route DNS queries to the nearest available Region, ensuring fast response times for all clients regardless of their location.

Anycast IP addresses

Two unique IPv4 or IPv6 addresses assigned to your global resolver that you configure on client devices and network equipment. These *anycast IP addresses* are the same globally, which simplifies DNS configuration across all your locations. Anycast IP addressing enables automatic routing of DNS requests to the nearest global resolver, optimizing response times and improving service reliability.

DNS views

Configuration templates that let you apply different DNS policies to different groups of clients in your network. Use *DNS views* to implement split-horizon DNS—for example, apply strict filtering and token authentication for remote locations, while using IP-based access and different security policies for branch offices.

Core network endpoint

A *core network endpoint* connects a global resolver to an AWS Cloud WAN (Cloud WAN) core network. After you connect the core network, you can use the core network's segments as access sources for DNS resolution. A global resolver can have one or more core network endpoints.

DNS client authentication

Select the authentication method that works best for your deployment:

Token based authentication

Secure DNS connections using encrypted tokens for DNS-over-HTTPS (DoH) and DNS-over-TLS (DoT). You can generate unique access tokens for individual clients or device groups, set expiration periods, and revoke tokens as needed.

Access source-based authentication

Control access using IP address and CIDR range allowlists. You can configure your branch office public IP addresses or network ranges, then specify which DNS protocols (DNS-over-port-53, DoT, or DoH) each location can use based on your security requirements.

DNS protocol selection

Choose the appropriate *DNS protocol* based on your security and compatibility needs:

- **DNS-over-port-53 (Do53)** - Use for maximum compatibility with existing network infrastructure
- **DNS-over-TLS (DoT)** - Use when you need encrypted DNS with dedicated port separation for network monitoring
- **DNS-over-HTTPS (DoH)** - Use when you need to bypass network restrictions, as traffic appears as regular HTTPS

Split-traffic DNS resolution

Route 53 Global Resolver enables organizations to seamlessly resolve both private and public domains from any location, eliminating the need for complex VPN configurations or Region-specific DNS settings.

Hybrid DNS resolution

Hybrid DNS resolution allows Route 53 Global Resolver to simultaneously resolve queries from on-premises users and applications to private applications on AWS.

Global private zone access

Global private zone access extends the reach of Amazon Route 53 private hosted zones beyond VPC boundaries. Authorized clients anywhere on the internet can resolve private domain names, enabling distributed teams to access internal resources without traditional network connectivity requirements.

Seamless failover

Seamless failover ensures continuous access to both private and public resources even when individual AWS Regions become unavailable. The anycast architecture automatically routes queries to healthy regions while maintaining consistent resolution behavior.

Core network traffic resolution

Core network traffic resolution extends split-traffic DNS resolution to clients that reach the global resolver through an AWS Cloud WAN core network. In addition to resolving queries from clients over the internet, Route 53 Global Resolver resolves DNS queries for clients that connect through a core network. Traffic from a connected core network's segments receives the same split-traffic resolution of both public and private domains.

High availability and global presence

Route 53 Global Resolver provides enterprise-grade availability through distributed architecture and automatic failover capabilities.

Multi-region deployment

Multi-region deployment distributes Route 53 Global Resolver instances across at least 2 AWS Regions to help ensure high availability and allow failover during service outages. You can select specific Regions based on your geographic requirements and compliance needs.

Automatic geographic optimization

Automatic geographic optimization routes DNS queries to the nearest available AWS Region based on network topology and latency. This reduces response times and improves user experience for globally distributed organizations.

Built-in redundancy

Built-in redundancy helps maintain service continuity through automatic failover to alternate regions when primary regions become unavailable. Clients continue to use the same anycast IP addresses while traffic is transparently rerouted.

DNS resolution and forwarding

Private hosted zone resolution

Private hosted zone resolution enables Route 53 Global Resolver to resolve DNS queries for Route 53 private hosted zones across AWS Regions. This allows authorized clients to resolve domains for applications and resources hosted by Route 53 from anywhere on the internet.

Split-horizon DNS

Split-horizon DNS provides different DNS responses based on the client making the query. Route 53 Global Resolver can resolve public domains on the internet while simultaneously resolving private domains, providing seamless access to both public and private resources.

DNSSEC validation

DNSSEC validation verifies the authenticity and integrity of DNS responses from public nameservers for DNSSEC-signed domains. This validation ensures DNS responses haven't been tampered with during transmission, providing protection against DNS spoofing and cache poisoning attacks.

EDNS Client Subnet (ECS)

EDNS Client Subnet is an optional feature that forwards client subnet information in DNS queries to authoritative nameservers. This enables more accurate geographic-based DNS responses, potentially reducing latency by directing clients to nearer content delivery networks or servers. For DNS-over-TLS (DoT) and DNS-over-HTTPS (DoH) connections, you can use EDNS0 to pass the client IP address information. When ECS is enabled on Global Resolver, the service automatically injects the client IP if not provided in the query.

DNS filtering and domain lists

Route 53 Global Resolver provides domain-based filtering using domain lists managed by AWS to block or allow specific domains.

DNS filtering rules

DNS filtering rules define how Route 53 Global Resolver handles DNS queries based on domain matching criteria. Rules are evaluated in priority order and can specify actions (ALLOW, BLOCK, or ALERT) for queries to specific domains or domain categories.

Domain lists

Domain lists are collections of domains used in filtering rules. They can be:

- **Custom domain lists** - Domain collections you create and maintain
- **AWS managed domain lists** - Pre-configured threat lists and content categories maintained by AWS that leverage threat intelligence to identify malicious domains. Available threat lists include:
  - Malware domains - Domains known to host or distribute malware
  - Botnet command and control - Domains used by botnets for command and control communications
  - Spam - Domains associated with spam and unwanted email campaigns
  - Phishing - Domains used in phishing attacks to steal credentials and personal information
  - Amazon GuardDuty threat list - Domains identified by GuardDuty threat intelligence

  Available content categories include social media, gambling, and other categories that help organizations control access to specific types of content.

  Individual domain specifications in managed lists cannot be viewed or edited to protect intellectual property and maintain security effectiveness.

Advanced DNS threat detection

Route 53 Global Resolver uses dynamic algorithmic analysis to detect advanced DNS threats such as DNS tunneling and Domain Generation Algorithms. Unlike domain lists that match known bad domains, algorithmic detection analyzes DNS query patterns in real-time to identify suspicious behavior.

DNS tunneling detection

DNS tunneling is used by attackers to exfiltrate data from the client by using the DNS tunnel without making a network connection to the client.

Domain Generation Algorithm (DGA) detection

Domain Generation Algorithms (DGAs) are used by attackers to create large numbers of domain names for its command-and-control servers.

Confidence thresholds

Each detection algorithm outputs a confidence score that determines rule triggering. Higher confidence thresholds reduce false positives but might miss sophisticated attacks. Lower thresholds increase detection sensitivity but require additional alert analysis to filter false positives.

Action limitations

Advanced threat protection rules support only `ALERT` and `BLOCK` actions. The `ALLOW` action is not supported because algorithmic detection cannot definitively classify benign traffic, only identify potentially malicious patterns.

Monitoring and logging

Query logs

Query logs provide detailed information about DNS queries processed by Route 53 Global Resolver, including source IP, queried domain, response code, policy actions taken, and timestamps. Logs can be delivered to Amazon CloudWatch, Amazon Data Firehose, or Amazon Simple Storage Service for analysis and compliance reporting.

OCSF format

Open Cybersecurity Schema Framework (OCSF) format is a standardized logging format used by Route 53 Global Resolver for DNS query logs. This format provides consistent, structured data that integrates easily with security information and event management (SIEM) systems and other security tools.

Log destinations

Log destinations determine where DNS query logs are delivered, each with different characteristics:

- **Amazon Simple Storage Service** - Cost-effective long-term storage ideal for compliance and batch analysis. Integrates with analytics tools like Amazon Athena and Amazon EMR.
- **Amazon CloudWatch Logs** - Real-time monitoring and alerting with integration to Amazon CloudWatch alarms and dashboards. Supports log insights for ad-hoc queries.
- **Amazon Data Firehose** - Real-time streaming to external systems with built-in data transformation capabilities. Supports automatic scaling and buffering.

Observability Region

The *Observability Region* determines where DNS query logs are delivered.

How Route 53 Global Resolver works

Route 53 Global Resolver enables split-traffic DNS resolution between public and private domains, provides high availability through through the two or more AWS Regions you choose, and secures DNS queries by intercepting requests and applying DNS filtering policies. Understanding this process helps you troubleshoot issues and optimize your deployment for performance, availability, and security.

What happens when clients make DNS queries

When someone at your location tries to query domain, Route 53 Global Resolver processes their DNS request through multiple security layers.

The DNS query processing involves these sequential steps:

1. **Query reception** - Client devices send DNS queries to Route 53 Global Resolver anycast IP addresses. The anycast routing automatically directs queries to the nearest AWS Region.
2. **Authentication** - Route 53 Global Resolver authenticates the client using configured authentication methods (token-based for DoH/DoT or IP Access Source for all protocols).
3. **Policy evaluation** - The service evaluates DNS queries against configured security policies and domain lists to determine the appropriate action (allow, block, or alert). For queries targeting private hosted zones, Route 53 Global Resolver checks if the client is authorized to access the private domain based on the DNS view rule managed by your administrator before proceeding with resolution.
4. **Resolution** - For allowed queries, Route 53 Global Resolver performs DNS resolution using public DNS resolvers or private hosted zone resolution as appropriate.
5. **Response delivery** - The service returns the DNS response to the client and logs the query details for monitoring and analysis.

Global anycast architecture

Route 53 Global Resolver uses anycast IP addresses to provide global availability and automatic geographic routing.

Route 53 Global Resolver uses anycast IP addresses to provide:

- **Automatic geographic routing** - DNS queries are automatically routed to the closest AWS Region for optimal performance.
- **Built-in redundancy** - If a Region becomes unavailable, traffic automatically fails over to the next closest Region.
- **Consistent IP addresses** - Clients use the same anycast IP addresses regardless of their location, simplifying configuration.

Private anycast architecture with Cloud WAN

When clients reach the global resolver over private connections through an AWS Cloud WAN core network, Route 53 Global Resolver uses the same anycast-based geographic routing to direct queries to the closest AWS Region.

Private IP addresses are resolved with the same routing and availability benefits as public internet traffic, over private Cloud WAN connectivity instead of the internet:

- **Automatic geographic routing** - DNS queries from a connected core network are automatically routed to the closest AWS Region for optimal performance.
- **Built-in redundancy** - If a Region becomes unavailable, traffic automatically fails over to the next closest Region.
- **Consistent endpoints** - Clients use the same anycast IP addresses regardless of their location, simplifying configuration.

DNS filtering and security

Route 53 Global Resolver provides comprehensive DNS filtering and security through multiple layers. The DNS filtering and security architecture diagram illustrates how queries are processed through authentication, policy evaluation, and resolution layers.

Route 53 Global Resolver provides comprehensive DNS security through:

- **Domain-based filtering** - Block or allow queries based on domain names using custom or AWS managed domain lists.
- **Threat intelligence integration** - Leverage AWS managed threat intelligence to automatically block known malicious domains.

- **Advanced threat detection** - Detect and block DNS tunneling attempts and Domain Generation Algorithm (DGA) patterns.
- **Real-time monitoring** - Generate alerts and logs for security events and policy violations.

DNS security and split-horizon use cases for Route 53 Global Resolver

Route 53 Global Resolver addresses three primary DNS challenges for organizations:

Enabling split-traffic between public and private DNS resolution

Enable global access to private hosted zones (PHZs) on Amazon Route 53 from any location while simultaneously resolving public domains on the internet. Allow remote locations and branch offices to resolve internal application names without complex VPN configurations or Region-specific forwarding. Implement split-horizon DNS to provide different DNS responses based on the client making the query, helping remote clients resolve queries for private and public domains.

Securing DNS traffic from DNS exfiltration attacks

Protect remote locations and branch offices from DNS-based data exfiltration attacks by filtering queries to malicious domains. Improve privacy by encrypting DNS traffic in-transit using DNS-over-HTTPS (DoH) and DNS-over-TLS (DoT) so that only authorized clients can access your DNS services. Apply security policies to block threats like DNS tunneling and Domain Generation Algorithms (DGAs). Validate DNS response authenticity using DNSSEC (Domain Name System Security Extensions) for DNSSEC-signed domains to protect against DNS spoofing and cache poisoning attacks.

High availability and global presence

Achieve high availability through global deployment and maintain consistent DNS configuration worldwide from a single management interface. Route 53 Global Resolver runs across the AWS Regions you choose, using anycast IP addresses that automatically route queries to the nearest available Region for optimal performance and reliability. Global enterprises can configure and manage DNS policies centrally while providing clients with a single set of IP addresses that work globally with automatic geographic optimization. Built-in redundancy is designed to help maintain service continuity even if individual Regions become unavailable.

Additional capabilities support these primary use cases:

Implementing DNS filtering and content policies

Govern internet access across multiple locations by creating custom domain lists or using AWS Managed Domain Lists. To help you implement filtering and content policies according to your needs, Managed Domain Lists include multiple categories of DNS threats that cover several domains. Configure Access Source using IP allowlists or Access Tokens, and set up different filtering policies for different office locations or client groups.

Flexible authentication for different deployment scenarios

Choose the authentication method that works best for your deployment: token-based authentication or IP based authentication using the source CIDR range allowlists.

Connecting to AWS Cloud WAN core networks

Connect a global resolver to an AWS Cloud WAN (Cloud WAN) core network with a core network endpoint. After you connect the core network, you can use its segments as access sources for DNS resolution.

Maintaining visibility and compliance

Monitor DNS activity across your organization by delivering logs to Amazon CloudWatch, Firehose, or Amazon Simple Storage Service. Choose a single destination Region for centralized log storage to support security audits, compliance requirements, and threat investigation.

AWS services that integrate with Route 53 Global Resolver for enhanced DNS security

Route 53 Global Resolver integrates seamlessly with other AWS services to provide comprehensive DNS security and management capabilities. These integrations enhance Route 53 Global Resolver's core functionality and enable advanced use cases:

Amazon Route 53

Route 53 Global Resolver integrates with Amazon Route 53 to resolve private hosted zones globally. For more information, see the Amazon Route 53 Developer Guide.

Amazon CloudWatch

Route 53 Global Resolver can deliver query logs to Amazon CloudWatch for monitoring and analysis. For more information, see the Amazon CloudWatch User Guide.

Amazon Simple Storage Service

Route 53 Global Resolver can store query logs in Amazon Simple Storage Service buckets for long-term storage and analysis. For more information, see the Amazon S3 User Guide.

Amazon Data Firehose

Route 53 Global Resolver can stream query logs to Amazon Data Firehose for real-time processing. For more information, see the Amazon Data Firehose Developer Guide.

Route 53 Resolver DNS Firewall

Route 53 Global Resolver complements Route 53 Resolver DNS Firewall by providing global DNS filtering capabilities. For more information, see the Route 53 Resolver DNS Firewall documentation.

AWS Global Accelerator

Route 53 Global Resolver works with AWS Global Accelerator to provide optimized global network performance for DNS resolution. For more information, see the AWS Global Accelerator Developer Guide.

AWS Cloud WAN

Route 53 Global Resolver can connect to an AWS Cloud WAN core network through a core network endpoint. After you connect the core network, its segments can be used as access sources, so clients that reach the global resolver from those segments are authorized for DNS resolution. For more information, see Connecting a global resolver to a core network.

How to access and manage Route 53 Global Resolver through AWS interfaces

Route 53 Global Resolver provides multiple access methods to accommodate different management preferences and automation requirements. You can manage Route 53 Global Resolver through these interfaces:

AWS Management Console

The AWS Management Console provides a web-based interface for Route 53 Global Resolver. You can use the console to configure DNS filtering policies, manage authentication settings, and monitor query logs.

AWS Command Line Interface (AWS CLI)

The AWS CLI provides direct access to Route 53 Global Resolver API operations. You can use the AWS CLI to automate Route 53 Global Resolver tasks and integrate them into your scripts and workflows.

AWS SDKs

AWS provides SDKs for many programming languages. You can use the SDKs to integrate Cloud Resolver functionality into your applications.

REST API

Route 53 Global Resolver provides a REST API that you can use to programmatically manage your DNS resolution and filtering policies.

Supported AWS Regions

Route 53 Global Resolver is available in the following AWS Regions:

- US East (N. Virginia)
- US East (Ohio)
- US West (N. California)
- US West (Oregon)
- Europe (Frankfurt)
- Europe (Ireland)
- Europe (London)
- Asia Pacific (Mumbai)
- Asia Pacific (Singapore)
- Asia Pacific (Tokyo)
- Asia Pacific (Seoul)

- Asia Pacific (Sydney)
- Africa (Cape Town)
- Asia Pacific (Hong Kong)
- Asia Pacific (Taipei)
- Asia Pacific (Osaka)
- Asia Pacific (Hyderabad)
- Asia Pacific (Jakarta)
- Asia Pacific (Melbourne)
- Asia Pacific (Malaysia)
- Asia Pacific (Thailand)
- Canada (Central)
- Canada West (Calgary)
- Europe (Zurich)
- Europe (Stockholm)
- Europe (Milan)
- Europe (Spain)
- Europe (Paris)
- Mexico (Central)
- South America (São Paulo)

Opt-in Region considerations for Route 53 Global Resolver

Some AWS Regions are active by default for your AWS account. Certain Regions are activated only when you manually select them. This document refers to those Regions as opt-in Regions. In contrast, Regions that are active by default, as soon as your AWS account is created, are referred to as commercial Regions, default Regions, or simply, Regions.

AWS Regions introduced after March 20, 2019 are deployed as opt-in Regions. In an opt-in Region, your account is not enabled within that Region. Your Route 53 Global Resolver data is not replicated to the Region unless you choose to opt into use of that Region. For more information about opt-in Regions, see Managing AWS Regions.

When using Route 53 Global Resolver in opt-in Regions, here are some considerations to keep in mind.

Behavioral considerations

- To select an opt-in Region for your Global Resolver deployment, your account must be opted-in to the Region.
- You can select any set of default or opt-in Regions for your Global Resolver.
- If you have selected an AWS opt-in Region for a Global Resolver, when you opt-out your account from that Region, the service will stop propagating your global resolver data to that Region. It will also trigger deletion of all your resources in that Region. This will cause impairment to your DNS traffic destined to that Region as resources get cleaned up, but traffic will eventually direct away to your other selected Regions.
- Similarly, if you have selected only a set of opt-in Regions for your Global Resolver and you opt-out your account from all the Regions, it will effectively delete your Global Resolver.
- You might also select any default or opt-in Region for your Observability Region where your DNS logs will be delivered. If you choose to opt-out your account, the service will stop sending DNS logs to your destinations in that Region. To prevent impairment of log delivery, we recommend setting up a new log delivery destination in a separate Region and update the observability Region in your Global Resolver accordingly.
- Global Resolver does not support expanding and contracting Regions in your Global Resolver at this time. We recommend you plan ahead which Regions you intend to opt-in and select before creating a Global Resolver.

Setting up account access for Route 53 Global Resolver

Before you start using Route 53 Global Resolver, you need an AWS account and the appropriate permissions to access Route 53 Global Resolver resources. This includes creating IAM users and roles with the necessary permissions.

This section guides you through the steps required to configure users and roles to access Route 53 Global Resolver.

Topics

- Creating policies and roles
- Network considerations

Creating policies and roles

Configure AWS Identity and Access Management (IAM) permissions so your team can deploy and manage Route 53 Global Resolver resources. You can use administrative permissions for full access or read-only permissions for monitoring and viewing configurations.

All Route 53 Global Resolver API operations require appropriate IAM permissions. If you don't have the required permissions, API calls will return `AccessDeniedException` (401) or `UnauthorizedException` (401) errors.

Administrative permissions

If you're setting up Route 53 Global Resolver for the first time or managing all aspects of the service, you need administrative permissions. You can use these AWS managed policies:

- `AmazonRoute53GlobalResolverFullAccess` - Provides full access to Route 53 Global Resolver resources, including creating, updating, and deleting global resolvers, DNS views, firewall rules, and domain lists
- `AmazonRoute53FullAccess` - Required if you plan to use private hosted zone forwarding
- `CloudWatchLogsFullAccess` - Required if you plan to send logs to Amazon CloudWatch
- `AmazonS3FullAccess` - Required if you plan to import firewall domain lists from Amazon S3 or send logs to Amazon S3

Read-only permissions

If you only need to view Route 53 Global Resolver configurations and logs, you can use these AWS managed policies:

- `AmazonRoute53GlobalResolverReadOnlyAccess` - Provides read-only access to Route 53 Global Resolver resources, including viewing global resolvers, DNS views, firewall rules, domain lists, and access sources
- `AmazonRoute53ReadOnlyAccess` - Required to view private hosted zone associations
- `CloudWatchReadOnlyAccess` - Required to view logs in Amazon CloudWatch
- `AmazonS3ReadOnlyAccess` - Required to view firewall domain list files stored in Amazon S3

Network considerations

Before implementing Route 53 Global Resolver, consider the following network requirements:

Client IP ranges

This is only required when using access source-based authentication. Identify the IP address ranges (CIDR blocks) for all clients that will use Route 53 Global Resolver. You'll need these for configuring rules for your access source.

DNS protocols

Determine which DNS protocols your clients will use:

- **Do53** - Standard DNS over port 53 (UDP/TCP)
- **DoH** - DNS-over-HTTPS for encrypted queries
- **DoT** - DNS-over-TLS for encrypted queries

Firewall and security groups

Ensure your network firewalls and security groups allow outbound traffic to Route 53 Global Resolver anycast IP addresses on the appropriate ports (53 for Do53, 443 for DoH, 853 for DoT).

Tutorial: Create your first Route 53 Global Resolver

This getting started guide demonstrates the basic components of Route 53 Global Resolver and optionally creating a simple DNS filtering setup. This tutorial covers the core concepts but doesn't include production requirements like client configuration, logging, or private domain resolution.

When you're finished, you'll have a basic Route 53 Global Resolver setup that can filter DNS queries and block malicious domains.

The following sections describe how to quickly get started with DNS security and filtering using Route 53 Global Resolver.

Prerequisites

Before you can use Route 53 Global Resolver, you need an AWS account and the appropriate permissions to access, view, and edit Route 53 Global Resolver components. Your system administrator must complete the steps in the section called “Setting up account access”, and then return to this tutorial.

Step 1: Create a global resolver

First, create a global resolver instance and select the AWS Regions where it will operate.

1. Open the Route 53 Global Resolver console at https://console.aws.amazon.com/route53globalresolver/.
2. Choose **Create global resolver**.
3. For **Name**, enter a descriptive name for your global resolver.
4. For **Description**, optionally enter a description.
5. For **Regions**, select two or more AWS Regions where you want to instantiate the global resolver. Choose Regions closest to your clients for optimal performance.
6. For **IP address type**, choose the IP address type for this resolver.
   - **IPv4** - Includes only IPv4 addresses.
   - **Dualstack** - Includes IPv4 and IPv6 addresses.
7. Optionally, add tags to help organize and manage your resources.
8. Choose **Create global resolver**.

You'll receive anycast IPv4 addresses immediately that your clients can use to reach the resolver. The global resolver creation process takes a few minutes to complete before the addresses become functional.

Step 2: Create a DNS view and configure authentication

Create a DNS view to organize your clients and configure authentication using IP Access Sources. This tutorial uses IP-based authentication. You can also use access tokens for DoH/DoT protocols.

1. In the Route 53 Global Resolver console, choose your global resolver.
2. Choose **Create DNS view**.
3. For **Name**, enter a descriptive name for your DNS view.
4. For **Description**, optionally enter a description.
5. Choose **Create DNS view**.
6. After the DNS view is created, choose **Access source** and then **Create access source**.
7. For **CIDR block**, enter the IP address range for your clients (for example, `203.0.113.0/24`).

8. For **Protocol**, choose **Do53** (DNS over port 53) for basic setup.
9. Choose **Create Access Source rule**.

Step 3: Configure DNS filtering rules (optional)

Set up basic DNS filtering rules to block access to malicious domains.

1. In your DNS view, choose **Firewall rules** and then **Create firewall rule**.
2. For **Name**, enter a descriptive name for the rule.
3. For **Priority**, enter `100` (lower numbers have higher priority).
4. For **Action**, choose **Block**.
5. For **Domain list type**, choose **AWS Managed Domain List**.
6. For **Managed domain list**, choose **AmazonGuardDutyThreatList** and **Malware and Botnet Command and Control** to block known malicious domains (you can add other managed lists or create custom lists later).
7. Choose **Create firewall rule**.

Step 4: Test your configuration

Test that your Route 53 Global Resolver configuration is working correctly.

1. From a client machine within your configured CIDR range, test DNS resolution using the anycast IP addresses provided by your global resolver:

   ```
   nslookup example.com <anycast-ip-address>
   ```

2. Verify that legitimate domains resolve correctly.
3. Test that blocked domains are properly filtered. You can create a custom domain list with a test domain to verify your firewall rules are working correctly. For more information about Managed Domain Lists, see Managed Domain Lists.
4. Check the Route 53 Global Resolver console for query logs and filtering activity.

For comprehensive testing procedures and troubleshooting, see *Troubleshooting Route 53 Global Resolver*.

Step 5: Monitoring DNS activity

Configure logging for your DNS activity.

1. Choose an Observability Region.
2. Select the destination for query logs.

For comprehensive testing procedures and troubleshooting, see *Testing and troubleshooting Route 53 Global Resolver*.

Step 6: Clean up (optional)

If you created this configuration for testing purposes and don't want to continue using Route 53 Global Resolver, clean up the resources to avoid ongoing charges.

1. In the Route 53 Global Resolver console, delete any firewall rules you created.
2. Delete any Access Source rules you created.
3. Delete the DNS view.
4. Delete the global resolver.

> **Important**
>
> Deleting these resources will stop DNS resolution for any clients configured to use them. Update your client configurations before deleting the resolver or removing access rules.

Next steps

Now that you have a basic Route 53 Global Resolver configuration, you can explore additional features:

- Configure client devices to use your resolver (required for production). Update your client DNS settings to use the anycast IP addresses provided by your global resolver.
- Set up logging for monitoring and compliance (recommended for production). Configure logging to Amazon CloudWatch, Amazon S3, or Amazon Data Firehose for monitoring and analysis. For more information, see Monitoring DNS activity and performance with Route 53 Global Resolver.

- Configure private hosted zone forwarding for internal domains (required if you have private AWS resources). For more information, see *Working with private hosted zones*.
- Set up encrypted DNS connectivity using DNS-over-HTTPS (DoH) or DNS-over-TLS (DoT). For more information, see *Configuring encrypted DNS*.
- Create custom domain lists and advanced filtering rules. For more information, see *DNS filtering*.
- Connect your global resolver to an AWS Cloud WAN core network. Create a core network endpoint so that the core network's segments can be used as access sources for DNS resolution. For more information, see Connecting a global resolver to a core network.

Tutorial: Create your first Route 53 Global Resolver with AWS Cloud WAN

This tutorial demonstrates how to create a private-only Route 53 Global Resolver. You create a global resolver with public IP connectivity turned off, connect it to an AWS Cloud WAN core network, and allow DNS queries from a core network segment. As a result, the resolver is reachable only over your private Cloud WAN connections and not over the internet.

When you're finished, you have a global resolver that authorized clients can reach only from a connected Cloud WAN segment.

The following sections describe how to get started with a private-only Route 53 Global Resolver using AWS Cloud WAN.

Prerequisites

Before you begin, make sure that you have the following:

- Completion of the setup steps. Your system administrator must complete the steps in the section called “Setting up account access”, and then return to this tutorial.
- An existing AWS Cloud WAN core network that you own. We recommend that its core network policy is at version `2026.10` or later so that DNS queries route to the closest AWS Region. For more information, see Core network policy versions.
- An AWS account and the appropriate permissions to access, view, and edit Route 53 Global Resolver components.

Step 1: Create a global resolver with public IP connectivity disabled

First, create a global resolver instance, select the AWS Regions where it will operate, and turn off public IP connectivity.

1. Open the Route 53 Global Resolver console at https://console.aws.amazon.com/route53globalresolver/.
2. Choose **Create global resolver**.
3. For **Name**, enter a descriptive name for your global resolver.
4. For **Description**, optionally enter a description.
5. For **Regions**, select two or more AWS Regions where you want to instantiate the global resolver. Choose Regions closest to your clients for optimal performance.
6. For **Public IP connectivity**, disable public IP connectivity. When public IP connectivity is disabled, the global resolver is reachable only from a connected core network, and not over the internet.
7. Optionally, add tags to help organize and manage your resources.
8. Choose **Create global resolver**.

The global resolver creation process takes a few minutes to complete. Because public IP connectivity is disabled, you connect a Cloud WAN core network in the next step to make the resolver reachable from your private connections.

Step 2: Connect a Cloud WAN core network

Create a core network endpoint to connect your global resolver to an AWS Cloud WAN core network. A core network endpoint requires a minimum of two AWS Regions, which must be among the resolver's Regions, and supports up to five private IP addresses.

1. In the Route 53 Global Resolver console, choose your global resolver.
2. Choose the **Core network endpoints** tab, and then choose **Create core network endpoint**.
3. For **Core network**, select the Cloud WAN core network to connect to.
4. Choose **Create core network endpoint**.

After you connect the core network, its segments can be used as access sources for DNS resolution. For more information about core network endpoints, see Connecting a global resolver to a core network.

Step 3: Create a DNS view and a segment-based access source

Create a DNS view to organize your clients, and then create an access source that selects a Cloud WAN segment. Clients that reach the resolver from that segment are authorized to resolve DNS queries.

1. In the Route 53 Global Resolver console, choose your global resolver.
2. Choose **Create DNS view**.
3. For **Name**, enter a descriptive name for your DNS view.
4. Choose **Create DNS view**.
5. After the DNS view is created, choose **Access source** and then **Create access source**.
6. For the access source type, select **Cloud WAN segment** instead of a CIDR range.
7. For **Segment**, select the core network segment whose clients you want to authorize.
8. Choose **Create Access Source rule**.

For more information about access sources, see Managing access controls with access sources and tokens in Route 53 Global Resolver.

Next steps

Now that you have a private-only Route 53 Global Resolver connected to a Cloud WAN core network, you can explore additional features:

- Configure client devices to use your resolver (required for production). Update your client DNS settings to use the private IP addresses of the core network endpoint.
- Set up logging for monitoring and compliance (recommended for production). Configure logging to Amazon CloudWatch, Amazon S3, or Amazon Data Firehose for monitoring and analysis. For more information, see Monitoring DNS activity and performance with Route 53 Global Resolver.
- Add DNS filtering rules to block malicious domains. For more information, see *DNS filtering*.

Managing DNS views with Route 53 Global Resolver

You can manage ongoing Route 53 Global Resolver operations including updating DNS views to control which client device groups can resolve to internal resources and what domains to filter.

Managing DNS views

After creating DNS views, you can update their configuration, enable or disable them, and manage their lifecycle.

Creating DNS views for client device groups

A DNS view is a logical grouping defines security policies for a group of client devices, such as remote workers, branch office devices, or on-premises equipment. Each view has its own authentication requirements, filtering rules, and private hosted zone associations.

To create a DNS view

1. Open the console at https://console.aws.amazon.com/route53globalresolver/.
2. Choose your global resolver from the list.
3. Choose the **DNS views** tab.
4. Choose **Create DNS view**.
5. In the **DNS view details** section:
   a. For **DNS view name**, enter a descriptive name for your DNS view (up to 128 characters).
   b. (Optional) For **Description**, enter a description for your DNS view (up to 255 characters).
6. In the **DNS query handling** section, configure the following settings:
   - **DNSSEC validation** - Choose **Enable** or **Disable**. DNSSEC validation enables the Global Resolver to verify the authenticity of DNS responses.
   - **Firewall rules fail open behavior** - Choose **Enable** to allow queries to proceed when DNS Firewall cannot evaluate them, or **Disable** to block such queries.
   - **EDNS0 client subnet** - Choose **Enable** to improve client location accuracy for traffic routing to nearby resources and efficient caching, or **Disable** to turn off this feature.
7. Choose **Create DNS view**.

After creating the DNS view, you can configure access controls, firewall rules, and private hosted zone associations.

Editing DNS views

You can modify DNS view settings after creation, including DNS query handling options and associated resources.

To edit a DNS view

1. In the console, navigate to your global resolver.
2. Choose the **DNS views** tab.
3. Select the DNS view you want to edit and choose **Edit**.
4. Modify the DNS view settings as needed and choose **Save changes**.

Enabling and disabling DNS views

You can temporarily disable a DNS view without deleting it. When disabled, the global resolver stops serving requests for client devices associated with that DNS view.

> **Warning**
>
> Disabling a DNS view immediately stops DNS resolution for all client devices associated with that view. Ensure you have alternative DNS resolution configured for affected client devices.

Deleting DNS views

Before you can delete a DNS view, you must first delete all associated resources, including Access Source rules, access tokens, firewall rules, and private hosted zone associations.

> **Warning**
>
> Deleting a DNS view is irreversible and will immediately stop DNS resolution for all client devices associated with that view.

Managing private hosted zone associations

You can view, update, and remove private hosted zone associations as needed to control which client device groups have access to internal resources.

Viewing associations

To view all private hosted zone associations for a DNS view, navigate to your DNS view and check the **Private hosted zones** section to see all associated zones with their status and association details.

Updating associations

You can update the name of a private hosted zone association by selecting the association, choosing **Edit**, updating the association name, and saving changes.

Removing associations

When you remove a private hosted zone association, Route 53 Global Resolver stops using that zone to resolve DNS queries for the associated DNS view.

> **Warning**
>
> Removing a private hosted zone association immediately affects DNS resolution. Queries for domains in the disassociated zone will be resolved using public DNS instead of the private zone records.

Configure settings for DNS views in Route 53 Global Resolver

Route 53 Global Resolver allows you to configure different DNS policies and access controls for different groups of client devices based on their security requirements and access needs. Set up DNS policies and access controls in Route 53 Global Resolver for different groups of client devices based on their security requirements and access needs.

Configuring DNS settings for client groups

Each DNS view has several settings that control how DNS queries are processed and resolved for different client device groups.

DNSSEC validation

DNSSEC validation helps verify that DNS responses for public domains are authentic and haven't been tampered with. When you enable DNSSEC validation, Route 53 Global Resolver checks DNSSEC signatures and returns SERVFAIL for domains with invalid signatures.

Consider enabling DNSSEC validation if:

- Your organization needs cryptographic verification of DNS responses
- You want protection against DNS spoofing and cache poisoning attacks
- You have compliance requirements that require DNSSEC validation

> **Note**
>
> DNSSEC validation only applies to public domains. Private hosted zones use their own authentication mechanisms.

EDNS Client Subnet (ECS)

EDNS Client Subnet includes information about the client's network location in DNS queries sent to authoritative servers. This allows content delivery networks and geographically distributed services to provide location-appropriate responses.

ECS can help you:

- Get better performance from geographically distributed services
- Improve content delivery network routing accuracy
- Better comply with regional content restrictions

Privacy considerations:

- ECS reveals partial client IP information to authoritative servers (maximum /24 for IPv4 and /48 for IPv6)
- Consider your organization's privacy requirements before enabling

Firewall fail open

The firewall fail open setting determines what happens when DNS firewall rules cannot be evaluated due to service impairment or configuration issues.

Disabled (default)

DNS queries are blocked when firewall rules can't be evaluated. This gives you maximum security but might affect availability during service issues.

Enabled

DNS queries are allowed when firewall rules can't be evaluated. This prioritizes availability over security during service issues.

Best practices for organizing client device groups

Follow these best practices when designing DNS views for different client device groups:

View organization strategies

- **Separate by security requirements** - Create different views for client devices with different security clearances or access levels
- **Organize by location** - Use separate views for different geographic locations or network segments
- **Group by device type** - Create dedicated views for servers, workstations, mobile devices, or IoT devices
- **Use descriptive names** - Choose names that clearly indicate the view's purpose and target client devices

Security considerations

- **Principle of least privilege** - Configure each view with the minimum access required for its client devices
- **Default deny** - Start with restrictive firewall rules and add exceptions as needed
- **Regular review** - Periodically review and update DNS view configurations
- **Monitor usage** - Use DNS query logs to monitor and analyze DNS view usage patterns

Managing access controls with access sources and tokens in Route 53 Global Resolver

Route 53 Global Resolver provides two primary methods for controlling client device access: access sources for IP-based authentication and access tokens for token-based authentication. This chapter covers both approaches, helping you choose the right authentication method for your environment and implement comprehensive access controls.

Topics

- Access control methods
- Configuring access sources and access source rules
- Managing access tokens for encrypted authentication
- Access control best practices and security considerations

Access control methods

Route 53 Global Resolver offers two distinct authentication methods to control client access to your DNS infrastructure. Each method serves different use cases and environments.

Access sources

: Access sources use IP-based authentication to control DNS queries. You configure access source rules that allow or deny queries based on client IP addresses. This method works well for environments with predictable IP ranges, such as branch offices or VPN connections. Access sources support all DNS protocols (Do53, DoT, and DoH) and provide straightforward configuration for network administrators.

Access tokens

: Access tokens use token-based authentication with encrypted, time-limited credentials for DoH and DoT protocols. This method suits mobile clients and environments where IP addresses change frequently. You can renew tokens before expiration and they offer enhanced security through encryption.

Consider these factors when selecting your authentication approach:

Authentication method comparison

Factor	Access sources	Access tokens
Use case	Fixed IP ranges, office networks, VPN users	Mobile devices, dynamic IPs, remote workers
Security level	Network-based, relies on IP trust	Encrypted credentials, time-limited
Management complexity	Simple IP range management	Token lifecycle and distribut ion
Protocol support	Do53, DoT, DoH	DoT, DoH only

You can use both methods simultaneously to create layered security. For example, use access sources for office networks and tokens for remote workers.

Configuring access sources and access source rules

Access sources control which clients can query your DNS infrastructure and which protocols they can use. You define an access source in one of two ways: by an IP address range (CIDR) or by an AWS Cloud WAN (Cloud WAN) core network segment. You create access source rules that specify the IP ranges or segments that can query your DNS infrastructure and the protocols they can use.

A segment-based access source is available only after you connect the Cloud WAN core network to your global resolver with a core network endpoint. For more information about core network endpoints and segment-based access sources, see Connecting a global resolver to a core network and Using a Cloud WAN segment as an access source.

Creating access source rules

Follow these steps to create an access source rule that allows specific IP ranges to query your DNS infrastructure.

1. Open the Route 53 Global Resolver console and navigate to your DNS view.
2. In the **Access source** section, choose **Create access source rule**.
3. For **Rule name**, enter a descriptive name that identifies the purpose of this rule, such as `office-network` or `vpn-users`.

4. For **IP address type**, choose **IPV4** or **IPV6**.
5. For **CIDR block**, specify the IP addresses that should have access. You can use CIDR notation for IP ranges: `203.0.113.0/24` or `2001:db8::/112`, or individual IP addresses: `203.0.113.5/32` or `2001:db8::1/128`.
6. For **Protocol**, select the DNS protocols this rule applies to:
   - **Do53** - Standard DNS over UDP/TCP (port 53)
   - **DoT** - DNS over TLS (port 853)
   - **DoH** - DNS over HTTPS (port 443)
7. Choose **Create access source rule**.

Client devices from the specified IP ranges can now query your DNS infrastructure using the selected protocols.

Using a Cloud WAN segment as an access source

Instead of an IP address range, you can define an access source by an AWS Cloud WAN (Cloud WAN) core network segment. Clients that reach the global resolver from that segment are authorized the same way as clients that match an IP-based access source.

Before you can use a segment as an access source, you must connect the Cloud WAN core network to your global resolver with a core network endpoint. After you connect the core network, its segments become available as access sources in the AWS Regions where your global resolver runs. For the steps to connect a core network and the full details of core network endpoints, see Connecting a global resolver to a core network.

Follow these steps to create an access source rule that uses a Cloud WAN segment.

1. Open the Route 53 Global Resolver console and navigate to your DNS view.
2. In the **Access source** section, choose **Create access source rule**.
3. For **Rule name**, enter a descriptive name that identifies the purpose of this rule, such as `cloudwan-production`.
4. For **IP address type**, choose **NONE**. A segment-based access source isn't defined by an IP range, so you don't choose IPV4 or IPV6.
5. For **Segment**, select the Cloud WAN core network segment that should have access instead of entering a CIDR block. Only segments from core networks that you have connected with a core network endpoint are available.

6. For **Protocol**, select the DNS protocols this rule applies to:
   - **Do53** - Standard DNS over UDP/TCP (port 53)
   - **DoT** - DNS over TLS (port 853)
   - **DoH** - DNS over HTTPS (port 443)
7. Choose **Create access source rule**.

Client devices that reach the global resolver from the selected segment can now query your DNS infrastructure using the selected protocols. If you delete the core network endpoint for that core network, its segments are no longer available as access sources.

Understanding rule evaluation and priority

Route 53 Global Resolver evaluates access source rules when identifying the correct view to use.

- Rules are processed from most specific to least specific IP ranges, where the most-specific matching rule takes precedence.
- If no rules match, the request is denied by default.

Test your access source configuration by querying from different IP addresses to make sure the rules work as expected.

Managing access tokens for encrypted authentication

Access tokens provide encrypted authentication for DoH and DoT protocols. Unlike IP-based access sources, tokens work regardless of client location and offer enhanced security through encryption and expiration controls.

Creating access tokens

Follow these steps to create access tokens to authenticate client devices that use DoH or DoT protocols.

1. Open the Route 53 Global Resolver console and navigate to your DNS view.
2. In the **Access source** section, choose **Create access token**.
3. For **Name**, enter a descriptive name that identifies the token's purpose, such as `mobile-devices` or `remote-workers-q4`.

4. For **Expiration**, set when the token should expire. We recommend 90 days or less for security. Consider your token distribution and renewal capabilities when setting the expiration period.
5. Choose **Create access token**.
6. Distribute the token securely to your client devices using your organization's secure communication channels.

Configuring client devices with access tokens

Configure client devices to use access tokens for authentication with your Route 53 Global Resolver infrastructure.

DoH configuration

To configure DoH with access tokens, you need your global resolver's DNS name or IP addresses:

1. Use the GetGlobalResolver API to retrieve connectivity details for your resolver.
2. Note the `ipv4Addresses` (for example, 3.3.3.3, 3.3.3.4) and `dnsName` (for example, a1bc234567890a.route53globalresolver.global.on.aws).
3. Include the token as a URL parameter in the DoH endpoint using the DNS name:

```
https://a1bc234567890a.route53globalresolver.global.on.aws/dns-query?token=<token-value>
```

Replace `<token-value>` with the actual token that you generated.

DoT configuration

For DoT queries with access tokens, include the token in an EDNS0 option with the following specifications:

- **Option Code:** `0xffa0`
- **Option Data:** The access token in string format

The specific implementation depends on your DoT client software and how it handles EDNS0 options.

Token lifecycle management

Manage token expiration and renewal to maintain secure access for your client devices.

- **Monitor expiration dates** - Track token expiration dates and plan renewals in advance.
- **Renew before expiration** - Create new tokens before old ones expire to avoid service interruption.
- **Rotate tokens regularly** - Replace tokens periodically even before expiration for enhanced security.
- **Revoke compromised tokens** - Delete tokens immediately if you suspect they have been compromised.

Consider implementing automated token renewal processes for large deployments to reduce administrative overhead.

Platform configuration examples

Use these platform-specific examples to configure client devices with your Route 53 Global Resolver access tokens and connection details.

Windows configuration

Follow these steps to configure Windows clients to use DoH with access tokens using the netsh command.

1. Open Command Prompt as an administrator.
2. Enable the global DoH setting:

```
netsh dns add global doh=yes
```

3. Register DoH servers with access tokens. Replace the example values with your actual resolver details:

```
netsh dns add encryption server=3.3.3.3 dohtemplate=https://
a1bc234567890a.route53globalresolver.global.on.aws/dns-query?token=<your-token>
 autoupgrade=yes
netsh dns add encryption server=3.3.3.4 dohtemplate=https://
a1bc234567890a.route53globalresolver.global.on.aws/dns-query?token=<your-token>
 autoupgrade=yes
```

4. Flush the DNS cache:

```
ipconfig /flushdns
```

5. Verify the configuration:

```
netsh dns show global
```

macOS configuration

Follow these steps to configure macOS clients using a mobile configuration profile for DoH with access tokens.

Create a mobile configuration profile with the following structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/
PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>PayloadContent</key>
  <array>
    <dict>
      <key>DNSSettings</key>
      <dict>
        <key>DNSProtocol</key>
        <string>HTTPS</string>
        <key>ServerAddresses</key>
        <array>
          <string>3.3.3.3</string>
          <string>3.3.3.4</string>
        </array>
        <key>ServerURL</key>
        <string>https://a1bc234567890a.route53globalresolver.global.on.aws/dns-query/
token/your-token</string>
      </dict>
      <key>PayloadType</key>
      <string>com.apple.dnsSettings.managed</string>
      <key>PayloadDisplayName</key>
      <string>Global Resolver DoH</string>
      <key>PayloadIdentifier</key>
      <string>com.example.dnssettings</string>
      <key>PayloadUUID</key>
      <string>uuid-1</string>
      <key>PayloadVersion</key>
      <integer>1</integer>
```

```
    </dict>
  </array>
  <key>PayloadType</key>
  <string>Configuration</string>
  <key>PayloadIdentifier</key>
  <string>com.example.dnssettings.profile</string>
  <key>PayloadUUID</key>
  <string>uuid-2</string>
  <key>PayloadVersion</key>
  <integer>1</integer>
  <key>PayloadDisplayName</key>
  <string>Global Resolver DoH</string>
</dict>
</plist>
```

Note

On macOS, pass the token as a path segment (`/dns-query/token/<your-token>`), not a query parameter. macOS strips query parameters from the DoH server URL, so the token never reaches Global Resolver.

Replace the example values:

- *a1bc234567890a.route53globalresolver.global.on.aws*: your Global Resolver's DNS name.
- *3.3.3.3* and *3.3.3.4*: your Global Resolver's IPv4 addresses. For dual-stack resolvers, you can also list its IPv6 addresses.
- *your-token*: your access token value, as returned by `GetAccessToken`. Don't use the token ID.
- *com.example.dnssettings* and *com.example.dnssettings.profile*: reverse-DNS identifiers for the DNS payload and the profile. *uuid-1* and *uuid-2*: unique UUIDs for each (for example, generate them with `uuidgen`).

Install the profile manually on the device, or deploy it through your mobile device management (MDM) solution.

Access control best practices and security considerations

Follow these best practices to maintain secure and effective access controls for your Route 53 Global Resolver infrastructure.

Security best practices

Implement these security measures to protect your DNS infrastructure:

- **Use layered authentication** - Combine access sources for trusted networks with tokens for mobile users. This approach provides defense in depth and accommodates different client scenarios.
- **Implement least privilege access** - Grant access only to the IP ranges and protocols that clients actually need. Avoid overly broad access source rules that could expose your infrastructure to unauthorized use.
- **Rotate tokens regularly** - Replace access tokens on a regular schedule, even before they expire. This practice limits the impact of compromised tokens and maintains security hygiene.
- **Monitor access patterns** - Review DNS query logs to identify unusual access patterns or potential security issues. Set up alerts for queries from unexpected IP ranges or using expired tokens.

Operational best practices

Follow these operational practices to maintain reliable access controls:

- **Document your access control strategy** - Maintain clear documentation of which access sources and tokens serve which client groups.
- **Test access controls regularly** - Verify that your access source rules and tokens work correctly from different client locations and scenarios.
- **Plan for token renewal** - Establish processes for distributing new tokens before old ones expire to avoid service disruptions.
- **Review access controls periodically** - Remove unused access source rules and expired tokens to maintain a clean configuration.

Securing DNS for clients with Route 53 Global Resolver

The following sections describe how to create rules to secure DNS queries made by your clients.

Topics

- Configure and manage DNS Firewall rules
- Managed Domain Lists for Route 53 Global Resolver
- Build lists of domains to block or allow
- DNS Firewall Advanced protections
- Manage DNS security policies in Route 53 Global Resolver

Configure and manage DNS Firewall rules

Creating and viewing firewall rules

Firewall rules define how Route 53 Global Resolver handles DNS queries based on domain lists, managed domain lists, content categories, or advanced threat protection. Each rule specifies a priority, target domains, and an action to take.

Best practices for rule priority:

- Use priority 100-999 for high-priority allow rules (trusted domains)
- Use priority 1000-4999 for block rules (known threats)
- Use priority 5000-9999 for alert rules (monitoring and analysis)
- Leave gaps between priorities to allow for future rule insertion

To create a DNS Firewall rule

1. In the Route 53 Global Resolver console, navigate to your DNS view.
2. Choose the **Firewall rules** tab.
3. Choose **Create firewall rule**.
4. In the **Rule details** section:
   a. For **Rule name**, enter a descriptive name for the rule (up to 128 characters).
   b. (Optional) For **Rule description**, enter a description for the rule (up to 255 characters).
5. In the **Rule configuration** section, choose the **Rule configuration type**:

- **Customer managed domain lists** - Use a domain list that you create and manage
- **AWS managed domain lists** - Use domain lists provided by Amazon that you can utilize
- **DNS Firewall Advanced protections** - Choose from a range of managed protections and specify a confidence threshold

6. For **Rule action**, choose the action to take when the rule matches:
   - **Allow** - The DNS query is resolved
   - **Alert** - Allows the DNS query but creates an alert
   - **Block** - The DNS query is blocked
7. Choose **Create firewall rule**.

Use the following procedure to view the rules assigned to them. You can also update the rule and rule settings.

To view and update a rule

1. In the Route 53 Global Resolver console, navigate to your DNS View.
2. Choose the **DNS Firewall rules** tab.
3. Choose the rule you want to view or edit, and choose **Edit**.
4. In the **Rule** page, you can view and edit settings.

For information about the values for rules, see the section called “Rule settings in DNS Firewall”.

To delete a rule

1. In the Route 53 Global Resolver console, navigate to your DNS View.
2. Choose the **DNS Firewall rules** tab.
3. Choose the rule you want to delete, and choose **Delete**, and confirm the deletion.

Rule settings in DNS Firewall

When you create or edit a DNS Firewall rule in your DNS View, you specify the following values:

Name

A unique identifier for the rule in the DNS View.

(Optional) Description

A short description that provides more information about the rule.

Domain list

The list of domains that the rule inspects for. You can create and manage your own domain list or you can subscribe to a domain list that AWS manages for you.

A rule can contain ether a domain list or a DNS Firewall Advanced protection, but not both.

Query type (domain lists only)

The list of DNS query types that the rule inspects for. The following are the valid values:

- A: Returns an IPv4 address.
- AAAA: Returns an Ipv6 address.
- CAA: Restricts CAs that can create SSL/TLS certifications for the domain.
- CNAME: Returns another domain name.
- DS: Record that identifies the DNSSEC signing key of a delegated zone.
- MX: Specifies mail servers.
- NAPTR: Regular-expression-based rewriting of domain names.
- NS: Authoritative name servers.
- PTR: Maps an IP address to a domain name.
- SOA: Start of authority record for the zone.
- SPF: Lists the servers authorized to send emails from a domain.
- SRV: Application specific values that identify servers.
- TXT: Verifies email senders and application-specific values.

A query type you define by using the DNS type ID, for example 28 for AAAA. The values must be defined as TYPENUMBER, where the NUMBER can be 1-65334, for example, TYPE28. For more information, see List of DNS record types.

You can create one query type per rule.

DNS Firewall Advanced protection

Detects suspicious DNS queries based on known threat signatures in DNS queries. You can choose protection from:

- Domain Generation Algorithms (DGAs)

DGAs are used by attackers to generate a large number of domains to launch malware attacks.

- DNS tunneling

  DNS tunneling is used by attackers to exfiltrate data from the client by using the DNS tunnel without making a network connection to the client.

In a DNS Firewall Advanced rule you can choose to either block, or alert on a query that matches the threat.

For more information, see DNS Firewall Advanced protections.

A rule can contain ether a DNS Firewall Advanced protection or a domain list, but not both.

Confidence threshold (DNS Firewall Advanced only)

The confidence threshold for DNS Firewall Advanced. You must provide this value when you create a DNS Firewall Advanced rule. The confidence level values mean:

- High – Detects only the most well corroborated threats with a low rate of false positives.
- Medium – Provides a balance between detecting threats and false positives.
- Low – Provides the highest detection rate for threats, but also increases false positives.

For more information, see Rule settings in DNS Firewall.

Action

How you want DNS Firewall to handle a DNS query whose domain name matches the specifications in the rule's domain list. For more information, see the section called “Rule actions in DNS Firewall”.

Priority

Unique positive integer setting for the rule within the DNS View that determines processing order. DNS Firewall inspects DNS queries against the rules in a DNS View starting with the lowest numeric priority setting and going up. You can change a rule's priority at any time, for example to change the order of processing or make space for other rules.

Rule actions in DNS Firewall

When DNS Firewall finds a match between a DNS query and a domain specification in a rule, it applies the action that's specified in the rule to the query.

You are required to specify one of the following options in each rule that you create:

- **Allow** – Stop inspecting the query and permit it to go through. Not available for DNS Firewall Advanced.
- **Alert** – Stop inspecting the query, permit it to go through, and log an alert for the query in the Route 53 Resolver logs.
- **Block** – Discontinue inspection of the query, block it from going to its intended destination, and log the block action for the query in the Route 53 Resolver logs.

  Reply with the configured block response, from the following:

  - **NODATA** – Respond indicating that the query was successful, but no response is available for it.
  - **NXDOMAIN**– Respond indicating that the query's domain name doesn't exist.
  - **OVERRIDE**– Provide a custom override in the response. This option requires the following additional settings:
    - **Record value** – The custom DNS record to send back in response to the query.
    - **Record type**– The DNS record's type. This determines the format of the record value. This must be `CNAME`.
    - **Time to live in seconds**– The recommended amount of time for the DNS resolver or web browser to cache the override record and use it in response to this query, if it is received again. By default, this is zero, and the record isn't cached.

Managed Domain Lists for Route 53 Global Resolver

Managed Domain Lists contain domain names that are associated with malicious activity or other potential threats. AWS maintains these lists to enable Route 53 Global Resolver customers to check internet-bound DNS queries against them when using DNS Firewall.

Keeping up to date on the constantly changing threat landscape can be time consuming and expensive. Managed Domain Lists can save you time when you implement and use DNS Firewall on Global Resolver. AWS automatically updates the lists when new vulnerabilities and threats emerge.

Managed domain lists are categorized into Threat and Content categories, designed to help protect you from common web threats and also block query resolution to domain not safe-for-work.

As a best practice, before using a Managed Domain List in production, test it in a non-production environment, with the rule action set to `Alert`. Evaluate the rule using Amazon CloudWatch

metrics combined with DNS Firewall sampled requests or Global Resolver logs. When you're satisfied that the rule does what you want, change the action setting as needed.

Available AWS Managed Domain Lists

This section describes the Managed Domain Lists that are currently available for Global Resolver. AWS provides the following Managed Domain Lists, for all users of Global Resolver, classified by **Threat** or **Content** Type.

Threat Categories

Threat type
Malware
Botnet/Command and Control
Aggregate Threat List
Amazon GuardDuty Threat List
Phishing
Spam

Content Categories

Content type
Violence and Hate Speech
For Kids
Online Ads
Science
Family and Parenting
Pets

Content type
Career and Job Search
Religion
Lifestyle
Home and Garden
Criminal and Illegal Activities
Sports and Recreation
Vehicles
Financial Services
Real Estate
Hobbies and Interests
Travel
Food and Dining
Government and Legal
Education
Fashion
Health
Shopping
Adult and Mature Content
Technology and Internet
Business and Economy

Content type
News
Search Engines and Portals
Arts and Culture
Entertainment
Military
Social Networking
Proxy Avoidance
Redirect
Email
Translation
Child Abuse
Abortion
Gambling
Hacking
Marijuana
Cryptocurrency
Dating
Artificial Intelligence and Machine Learning
Parked Domains
Private IP Address

Managed Domain Lists cannot be downloaded or browsed. To protect intellectual property, you can't view or edit the individual domain specifications within the Managed Domain Lists. This restriction also helps to prevent malicious users from designing threats that specifically circumvent published lists.

Build lists of domains to block or allow

You can create your own domain lists to specify domain categories that you either don't find in the managed domain list offerings or that you prefer to handle on your own.

In addition to the procedures described in this section, in the console, you can create a domain list in the context of DNS Firewall rule management, when you create or update a rule.

Each domain specification in your domain list must satisfy the following requirements:

- It can optionally start with `*` (asterisk).
- With the exception of the optional starting asterisk and a period, as a delimiter between labels, it must only contain the following characters: `A-Z`, `a-z`, `0-9`, `-` (hyphen).
- It must be from 1-255 characters in length.

To create a domain list

1. In the Route 53 Global Resolver console, navigate to your Global Resolver.
2. Choose the **Domain lists** tab.
3. Choose **Create domain list**.
4. Provide a name and optional description for your domain list, along with any tags, and select **Create domain list**.
5. After the domain list is created and operational, you can begin adding domains to your domain list by selecting **Add domains**.
6. If you choose to **Upload a list of domains from an Amazon S3 bucket**, enter the URI of the Amazon S3 bucket where you created a domain list. This domain list should have one domain name per line.
7. Otherwise, enter your domain specifications in the text box, one per line.
8. Choose **Add domains**.

To delete a domain list

1. In the Route 53 Global Resolver console, navigate to your Global Resolver.
2. Choose the **Domain lists** tab.
3. Select the domain list that you want to delete, then choose **Delete**, and confirm the deletion.

DNS Firewall Advanced protections

DNS Firewall Advanced detects suspicious DNS queries based on known threat signatures in DNS queries. You can specify a threat type in a rule that you use in a DNS Firewall rule, associated with a DNS View.

DNS Firewall Advanced works by identifying suspicious DNS threat signatures by inspecting a range of key identifiers in the DNS payload including the timestamp of requests, frequency of request and responses, the DNS query strings, and the length, type or size of both outbound and inbound DNS queries. Based on the type of threat signature, you can configure policies to block, or simply log and alert on the query. By using an expanded set of threat identifiers, you can protect against DNS threats from domain sources that might yet be unclassified by threat intelligence feeds maintained by the broader security community.

Currently, DNS Firewall Advanced offers protections from:

- Domain Generation Algorithms (DGAs)

  DGAs are used by attackers to generate a large number of domains to launch malware attacks.
- Dictionary DGA

  Detect DGA attacks that use domain names associated with dictionary words in large numbers to perform malicious command and control DNS communications.
- DNS tunneling

  DNS tunneling is used by attackers to exfiltrate data from the client by using the DNS tunnel without making a network connection to the client.

To learn how to create rules, see the section called “Configure and manage DNS Firewall rules”.

Mitigating false positive scenarios

If you are encountering false-positive scenarios in rules that use DNS Firewall Advanced protections to block queries, perform the following steps:

1. In the Global Resolver logs, identify the rule and DNS Firewall Advanced protections that are causing the false positive. You do this by finding the log for the query that DNS Firewall is blocking, but that you want to allow through. The log record lists the DNS View, the rule, rule action, and the DNS Firewall Advanced protection.
2. Create a new rule in the DNS View that explicitly allows the blocked query through. When you create the rule, you can define your own domain list with just the domain specification that you want to allow. Follow the guidance for rule management at the section called “Configure and manage DNS Firewall rules”.
3. Prioritize the new rule inside the rule so that it runs before the rule that's using the managed list. To do this, give the new rule a lower numeric priority setting.

When you have updated your rules, the new rule explicitly allows the domain name that you want to allow before the blocking rule runs.

Manage DNS security policies in Route 53 Global Resolver

Managing global resolvers

After creating a global resolver, you can view its details, edit its configuration, and manage associated resources from the Global Resolvers page.

Viewing resolver details

The Global Resolvers page displays a list of all your resolvers with key information including resolver name, deployed regions, associated DNS views, observability region, and operational status.

Editing global resolvers

You can modify the resolver name and description after creation. You cannot modify the regions where a global resolver is deployed after creation.

Managing firewall rules

After creating firewall rules, you can modify their priority, update their configuration, or delete them as needed.

Rule priority and evaluation order

Firewall rules are evaluated in priority order, with lower numbers processed first. When a query matches multiple rules, only the first matching rule's action is applied.

Updating firewall rules

You can update most aspects of a firewall rule after creation, including its priority, action, and target domains.

Configuring private hosted zone associations with Route 53 Global Resolver

Route 53 Global Resolver enables you to resolve records associated with your Route 53 private hosted zones from your authorized clients. You can associate the Route 53 private hosted zones with the DNS view that your clients authorized with the DNS view is allowed to resolve.

Topics

- Manage private hosted zone associations with Route 53 Global Resolver

Manage private hosted zone associations with Route 53 Global Resolver

Creating private hosted zones for internal applications

Before you can associate a private hosted zone with a DNS view, you must first create the zone using the Amazon Route 53 console or API.

To create a private hosted zone

1. Open the Amazon Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose **Create hosted zone**.
4. For **Domain name**, enter the domain name for your private zone (for example, `internal.example.com`).
5. For **Type**, choose **Private hosted zone**.
6. For **VPCs to associate with the hosted zone**, choose the VPCs you want to associated with the hosted zone. Optionally, you can use the `CreateHostedZone` API operation to skip entering a value since you'll be associating the zone with Route 53 Global Resolver instead.

7. Choose **Create hosted zone**.

After creating the private hosted zone, add the DNS records you need for your internal services.

Associating private hosted zones with DNS views

To enable Route 53 Global Resolver to resolve queries for your private hosted zone, you must associate the zone with one or more DNS views.

To associate a private hosted zone with a DNS view

1. In the Route 53 Global Resolver console, navigate to your global resolver.
2. Choose the DNS view where you want to associate the private hosted zone.
3. In the **Private hosted zones** section, choose **Associate hosted zone**.
4. For **Hosted zone**, select the private hosted zone you want to associate.
5. For **Association name**, enter a descriptive name for this association.
6. Choose **Associate hosted zone**.

The association process typically takes a few minutes to complete. After this completes, Route 53 Global Resolver will use the records in the private hosted zone to answer DNS queries from client devices associated with the DNS view.

You can also associate a private hosted zone in your account with a DNS view that another AWS account has shared with you through AWS Resource Access Manager. For more information, see Sharing Route 53 Global Resolver DNS views between AWS accounts.

Sharing Route 53 Global Resolver DNS views between AWS accounts

You can share a Route 53 Global Resolver DNS view with other AWS accounts by using AWS Resource Access Manager (AWS RAM). AWS RAM lets you share your AWS resources with any AWS account or through AWS Organizations. For more information, see the Resource Access Manager User Guide.

When you share a DNS view, the receiving account (the *consumer*) can associate its own Route 53 private hosted zones with the view. The records in those zones then resolve through your global resolver. The consumer doesn't need to transfer ownership of the hosted zone or the DNS view. The

account that owns the DNS view (the *owner*) keeps full control of the global resolver and the DNS view. The owner can stop sharing the view or remove any association at any time.

> **Note**
>
> A shared DNS view applies the consumer's private hosted zone records to DNS resolution in every AWS Region where the owner's global resolver runs. Only share a DNS view with accounts that you trust.

Note the following:

Permissions that the owner grants to the consumer

When you share a DNS view, you choose a AWS RAM managed permission that defines what the consumer can do with the view. Route 53 Global Resolver provides the following AWS managed permissions for the `route53globalresolver:dns-view` resource type:

- **AWSRAMDefaultPermissionDNSView** (default) – Lets the consumer view the shared DNS view and associate its own private hosted zones with the view. This permission allows the `AssociateHostedZone` and `GetDNSView` actions.
- **AWSRAMPermissionDNSViewLifecycleManagement** – Lets the consumer view, update, enable, and disable the shared DNS view. Does not allow associating private hosted zones with the view.
- **AWSRAMPermissionDNSViewFullAccess** – Lets the consumer both associate private hosted zones and manage the lifecycle of the shared DNS view.

You can also create a customer managed permission that grants only some of these actions. For more information, see Creating customer managed permissions in the *AWS Resource Access Manager User Guide*.

Hosted zone associations are owned by the consumer

A private hosted zone association that a consumer creates on a shared DNS view belongs to the consumer's account. The owner of the DNS view can see and delete these associations. The consumer can manage the associations that it created.

Stopping sharing (unsharing) a DNS view

If you stop sharing a DNS view, the consumer can no longer create new hosted zone associations on the view or update the view. However, existing associations are *not* removed

automatically. They continue to affect DNS resolution until the owner or the consumer removes them. To stop a consumer's records from resolving right away, remove the consumer's hosted zone associations from the DNS view.

Deleting a shared DNS view

You can't delete a DNS view while it's shared. Stop sharing the view first, and then delete it.

Quotas

Hosted zone associations that a consumer creates on a shared DNS view count toward the owner's quota for associations per DNS view. For current Route 53 Global Resolver quotas, see Quotas for Route 53 Global Resolver.

Data in opt-in AWS Regions

The owner's global resolver might run in opt-in AWS Regions that the consumer hasn't turned on. When a consumer associates a private hosted zone with a shared DNS view, the hosted zone ID and domain name are copied to all Regions where the owner's global resolver runs. This includes opt-in Regions that the consumer hasn't turned on.

Sharing a DNS view that you own

To share a DNS view, you create a resource share in AWS RAM. A resource share specifies the resources to share, the managed permission to apply, and the principals (accounts, organizational units, or organization) to share with.

You can share a DNS view only if you own it. You can't share a DNS view that was shared with you.

- Console
- CLI

Sharing a DNS view (console)

To share a DNS view with another AWS account

1. Open the AWS RAM console at https://console.aws.amazon.com/ram/.

> **Note**
>
> A DNS view is a global resource. You must create the resource share in the US East (N. Virginia) (`us-east-1`) Region.

2. Choose **Create resource share**.
3. For **Name**, enter a descriptive name for the resource share.
4. Under **Resources**, choose the **Route 53 Global Resolver DNS Views** resource type, and then select the DNS view that you want to share.
5. For **Managed permissions**, choose the permission that grants the access you want. For more information, see the *Permissions that the owner grants to the consumer* list earlier in this topic.
6. For **Principals**, enter the AWS account ID, OU, or organization to share the DNS view with, and then choose **Add**.
7. Choose **Create resource share**.

If you share the DNS view with an account outside your organization in AWS Organizations, the consumer gets an invitation to the resource share. The consumer must accept the invitation before it can use the shared DNS view.

Sharing a DNS view (CLI)

To share a DNS view, use the AWS RAM `create-resource-share` command. Specify the ARN of the DNS view and the consumer account. Run the command in the US East (N. Virginia) (`us-east-1`) Region. The following example shares a DNS view with the account `222233334444` using the default managed permission.

```
aws ram create-resource-share --region us-east-1 --name MyDNSViewShare --
resource-arns arn:aws:route53globalresolver::111122223333:dns-view/dnsv-
abcdef1234567890 --principals 222233334444
```

To grant more access, include the `--permission-arns` parameter with the ARN of the `AWSRAMPermissionDNSViewLifecycleManagement` or `AWSRAMPermissionDNSViewFullAccess` managed permission. To find a permission's ARN, use the AWS RAM `list-permissions` command.

Using a DNS view that's shared with you

After another account shares a DNS view with you, accept any invitation that was sent. You can then associate your own Route 53 private hosted zones with the shared DNS view. The records in those hosted zones then resolve through the owner's global resolver for authorized clients.

To find DNS views shared with your account, open the AWS RAM console at https://console.aws.amazon.com/ram/ and choose **Shared with me**, **Shared resources**. To link a private hosted zone to a shared DNS view, follow the same steps as for a DNS view that you own. Specify the ARN of the shared DNS view. For more information, see Configuring private hosted zone associations with Route 53 Global Resolver.

Note the following about a DNS view that's shared with you:

- You can associate and disassociate your own private hosted zones. You can't modify or delete the DNS view itself unless the owner grants you a managed permission that allows those actions.
- The private hosted zone associations that you create belong to your account. You can view, update, and remove them.
- If the owner stops sharing the DNS view, you can no longer create new associations on the view. However, existing associations continue to work until you or the owner removes them.

Connecting a global resolver to a core network

A *core network endpoint* connects a global resolver to an AWS Cloud WAN (Cloud WAN) core network. After you connect the core network, you can use the core network's segments as access sources for DNS resolution. Clients that reach the global resolver from those segments are authorized the same way as other access sources that you configure.

A core network endpoint connects one global resolver to one Cloud WAN core network. At this time, a global resolver supports a single core network endpoint. For more information about Cloud WAN and core networks, see the AWS Cloud WAN section of the *AWS Network Manager User Guide*.

> **Note**
>
> Core networks use an attached core network policy. To enable closest-Region routing so that clients get the most out of a global resolver that you connect to a Cloud WAN core

network, update your core network policy to version `2026.10`. For more information, see Core network policy versions.

Note

When you connect a core network, the segments of that core network become available as access sources for every AWS Region where your global resolver runs. Only connect core networks that you trust as sources of DNS traffic.

Note the following:

Segments are used as access sources

After you connect a core network, its segments can be used as access sources for DNS resolution. You reference a segment as an access source in the same way that you reference an IP-based access source. For more information about access sources, see Managing access controls with access sources and tokens in Route 53 Global Resolver.

One endpoint connects one core network

Each core network endpoint connects one global resolver to one Cloud WAN core network.

Deleting a core network endpoint

When you delete a core network endpoint, the core network's segments are no longer available as access sources for the global resolver. Clients that relied on those segments for authorization can no longer resolve DNS queries through the global resolver. Update your access source configuration before you delete an endpoint that is in use.

Quotas

Core network endpoints are subject to Route 53 Global Resolver quotas. For current quotas, see Quotas for Route 53 Global Resolver.

How core network endpoints work

A Cloud WAN core network is a managed, global network that connects your Amazon Virtual Private Clouds, AWS Regions, and on-premises locations. A core network is divided into *segments*, which are isolated routing domains that group network traffic.

When you create a core network endpoint, the global resolver connects to the core network that you specify. Route 53 Global Resolver then makes the core network's segments available as access sources. You can reference a segment as an access source in your access control configuration, so that clients reaching the global resolver from that segment are authorized to resolve DNS queries. Because a global resolver runs across the AWS Regions that you choose, the connected segments are available as access sources in each of those Regions.

Managing core network endpoints

You can create a core network endpoint, view its details, list the endpoints for a global resolver, update an endpoint, and delete an endpoint. You can manage core network endpoints with the AWS Management Console, the AWS CLI, or the Route 53 Global Resolver API.

Create

Create a core network endpoint to connect a global resolver to a Cloud WAN core network. After you create the endpoint, the core network's segments can be used as access sources.

Get

Get a core network endpoint to view its details, such as the connected core network and the endpoint's current status.

List

List the core network endpoints to see every core network that a global resolver connects to.

Update

Update a core network endpoint to change its configuration.

Delete

Delete a core network endpoint to disconnect the global resolver from the core network. After you delete the endpoint, the core network's segments are no longer available as access sources.

Creating a core network endpoint

To create a core network endpoint, connect a global resolver to a Cloud WAN core network that you own. After you create the endpoint, the core network's segments can be used as access sources for DNS resolution.

> **Note**
>
> A core network endpoint requires a minimum of two AWS Regions, which must be among the Regions where the global resolver operates, and supports a maximum of five private IP addresses.

- Console
- CLI

Creating a core network endpoint (console)

To create a core network endpoint

1. Open the Route 53 Global Resolver console at https://console.aws.amazon.com/route53globalresolver/.
2. In the navigation pane, choose **Global resolvers**.
3. Select the global resolver that you want to connect to a core network.
4. Choose the **Core network endpoints** tab, and then choose **Create core network endpoint**.
5. For **Core network**, select the Cloud WAN core network to connect to.
6. Choose **Create core network endpoint**.

After the endpoint is created, you can reference the core network's segments as access sources. For more information about access sources, see Managing access controls with access sources and tokens in Route 53 Global Resolver.

Creating a core network endpoint (CLI)

You can also create and manage core network endpoints with the AWS CLI and the Route 53 Global Resolver API. Use the create operation to connect a global resolver to a core network, the get and list operations to view your endpoints, the update operation to change an endpoint, and the delete operation to disconnect a core network. For the command syntax and parameters, see the Route 53 Global Resolver section of the *AWS Command Line Interface Reference*.

Monitoring DNS activity and performance with Route 53 Global Resolver

Route 53 Global Resolver provides comprehensive visibility into DNS activity across your organization, enabling you to identify security threats, analyze client device behavior, and maintain compliance. This chapter covers both the monitoring tools available and detailed procedures for setting up DNS monitoring, configuring logging destinations, and analyzing DNS data to investigate threats and optimize performance.

AWS provides these monitoring tools to help you maintain secure, reliable DNS service:

- *Amazon CloudWatch Logs* enables you to monitor, store, and access your log files from Amazon EC2 instances, CloudTrail, and other sources. Route 53 Global Resolver can deliver DNS query logs directly to CloudWatch Logs for real-time monitoring and analysis. You can also archive your log data in highly durable storage. For more information, see the Amazon CloudWatch Logs User Guide.
  - You can use metric filters to search and filter log data coming into CloudWatch Logs and create CloudWatch metrics from the log events. Use these metrics to track DNS query volumes, response times, and security events. You can also create dashboards to monitor DNS performance across locations and set up alarms to notify you when query volumes spike or response times increase. For more information, see the Amazon CloudWatch User Guide.
- *Amazon EventBridge* can be used to automate your AWS services and respond automatically to system events, such as application availability issues or resource changes. Events from AWS services are delivered to EventBridge in near real time. You can write simple rules to indicate which events are of interest to you and which automated actions to take when an event matches a rule. For more information, see Amazon EventBridge User Guide.
- *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see the AWS CloudTrail User Guide.

For information about creating Amazon CloudWatch metrics, alarms, and analytics from Route 53 Global Resolver logs, see the section called “Monitoring with CloudWatch”.

Topics

- Gain visibility into DNS activity with Route 53 Global Resolver

- Configure DNS monitoring and logging with Route 53 Global Resolver
- Monitoring Route 53 Global Resolver with Amazon CloudWatch

Gain visibility into DNS activity with Route 53 Global Resolver

Route 53 Global Resolver provides comprehensive DNS query logging capabilities to monitor client device activity and identify security threats. Enable DNS query logging in Route 53 Global Resolver to see what websites client devices access, identify potential security threats, and analyze DNS resolution patterns. Logs capture comprehensive information about each query, including which security policies were applied.

What information is captured in DNS logs

Each DNS query log entry provides detailed information about client device activity and security policy enforcement:

- **Query information** - Domain name, query type, query class, and protocol used
- **Client device information** - Source IP address, DNS view, and authentication method
- **Response information** - Response code, answer records, and response time
- **Security actions** - Firewall rule matches, threat detection results, and actions taken
- **Metadata** - Timestamp, global resolver ID, Region, and trace information

OCSF format for security integration

DNS query logs use the Open Cybersecurity Schema Framework (OCSF), which provides a standardized format for security event data. This format enables:

- **Standardized analysis** - Consistent schema across different security tools
- **Improved interoperability** - Easy integration with SIEM and analytics platforms
- **Enhanced correlation** - Ability to correlate DNS events with other security data
- **Future compatibility** - Support for evolving security analysis requirements

OCSF log format examples

Route 53 Global Resolver DNS query logs follow the OCSF schema structure, providing detailed information about each DNS query, response, and security action. The following examples show the log format for allowed, denied, and alerted queries.

Route 53 Global Resolver DNS log - Example – Access allowed

This example shows a DNS query that was allowed through firewall rules. The log includes query details, response information, and enrichment data with Route 53 Global Resolver-specific identifiers.


```
{
    "action_id": 1,
    "action_name": "Allowed",
    "activity_id": 6,
    "activity_name": "Traffic",
    "category_name": "Network Activity",
    "category_uid": 4,
    "class_name": "DNS Activity",
    "class_uid": 4003,
    "cloud": {
        "provider": "AWS",
        "region": "us-east-1",
        "account": {
            "uid": "123456789012"
        }
    },
    "connection_info": {
        "direction": "Inbound",
        "direction_id": 1,
        "protocol_name": "udp",
        "protocol_num": 17,
        "protocol_ver_id": 4,
        "uid": "db21d1739ddb423a"
    },
    "disposition": "Allowed",
    "disposition_id": 1,
    "duration": 1,
    "end_time": 1761358379996,
    "answers": [{
        "rdata": "3.3.3.3",
        "type": "A",
```

```
        "class": "IN",
        "ttl": 300
    },
    {
        "rdata": "3.3.3.4",
        "type": "A",
        "class": "IN",
        "ttl": 300
    }],
    "src_endpoint": {
        "ip": "3.3.3.1",
        "port": 56576
    },
    "enrichments": [{
        "name": "global-resolver",
        "value": "gr-a1b2c3d4fexample",
        "data": {
            "dns_view_id": "dnsv-a1b2c3d4fexample",
            "firewall_rule_id": "fr-a1b2c3d4fexample",
            "firewall_domain_list_id": "fdl-a1b2c3d4fexample",
            "access_source_cidr": "3.3.3.0/24"
        }
    }],
    "message": "",
    "metadata": {
        "version": "1.2.0",
        "product": {
            "name": "Global Resolver",
            "vendor_name": "AWS",
            "feature": {
                "name": "DNS"
            }
        }
    },
    "query": {
        "hostname": "example.com.",
        "class": "IN",
        "type": "A",
        "opcode": "Query",
        "opcode_id": 0
    },
    "query_time": 1761358379995,
    "rcode": "NOERROR",
    "rcode_id": 0,
```

```
    "response_time": 1761358379995,
    "severity": "Informational",
    "severity_id": 1,
    "start_time": 1761358379995,
    "status": "Success",
    "status_id": 1,
    "time": 1761358379995,
    "type_name": "DNS Activity: Traffic",
    "type_uid": 400306
}
```


Route 53 Global Resolver DNS log - Example – Access denied

This example shows a DNS query that was blocked by firewall rules. The log includes the denial action, empty answers array, and REFUSED response code indicating the query was not processed.


```
{
    "action_id": 2,
    "action_name": "Denied",
    "activity_id": 6,
    "activity_name": "Traffic",
    "category_name": "Network Activity",
    "category_uid": 4,
    "class_name": "DNS Activity",
    "class_uid": 4003,
    "cloud": {
        "provider": "AWS",
        "region": "us-west-2",
        "account": {
            "uid": "123456789012"
        }
    },
    "connection_info": {
        "direction": "Inbound",
        "direction_id": 1,
        "protocol_name": "tcp",
        "protocol_num": 6,
        "protocol_ver_id": 4,
        "uid": "9fdc6fbc09794d5e"
    },
    "disposition": "Blocked",
    "disposition_id": 2,
    "duration": 1,
```

```
    "end_time": 1761358379996,
    "answers": [],
    "src_endpoint": {
        "ip": "3.3.3.3",
        "port": 28276
    },
    "enrichments": [
        {
            "name": "global-resolver",
            "value": "gr-a1b2c3d4fexample",
            "data": {
                "dns_view_id": "dnsv-a1b2c3d4fexample",
                "firewall_rule_id": "fr-a1b2c3d4fexample",
                "firewall_domain_list_id": "fdl-a1b2c3d4fexample",
                "token_id": "at-a1b2c3d4fexample",
                "token_name": "device-123456",
                "token_expiration_time": "1789419206",
                "token_ttl_days": 134
            }
        }
    ],
    "message": "",
    "metadata": {
        "version": "1.2.0",
        "product": {
            "name": "Global Resolver",
            "vendor_name": "AWS",
            "feature": {
                "name": "DNS"
            }
        }
    },
    "query": {
        "hostname": "example.com.",
        "class": "IN",
        "type": "A",
        "opcode": "Query",
        "opcode_id": 0
    },
    "query_time": 1761358379995,
    "rcode": "REFUSED",
    "rcode_id": 5,
    "response_time": 1761358379995,
    "severity": "Informational",
```

```
    "severity_id": 1,
    "start_time": 1761358379995,
    "status": "Failure",
    "status_id": 1,
    "time": 1761358379995,
    "type_name": "DNS Activity: Traffic",
    "type_uid": 400306
}
```


Route 53 Global Resolver DNS log - Example – Access alert

This example shows a DNS query that triggered an alert rather than being blocked. The log includes the `disposition` value of `Alert`, indicating the query was allowed through and resolved, but flagged for review.


```
{
    "action_id": 1,
    "action_name": "Allowed",
    "activity_id": 6,
    "activity_name": "Traffic",
    "category_name": "Network Activity",
    "category_uid": 4,
    "class_name": "DNS Activity",
    "class_uid": 4003,
    "cloud": {
        "provider": "AWS",
        "region": "eu-west-1",
        "account": {
            "uid": "123456789012"
        }
    },
    "connection_info": {
        "direction": "Inbound",
        "direction_id": 1,
        "protocol_name": "tcp",
        "protocol_num": 6,
        "protocol_ver_id": 4,
        "uid": "b7e4a1f2c6d84e39"
    },
    "disposition": "Alert",
    "disposition_id": 19,
    "duration": 1,
    "end_time": 1761358379996,
```

```
    "answers": [{
        "rdata": "3.3.3.3",
        "type": "A",
        "class": "IN",
        "ttl": 300
    },
    {
        "rdata": "3.3.3.4",
        "type": "A",
        "class": "IN",
        "ttl": 300
    }],
    "src_endpoint": {
        "ip": "3.3.3.2",
        "port": 58492
    },
    "enrichments": [{
        "name": "global-resolver",
        "value": "gr-a1b2c3d4fexample",
        "data": {
            "dns_view_id": "dnsv-a1b2c3d4fexample",
            "firewall_rule_id": "fr-a1b2c3d4fexample",
            "firewall_domain_list_id": "fdl-a1b2c3d4fexample",
            "access_source_cidr": "3.3.3.2/32"
        }
    }],
    "message": "",
    "metadata": {
        "version": "1.2.0",
        "product": {
            "name": "Global Resolver",
            "vendor_name": "AWS",
            "feature": {
                "name": "DNS"
            }
        }
    },
    "query": {
        "hostname": "example.com.",
        "class": "IN",
        "type": "A",
        "opcode": "Query",
        "opcode_id": 0
    },
```

```
    "query_time": 1761358379995,
    "rcode": "NOERROR",
    "rcode_id": 0,
    "response_time": 1761358379995,
    "severity": "Informational",
    "severity_id": 1,
    "start_time": 1761358379995,
    "status": "Success",
    "status_id": 1,
    "time": 1761358379995,
    "type_name": "DNS Activity: Traffic",
    "type_uid": 400306
}
```


Configure DNS monitoring and logging with Route 53 Global Resolver

Configure DNS monitoring in Route 53 Global Resolver to capture detailed information about DNS queries, responses, and security actions. This section covers the steps to set up logging destinations and configure monitoring tools.

Setting the observability Region

Before configuring DNS logging, you must set an observability Region where logs and metrics will be stored. This region determines where your monitoring data is processed and stored.

1. Open the Route 53 Global Resolver console at https://console.aws.amazon.com/route53globalresolver/.
2. In the navigation pane, choose **Settings**.
3. In the **Observability region** section, choose **Set region**.
4. Select the AWS Region where you want to store monitoring data, then choose **Set region**.

After setting the observability region, you can configure log delivery destinations in that Region.

After you configure DNS monitoring and log delivery, you can create Amazon CloudWatch metrics and alarms from your Route 53 Global Resolver logs. For more information, see the section called "Monitoring with CloudWatch".

Monitoring Route 53 Global Resolver with Amazon CloudWatch

Route 53 Global Resolver DNS publishes query log data directly to Amazon CloudWatch Logs. With CloudWatch Logs, you can search and analyze your log data, create metric filters that define patterns to look for in the data, and set alarms that send notifications based on those metric filters. For more information about Amazon CloudWatch Logs, see What is Amazon CloudWatch Logs?

You can process Route 53 Global Resolver DNS query log records as you would with any other log events that CloudWatch Logs collects. For more information about monitoring log data and metric filters, see Creating metrics from log events using filters in the *Amazon CloudWatch Logs User Guide*.

Topics

- Example: Create a Amazon CloudWatch metric filter and alarm for blocked DNS queries
- Example metric filters
- IAM role for publishing CloudWatch metrics from DNS query logs
- Using Amazon CloudWatch Contributor Insights with Route 53 Global Resolver data
- Analyzing Route 53 Global Resolver logs with Amazon CloudWatch Logs Insights

Example: Create a Amazon CloudWatch metric filter and alarm for blocked DNS queries

The following example shows you how to create a metric filter that counts blocked DNS queries in your Route 53 Global Resolver logs. It also shows you how to create an alarm that notifies you when 10 or more blocked queries occur within a 1-hour period.

Step 1: Create the metric filter

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. In the navigation pane, choose **Logs**, then **Log groups**.
3. Select your Route 53 Global Resolver log group, which is located in the observability Region that you set for Route 53 Global Resolver, and then choose **Actions**, **Create metric filter**.
4. For **Filter pattern**, enter the following. This matches any log entry in which a firewall rule blocked the DNS query: `{ $.disposition = "Blocked" }`

5. To verify the pattern works, select a log stream under **Select log data to test** and choose **Test pattern**.
6. Choose **Next**.
7. Provide a filter name, set the metric namespace to `Route53GlobalResolver`, and provide a metric name.
8. Set the metric value to 1 so that each blocked query increments the count.
9. Choose **Dimensions** and add a custom dimension. The dimension name and value can be one of the following.

   Metric filter dimensions for Route 53 Global Resolver logs

Dimension name	Value
`DNSViewId`	`$.enrichments[0].data.dns_view_id`
`AccessTokenId`	`$.enrichments[0].data.token_id`
`AccessSourceCidr`	`$.enrichments[0].data.access_source_cidr`
`GlobalResolverId`	`$.enrichments[0].value`

10. Choose **Next**, then **Create metric filter**.

Step 2: Create the alarm

1. In the navigation pane, choose **Alarms**, **All alarms**.
2. Choose **Create alarm**.
3. Find and select the metric you created, then choose **Select metric**.
4. Configure the alarm as follows, then choose **Next**:
   - For **Statistic**, choose **Sum** to count all blocked queries in the window.
   - For **Period**, choose **1 hour**.
   - For **Whenever**, choose **Greater/Equal** and enter 10 for the threshold.
   - For **Additional configuration**, **Datapoints to alarm**, leave the default of 1.
5. Choose or create an Amazon SNS topic to receive the notification. Choose **Next**.

6. Enter a name and description for the alarm and choose **Next**.
7. Review the configuration and choose **Create alarm**.

Example metric filters

This section contains example metric filter patterns and Amazon CloudWatch Logs Insights queries for Route 53 Global Resolver DNS query logs.

Blocked queries by Global Resolver

```
{
 "logGroupName": "/aws/vendedlogs/route53globalresolver/global-resolver/
GLOBAL_RESOLVER_LOGS/gr-<resolverId>",
 "filterName": "BlockedQueriesByGlobalResolverId",
 "filterPattern": "{ $.disposition = \"Blocked\" }",
 "metricTransformations": [
   {
     "metricName": "BlockedQueriesByGlobalResolverId",
     "metricNamespace": "Route53GlobalResolver",
     "metricValue": "1",
     "dimensions": { "GlobalResolverId": "$.enrichments[0].value" }
   }
 ]
}
```

Allowed queries by Access Source

```
{
 "logGroupName": "/aws/vendedlogs/route53globalresolver/global-resolver/
GLOBAL_RESOLVER_LOGS/gr-<resolverId>",
 "filterName": "AllowedQueriesByAccessSourceCidr",
 "filterPattern": "{ $.disposition = \"Allowed\" }",
 "metricTransformations": [
   {
     "metricName": "AllowedQueriesByAccessSourceCidr",
     "metricNamespace": "Route53GlobalResolver",
     "metricValue": "1",
     "dimensions": { "AccessSourceCidr": "$.enrichments[0].data.access_source_cidr" }
   }
 ]
}
```

Alert queries by Access Token

```
{
 "logGroupName": "/aws/vendedlogs/route53globalresolver/global-resolver/
GLOBAL_RESOLVER_LOGS/gr-<resolverId>",
 "filterName": "AlertQueriesByAccessTokenId",
 "filterPattern": "{ $.disposition = \"Alert\" }",
 "metricTransformations": [
   {
     "metricName": "AlertQueriesByAccessTokenId",
     "metricNamespace": "Route53GlobalResolver",
     "metricValue": "1",
     "dimensions": { "AccessTokenId": "$.enrichments[0].data.token_id" }
   }
 ]
}
```

Total queries by DNS View

```
{
 "logGroupName": "/aws/vendedlogs/route53globalresolver/global-resolver/
GLOBAL_RESOLVER_LOGS/gr-<resolverId>",
 "filterName": "TotalQueriesByDNSViewId",
 "filterPattern": "{ $.query_time = * }",
 "metricTransformations": [
   {
     "metricName": "TotalQueriesByDNSViewId",
     "metricNamespace": "Route53GlobalResolver",
     "metricValue": "1",
     "dimensions": { "DNSViewId": "$.enrichments[0].data.dns_view_id" }
   }
 ]
}
```

SERVFAIL by Global Resolver

```
{
 "logGroupName": "/aws/vendedlogs/route53globalresolver/global-resolver/
GLOBAL_RESOLVER_LOGS/gr-<resolverId>",
 "filterName": "ServfailQueriesByGlobalResolverId",
 "filterPattern": "{ $.rcode = \"SERVFAIL\" }",
 "metricTransformations": [
```

```
   {
     "metricName": "ServfailQueriesByGlobalResolverId",
     "metricNamespace": "Route53GlobalResolver",
     "metricValue": "1",
     "dimensions": { "GlobalResolverId": "$.enrichments[0].value" }
   }
 ]
}
```

NXDOMAIN by Access Source

```
{
 "logGroupName": "/aws/vendedlogs/route53globalresolver/global-resolver/
GLOBAL_RESOLVER_LOGS/gr-<resolverId>",
 "filterName": "NxdomainQueriesByAccessSourceCidr",
 "filterPattern": "{ $.rcode = \"NXDOMAIN\" }",
 "metricTransformations": [
   {
     "metricName": "NxdomainQueriesByAccessSourceCidr",
     "metricNamespace": "Route53GlobalResolver",
     "metricValue": "1",
     "dimensions": { "AccessSourceCidr": "$.enrichments[0].data.access_source_cidr" }
   }
 ]
}
```

NXDOMAIN by Access Token

```
{
 "logGroupName": "/aws/vendedlogs/route53globalresolver/global-resolver/
GLOBAL_RESOLVER_LOGS/gr-<resolverId>",
 "filterName": "NxdomainQueriesByAccessTokenId",
 "filterPattern": "{ $.rcode = \"NXDOMAIN\" }",
 "metricTransformations": [
   {
     "metricName": "NxdomainQueriesByAccessTokenId",
     "metricNamespace": "Route53GlobalResolver",
     "metricValue": "1",
     "dimensions": { "AccessTokenId": "$.enrichments[0].data.token_id" }
   }
 ]
}
```

NOERROR by DNS View

```
{
 "logGroupName": "/aws/vendedlogs/route53globalresolver/global-resolver/
GLOBAL_RESOLVER_LOGS/gr-<resolverId>",
 "filterName": "NoerrorQueriesByDNSViewId",
 "filterPattern": "{ $.rcode = \"NOERROR\" }",
 "metricTransformations": [
   {
     "metricName": "NoerrorQueriesByDNSViewId",
     "metricNamespace": "Route53GlobalResolver",
     "metricValue": "1",
     "dimensions": { "DNSViewId": "$.enrichments[0].data.dns_view_id" }
   }
 ]
}
```

Amazon CloudWatch Logs Insights queries for P90 response time

P90 Response Time by Global Resolver

```
fields @timestamp,
      enrichments.0.value as GlobalResolverId,
      (response_time - query_time) as ResponseTime
| filter ispresent(response_time) and ispresent(query_time)
| stats pct(ResponseTime, 90) as P90ResponseTime by GlobalResolverId
```

Days until token expiry by Access Token

```
fields @timestamp,
      enrichments.0.data.token_id as AccessTokenId,
      enrichments.0.data.token_ttl_days as DaysUntilExpiry
| filter ispresent(enrichments.0.data.token_ttl_days)
| stats latest(DaysUntilExpiry) by AccessTokenId
```

IAM role for publishing CloudWatch metrics from DNS query logs

The AWS Identity and Access Management (IAM) policy that's attached to your IAM role or user must include at least the following permissions.

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [
        {
            "Sid": "CloudWatchMetricFilterPermissions",
            "Effect": "Allow",
            "Action": [
                "logs:PutMetricFilter",
                "logs:DescribeMetricFilters",
                "logs:DeleteMetricFilter",
                "logs:TestMetricFilter"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CloudWatchLogsInsightsPermissions",
            "Effect": "Allow",
            "Action": [
                "logs:StartQuery",
                "logs:StopQuery",
                "logs:GetQueryResults",
                "logs:DescribeQueries",
                "logs:DescribeLogGroups"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CloudWatchMetricsNamespacePermissions",
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData",
                "cloudwatch:GetMetricData",
                "cloudwatch:ListMetrics"
            ],
            "Resource": "*"
        }
    ]
}
```

Using Amazon CloudWatch Contributor Insights with Route 53 Global Resolver data

Amazon CloudWatch Contributor Insights analyzes log data to create time series that display contributor data. You can see metrics about the top-N contributors, the total number of unique contributors, and their usage. This information helps you find the highest-impact contributors to a

system. With Route 53 Global Resolver DNS query logs, you can use Contributor Insights to identify your top query sources, the most-blocked domains, the noisiest access tokens, and other high-impact contributors to your DNS traffic. For more information, see Using Contributor Insights to analyze high-cardinality data.

1. Before you begin, configure DNS query logging for your global resolver to deliver query logs to Amazon CloudWatch Logs. For more information, see the section called "Configure DNS monitoring".
2. In the console, navigate to **Route 53**, **Route 53 Global Resolver**, then **Global Resolvers**. Select the global resolver you want, and then choose the **Analytics** tab.

   > **Note**
   >
   > You can access both Contributor Insights and Logs Insights from the **Analytics** tab. You can also select an individual rule group to view its analytics.

3. Choose **Contributor Insights**, then **Create rule**. For more information, see Create a Contributor Insights rule.
4. Choose the log group or log groups that contain your Route 53 Global Resolver query logs.
5. For **Rule type**, choose **Sample rule**. Then choose **Select sample rule**, and choose one of the Route 53 Global Resolver Logs sample rules.

   > **Note**
   >
   > Include the resource ID in the rule name so that you can identify which resource the rule applies to. The resource ID can be a DNS view ID, Amazon VPC ID, account ID, and so on.

6. Selecting one of the sample rules populates all the required fields on the CloudWatch rule page.
7. To view the resulting metrics, open the CloudWatch console, choose **Metrics**, then **All metrics**, and look for the Route 53 Resolver namespace, which includes DNS Firewall metrics.

Analyzing Route 53 Global Resolver logs with Amazon CloudWatch Logs Insights

With Amazon CloudWatch Logs Insights, you can interactively search and analyze your Route 53 Global Resolver log data with queries. You can run queries to respond to operational issues

and identify patterns in your DNS traffic. For more information, see Analyzing log data with CloudWatch Logs Insights.

1. Before you begin, configure DNS query logging for your global resolver to deliver query logs to Amazon CloudWatch Logs. For more information, see the section called “Configure DNS monitoring”.
2. In the console, navigate to **Route 53**, **Route 53 Global Resolver**, then **Global Resolvers**. Select the global resolver you want, choose the **Analytics** tab, and then choose **Logs Insights**.
3. Select one or more log groups that you want to analyze. When you configure DNS query logging, your global resolver delivers query logs to these log groups.
4. Choose the dimensions to analyze. You can choose from two dimensions: **Metric** and **Action**. Alternatively, you can enter your own CloudWatch Logs Insights query to analyze the log data in other ways.

Troubleshooting DNS issues with Route 53 Global Resolver

Route 53 Global Resolver provides comprehensive DNS resolution capabilities, but occasionally you might need to troubleshoot connectivity, performance, or configuration issues. Use DNS logs, monitoring data, and diagnostic techniques to identify and resolve issues affecting client device DNS resolution with Route 53 Global Resolver. This chapter provides systematic approaches to troubleshooting common DNS problems and optimizing performance.

Topics

- Diagnose DNS connectivity issues with Route 53 Global Resolver
- Resolve DNS performance issues with Route 53 Global Resolver
- Troubleshoot configuration issues in Route 53 Global Resolver
- Troubleshooting internal resource access

Diagnose DNS connectivity issues with Route 53 Global Resolver

Route 53 Global Resolver provides reliable DNS resolution, but connectivity issues can occasionally occur due to configuration, authentication, or network problems. When client devices cannot resolve domain names using Route 53 Global Resolver, use these systematic approaches to identify and resolve connectivity problems.

Client devices cannot resolve domains

Follow these steps to diagnose resolution failures:

1. **Verify queries reach the global resolver**
   - Check DNS query logs for queries from the affected client device IP address
   - Confirm the client device is configured with the correct anycast IP addresses
   - Test network connectivity from the client device to the anycast IPs
2. **Check client device authentication**
   - Verify the client device is authenticated to the correct DNS view
   - Check Access Source rules for the client device's IP address or CIDR block
   - Confirm access tokens are valid and not expired (for token-based authentication)
3. **Review firewall rules**
   - Check if firewall rules are blocking the queries
   - Review rule priority and make sure that allow rules have higher priority than block rules
   - Verify firewall fail-open behavior settings
4. **Confirm DNS view associations**
   - Verify private hosted zone associations for internal domains
   - Check that DNS records exist in the associated private hosted zones
   - Ensure domain names in queries exactly match zone names

Intermittent resolution failures

For sporadic DNS resolution issues, investigate these potential causes:

Authentication issues

- Check for access token expiration and renewal patterns
- Review authentication logs for failed authentication attempts
- Verify client device clock synchronization for token validation

Network connectivity

- Monitor for network path changes or routing issues
- Check for firewall or NAT device configuration changes
- Verify consistent anycast routing to the nearest Region

Service health

- Check AWS Service Health Dashboard for Route 53 Global Resolver issues
- Review CloudWatch metrics for error rate spikes
- Monitor private hosted zone association status

Unexpected public resolution

When queries resolve to public DNS instead of private hosted zones:

1. **Verify private hosted zone configuration**
   - Confirm the private hosted zone contains the expected DNS records
   - Check that record names exactly match the queried domain
   - Verify record types match the query type (A, AAAA, CNAME, etc.)
2. **Check DNS view associations**
   - Verify the private hosted zone is associated with the correct DNS view
   - Confirm the client device is authenticated to the DNS view
   - Check association status in the console
3. **Review firewall rules**
   - Check for firewall rules that might be blocking private zone queries
   - Verify rule evaluation order and priority
   - Review DNS query logs for firewall actions

Resolve DNS performance issues with Route 53 Global Resolver

Route 53 Global Resolver is designed for optimal performance with global anycast architecture, but various factors can affect DNS resolution speed. Address slow DNS resolution and optimize query response times for better client device performance using Route 53 Global Resolver.

Slow DNS resolution

To diagnose and resolve slow DNS response times:

1. **Analyze query response times**
   - Use DNS query logs to identify queries with high response times

- Compare response times across different domains and query types
- Monitor response time trends over time

2. **Check for high query volumes**
   - Monitor CloudWatch metrics for query volume spikes
   - Identify client devices or DNS views generating excessive queries
   - Look for query patterns that might indicate misconfigurations
3. **Review cache performance**
   - Analyze cache hit rates for frequently queried domains
   - Review TTL settings for DNS records
   - Consider adjusting TTL values based on query patterns
4. **Verify optimal Region selection**
   - Ensure global resolver regions are close to client device locations
   - Monitor anycast routing to confirm queries reach the nearest Region
   - Consider adding regions if client devices are geographically distant

Performance optimization strategies

Use these strategies to optimize DNS performance:

Cache optimization

- Adjust TTL values based on query patterns and change frequency
- Use longer TTLs for stable records, shorter TTLs for frequently changing records
- Monitor cache hit rates and adjust TTLs accordingly

Region optimization

- Deploy global resolver in regions closest to client device concentrations
- Monitor query routing patterns and response times by region
- Consider adding regions for better geographic coverage

Protocol optimization

- Choose appropriate DNS protocols based on security and performance requirements
- Consider DNS-over-HTTPS (DoH) for encrypted connections with caching benefits
- Use DNS-over-TLS (DoT) for encrypted connections with lower overhead

Rule optimization

- Review and optimize firewall rule priority and complexity
- Place frequently matched rules higher in priority order
- Simplify complex rule conditions where possible

Troubleshoot configuration issues in Route 53 Global Resolver

Route 53 Global Resolver offers extensive configuration options for DNS views, authentication, and firewall rules, which can sometimes lead to configuration conflicts or mismatches. Identify and resolve common Route 53 Global Resolver configuration problems that affect DNS resolution.

Authentication problems

Common authentication issues and solutions:

Access Source rule mismatches

- Verify client device IP addresses match configured CIDR blocks
- Check for NAT or proxy devices changing source IP addresses
- Ensure Access Source rules cover all expected client device IP ranges

Token authentication failures

- Verify tokens are correctly configured on client devices
- Check token expiration dates and renewal processes
- Ensure client device clocks are synchronized for token validation

Protocol mismatches

- Verify client devices use protocols allowed by Access Source rules
- Check that DoH and DoT configurations match token protocols
- Ensure firewall rules don't block required protocols

DNS view configuration issues

Common DNS view configuration problems:

Incorrect DNS view associations

- Verify client devices are authenticated to the intended DNS view

- Check that private hosted zones are associated with the correct DNS views
- Review firewall rules applied to each DNS view

DNS settings conflicts

- Review DNSSEC validation settings for compatibility with client devices
- Check EDNS Client Subnet settings for privacy and performance balance
- Verify firewall fail-open behavior aligns with security requirements

Firewall rule issues

Common firewall rule configuration problems:

Rule priority conflicts

- Review rule evaluation order and verify correct priority assignment
- Check for block rules with higher priority than intended allow rules
- Test rule changes in a controlled environment before production deployment

Domain list mismatches

- Verify domain specifications in custom domain lists
- Check wildcard patterns for correct syntax and coverage
- Ensure domain lists are updated and synchronized

Troubleshooting internal resource access

Common issues and solutions when managing internal resource access:

Client devices not resolving to private zone records

- Verify the private hosted zone is associated with the correct DNS view
- Check that the client device is authenticated to the correct DNS view
- Ensure the domain name in the query exactly matches the zone name
- Verify the DNS records exist in the private hosted zone

Intermittent resolution failures

- Check the association status in the console
- Review DNS query logs for error patterns
- Verify network connectivity between Route 53 Global Resolver and Amazon Route 53

Unexpected public resolution

- Confirm the private hosted zone contains the expected records
- Check for firewall rules that might be blocking the query
- Verify the query is coming from a client device associated with the DNS view

Quotas for Route 53 Global Resolver

Your AWS account has default quotas, formerly referred to as limits, for each AWS service. Unless otherwise noted, each quota is Region-specific. You can request increases for some quotas. Other quotas can't be increased.

To view the quotas for Route 53 Global Resolver, open the Service Quotas console. In the navigation pane, choose **AWS services**, and then select **Route 53 Global Resolver**.

To request a quota increase, see Requesting a Quota Increase in the *Service Quotas User Guide*. If the quota is not yet available in Service Quotas, use the limit increase form.

Your AWS account has the following quotas related to Route 53 Global Resolver.

Soft quotas

The following table describes quotas in Route 53 Global Resolver that can be increased. To change quotas, see Requesting a Quota Increase in the *Service Quotas User Guide*.

For some customers, your account quota might be below these published quotas. If you got a *Resource limit exceeded* error wrongfully, use the Service Quotas console to request a quota increase.

Global Resolver soft quotas

Resource	Default quota	Adjustable
Global resolvers per account	2	Yes
DNS views per global resolver	5	Yes
DNS views per account	50	Yes
Domain lists per account	1,000	Yes

Resource	Default quota	Adjustable
Domains per domain list	100,000	Yes
Domains per account	100,000	Yes
Domains in a file imported from S3	10,000	Yes
Number of domains referenced across all Firewall Rules	1,000,000	Yes
Firewall rules per DNS view	100	Yes
Access tokens per global resolver	5,000	Yes
Access Sources per global resolver	1,000	Yes
Access Sources CIDR size per global resolver	65,000	No
IPv6 access source /48 blocks per global resolver	1	Yes
Private hosted zones per DNS view	1,000	Yes

Creating Amazon Route 53 and Route 53 VPC Resolver resources with AWS CloudFormation

Amazon Route 53 and Route 53 VPC Resolver are integrated with AWS CloudFormation, a service that helps you to model and set up your AWS resources so that you can spend less time creating and managing your resources and infrastructure. You create a template that describes all the AWS resources that you want, and CloudFormation provisions and configures those resources for you.

When you use CloudFormation, you can reuse your template to set up your Route 53 and VPC Resolver resources consistently and repeatedly. Describe your resources once, and then provision the same resources over and over in multiple AWS accounts and Regions.

Route 53, VPC Resolver, and CloudFormation templates

To provision and configure resources for Route 53, VPC Resolver, and related services, you must understand CloudFormation templates. Templates are formatted text files in JSON or YAML. These templates describe the resources that you want to provision in your CloudFormation stacks. If you're unfamiliar with JSON or YAML, you can use CloudFormation Designer to help you get started with CloudFormation templates. For more information, see What is CloudFormation Designer? in the *AWS CloudFormation User Guide*.

Route 53 supports creating the following resource types in CloudFormation:

- `AWS::Route53::DNSSEC`
- `AWS::Route53::HealthCheck`
- `AWS::Route53::HostedZone`
- `AWS::Route53::KeySigningKey`
- `AWS::Route53::RecordSet`
- `AWS::Route53::RecordSetGroup`

For more information, including examples of JSON and YAML templates for Route 53 resources, see the Amazon Route 53 resource type reference in the *AWS CloudFormation User Guide*.

VPC Resolver supports creating the following resource types in CloudFormation:

- `AWS::Route53Resolver::FirewallDomainList`

- `AWS::Route53Resolver::FirewallDomainList`
- `AWS::Route53Resolver::FirewallRuleGroupAssociation`
- `AWS::Route53Resolver::ResolverDNSSECConfig`
- `AWS::Route53Resolver::ResolverEndpoint`
- `AWS::Route53Resolver::ResolverQueryLoggingConfig`
- `AWS::Route53Resolver::ResolverQueryLoggingConfigAssociation`
- `AWS::Route53Resolver::ResolverRule`
- `AWS::Route53Resolver::ResolverRuleAssociation`

For more information, including examples of JSON and YAML templates for VPC Resolver resources, see the Route 53 VPC Resolver resource type reference in the *AWS CloudFormation User Guide*.

Best practices for Route 53 and CloudFormation

When using CloudFormation to manage Route 53 resources, follow these best practices to avoid common issues and ensure reliable deployments.

Understanding eventual consistency

Route 53 uses an eventually consistent model for DNS changes. This can affect CloudFormation operations, particularly during rollbacks and rapid successive changes.

> **Important**
>
> When CloudFormation attempts to roll back DNS record changes, the rollback might fail due to the Route 53 eventual consistency model. If CloudFormation tries to recreate a record that was recently deleted but still appears to exist due to eventual consistency, you might encounter `InvalidChangeBatch` errors that leave your DNS in a broken state.

To minimize issues related to eventual consistency:

- **Plan changes carefully** - Avoid making rapid successive changes to the same DNS records
- **Test in non-production first** - Always test DNS changes in development environments before applying to production

- **Monitor deployments** - Watch CloudFormation stack events closely during DNS-related deployments. For monitoring guidance, see Monitoring Amazon Route 53.
- **Have rollback procedures ready** - Prepare manual recovery procedures in case automatic rollbacks fail

DNS record ordering and logical IDs

When creating multiple DNS records in CloudFormation, be careful about record ordering and logical ID assignment.

> **Warning**
>
> If you define DNS records in arrays or lists within your CloudFormation template, inserting new records in the middle of the list can cause CloudFormation to reassign logical IDs to existing records. This triggers record replacements that can lead to service disruptions and rollback failures.

Best practices for DNS record management:

- **Use explicit logical IDs** - Always assign explicit, meaningful logical IDs to DNS records rather than relying on array indices. For more information about CloudFormation logical IDs, see Resources section structure in the *AWS CloudFormation User Guide*
- **Append new records** - When adding new DNS records to existing lists, append them to the end rather than inserting them in the middle
- **Group related records** - Consider using `AWS::Route53::RecordSetGroup` for managing related records together. For more information, see AWS::Route53::RecordSetGroup in the *AWS CloudFormation User Guide*.
- **Review change sets** - Always review CloudFormation change sets before deployment to identify unexpected record replacements. For more information, see Updating stacks using change sets in the *AWS CloudFormation User Guide*.

Handling rollback failures

If a CloudFormation rollback fails due to DNS-related issues, you might need to perform manual recovery.

To perform manual recovery from failed DNS rollbacks

1. Identify the failed DNS records by reviewing CloudFormation stack events and Route 53 hosted zone records
2. Manually create or update the missing DNS records through the Route 53 console or API. For information about creating records, see Working with records.
3. After DNS is restored, update your CloudFormation template to match the current state
4. Deploy the corrected template to bring CloudFormation back in sync with the actual resources

To prevent rollback failures:

- Avoid making changes that could trigger DNS record replacements during high-traffic periods
- Implement health checks and monitoring to detect DNS issues quickly. For information about health checks, see Creating and updating health checks.
- Consider using blue-green deployment strategies for critical DNS changes. For more information about deployment best practices, see Best practices for Amazon Route 53.
- Document emergency procedures for manual DNS recovery

Learn more about CloudFormation

To learn more about CloudFormation, see the following resources:

- AWS CloudFormation
- AWS CloudFormation User Guide
- CloudFormation API Reference
- AWS CloudFormation Command Line Interface User Guide

Code examples for Route 53 using AWS SDKs

The following code examples show how to use Route 53 with an AWS software development kit (SDK).

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Code examples

- Code examples for Route 53 using AWS SDKs
  - Basic examples for Route 53 using AWS SDKs
    - Actions for Route 53 using AWS SDKs
      - Use ChangeResourceRecordSets with a CLI
      - Use CreateHostedZone with a CLI
      - Use DeleteHostedZone with a CLI
      - Use GetHostedZone with a CLI
      - Use ListHostedZones with an AWS SDK or CLI
      - Use ListHostedZonesByName with a CLI
      - Use ListQueryLoggingConfigs with a CLI
- Code examples for Route 53 domain registration using AWS SDKs
  - Basic examples for Route 53 domain registration using AWS SDKs
    - Hello Route 53 domain registration
    - Learn the basics of Route 53 domain registration with an AWS SDK
    - Actions for Route 53 domain registration using AWS SDKs
      - Use CheckDomainAvailability with an AWS SDK or CLI
      - Use CheckDomainTransferability with an AWS SDK or CLI
      - Use GetDomainDetail with an AWS SDK or CLI
      - Use GetDomainSuggestions with an AWS SDK or CLI
      - Use GetOperationDetail with an AWS SDK or CLI
      - Use ListDomains with an AWS SDK or CLI
      - Use ListOperations with an AWS SDK or CLI

- Use ListPrices with an AWS SDK
- Use RegisterDomain with an AWS SDK or CLI
- Use ViewBilling with an AWS SDK or CLI

Code examples for Route 53 using AWS SDKs

The following code examples show how to use Route 53 with an AWS software development kit (SDK).

Actions are code excerpts from larger programs and must be run in context. While actions show you how to call individual service functions, you can see actions in context in their related scenarios.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Route 53

- Basic examples for Route 53 using AWS SDKs
  - Actions for Route 53 using AWS SDKs
    - Use ChangeResourceRecordSets with a CLI
    - Use CreateHostedZone with a CLI
    - Use DeleteHostedZone with a CLI
    - Use GetHostedZone with a CLI
    - Use ListHostedZones with an AWS SDK or CLI
    - Use ListHostedZonesByName with a CLI
    - Use ListQueryLoggingConfigs with a CLI

Basic examples for Route 53 using AWS SDKs

The following code examples show how to use the basics of Amazon Route 53 with AWS SDKs.

Examples

- Actions for Route 53 using AWS SDKs
  - Use ChangeResourceRecordSets with a CLI
  - Use CreateHostedZone with a CLI

- Use DeleteHostedZone with a CLI
- Use GetHostedZone with a CLI
- Use ListHostedZones with an AWS SDK or CLI
- Use ListHostedZonesByName with a CLI
- Use ListQueryLoggingConfigs with a CLI

Actions for Route 53 using AWS SDKs

The following code examples demonstrate how to perform individual Route 53 actions with AWS SDKs. Each example includes a link to GitHub, where you can find instructions for setting up and running the code.

The following examples include only the most commonly used actions. For a complete list, see the Amazon Route 53 API Reference.

Examples

- Use ChangeResourceRecordSets with a CLI
- Use CreateHostedZone with a CLI
- Use DeleteHostedZone with a CLI
- Use GetHostedZone with a CLI
- Use ListHostedZones with an AWS SDK or CLI
- Use ListHostedZonesByName with a CLI
- Use ListQueryLoggingConfigs with a CLI

Use `ChangeResourceRecordSets` with a CLI

The following code examples show how to use `ChangeResourceRecordSets`.

CLI

AWS CLI

To create, update, or delete a resource record set

The following `change-resource-record-sets` example creates a resource record set using the `hosted-zone-id` `Z1R8UBAEXAMPLE` and the JSON-formatted configuration in the file `C:\awscli\route53\change-resource-record-sets.json`.

```
aws route53 change-resource-record-sets \
    --hosted-zone-id Z1R8UBAEXAMPLE \
    --change-batch file://C:\awscli\route53\change-resource-record-sets.json
```

Contents of `C:\awscli\route53\change-resource-record-sets.json`:

```
{
    "Comment": "Create a basic A record for www.example.com",
    "Changes": [
        {
            "Action": "CREATE",
            "ResourceRecordSet": {
                "Name": "www.example.com",
                "Type": "A",
                "TTL": 300,
                "ResourceRecords": [
                    {
                        "Value": "192.0.2.44"
                    }
                ]
            }
        }
    ]
}
```

Output:

```
{
    "ChangeInfo": {
        "Id": "/change/CWPIK4URU2I5S",
        "Status": "PENDING",
        "SubmittedAt": "2025-01-15T18:21:11.930000+00:00"
    }
}
```

The configuration in the JSON file depends on the kind of resource record set you want to create:

BasicWeightedAliasWeighted AliasLatencyLatency AliasFailoverFailover Alias

Basic Syntax:

```
{
    "Comment": "optional comment about the changes in this change batch request",
    "Changes": [
        {
            "Action": "CREATE"|"DELETE"|"UPSERT",
            "ResourceRecordSet": {
                "Name": "DNS domain name",
                "Type":
 "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
                "TTL": time to live in seconds,
                "ResourceRecords": [
                    {
                        "Value": "applicable value for the record type"
                    },
                    {...}
                ]
            }
        },
        {...}
    ]
}
```

Weighted Syntax:

```
{
    "Comment": "optional comment about the changes in this change batch request",
    "Changes": [
        {
            "Action": "CREATE"|"DELETE"|"UPSERT",
            "ResourceRecordSet": {
                "Name": "DNS domain name",
                "Type":
 "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
                "SetIdentifier": "unique description for this resource record
 set",
                "Weight": value between 0 and 255,
                "TTL": time to live in seconds,
                "ResourceRecords": [
                    {
                        "Value": "applicable value for the record type"
                    },
                    {...}
                ],
```

```
                "HealthCheckId": "optional ID of an Amazon Route 53 health check"
            }
        },
        {...}
    ]
}
```

Alias Syntax:

```
{
    "Comment": "optional comment about the changes in this change batch request",
    "Changes": [
        {
            "Action": "CREATE"|"DELETE"|"UPSERT",
            "ResourceRecordSet": {
                "Name": "DNS domain name",
                "Type":
 "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
                "AliasTarget": {
                    "HostedZoneId": "hosted zone ID for your CloudFront
 distribution, Amazon S3 bucket, Elastic Load Balancing load balancer, or Amazon
 Route 53 hosted zone",
                    "DNSName": "DNS domain name for your CloudFront distribution,
 Amazon S3 bucket, Elastic Load Balancing load balancer, or another resource
 record set in this hosted zone",
                    "EvaluateTargetHealth": true|false
                },
                "HealthCheckId": "optional ID of an Amazon Route 53 health check"
            }
        },
        {...}
    ]
}
```

Weighted Alias Syntax:

```
{
    "Comment": "optional comment about the changes in this change batch request",
    "Changes": [
        {
            "Action": "CREATE"|"DELETE"|"UPSERT",
            "ResourceRecordSet": {
                "Name": "DNS domain name",
```

```
                "Type":
 "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
                "SetIdentifier": "unique description for this resource record
 set",
                "Weight": value between 0 and 255,
                "AliasTarget": {
                    "HostedZoneId": "hosted zone ID for your CloudFront
 distribution, Amazon S3 bucket, Elastic Load Balancing load balancer, or Amazon
 Route 53 hosted zone",
                    "DNSName": "DNS domain name for your CloudFront distribution,
 Amazon S3 bucket, Elastic Load Balancing load balancer, or another resource
 record set in this hosted zone",
                    "EvaluateTargetHealth": true|false
                },
                "HealthCheckId": "optional ID of an Amazon Route 53 health check"
            }
        },
        {...}
    ]
}
```

Latency Syntax:

```
{
    "Comment": "optional comment about the changes in this change batch request",
    "Changes": [
        {
            "Action": "CREATE"|"DELETE"|"UPSERT",
            "ResourceRecordSet": {
                "Name": "DNS domain name",
                "Type":
 "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
                "SetIdentifier": "unique description for this resource record
 set",
                "Region": "Amazon EC2 region name",
                "TTL": time to live in seconds,
                "ResourceRecords": [
                    {
                        "Value": "applicable value for the record type"
                    },
                    {...}
                ],
                "HealthCheckId": "optional ID of an Amazon Route 53 health check"
```

```
            }
        },
        {...}
    ]
}
```

Latency Alias Syntax:

```
{
    "Comment": "optional comment about the changes in this change batch request",
    "Changes": [
        {
            "Action": "CREATE"|"DELETE"|"UPSERT",
            "ResourceRecordSet": {
                "Name": "DNS domain name",
                "Type":
 "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
                "SetIdentifier": "unique description for this resource record
 set",
                "Region": "Amazon EC2 region name",
                "AliasTarget": {
                    "HostedZoneId": "hosted zone ID for your CloudFront
 distribution, Amazon S3 bucket, Elastic Load Balancing load balancer, or Amazon
 Route 53 hosted zone",
                    "DNSName": "DNS domain name for your CloudFront distribution,
 Amazon S3 bucket, Elastic Load Balancing load balancer, or another resource
 record set in this hosted zone",
                    "EvaluateTargetHealth": true|false
                },
                "HealthCheckId": "optional ID of an Amazon Route 53 health check"
            }
        },
        {...}
    ]
}
```

Failover Syntax:

```
{
    "Comment": "optional comment about the changes in this change batch request",
    "Changes": [
        {
            "Action": "CREATE"|"DELETE"|"UPSERT",
```

```
            "ResourceRecordSet": {
                "Name": "DNS domain name",
                "Type":
 "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
                "SetIdentifier": "unique description for this resource record
 set",
                "Failover": "PRIMARY" | "SECONDARY",
                "TTL": time to live in seconds,
                "ResourceRecords": [
                    {
                        "Value": "applicable value for the record type"
                    },
                    {...}
                ],
                "HealthCheckId": "ID of an Amazon Route 53 health check"
            }
        },
        {...}
    ]
}
```

Failover Alias Syntax:

```
{
    "Comment": "optional comment about the changes in this change batch request",
    "Changes": [
        {
            "Action": "CREATE"|"DELETE"|"UPSERT",
            "ResourceRecordSet": {
                "Name": "DNS domain name",
                "Type":
 "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
                "SetIdentifier": "unique description for this resource record
 set",
                "Failover": "PRIMARY" | "SECONDARY",
                "AliasTarget": {
                    "HostedZoneId": "hosted zone ID for your CloudFront
 distribution, Amazon S3 bucket, Elastic Load Balancing load balancer, or Amazon
 Route 53 hosted zone",
                    "DNSName": "DNS domain name for your CloudFront distribution,
 Amazon S3 bucket, Elastic Load Balancing load balancer, or another resource
 record set in this hosted zone",
                    "EvaluateTargetHealth": true|false
```

```
                },
                "HealthCheckId": "optional ID of an Amazon Route 53 health check"
            }
        },
        {...}
    ]
}
```

For more information, see Working with records in the *Amazon Route 53 Developer Guide*.

- For API details, see ChangeResourceRecordSets in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: This example creates an A record for www.example.com and changes the A record for test.example.com from 192.0.2.3 to 192.0.2.1. Note that values for changes TXT-type records must be in double quotes. See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "www.example.com"
$change1.ResourceRecordSet.Type = "TXT"
$change1.ResourceRecordSet.TTL = 600
$change1.ResourceRecordSet.ResourceRecords.Add(@{Value="item 1 item 2 item 3"})

$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "DELETE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change2.ResourceRecordSet.Name = "test.example.com"
$change2.ResourceRecordSet.Type = "A"
$change2.ResourceRecordSet.TTL = 600
$change2.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.3"})

$change3 = New-Object Amazon.Route53.Model.Change
$change3.Action = "CREATE"
$change3.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change3.ResourceRecordSet.Name = "test.example.com"
$change3.ResourceRecordSet.Type = "A"
```

```
$change3.ResourceRecordSet.TTL = 600
$change3.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.1"})

$params = @{
    HostedZoneId="Z1PA6795UKMFR9"
 ChangeBatch_Comment="This change batch creates a TXT record for www.example.com.
 and changes the A record for test.example.com. from 192.0.2.3 to 192.0.2.1."
 ChangeBatch_Change=$change1,$change2,$change3
}

Edit-R53ResourceRecordSet @params
```

Example 2: This example shows how to create alias resource record sets. 'Z222222222' is the ID of the Amazon Route 53 hosted zone in which you're creating the alias resource record set. 'example.com' is the zone apex for which you want to create an alias and 'www.example.com' is a subdomain for which you also want to create an alias. 'Z1111111111111' is an example of a hosted zone ID for the load balancer and 'example-load-balancer-1111111111.us-east-1.elb.amazonaws.com' is an example of a load balancer domain name with which Amazon Route 53 responds to queries for example.com and www.example.com. See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "example.com"
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change1.ResourceRecordSet.AliasTarget.HostedZoneId = "Z1111111111111"
$change1.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-1111111111.us-east-1.elb.amazonaws.com."
$change1.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $true

$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "CREATE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "www.example.com"
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
```

```
$change1.ResourceRecordSet.AliasTarget.HostedZoneId = "Z1111111111111"
$change1.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-1111111111.us-east-1.elb.amazonaws.com."
$change1.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $false

$params = @{
    HostedZoneId="Z222222222"
 ChangeBatch_Comment="This change batch creates two alias resource record sets,
 one for the zone apex, example.com, and one for www.example.com, that both point
 to example-load-balancer-1111111111.us-east-1.elb.amazonaws.com."
 ChangeBatch_Change=$change1,$change2
}

Edit-R53ResourceRecordSet @params
```

Example 3: This example creates two A records for www.example.com. One-fourth of the time (1/(1+3)), Amazon Route 53 responds to queries for www.example.com with the two values for the first resource record set (192.0.2.9 and 192.0.2.10). Three-fourths of the time (3/(1+3)) Amazon Route 53 responds to queries for www.example.com with the two values for the second resource record set (192.0.2.11 and 192.0.2.12). See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "www.example.com"
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.SetIdentifier = "Rack 2, Positions 4 and 5"
$change1.ResourceRecordSet.Weight = 1
$change1.ResourceRecordSet.TTL = 600
$change1.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.9"})
$change1.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.10"})

$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "CREATE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change2.ResourceRecordSet.Name = "www.example.com"
$change2.ResourceRecordSet.Type = "A"
$change2.ResourceRecordSet.SetIdentifier = "Rack 5, Positions 1 and 2"
$change2.ResourceRecordSet.Weight = 3
$change2.ResourceRecordSet.TTL = 600
```

```
$change2.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.11"})
$change2.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.12"})

$params = @{
    HostedZoneId="Z1PA6795UKMFR9"
    ChangeBatch_Comment="This change creates two weighted resource record sets,
 each of which has two values."
    ChangeBatch_Change=$change1,$change2
}

Edit-R53ResourceRecordSet @params
```

Example 4: This example shows how to create weighted alias resource record sets assuming that example.com is the domain for which you want to create weighted alias resource record sets. SetIdentifier differentiates the two weighted alias resource record sets from one another. This element is required because the Name and Type elements have the same values for both resource record sets. Z1111111111111 and Z3333333333333 are examples of hosted zone IDs for the ELB load balancer specified by the value of DNSName. example-load-balancer-2222222222.us-east-1.elb.amazonaws.com and example-load-balancer-4444444444.us-east-1.elb.amazonaws.com are examples of Elastic Load Balancing domains from which Amazon Route 53 responds to queries for example.com. See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "example.com"
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.SetIdentifier = "1"
$change1.ResourceRecordSet.Weight = 3
$change1.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change1.ResourceRecordSet.AliasTarget.HostedZoneId = "Z1111111111111"
$change1.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-2222222222.us-east-1.elb.amazonaws.com."
$change1.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $true

$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "CREATE"
```

```
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change2.ResourceRecordSet.Name = "example.com"
$change2.ResourceRecordSet.Type = "A"
$change2.ResourceRecordSet.SetIdentifier = "2"
$change2.ResourceRecordSet.Weight = 1
$change2.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change2.ResourceRecordSet.AliasTarget.HostedZoneId = "Z3333333333333"
$change2.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-4444444444.us-east-1.elb.amazonaws.com."
$change2.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $false

$params = @{
    HostedZoneId="Z5555555555"
    ChangeBatch_Comment="This change batch creates two weighted alias resource
 record sets. Amazon Route 53 responds to queries for example.com with the first
 ELB domain 3/4ths of the times and the second one 1/4th of the time."
    ChangeBatch_Change=$change1,$change2
}

Edit-R53ResourceRecordSet @params
```

Example 5: This example creates two latency alias resource record sets, one for an ELB load balancer in the US West (Oregon) region (us-west-2), and another for a load balancer in the Asia Pacific (Singapore) region (ap-southeast-1). See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "example.com"
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.SetIdentifier = "Oregon load balancer 1"
$change1.ResourceRecordSet.Region = us-west-2
$change1.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change1.ResourceRecordSet.AliasTarget.HostedZoneId = "Z1111111111111"
$change1.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-2222222222.us-west-2.elb.amazonaws.com"
$change1.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $true

$change2 = New-Object Amazon.Route53.Model.Change
```

```
$change2.Action = "CREATE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change2.ResourceRecordSet.Name = "example.com"
$change2.ResourceRecordSet.Type = "A"
$change2.ResourceRecordSet.SetIdentifier = "Singapore load balancer 1"
$change2.ResourceRecordSet.Region = ap-southeast-1
$change2.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change2.ResourceRecordSet.AliasTarget.HostedZoneId = "Z2222222222222"
$change2.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-1111111111.ap-southeast-1.elb.amazonaws.com"
$change2.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $true

$params = @{
    HostedZoneId="Z5555555555"
    ChangeBatch_Comment="This change batch creates two latency resource
 record sets, one for the US West (Oregon) region and one for the Asia Pacific
 (Singapore) region."
    ChangeBatch_Change=$change1,$change2
}

Edit-R53ResourceRecordSet @params
```

- For API details, see ChangeResourceRecordSets in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: This example creates an A record for www.example.com and changes the A record for test.example.com from 192.0.2.3 to 192.0.2.1. Note that values for changes TXT-type records must be in double quotes. See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "www.example.com"
$change1.ResourceRecordSet.Type = "TXT"
$change1.ResourceRecordSet.TTL = 600
$change1.ResourceRecordSet.ResourceRecords = @()
$change1.ResourceRecordSet.ResourceRecords.Add(@{Value="item 1 item 2 item 3"})
```

```
$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "DELETE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change2.ResourceRecordSet.Name = "test.example.com"
$change2.ResourceRecordSet.Type = "A"
$change2.ResourceRecordSet.TTL = 600
$change2.ResourceRecordSet.ResourceRecords = @()
$change2.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.3"})

$change3 = New-Object Amazon.Route53.Model.Change
$change3.Action = "CREATE"
$change3.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change3.ResourceRecordSet.Name = "test.example.com"
$change3.ResourceRecordSet.Type = "A"
$change3.ResourceRecordSet.TTL = 600
$change3.ResourceRecordSet.ResourceRecords = @()
$change3.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.1"})

$params = @{
    HostedZoneId="Z1PA6795UKMFR9"
 ChangeBatch_Comment="This change batch creates a TXT record for www.example.com.
 and changes the A record for test.example.com. from 192.0.2.3 to 192.0.2.1."
 ChangeBatch_Change=$change1,$change2,$change3
}

Edit-R53ResourceRecordSet @params
```

Example 2: This example shows how to create alias resource record sets. 'Z222222222' is the ID of the Amazon Route 53 hosted zone in which you're creating the alias resource record set. 'example.com' is the zone apex for which you want to create an alias and 'www.example.com' is a subdomain for which you also want to create an alias. 'Z1111111111111' is an example of a hosted zone ID for the load balancer and 'example-load-balancer-1111111111.us-east-1.elb.amazonaws.com' is an example of a load balancer domain name with which Amazon Route 53 responds to queries for example.com and www.example.com. See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "example.com"
```

```
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change1.ResourceRecordSet.AliasTarget.HostedZoneId = "Z1111111111111"
$change1.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-1111111111.us-east-1.elb.amazonaws.com."
$change1.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $true

$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "CREATE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "www.example.com"
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change1.ResourceRecordSet.AliasTarget.HostedZoneId = "Z1111111111111"
$change1.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-1111111111.us-east-1.elb.amazonaws.com."
$change1.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $false

$params = @{
    HostedZoneId="Z222222222"
 ChangeBatch_Comment="This change batch creates two alias resource record sets,
 one for the zone apex, example.com, and one for www.example.com, that both point
 to example-load-balancer-1111111111.us-east-1.elb.amazonaws.com."
 ChangeBatch_Change=$change1,$change2
}

Edit-R53ResourceRecordSet @params
```

Example 3: This example creates two A records for www.example.com. One-fourth of the time (1/(1+3)), Amazon Route 53 responds to queries for www.example.com with the two values for the first resource record set (192.0.2.9 and 192.0.2.10). Three-fourths of the time (3/(1+3)) Amazon Route 53 responds to queries for www.example.com with the two values for the second resource record set (192.0.2.11 and 192.0.2.12). See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "www.example.com"
```

```
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.SetIdentifier = "Rack 2, Positions 4 and 5"
$change1.ResourceRecordSet.Weight = 1
$change1.ResourceRecordSet.TTL = 600
$change1.ResourceRecordSet.ResourceRecords = @()
$change1.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.9"})
$change1.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.10"})

$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "CREATE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change2.ResourceRecordSet.Name = "www.example.com"
$change2.ResourceRecordSet.Type = "A"
$change2.ResourceRecordSet.SetIdentifier = "Rack 5, Positions 1 and 2"
$change2.ResourceRecordSet.Weight = 3
$change2.ResourceRecordSet.TTL = 600
$change2.ResourceRecordSet.ResourceRecords = @()
$change2.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.11"})
$change2.ResourceRecordSet.ResourceRecords.Add(@{Value="192.0.2.12"})

$params = @{
    HostedZoneId="Z1PA6795UKMFR9"
    ChangeBatch_Comment="This change creates two weighted resource record sets,
 each of which has two values."
    ChangeBatch_Change=$change1,$change2
}

Edit-R53ResourceRecordSet @params
```

Example 4: This example shows how to create weighted alias resource record sets assuming that example.com is the domain for which you want to create weighted alias resource record sets. SetIdentifier differentiates the two weighted alias resource record sets from one another. This element is required because the Name and Type elements have the same values for both resource record sets. Z1111111111111 and Z3333333333333 are examples of hosted zone IDs for the ELB load balancer specified by the value of DNSName. example-load-balancer-2222222222.us-east-1.elb.amazonaws.com and example-load-balancer-4444444444.us-east-1.elb.amazonaws.com are examples of Elastic Load Balancing domains from which Amazon Route 53 responds to queries for example.com. See the Amazon Route 53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "example.com"
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.SetIdentifier = "1"
$change1.ResourceRecordSet.Weight = 3
$change1.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change1.ResourceRecordSet.AliasTarget.HostedZoneId = "Z1111111111111"
$change1.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-2222222222.us-east-1.elb.amazonaws.com."
$change1.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $true

$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "CREATE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change2.ResourceRecordSet.Name = "example.com"
$change2.ResourceRecordSet.Type = "A"
$change2.ResourceRecordSet.SetIdentifier = "2"
$change2.ResourceRecordSet.Weight = 1
$change2.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change2.ResourceRecordSet.AliasTarget.HostedZoneId = "Z3333333333333"
$change2.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-4444444444.us-east-1.elb.amazonaws.com."
$change2.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $false

$params = @{
    HostedZoneId="Z5555555555"
    ChangeBatch_Comment="This change batch creates two weighted alias resource
 record sets. Amazon Route 53 responds to queries for example.com with the first
 ELB domain 3/4ths of the times and the second one 1/4th of the time."
    ChangeBatch_Change=$change1,$change2
}

Edit-R53ResourceRecordSet @params
```

Example 5: This example creates two latency alias resource record sets, one for an ELB load balancer in the US West (Oregon) region (us-west-2), and another for a load balancer in the Asia Pacific (Singapore) region (ap-southeast-1). See the Amazon Route

53 documentation for more details. You can use the Get-R53Change cmdlet to poll to determine when the changes are complete.

```
$change1 = New-Object Amazon.Route53.Model.Change
$change1.Action = "CREATE"
$change1.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change1.ResourceRecordSet.Name = "example.com"
$change1.ResourceRecordSet.Type = "A"
$change1.ResourceRecordSet.SetIdentifier = "Oregon load balancer 1"
$change1.ResourceRecordSet.Region = us-west-2
$change1.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change1.ResourceRecordSet.AliasTarget.HostedZoneId = "Z1111111111111"
$change1.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-2222222222.us-west-2.elb.amazonaws.com"
$change1.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $true

$change2 = New-Object Amazon.Route53.Model.Change
$change2.Action = "CREATE"
$change2.ResourceRecordSet = New-Object Amazon.Route53.Model.ResourceRecordSet
$change2.ResourceRecordSet.Name = "example.com"
$change2.ResourceRecordSet.Type = "A"
$change2.ResourceRecordSet.SetIdentifier = "Singapore load balancer 1"
$change2.ResourceRecordSet.Region = ap-southeast-1
$change2.ResourceRecordSet.AliasTarget = New-Object
 Amazon.Route53.Model.AliasTarget
$change2.ResourceRecordSet.AliasTarget.HostedZoneId = "Z2222222222222"
$change2.ResourceRecordSet.AliasTarget.DNSName = "example-load-
balancer-1111111111.ap-southeast-1.elb.amazonaws.com"
$change2.ResourceRecordSet.AliasTarget.EvaluateTargetHealth = $true

$params = @{
    HostedZoneId="Z5555555555"
    ChangeBatch_Comment="This change batch creates two latency resource
 record sets, one for the US West (Oregon) region and one for the Asia Pacific
 (Singapore) region."
    ChangeBatch_Change=$change1,$change2
}

Edit-R53ResourceRecordSet @params
```

- For API details, see ChangeResourceRecordSets in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `CreateHostedZone` with a CLI

The following code examples show how to use `CreateHostedZone`.

CLI

AWS CLI

To create a hosted zone

The following `create-hosted-zone` example adds a hosted zone named `example.com` using the caller reference `2014-04-01-18:47`. The optional comment includes a space, so it must be enclosed in quotation marks.

```
aws route53 create-hosted-zone \
    --name example.com \
    --caller-reference 2014-04-01-18:47 \
    --hosted-zone-config Comment="command-line version"
```

Output:

```
{
    "Location": "https://route53.amazonaws.com/2013-04-01/hostedzone/
Z1D633PJEXAMPLE",
    "HostedZone": {
        "Id": "/hostedzone/Z1D633PJEXAMPLE",
        "Name": "example.com.",
        "CallerReference": "2014-04-01-18:47",
        "Config": {
            "Comment": "command-line version",
            "PrivateZone": false
        },
        "ResourceRecordSetCount": 2
    },
    "ChangeInfo": {
        "Id": "/change/C1PA6795EXAMPLE",
        "Status": "PENDING",
        "SubmittedAt": "2014-04-01T18:47:10.123Z"
    },
```

```
    "DelegationSet": {
        "NameServers": [
            "ns-2048.awsdns-64.com",
            "ns-2049.awsdns-65.net",
            "ns-2050.awsdns-66.org",
            "ns-2051.awsdns-67.co.uk"
        ]
    }
}
```

For more information, see Creating a public hosted zone in the *Amazon Route 53 Developer Guide*.

- For API details, see CreateHostedZone in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Creates a new hosted zone named 'example.com', associated with a reusable delegation set. Note that you must supply a value for the CallerReference parameter so that requests that need to be retried if necessary without the risk of executing the operation twice. Because the hosted zone is being created in a VPC it is automatically private and you should not set the -HostedZoneConfig_PrivateZone parameter.

```
$params = @{
    Name="example.com"
    CallerReference="myUniqueIdentifier"
    HostedZoneConfig_Comment="This is my first hosted zone"
    DelegationSetId="NZ8X2CISAMPLE"
    VPC_VPCId="vpc-1a2b3c4d"
    VPC_VPCRegion="us-east-1"
}

New-R53HostedZone @params
```

- For API details, see CreateHostedZone in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Creates a new hosted zone named 'example.com', associated with a reusable delegation set. Note that you must supply a value for the CallerReference parameter

so that requests that need to be retried if necessary without the risk of executing the operation twice. Because the hosted zone is being created in a VPC it is automatically private and you should not set the -HostedZoneConfig_PrivateZone parameter.

```
$params = @{
    Name="example.com"
    CallerReference="myUniqueIdentifier"
    HostedZoneConfig_Comment="This is my first hosted zone"
    DelegationSetId="NZ8X2CISAMPLE"
    VPC_VPCId="vpc-1a2b3c4d"
    VPC_VPCRegion="us-east-1"
}

New-R53HostedZone @params
```

- For API details, see CreateHostedZone in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `DeleteHostedZone` with a CLI

The following code examples show how to use `DeleteHostedZone`.

CLI

AWS CLI

To delete a hosted zone

The following `delete-hosted-zone` example deletes the hosted zone with an `id` of `Z36KTIQEXAMPLE`.

```
aws route53 delete-hosted-zone \
    --id Z36KTIQEXAMPLE
```

Output:

```
{
```

```
    "ChangeInfo": {
        "Id": "/change/C2682N5HEXAMPLE",
        "Status": "PENDING",
        "SubmittedAt": "2014-04-01T19:12:45.384Z"
    }
}
```

For more information, see Deleting a public hosted zone in the *Amazon Route 53 Developer Guide*.

- For API details, see DeleteHostedZone in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Deletes the hosted zone with the specified ID. You will be prompted for confirmation before the command proceeds unless you add the -Force switch parameter.

```
Remove-R53HostedZone -Id Z1PA6795UKMFR9
```

- For API details, see DeleteHostedZone in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Deletes the hosted zone with the specified ID. You will be prompted for confirmation before the command proceeds unless you add the -Force switch parameter.

```
Remove-R53HostedZone -Id Z1PA6795UKMFR9
```

- For API details, see DeleteHostedZone in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `GetHostedZone` with a CLI

The following code examples show how to use `GetHostedZone`.

CLI

AWS CLI

To get information about a hosted zone

The following `get-hosted-zone` example gets information about the hosted zone with an `id` of `Z1R8UBAEXAMPLE`.

```
aws route53 get-hosted-zone \
    --id Z1R8UBAEXAMPLE
```

Output:

```
{
    "HostedZone": {
        "Id": "/hostedzone/Z1R8UBAEXAMPLE",
        "Name": "example.com.",
        "CallerReference": "2014-04-01-18:47",
        "Config": {
            "Comment": "command-line version",
            "PrivateZone": false
        },
        "ResourceRecordSetCount": 2
    },
    "DelegationSet": {
        "NameServers": [
            "ns-2048.awsdns-64.com",
            "ns-2049.awsdns-65.net",
            "ns-2050.awsdns-66.org",
            "ns-2051.awsdns-67.co.uk"
        ]
    }
}
```

For more information, see Working with hosted zones in the *Amazon Route 53 Developer Guide*.

- For API details, see GetHostedZone in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns details of the hosted zone with ID Z1D633PJN98FT9.

```
Get-R53HostedZone -Id Z1D633PJN98FT9
```

- For API details, see GetHostedZone in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns details of the hosted zone with ID Z1D633PJN98FT9.

```
Get-R53HostedZone -Id Z1D633PJN98FT9
```

- For API details, see GetHostedZone in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `ListHostedZones` with an AWS SDK or CLI

The following code examples show how to use `ListHostedZones`.

CLI

AWS CLI

To list the hosted zones associated with the current AWS account

The following `list-hosted-zones` example lists summary information about the first 100 hosted zones that are associated with the current AWS account.

```
aws route53 list-hosted-zones
```

Output:

```
{
    "HostedZones": [
```

```
        {
            "Id": "/hostedzone/Z1R8UBAEXAMPLE",
            "Name": "example.com.",
            "CallerReference": "2014-04-01-18:47",
            "Config": {
                "Comment": "command-line version",
                "PrivateZone": false
            },
            "ResourceRecordSetCount": 2
        }
    ]
}
```

If you have more than 100 hosted zones, or if you want to list them in groups smaller than 100, include the `--max-items` parameter. For example, to list hosted zones one at a time, use the following command.

```
aws route53 list-hosted-zones \
    --max-items 1
```

To view information about the next hosted zone, take the value of `NextToken` from the response to the previous command, and include it in the `--starting-token` parameter, as shown in the following example.

```
aws route53 list-hosted-zones \
    --max-items 1 \
    --starting-token Z3M3LMPEXAMPLE
```

For more information, see Listing public hosted zones in the *Amazon Route 53 Developer Guide*.

- For API details, see ListHostedZones in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Outputs all of your public and private hosted zones.

```
Get-R53HostedZoneList
```

Example 2: Outputs all of the hosted zones that are associated with the reusable delegation set that has the ID NZ8X2CISAMPLE

```
Get-R53HostedZoneList -DelegationSetId NZ8X2CISAMPLE
```

- For API details, see ListHostedZones in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Outputs all of your public and private hosted zones.

```
Get-R53HostedZoneList
```

Example 2: Outputs all of the hosted zones that are associated with the reusable delegation set that has the ID NZ8X2CISAMPLE

```
Get-R53HostedZoneList -DelegationSetId NZ8X2CISAMPLE
```

- For API details, see ListHostedZones in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

Rust

SDK for Rust

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
async fn show_host_info(client: &aws_sdk_route53::Client) -> Result<(),
 aws_sdk_route53::Error> {
    let hosted_zone_count = client.get_hosted_zone_count().send().await?;

    println!(
        "Number of hosted zones in region : {}",
        hosted_zone_count.hosted_zone_count(),
    );
```

```
    let hosted_zones = client.list_hosted_zones().send().await?;

    println!("Zones:");

    for hz in hosted_zones.hosted_zones() {
        let zone_name = hz.name();
        let zone_id = hz.id();

        println!("  ID :   {}", zone_id);
        println!("  Name : {}", zone_name);
        println!();
    }

    Ok(())
}
```

- For API details, see ListHostedZones in *AWS SDK for Rust API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `ListHostedZonesByName` with a CLI

The following code examples show how to use `ListHostedZonesByName`.

CLI

AWS CLI

Example 1: To list hosted zones ordered by domain name

The following `list-hosted-zones-by-name` example lists up to 100 hosted zones ordered by domain name.

```
aws route53 list-hosted-zones-by-name
```

Output:

```
{
```

```
    "HostedZones": [
        {
            "ResourceRecordSetCount": 2,
            "CallerReference": "test20150527-2",
            "Config": {
                "Comment": "test2",
                "PrivateZone": false
            },
            "Id": "/hostedzone/Z119WBBTVP5WFX",
            "Name": "2.example.com."
        },
        {
            "ResourceRecordSetCount": 2,
            "CallerReference": "test20150527-1",
            "Config": {
                "Comment": "test",
                "PrivateZone": false
            },
            "Id": "/hostedzone/Z3P5QSUBK4POTI",
            "Name": "www.example.com."
        }
    ],
    "IsTruncated": false,
    "MaxItems": "100"
}
```

Example 2: To list hosted zones beginning with a specified domain name

The following `list-hosted-zones-by-name` example lists hosted zones ordered by name, beginning with `www.example.com`.

```
aws route53 list-hosted-zones-by-name \
    --dns-name www.example.com
```

Output:

```
{
    "HostedZones": [
        {
            "ResourceRecordSetCount": 2,
            "CallerReference": "mwunderl20150527-1",
            "Config": {
```

```
                "Comment": "test",
                "PrivateZone": false
            },
            "Id": "/hostedzone/Z3P5QSUBK4POTI",
            "Name": "www.example.com."
        }
    ],
    "DNSName": "www.example.com",
    "IsTruncated": false,
    "MaxItems": "100"
}
```

For more information, see Listing public hosted zones in the *Amazon Route 53 Developer Guide*.

- For API details, see ListHostedZonesByName in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: Returns all of your public and private hosted zones in ASCII order by domain name.

```
Get-R53HostedZonesByName
```

Example 2: Returns your public and private hosted zones, in ASCII order by domain name, starting at the specified DNS name.

```
Get-R53HostedZonesByName -DnsName example2.com
```

- For API details, see ListHostedZonesByName in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: Returns all of your public and private hosted zones in ASCII order by domain name.

```
Get-R53HostedZonesByName
```

Example 2: Returns your public and private hosted zones, in ASCII order by domain name, starting at the specified DNS name.

```
Get-R53HostedZonesByName -DnsName example2.com
```

- For API details, see ListHostedZonesByName in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `ListQueryLoggingConfigs` with a CLI

The following code examples show how to use `ListQueryLoggingConfigs`.

CLI

AWS CLI

To list query logging configurations

The following `list-query-logging-configs` example lists information about the first 100 query logging configurations in your AWS account, for the hosted zone `Z1OX3WQEXAMPLE`.

```
aws route53 list-query-logging-configs \
    --hosted-zone-id Z1OX3WQEXAMPLE
```

Output:

```
{
    "QueryLoggingConfigs": [
        {
            "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "HostedZoneId": "Z1OX3WQEXAMPLE",
            "CloudWatchLogsLogGroupArn": "arn:aws:logs:us-
east-1:123456789012:log-group:/aws/route53/example.com:*"
        }
    ]
```

```
}
```

For more information, see Logging DNS queries in the *Amazon Route 53 Developer Guide*.

- For API details, see ListQueryLoggingConfigs in *AWS CLI Command Reference*.

PowerShell

Tools for PowerShell V4

Example 1: This example returns all the configurations for DNS query logging that are associated with the current AWS account.

```
Get-R53QueryLoggingConfigList
```

Output:

```
Id                                   HostedZoneId   CloudWatchLogsLogGroupArn
--                                   ------------   ------------------------
59b0fa33-4fea-4471-a88c-926476aaa40d Z385PDS6EAAAZR arn:aws:logs:us-
east-1:111111111112:log-group:/aws/route53/example1.com:*
ee528e95-4e03-4fdc-9d28-9e24ddaaa063 Z94SJHBV1AAAAZ arn:aws:logs:us-
east-1:111111111112:log-group:/aws/route53/example2.com:*
e38dddda-ceb6-45c1-8cb7-f0ae56aaaa2b Z3MEQ8T7AAA1BF arn:aws:logs:us-
east-1:111111111112:log-group:/aws/route53/example3.com:*
```

- For API details, see ListQueryLoggingConfigs in *AWS Tools for PowerShell Cmdlet Reference (V4)*.

Tools for PowerShell V5

Example 1: This example returns all the configurations for DNS query logging that are associated with the current AWS account.

```
Get-R53QueryLoggingConfigList
```

Output:

```
Id                                   HostedZoneId   CloudWatchLogsLogGroupArn
--                                   ------------   ------------------------
```

```
59b0fa33-4fea-4471-a88c-926476aaa40d Z385PDS6EAAAZR arn:aws:logs:us-
east-1:111111111112:log-group:/aws/route53/example1.com:*
ee528e95-4e03-4fdc-9d28-9e24ddaaa063 Z94SJHBV1AAAAZ arn:aws:logs:us-
east-1:111111111112:log-group:/aws/route53/example2.com:*
e38dddda-ceb6-45c1-8cb7-f0ae56aaaa2b Z3MEQ8T7AAA1BF arn:aws:logs:us-
east-1:111111111112:log-group:/aws/route53/example3.com:*
```

- For API details, see ListQueryLoggingConfigs in *AWS Tools for PowerShell Cmdlet Reference (V5)*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Code examples for Route 53 domain registration using AWS SDKs

The following code examples show how to use Route 53 domain registration with an AWS software development kit (SDK).

Basics are code examples that show you how to perform the essential operations within a service.

Actions are code excerpts from larger programs and must be run in context. While actions show you how to call individual service functions, you can see actions in context in their related scenarios.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Route 53 domain registration

- Basic examples for Route 53 domain registration using AWS SDKs
  - Hello Route 53 domain registration
  - Learn the basics of Route 53 domain registration with an AWS SDK
  - Actions for Route 53 domain registration using AWS SDKs
    - Use CheckDomainAvailability with an AWS SDK or CLI
    - Use CheckDomainTransferability with an AWS SDK or CLI
    - Use GetDomainDetail with an AWS SDK or CLI

- Use GetDomainSuggestions with an AWS SDK or CLI
- Use GetOperationDetail with an AWS SDK or CLI
- Use ListDomains with an AWS SDK or CLI
- Use ListOperations with an AWS SDK or CLI
- Use ListPrices with an AWS SDK
- Use RegisterDomain with an AWS SDK or CLI
- Use ViewBilling with an AWS SDK or CLI

Basic examples for Route 53 domain registration using AWS SDKs

The following code examples show how to use the basics of Amazon Route 53 domain registration with AWS SDKs.

Examples

- Hello Route 53 domain registration
- Learn the basics of Route 53 domain registration with an AWS SDK
- Actions for Route 53 domain registration using AWS SDKs
  - Use CheckDomainAvailability with an AWS SDK or CLI
  - Use CheckDomainTransferability with an AWS SDK or CLI
  - Use GetDomainDetail with an AWS SDK or CLI
  - Use GetDomainSuggestions with an AWS SDK or CLI
  - Use GetOperationDetail with an AWS SDK or CLI
  - Use ListDomains with an AWS SDK or CLI
  - Use ListOperations with an AWS SDK or CLI
  - Use ListPrices with an AWS SDK
  - Use RegisterDomain with an AWS SDK or CLI
  - Use ViewBilling with an AWS SDK or CLI

Hello Route 53 domain registration

The following code examples show how to get started using Route 53 domain registration.

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
public static class HelloRoute53Domains
{
    static async Task Main(string[] args)
    {
        // Use the AWS .NET Core Setup package to set up dependency injection for
 the Amazon Route 53 domain registration service.
        // Use your AWS profile name, or leave it blank to use the default
 profile.
        using var host = Host.CreateDefaultBuilder(args)
            .ConfigureServices((_, services) =>
                services.AddAWSService<IAmazonRoute53Domains>()
            ).Build();

        // Now the client is available for injection.
        var route53Client =
 host.Services.GetRequiredService<IAmazonRoute53Domains>();

        // You can use await and any of the async methods to get a response.
        var response = await route53Client.ListPricesAsync(new ListPricesRequest
 { Tld = "com" });
        Console.WriteLine($"Hello Amazon Route 53 Domains! Following are prices
 for .com domain operations:");
        var comPrices = response.Prices.FirstOrDefault();
        if (comPrices != null)
        {
            Console.WriteLine($"\tRegistration:
 {comPrices.RegistrationPrice?.Price} {comPrices.RegistrationPrice?.Currency}");
            Console.WriteLine($"\tRenewal: {comPrices.RenewalPrice?.Price}
 {comPrices.RenewalPrice?.Currency}");
        }
    }
```

```
}
```

- For API details, see ListPrices in *AWS SDK for .NET API Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.route53domains.Route53DomainsClient;
import software.amazon.awssdk.services.route53.model.Route53Exception;
import software.amazon.awssdk.services.route53domains.model.DomainPrice;
import software.amazon.awssdk.services.route53domains.model.ListPricesRequest;
import software.amazon.awssdk.services.route53domains.model.ListPricesResponse;
import java.util.List;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 *
 * This Java code examples performs the following operation:
 *
 * 1. Invokes ListPrices for at least one domain type, such as the “com” type
 * and displays the prices for Registration and Renewal.
 *
 */
public class HelloRoute53 {
    public static final String DASHES = new String(new char[80]).replace("\0",
 "-");
```

```
   public static void main(String[] args) {
       final String usage = "\n" +
               "Usage:\n" +
               "    <hostedZoneId> \n\n" +
               "Where:\n" +
               "    hostedZoneId - The id value of an existing hosted zone. \n";

       if (args.length != 1) {
           System.out.println(usage);
           System.exit(1);
       }

       String domainType = args[0];
       Region region = Region.US_EAST_1;
       Route53DomainsClient route53DomainsClient =
Route53DomainsClient.builder()
               .region(region)
               .build();

       System.out.println(DASHES);
       System.out.println("Invokes ListPrices for at least one domain type.");
       listPrices(route53DomainsClient, domainType);
       System.out.println(DASHES);
   }

   public static void listPrices(Route53DomainsClient route53DomainsClient,
String domainType) {
       try {
           ListPricesRequest pricesRequest = ListPricesRequest.builder()
                   .maxItems(10)
                   .tld(domainType)
                   .build();

           ListPricesResponse response =
route53DomainsClient.listPrices(pricesRequest);
           List<DomainPrice> prices = response.prices();
           for (DomainPrice pr : prices) {
               System.out.println("Name: " + pr.name());
               System.out.println(
                       "Registration: " + pr.registrationPrice().price() + " " +
pr.registrationPrice().currency());
               System.out.println("Renewal: " + pr.renewalPrice().price() + " "
+ pr.renewalPrice().currency());
```

```
                System.out.println("Transfer: " + pr.transferPrice().price() + "
 " + pr.transferPrice().currency());
                System.out.println("Transfer: " + pr.transferPrice().price() + "
 " + pr.transferPrice().currency());
                System.out.println("Change Ownership: " +
 pr.changeOwnershipPrice().price() + " "
                        + pr.changeOwnershipPrice().currency());
                System.out.println(
                        "Restoration: " + pr.restorationPrice().price() + " " +
 pr.restorationPrice().currency());
                System.out.println(" ");
            }

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
}
```

- For API details, see ListPrices in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/**
 Before running this Kotlin code example, set up your development environment,
 including your credentials.

 For more information, see the following documentation topic:
 https://docs.aws.amazon.com/sdk-for-kotlin/latest/developer-guide/setup.html
 */
suspend fun main(args: Array<String>) {
```

```
    val usage = """
        Usage:
           <domainType>

       Where:
           domainType - The domain type (for example, com).
    """

    if (args.size != 1) {
        println(usage)
        exitProcess(0)
    }

    val domainType = args[0]
    println("Invokes ListPrices using a Paginated method.")
    listPricesPaginated(domainType)
}

suspend fun listPricesPaginated(domainType: String) {
    val pricesRequest =
        ListPricesRequest {
            maxItems = 10
            tld = domainType
        }

    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .listPricesPaginated(pricesRequest)
            .transform { it.prices?.forEach { obj -> emit(obj) } }
            .collect { pr ->
                println("Registration: ${pr.registrationPrice}
 ${pr.registrationPrice?.currency}")
                println("Renewal: ${pr.renewalPrice?.price}
 ${pr.renewalPrice?.currency}")
                println("Transfer: ${pr.transferPrice?.price}
 ${pr.transferPrice?.currency}")
                println("Restoration: ${pr.restorationPrice?.price}
 ${pr.restorationPrice?.currency}")
            }
    }
}
```

- For API details, see ListPrices in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Learn the basics of Route 53 domain registration with an AWS SDK

The following code examples show how to:

- List current domains, and list operations in the past year.
- View billing for the past year, and view prices for domain types.
- Get domain suggestions.
- Check domain availability and transferability.
- Optionally, request a domain registration.
- Get an operation detail.
- Optionally, get a domain detail.

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

Run an interactive scenario at a command prompt.

```
public static class Route53DomainScenario
{
    /*
    Before running this .NET code example, set up your development environment,
 including your credentials.

    This .NET example performs the following tasks:
```

```
        1. List current domains.
        2. List operations in the past year.
        3. View billing for the account in the past year.
        4. View prices for domain types.
        5. Get domain suggestions.
        6. Check domain availability.
        7. Check domain transferability.
        8. Optionally, request a domain registration.
        9. Get an operation detail.
       10. Optionally, get a domain detail.
  */

   private static Route53Wrapper _route53Wrapper = null!;
   private static IConfiguration _configuration = null!;

   static async Task Main(string[] args)
   {
       // Set up dependency injection for the Amazon service.
       using var host = Host.CreateDefaultBuilder(args)
           .ConfigureLogging(logging =>
               logging.AddFilter("System", LogLevel.Debug)
                   .AddFilter<DebugLoggerProvider>("Microsoft",
LogLevel.Information)
                   .AddFilter<ConsoleLoggerProvider>("Microsoft",
LogLevel.Trace))
                   .ConfigureServices((_, services) =>
           services.AddAWSService<IAmazonRoute53Domains>()
               .AddTransient<Route53Wrapper>()
           )
           .Build();

       _configuration = new ConfigurationBuilder()
           .SetBasePath(Directory.GetCurrentDirectory())
           .AddJsonFile("settings.json") // Load settings from .json file.
           .AddJsonFile("settings.local.json",
               true) // Optionally, load local settings.
           .Build();

       var logger = LoggerFactory.Create(builder =>
       {
           builder.AddConsole();
       }).CreateLogger(typeof(Route53DomainScenario));

       _route53Wrapper = host.Services.GetRequiredService<Route53Wrapper>();
```

```
        Console.WriteLine(new string('-', 80));
        Console.WriteLine("Welcome to the Amazon Route 53 domains example
scenario.");
        Console.WriteLine(new string('-', 80));

        try
        {
            await ListDomains();
            await ListOperations();
            await ListBillingRecords();
            await ListPrices();
            await ListDomainSuggestions();
            await CheckDomainAvailability();
            await CheckDomainTransferability();
            var operationId = await RequestDomainRegistration();
            await GetOperationalDetail(operationId);
            await GetDomainDetails();
        }
        catch (Exception ex)
        {
            logger.LogError(ex, "There was a problem executing the scenario.");
        }

        Console.WriteLine(new string('-', 80));
        Console.WriteLine("The Amazon Route 53 domains example scenario is
complete.");
        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// List account registered domains.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task ListDomains()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"1. List account domains.");
        var domains = await _route53Wrapper.ListDomains();
        for (int i = 0; i < domains.Count; i++)
        {
            Console.WriteLine($"\t{i + 1}. {domains[i].DomainName}");
        }
```

```
        if (!domains.Any())
        {
            Console.WriteLine("\tNo domains found in this account.");
        }

        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// List domain operations in the past year.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task ListOperations()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"2. List account domain operations in the past
year.");
        var operations = await _route53Wrapper.ListOperations(
            DateTime.Today.AddYears(-1));
        for (int i = 0; i < operations.Count; i++)
        {
            Console.WriteLine($"\tOperation Id: {operations[i].OperationId}");
            Console.WriteLine($"\tStatus: {operations[i].Status}");
            Console.WriteLine($"\tDate: {operations[i].SubmittedDate}");
        }
        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// List billing in the past year.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task ListBillingRecords()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"3. View billing for the account in the past year.");
        var billingRecords = await _route53Wrapper.ViewBilling(
            DateTime.Today.AddYears(-1),
            DateTime.Today);
        for (int i = 0; i < billingRecords.Count; i++)
        {
            Console.WriteLine($"\tBill Date:
{billingRecords[i].BillDate.ToShortDateString()}");
            Console.WriteLine($"\tOperation: {billingRecords[i].Operation}");
```

```
            Console.WriteLine($"\tPrice: {billingRecords[i].Price}");
        }
        if (!billingRecords.Any())
        {
            Console.WriteLine("\tNo billing records found in this account for the
past year.");
        }
        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// List prices for a few domain types.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task ListPrices()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"4. View prices for domain types.");
        var domainTypes = new List<string> { "net", "com", "org", "co" };

        var prices = await _route53Wrapper.ListPrices(domainTypes);
        foreach (var pr in prices)
        {
            Console.WriteLine($"\tName: {pr.Name}");
            Console.WriteLine($"\tRegistration: {pr.RegistrationPrice?.Price}
{pr.RegistrationPrice?.Currency}");
            Console.WriteLine($"\tRenewal: {pr.RenewalPrice?.Price}
{pr.RenewalPrice?.Currency}");
            Console.WriteLine($"\tTransfer: {pr.TransferPrice?.Price}
{pr.TransferPrice?.Currency}");
            Console.WriteLine($"\tChange Ownership:
{pr.ChangeOwnershipPrice?.Price} {pr.ChangeOwnershipPrice?.Currency}");
            Console.WriteLine($"\tRestoration: {pr.RestorationPrice?.Price}
{pr.RestorationPrice?.Currency}");
            Console.WriteLine();
        }
        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// List domain suggestions for a domain name.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task ListDomainSuggestions()
```

```
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"5. Get domain suggestions.");
        string? domainName = null;
        while (domainName == null || string.IsNullOrWhiteSpace(domainName))
        {
            Console.WriteLine($"Enter a domain name to get available domain
suggestions.");
            domainName = Console.ReadLine();
        }

        var suggestions = await _route53Wrapper.GetDomainSuggestions(domainName,
true, 5);
        foreach (var suggestion in suggestions)
        {
            Console.WriteLine($"\tSuggestion Name: {suggestion.DomainName}");
            Console.WriteLine($"\tAvailability: {suggestion.Availability}");
        }
        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// Check availability for a domain name.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task CheckDomainAvailability()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"6. Check domain availability.");
        string? domainName = null;
        while (domainName == null || string.IsNullOrWhiteSpace(domainName))
        {
            Console.WriteLine($"Enter a domain name to check domain
availability.");
            domainName = Console.ReadLine();
        }

        var availability = await
_route53Wrapper.CheckDomainAvailability(domainName);
        Console.WriteLine($"\tAvailability: {availability}");
        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
```

```
    /// Check transferability for a domain name.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task CheckDomainTransferability()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"7. Check domain transferability.");
        string? domainName = null;
        while (domainName == null || string.IsNullOrWhiteSpace(domainName))
        {
            Console.WriteLine($"Enter a domain name to check domain
 transferability.");
            domainName = Console.ReadLine();
        }

        var transferability = await
 _route53Wrapper.CheckDomainTransferability(domainName);
        Console.WriteLine($"\tTransferability: {transferability}");

        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
    /// Check transferability for a domain name.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task<string?> RequestDomainRegistration()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"8. Optionally, request a domain registration.");

        Console.WriteLine($"\tNote: This example uses domain request settings in
 settings.json.");
        Console.WriteLine($"\tTo change the domain registration settings, set the
 values in that file.");
        Console.WriteLine($"\tRemember, registering an actual domain will incur
 an account billing cost.");
        Console.WriteLine($"\tWould you like to begin a domain registration? (y/
n)");
        var ynResponse = Console.ReadLine();
        if (ynResponse != null && ynResponse.Equals("y",
 StringComparison.InvariantCultureIgnoreCase))
        {
            string domainName = _configuration["DomainName"];
```

```
            ContactDetail contact = new ContactDetail();
            contact.CountryCode =
CountryCode.FindValue(_configuration["Contact:CountryCode"]);
            contact.ContactType =
ContactType.FindValue(_configuration["Contact:ContactType"]);

            _configuration.GetSection("Contact").Bind(contact);

            var operationId = await _route53Wrapper.RegisterDomain(
                domainName,
                Convert.ToBoolean(_configuration["AutoRenew"]),
                Convert.ToInt32(_configuration["DurationInYears"]),
                contact);
            if (operationId != null)
            {
                Console.WriteLine(
                    $"\tRegistration requested. Operation Id: {operationId}");
            }

            return operationId;
        }

        Console.WriteLine(new string('-', 80));
        return null;
    }

    /// <summary>
    /// Get details for an operation.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task GetOperationalDetail(string? operationId)
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"9. Get an operation detail.");

        var operationDetails =
            await _route53Wrapper.GetOperationDetail(operationId);

        Console.WriteLine(operationDetails);

        Console.WriteLine(new string('-', 80));
    }

    /// <summary>
```

```
    /// Optionally, get details for a registered domain.
    /// </summary>
    /// <returns>Async task.</returns>
    private static async Task<string?> GetDomainDetails()
    {
        Console.WriteLine(new string('-', 80));
        Console.WriteLine($"10. Get details on a domain.");

        Console.WriteLine($"\tNote: you must have a registered domain to get
 details.");
        Console.WriteLine($"\tWould you like to get domain details? (y/n)");
        var ynResponse = Console.ReadLine();
        if (ynResponse != null && ynResponse.Equals("y",
 StringComparison.InvariantCultureIgnoreCase))
        {
            string? domainName = null;
            while (domainName == null)
            {
                Console.WriteLine($"\tEnter a domain name to get details.");
                domainName = Console.ReadLine();
            }

            var domainDetails = await
 _route53Wrapper.GetDomainDetail(domainName);
            Console.WriteLine(domainDetails);
        }

        Console.WriteLine(new string('-', 80));
        return null;
    }
}
```

Wrapper methods used by the scenario for Route 53 domain registration actions.

```
public class Route53Wrapper
{
    private readonly IAmazonRoute53Domains _amazonRoute53Domains;
    private readonly ILogger<Route53Wrapper> _logger;
    public Route53Wrapper(IAmazonRoute53Domains amazonRoute53Domains,
 ILogger<Route53Wrapper> logger)
    {
```

```
        _amazonRoute53Domains = amazonRoute53Domains;
        _logger = logger;
    }


    /// <summary>
    /// List prices for domain type operations.
    /// </summary>
    /// <param name="domainTypes">Domain types to include in the results.</param>
    /// <returns>The list of domain prices.</returns>
    public async Task<List<DomainPrice>> ListPrices(List<string> domainTypes)
    {
        var results = new List<DomainPrice>();
        var paginatePrices = _amazonRoute53Domains.Paginators.ListPrices(new
ListPricesRequest());
        // Get the entire list using the paginator.
        await foreach (var prices in paginatePrices.Prices)
        {
            results.Add(prices);
        }
        return results.Where(p => domainTypes.Contains(p.Name)).ToList();
    }


    /// <summary>
    /// Check the availability of a domain name.
    /// </summary>
    /// <param name="domain">The domain to check for availability.</param>
    /// <returns>An availability result string.</returns>
    public async Task<string> CheckDomainAvailability(string domain)
    {
        var result = await _amazonRoute53Domains.CheckDomainAvailabilityAsync(
            new CheckDomainAvailabilityRequest
            {
                DomainName = domain
            }
        );
        return result.Availability.Value;
    }


    /// <summary>
    /// Check the transferability of a domain name.
    /// </summary>
```

```
    /// <param name="domain">The domain to check for transferability.</param>
    /// <returns>A transferability result string.</returns>
    public async Task<string> CheckDomainTransferability(string domain)
    {
        var result = await _amazonRoute53Domains.CheckDomainTransferabilityAsync(
            new CheckDomainTransferabilityRequest
            {
                DomainName = domain
            }
        );
        return result.Transferability.Transferable.Value;
    }


    /// <summary>
    /// Get a list of suggestions for a given domain.
    /// </summary>
    /// <param name="domain">The domain to check for suggestions.</param>
    /// <param name="onlyAvailable">If true, only returns available domains.</
param>
    /// <param name="suggestionCount">The number of suggestions to return.
 Defaults to the max of 50.</param>
    /// <returns>A collection of domain suggestions.</returns>
    public async Task<List<DomainSuggestion>> GetDomainSuggestions(string domain,
 bool onlyAvailable, int suggestionCount = 50)
    {
        var result = await _amazonRoute53Domains.GetDomainSuggestionsAsync(
            new GetDomainSuggestionsRequest
            {
                DomainName = domain,
                OnlyAvailable = onlyAvailable,
                SuggestionCount = suggestionCount
            }
        );
        return result.SuggestionsList;
    }


    /// <summary>
    /// Get details for a domain action operation.
    /// </summary>
    /// <param name="operationId">The operational Id.</param>
    /// <returns>A string describing the operational details.</returns>
    public async Task<string> GetOperationDetail(string? operationId)
```

```
    {
        if (operationId == null)
            return "Unable to get operational details because ID is null.";
        try
        {
            var operationDetails =
                await _amazonRoute53Domains.GetOperationDetailAsync(
                    new GetOperationDetailRequest
                    {
                        OperationId = operationId
                    }
                );

            var details = $"\tOperation {operationId}:\n" +
                          $"\tFor domain {operationDetails.DomainName} on
 {operationDetails.SubmittedDate.ToShortDateString()}.\n" +
                          $"\tMessage is {operationDetails.Message}.\n" +
                          $"\tStatus is {operationDetails.Status}.\n";

            return details;
        }
        catch (AmazonRoute53DomainsException ex)
        {
            return $"Unable to get operation details. Here's why: {ex.Message}.";
        }
    }


    /// <summary>
    /// Initiate a domain registration request.
    /// </summary>
    /// <param name="contact">Contact details.</param>
    /// <param name="domainName">The domain name to register.</param>
    /// <param name="autoRenew">True if the domain should automatically renew.</
param>
    /// <param name="duration">The duration in years for the domain
 registration.</param>
    /// <returns>The operation Id.</returns>
    public async Task<string?> RegisterDomain(string domainName, bool autoRenew,
 int duration, ContactDetail contact)
    {
        // This example uses the same contact information for admin, registrant,
 and tech contacts.
        try
```

```
        {
            var result = await _amazonRoute53Domains.RegisterDomainAsync(
                new RegisterDomainRequest()
                {
                    AdminContact = contact,
                    RegistrantContact = contact,
                    TechContact = contact,
                    DomainName = domainName,
                    AutoRenew = autoRenew,
                    DurationInYears = duration,
                    PrivacyProtectAdminContact = false,
                    PrivacyProtectRegistrantContact = false,
                    PrivacyProtectTechContact = false
                }
            );
            return result.OperationId;
        }
        catch (InvalidInputException)
        {
            _logger.LogInformation($"Unable to request registration for domain
{domainName}");
            return null;
        }
    }


    /// <summary>
    /// View billing records for the account between a start and end date.
    /// </summary>
    /// <param name="startDate">The start date for billing results.</param>
    /// <param name="endDate">The end date for billing results.</param>
    /// <returns>A collection of billing records.</returns>
    public async Task<List<BillingRecord>> ViewBilling(DateTime startDate,
DateTime endDate)
    {
        var results = new List<BillingRecord>();
        var paginateBilling = _amazonRoute53Domains.Paginators.ViewBilling(
            new ViewBillingRequest()
            {
                Start = startDate,
                End = endDate
            });

        // Get the entire list using the paginator.
```

```
        await foreach (var billingRecords in paginateBilling.BillingRecords)
        {
            results.Add(billingRecords);
        }
        return results;
    }


    /// <summary>
    /// List the domains for the account.
    /// </summary>
    /// <returns>A collection of domain summary records.</returns>
    public async Task<List<DomainSummary>> ListDomains()
    {
        var results = new List<DomainSummary>();
        var paginateDomains = _amazonRoute53Domains.Paginators.ListDomains(
            new ListDomainsRequest());

        // Get the entire list using the paginator.
        await foreach (var domain in paginateDomains.Domains)
        {
            results.Add(domain);
        }
        return results;
    }


    /// <summary>
    /// List operations for the account that are submitted after a specified
date.
    /// </summary>
    /// <returns>A collection of operation summary records.</returns>
    public async Task<List<OperationSummary>> ListOperations(DateTime
submittedSince)
    {
        var results = new List<OperationSummary>();
        var paginateOperations = _amazonRoute53Domains.Paginators.ListOperations(
            new ListOperationsRequest()
            {
                SubmittedSince = submittedSince
            });

        // Get the entire list using the paginator.
        await foreach (var operations in paginateOperations.Operations)
```

```
        {
            results.Add(operations);
        }
        return results;
    }


    /// <summary>
    /// Get details for a domain.
    /// </summary>
    /// <returns>A string with detail information about the domain.</returns>
    public async Task<string> GetDomainDetail(string domainName)
    {
        try
        {
            var result = await _amazonRoute53Domains.GetDomainDetailAsync(
                new GetDomainDetailRequest()
                {
                    DomainName = domainName
                });
            var details = $"\tDomain {domainName}:\n" +
                          $"\tCreated on
 {result.CreationDate.ToShortDateString()}.\n" +
                          $"\tAdmin contact is {result.AdminContact.Email}.\n" +
                          $"\tAuto-renew is {result.AutoRenew}.\n";

            return details;
        }
        catch (InvalidInputException)
        {
            return $"Domain {domainName} was not found in your account.";
        }
    }
}
```

- For API details, see the following topics in *AWS SDK for .NET API Reference*.
  - CheckDomainAvailability
  - CheckDomainTransferability
  - GetDomainDetail
  - GetDomainSuggestions

- GetOperationDetail
- ListDomains
- ListOperations
- ListPrices
- RegisterDomain
- ViewBilling

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 *
 * This example uses pagination methods where applicable. For example, to list
 * domains, the
 * listDomainsPaginator method is used. For more information about pagination,
 * see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/
pagination.html
 *
 * This Java code example performs the following operations:
 *
 * 1. List current domains.
 * 2. List operations in the past year.
 * 3. View billing for the account in the past year.
```

```
 * 4. View prices for domain types.
 * 5. Get domain suggestions.
 * 6. Check domain availability.
 * 7. Check domain transferability.
 * 8. Request a domain registration.
 * 9. Get operation details.
 * 10. Optionally, get domain details.
 */

public class Route53Scenario {
    public static final String DASHES = new String(new char[80]).replace("\0",
 "-");

    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <domainType> <phoneNumber> <email> <domainSuggestion>
 <firstName> <lastName> <city>

                Where:
                    domainType - The domain type (for example, com).\s
                    phoneNumber - The phone number to use (for example,
 +91.9966564xxx)      email - The email address to use.      domainSuggestion -
 The domain suggestion (for example, findmy.accountants).\s
                    firstName - The first name to use to register a domain.\s
                    lastName -  The last name to use to register a domain.\s
                    city - the city to use to register a domain.\s
                    """;

        if (args.length != 7) {
            System.out.println(usage);
            System.exit(1);
        }

        String domainType = args[0];
        String phoneNumber = args[1];
        String email = args[2];
        String domainSuggestion = args[3];
        String firstName = args[4];
        String lastName = args[5];
        String city = args[6];
        Region region = Region.US_EAST_1;
```

```
        Route53DomainsClient route53DomainsClient =
Route53DomainsClient.builder()
                .region(region)
                .build();

        System.out.println(DASHES);
        System.out.println("Welcome to the Amazon Route 53 domains example
scenario.");
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("1. List current domains.");
        listDomains(route53DomainsClient);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("2. List operations in the past year.");
        listOperations(route53DomainsClient);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("3. View billing for the account in the past year.");
        listBillingRecords(route53DomainsClient);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("4. View prices for domain types.");
        listPrices(route53DomainsClient, domainType);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("5. Get domain suggestions.");
        listDomainSuggestions(route53DomainsClient, domainSuggestion);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("6. Check domain availability.");
        checkDomainAvailability(route53DomainsClient, domainSuggestion);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("7. Check domain transferability.");
        checkDomainTransferability(route53DomainsClient, domainSuggestion);
        System.out.println(DASHES);
```

```
        System.out.println(DASHES);
        System.out.println("8. Request a domain registration.");
        String opId = requestDomainRegistration(route53DomainsClient,
domainSuggestion, phoneNumber, email, firstName,
                lastName, city);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("9. Get operation details.");
        getOperationalDetail(route53DomainsClient, opId);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("10. Get domain details.");
        System.out.println("Note: You must have a registered domain to get
details.");
        System.out.println("Otherwise, an exception is thrown that states ");
        System.out.println("Domain xxxxxxx not found in xxxxxxx account.");
        getDomainDetails(route53DomainsClient, domainSuggestion);
        System.out.println(DASHES);
    }

    public static void getDomainDetails(Route53DomainsClient
route53DomainsClient, String domainSuggestion) {
        try {
            GetDomainDetailRequest detailRequest =
GetDomainDetailRequest.builder()
                    .domainName(domainSuggestion)
                    .build();

            GetDomainDetailResponse response =
route53DomainsClient.getDomainDetail(detailRequest);
            System.out.println("The contact first name is " +
response.registrantContact().firstName());
            System.out.println("The contact last name is " +
response.registrantContact().lastName());
            System.out.println("The contact org name is " +
response.registrantContact().organizationName());

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
```

```
   }

   public static void getOperationalDetail(Route53DomainsClient
route53DomainsClient, String operationId) {
       try {
           GetOperationDetailRequest detailRequest =
GetOperationDetailRequest.builder()
                   .operationId(operationId)
                   .build();

           GetOperationDetailResponse response =
route53DomainsClient.getOperationDetail(detailRequest);
           System.out.println("Operation detail message is " +
response.message());

       } catch (Route53Exception e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
   }

   public static String requestDomainRegistration(Route53DomainsClient
route53DomainsClient,
           String domainSuggestion,
           String phoneNumber,
           String email,
           String firstName,
           String lastName,
           String city) {

       try {
           ContactDetail contactDetail = ContactDetail.builder()
                   .contactType(ContactType.COMPANY)
                   .state("LA")
                   .countryCode(CountryCode.IN)
                   .email(email)
                   .firstName(firstName)
                   .lastName(lastName)
                   .city(city)
                   .phoneNumber(phoneNumber)
                   .organizationName("My Org")
                   .addressLine1("My Address")
                   .zipCode("123 123")
                   .build();
```

```
            RegisterDomainRequest domainRequest = RegisterDomainRequest.builder()
                    .adminContact(contactDetail)
                    .registrantContact(contactDetail)
                    .techContact(contactDetail)
                    .domainName(domainSuggestion)
                    .autoRenew(true)
                    .durationInYears(1)
                    .build();

            RegisterDomainResponse response =
route53DomainsClient.registerDomain(domainRequest);
            System.out.println("Registration requested. Operation Id: " +
response.operationId());
            return response.operationId();

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
        return "";
    }

    public static void checkDomainTransferability(Route53DomainsClient
route53DomainsClient, String domainSuggestion) {
        try {
            CheckDomainTransferabilityRequest transferabilityRequest =
CheckDomainTransferabilityRequest.builder()
                    .domainName(domainSuggestion)
                    .build();

            CheckDomainTransferabilityResponse response = route53DomainsClient
                    .checkDomainTransferability(transferabilityRequest);
            System.out.println("Transferability: " +
response.transferability().transferable().toString());

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static void checkDomainAvailability(Route53DomainsClient
route53DomainsClient, String domainSuggestion) {
```

```
        try {
            CheckDomainAvailabilityRequest availabilityRequest =
CheckDomainAvailabilityRequest.builder()
                    .domainName(domainSuggestion)
                    .build();

            CheckDomainAvailabilityResponse response = route53DomainsClient
                    .checkDomainAvailability(availabilityRequest);
            System.out.println(domainSuggestion + " is " +
response.availability().toString());

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static void listDomainSuggestions(Route53DomainsClient
route53DomainsClient, String domainSuggestion) {
        try {
            GetDomainSuggestionsRequest suggestionsRequest =
GetDomainSuggestionsRequest.builder()
                    .domainName(domainSuggestion)
                    .suggestionCount(5)
                    .onlyAvailable(true)
                    .build();

            GetDomainSuggestionsResponse response =
route53DomainsClient.getDomainSuggestions(suggestionsRequest);
            List<DomainSuggestion> suggestions = response.suggestionsList();
            for (DomainSuggestion suggestion : suggestions) {
                System.out.println("Suggestion Name: " +
suggestion.domainName());
                System.out.println("Availability: " + suggestion.availability());
                System.out.println(" ");
            }

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

```
   public static void listPrices(Route53DomainsClient route53DomainsClient,
String domainType) {
       try {
           ListPricesRequest pricesRequest = ListPricesRequest.builder()
                   .tld(domainType)
                   .build();

           ListPricesIterable listRes =
route53DomainsClient.listPricesPaginator(pricesRequest);
           listRes.stream()
                   .flatMap(r -> r.prices().stream())
                   .forEach(content -> System.out.println(" Name: " +
content.name() +
                           " Registration: " +
content.registrationPrice().price() + " "
                           + content.registrationPrice().currency() +
                           " Renewal: " + content.renewalPrice().price() + " " +
content.renewalPrice().currency()));

       } catch (Route53Exception e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
   }

   public static void listBillingRecords(Route53DomainsClient
route53DomainsClient) {
       try {
           Date currentDate = new Date();
           LocalDateTime localDateTime =
currentDate.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime();
           ZoneOffset zoneOffset = ZoneOffset.of("+01:00");
           LocalDateTime localDateTime2 = localDateTime.minusYears(1);
           Instant myStartTime = localDateTime2.toInstant(zoneOffset);
           Instant myEndTime = localDateTime.toInstant(zoneOffset);

           ViewBillingRequest viewBillingRequest = ViewBillingRequest.builder()
                   .start(myStartTime)
                   .end(myEndTime)
                   .build();

           ViewBillingIterable listRes =
route53DomainsClient.viewBillingPaginator(viewBillingRequest);
           listRes.stream()
```

```
                    .flatMap(r -> r.billingRecords().stream())
                    .forEach(content -> System.out.println(" Bill Date:: " +
content.billDate() +
                            " Operation: " + content.operationAsString() +
                            " Price: " + content.price()));

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static void listOperations(Route53DomainsClient route53DomainsClient)
{
        try {
            Date currentDate = new Date();
            LocalDateTime localDateTime =
currentDate.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime();
            ZoneOffset zoneOffset = ZoneOffset.of("+01:00");
            localDateTime = localDateTime.minusYears(1);
            Instant myTime = localDateTime.toInstant(zoneOffset);

            ListOperationsRequest operationsRequest =
ListOperationsRequest.builder()
                    .submittedSince(myTime)
                    .build();

            ListOperationsIterable listRes =
route53DomainsClient.listOperationsPaginator(operationsRequest);
            listRes.stream()
                    .flatMap(r -> r.operations().stream())
                    .forEach(content -> System.out.println(" Operation Id: " +
content.operationId() +
                            " Status: " + content.statusAsString() +
                            " Date: " + content.submittedDate()));

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    public static void listDomains(Route53DomainsClient route53DomainsClient) {
        try {
```

```
            ListDomainsIterable listRes =
 route53DomainsClient.listDomainsPaginator();
            listRes.stream()
                    .flatMap(r -> r.domains().stream())
                    .forEach(content -> System.out.println("The domain name is "
 + content.domainName()));

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
}
```

- For API details, see the following topics in *AWS SDK for Java 2.x API Reference*.
  - CheckDomainAvailability
  - CheckDomainTransferability
  - GetDomainDetail
  - GetDomainSuggestions
  - GetOperationDetail
  - ListDomains
  - ListOperations
  - ListPrices
  - RegisterDomain
  - ViewBilling

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/**
Before running this Kotlin code example, set up your development environment,
including your credentials.

For more information, see the following documentation topic:
https://docs.aws.amazon.com/sdk-for-kotlin/latest/developer-guide/setup.html

This Kotlin code example performs the following operations:

1. List current domains.
2. List operations in the past year.
3. View billing for the account in the past year.
4. View prices for domain types.
5. Get domain suggestions.
6. Check domain availability.
7. Check domain transferability.
8. Request a domain registration.
9. Get operation details.
10. Optionally, get domain details.
 */

val DASHES: String = String(CharArray(80)).replace("\u0000", "-")

suspend fun main(args: Array<String>) {
    val usage = """
        Usage:
            <domainType> <phoneNumber> <email> <domainSuggestion> <firstName>
 <lastName> <city>
        Where:
           domainType - The domain type (for example, com).
           phoneNumber - The phone number to use (for example, +1.2065550100)

           email - The email address to use.
           domainSuggestion - The domain suggestion (for example,
 findmy.example).
           firstName - The first name to use to register a domain.
           lastName -  The last name to use to register a domain.
           city - The city to use to register a domain.
    """

    if (args.size != 7) {
        println(usage)
```

```
        exitProcess(1)
    }

    val domainType = args[0]
    val phoneNumber = args[1]
    val email = args[2]
    val domainSuggestion = args[3]
    val firstName = args[4]
    val lastName = args[5]
    val city = args[6]

    println(DASHES)
    println("Welcome to the Amazon Route 53 domains example scenario.")
    println(DASHES)

    println(DASHES)
    println("1. List current domains.")
    listDomains()
    println(DASHES)

    println(DASHES)
    println("2. List operations in the past year.")
    listOperations()
    println(DASHES)

    println(DASHES)
    println("3. View billing for the account in the past year.")
    listBillingRecords()
    println(DASHES)

    println(DASHES)
    println("4. View prices for domain types.")
    listAllPrices(domainType)
    println(DASHES)

    println(DASHES)
    println("5. Get domain suggestions.")
    listDomainSuggestions(domainSuggestion)
    println(DASHES)

    println(DASHES)
    println("6. Check domain availability.")
    checkDomainAvailability(domainSuggestion)
    println(DASHES)
```

```
    println(DASHES)
    println("7. Check domain transferability.")
    checkDomainTransferability(domainSuggestion)
    println(DASHES)

    println(DASHES)
    println("8. Request a domain registration.")
    val opId = requestDomainRegistration(domainSuggestion, phoneNumber, email,
 firstName, lastName, city)
    println(DASHES)

    println(DASHES)
    println("9. Get operation details.")
    getOperationalDetail(opId)
    println(DASHES)

    println(DASHES)
    println("10. Get domain details.")
    println("Note: You must have a registered domain to get details.")
    println("Otherwise an exception is thrown that states ")
    println("Domain xxxxxxx not found in xxxxxxx account.")
    getDomainDetails(domainSuggestion)
    println(DASHES)
}

suspend fun getDomainDetails(domainSuggestion: String?) {
    val detailRequest =
        GetDomainDetailRequest {
            domainName = domainSuggestion
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response = route53DomainsClient.getDomainDetail(detailRequest)
        println("The contact first name is
 ${response.registrantContact?.firstName}")
        println("The contact last name is
 ${response.registrantContact?.lastName}")
        println("The contact org name is
 ${response.registrantContact?.organizationName}")
    }
}

suspend fun getOperationalDetail(opId: String?) {
```

```
    val detailRequest =
        GetOperationDetailRequest {
            operationId = opId
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response = route53DomainsClient.getOperationDetail(detailRequest)
        println("Operation detail message is ${response.message}")
    }
}

suspend fun requestDomainRegistration(
    domainSuggestion: String?,
    phoneNumberVal: String?,
    emailVal: String?,
    firstNameVal: String?,
    lastNameVal: String?,
    cityVal: String?,
): String? {
    val contactDetail =
        ContactDetail {
            contactType = ContactType.Company
            state = "LA"
            countryCode = CountryCode.In
            email = emailVal
            firstName = firstNameVal
            lastName = lastNameVal
            city = cityVal
            phoneNumber = phoneNumberVal
            organizationName = "My Org"
            addressLine1 = "My Address"
            zipCode = "123 123"
        }

    val domainRequest =
        RegisterDomainRequest {
            adminContact = contactDetail
            registrantContact = contactDetail
            techContact = contactDetail
            domainName = domainSuggestion
            autoRenew = true
            durationInYears = 1
        }
```

```
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response = route53DomainsClient.registerDomain(domainRequest)
        println("Registration requested. Operation Id: ${response.operationId}")
        return response.operationId
    }
}

suspend fun checkDomainTransferability(domainSuggestion: String?) {
    val transferabilityRequest =
        CheckDomainTransferabilityRequest {
            domainName = domainSuggestion
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response =
 route53DomainsClient.checkDomainTransferability(transferabilityRequest)
        println("Transferability: ${response.transferability?.transferable}")
    }
}

suspend fun checkDomainAvailability(domainSuggestion: String) {
    val availabilityRequest =
        CheckDomainAvailabilityRequest {
            domainName = domainSuggestion
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response =
 route53DomainsClient.checkDomainAvailability(availabilityRequest)
        println("$domainSuggestion is ${response.availability}")
    }
}

suspend fun listDomainSuggestions(domainSuggestion: String?) {
    val suggestionsRequest =
        GetDomainSuggestionsRequest {
            domainName = domainSuggestion
            suggestionCount = 5
            onlyAvailable = true
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
```

```
        val response =
 route53DomainsClient.getDomainSuggestions(suggestionsRequest)
        response.suggestionsList?.forEach { suggestion ->
            println("Suggestion Name: ${suggestion.domainName}")
            println("Availability: ${suggestion.availability}")
            println(" ")
        }
    }
}

suspend fun listAllPrices(domainType: String?) {
    val pricesRequest =
        ListPricesRequest {
            tld = domainType
        }

    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .listPricesPaginated(pricesRequest)
            .transform { it.prices?.forEach { obj -> emit(obj) } }
            .collect { pr ->
                println("Registration: ${pr.registrationPrice}
 ${pr.registrationPrice?.currency}")
                println("Renewal: ${pr.renewalPrice?.price}
 ${pr.renewalPrice?.currency}")
                println("Transfer: ${pr.transferPrice?.price}
 ${pr.transferPrice?.currency}")
                println("Restoration: ${pr.restorationPrice?.price}
 ${pr.restorationPrice?.currency}")
            }
    }
}

suspend fun listBillingRecords() {
    val currentDate = Date()
    val localDateTime =
 currentDate.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime()
    val zoneOffset = ZoneOffset.of("+01:00")
    val localDateTime2 = localDateTime.minusYears(1)
    val myStartTime = localDateTime2.toInstant(zoneOffset)
    val myEndTime = localDateTime.toInstant(zoneOffset)
    val timeStart: Instant? = myStartTime?.let { Instant(it) }
    val timeEnd: Instant? = myEndTime?.let { Instant(it) }
```

```
    val viewBillingRequest =
        ViewBillingRequest {
            start = timeStart
            end = timeEnd
        }

    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .viewBillingPaginated(viewBillingRequest)
            .transform { it.billingRecords?.forEach { obj -> emit(obj) } }
            .collect { billing ->
                println("Bill Date: ${billing.billDate}")
                println("Operation: ${billing.operation}")
                println("Price: ${billing.price}")
            }
    }
}

suspend fun listOperations() {
    val currentDate = Date()
    var localDateTime =
 currentDate.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime()
    val zoneOffset = ZoneOffset.of("+01:00")
    localDateTime = localDateTime.minusYears(1)
    val myTime: java.time.Instant? = localDateTime.toInstant(zoneOffset)
    val time2: Instant? = myTime?.let { Instant(it) }
    val operationsRequest =
        ListOperationsRequest {
            submittedSince = time2
        }

    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .listOperationsPaginated(operationsRequest)
            .transform { it.operations?.forEach { obj -> emit(obj) } }
            .collect { content ->
                println("Operation Id: ${content.operationId}")
                println("Status: ${content.status}")
                println("Date: ${content.submittedDate}")
            }
    }
```

```
}

suspend fun listDomains() {
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .listDomainsPaginated(ListDomainsRequest {})
            .transform { it.domains?.forEach { obj -> emit(obj) } }
            .collect { content ->
                println("The domain name is ${content.domainName}")
            }
    }
}
```

- For API details, see the following topics in *AWS SDK for Kotlin API reference*.
  - CheckDomainAvailability
  - CheckDomainTransferability
  - GetDomainDetail
  - GetDomainSuggestions
  - GetOperationDetail
  - ListDomains
  - ListOperations
  - ListPrices
  - RegisterDomain
  - ViewBilling

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Actions for Route 53 domain registration using AWS SDKs

The following code examples demonstrate how to perform individual Route 53 domain registration actions with AWS SDKs. Each example includes a link to GitHub, where you can find instructions for setting up and running the code.

The following examples include only the most commonly used actions. For a complete list, see the Amazon Route 53 domain registration API Reference.

Examples

- Use CheckDomainAvailability with an AWS SDK or CLI
- Use CheckDomainTransferability with an AWS SDK or CLI
- Use GetDomainDetail with an AWS SDK or CLI
- Use GetDomainSuggestions with an AWS SDK or CLI
- Use GetOperationDetail with an AWS SDK or CLI
- Use ListDomains with an AWS SDK or CLI
- Use ListOperations with an AWS SDK or CLI
- Use ListPrices with an AWS SDK
- Use RegisterDomain with an AWS SDK or CLI
- Use ViewBilling with an AWS SDK or CLI

Use `CheckDomainAvailability` with an AWS SDK or CLI

The following code examples show how to use `CheckDomainAvailability`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/// <summary>
```

```
    /// Check the availability of a domain name.
    /// </summary>
    /// <param name="domain">The domain to check for availability.</param>
    /// <returns>An availability result string.</returns>
    public async Task<string> CheckDomainAvailability(string domain)
    {
        var result = await _amazonRoute53Domains.CheckDomainAvailabilityAsync(
            new CheckDomainAvailabilityRequest
            {
                DomainName = domain
            }
        );
        return result.Availability.Value;
    }
```

- For API details, see CheckDomainAvailability in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To determine whether you can register a domain name with Route 53

The following `check-domain-availability` example returns information about whether the domain name `example.com` is available to be registered using Route 53.

This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains check-domain-availability \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "Availability": "UNAVAILABLE"
}
```

Route 53 supports a large number of top-level domains (TLDs), such as `.com` and `.jp`, but we don't support all available TLDs. If you check the availability of a domain and Route 53 doesn't support the TLD, `check-domain-availability` returns the following message.

```
An error occurred (UnsupportedTLD) when calling the CheckDomainAvailability
 operation: <top-level domain> tld is not supported.
```

For a list of the TLDs that you can use when registering a domain with Route 53, see Domains That You Can Register with Amazon Route 53 in the *Amazon Route 53 Developer Guide*. For more information about registering domains with Amazon Route 53, see Registering a New Domain in the *Amazon Route 53 Developer Guide*.

- For API details, see CheckDomainAvailability in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void checkDomainAvailability(Route53DomainsClient
route53DomainsClient, String domainSuggestion) {
       try {
           CheckDomainAvailabilityRequest availabilityRequest =
CheckDomainAvailabilityRequest.builder()
                   .domainName(domainSuggestion)
                   .build();

           CheckDomainAvailabilityResponse response = route53DomainsClient
                   .checkDomainAvailability(availabilityRequest);
           System.out.println(domainSuggestion + " is " +
response.availability().toString());

       } catch (Route53Exception e) {
           System.err.println(e.getMessage());
           System.exit(1);
```

```
        }
    }
```

- For API details, see CheckDomainAvailability in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun checkDomainAvailability(domainSuggestion: String) {
    val availabilityRequest =
        CheckDomainAvailabilityRequest {
            domainName = domainSuggestion
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response =
 route53DomainsClient.checkDomainAvailability(availabilityRequest)
        println("$domainSuggestion is ${response.availability}")
    }
}
```

- For API details, see CheckDomainAvailability in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `CheckDomainTransferability` with an AWS SDK or CLI

The following code examples show how to use `CheckDomainTransferability`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/// <summary>
/// Check the transferability of a domain name.
/// </summary>
/// <param name="domain">The domain to check for transferability.</param>
/// <returns>A transferability result string.</returns>
public async Task<string> CheckDomainTransferability(string domain)
{
    var result = await _amazonRoute53Domains.CheckDomainTransferabilityAsync(
        new CheckDomainTransferabilityRequest
        {
            DomainName = domain
        }
    );
    return result.Transferability.Transferable.Value;
}
```

- For API details, see CheckDomainTransferability in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To determine whether a domain can be transferred to Route 53

The following `check-domain-transferability` example returns information about whether you can transfer the domain name `example.com` to Route 53.

This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains check-domain-transferability \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "Transferability": {
        "Transferable": "UNTRANSFERABLE"
    }
}
```

For more information, see Transferring Registration for a Domain to Amazon Route 53 in the *Amazon Route 53 Developer Guide*.

- For API details, see CheckDomainTransferability in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void checkDomainTransferability(Route53DomainsClient
route53DomainsClient, String domainSuggestion) {
       try {
           CheckDomainTransferabilityRequest transferabilityRequest =
CheckDomainTransferabilityRequest.builder()
                   .domainName(domainSuggestion)
```

```
                    .build();

            CheckDomainTransferabilityResponse response = route53DomainsClient
                    .checkDomainTransferability(transferabilityRequest);
            System.out.println("Transferability: " +
 response.transferability().transferable().toString());

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

- For API details, see CheckDomainTransferability in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun checkDomainTransferability(domainSuggestion: String?) {
    val transferabilityRequest =
        CheckDomainTransferabilityRequest {
            domainName = domainSuggestion
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response =
 route53DomainsClient.checkDomainTransferability(transferabilityRequest)
        println("Transferability: ${response.transferability?.transferable}")
    }
}
```

- For API details, see CheckDomainTransferability in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `GetDomainDetail` with an AWS SDK or CLI

The following code examples show how to use `GetDomainDetail`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Get details for a domain.
    /// </summary>
    /// <returns>A string with detail information about the domain.</returns>
    public async Task<string> GetDomainDetail(string domainName)
    {
        try
        {
            var result = await _amazonRoute53Domains.GetDomainDetailAsync(
                new GetDomainDetailRequest()
                {
                    DomainName = domainName
                });
            var details = $"\tDomain {domainName}:\n" +
                          $"\tCreated on
{result.CreationDate.ToShortDateString()}.\n" +
                          $"\tAdmin contact is {result.AdminContact.Email}.\n" +
```

```
                        $"\tAuto-renew is {result.AutoRenew}.\n";

            return details;
        }
        catch (InvalidInputException)
        {
            return $"Domain {domainName} was not found in your account.";
        }
    }
```

- For API details, see GetDomainDetail in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To get detailed information about a specified domain

The following `get-domain-detail` example displays detailed information about the specified domain.

This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains get-domain-detail \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "DomainName": "example.com",
    "Nameservers": [
        {
            "Name": "ns-2048.awsdns-64.com",
            "GlueIps": []
        },
        {
            "Name": "ns-2049.awsdns-65.net",
            "GlueIps": []
        },
```

```
        {
            "Name": "ns-2050.awsdns-66.org",
            "GlueIps": []
        },
        {
            "Name": "ns-2051.awsdns-67.co.uk",
            "GlueIps": []
        }
    ],
    "AutoRenew": true,
    "AdminContact": {
        "FirstName": "Saanvi",
        "LastName": "Sarkar",
        "ContactType": "COMPANY",
        "OrganizationName": "Example",
        "AddressLine1": "123 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "ssarkar@example.com",
        "ExtraParams": []
    },
    "RegistrantContact": {
        "FirstName": "Alejandro",
        "LastName": "Rosalez",
        "ContactType": "COMPANY",
        "OrganizationName": "Example",
        "AddressLine1": "123 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "arosalez@example.com",
        "ExtraParams": []
    },
    "TechContact": {
        "FirstName": "Wang",
        "LastName": "Xiulan",
        "ContactType": "COMPANY",
        "OrganizationName": "Example",
        "AddressLine1": "123 Main Street",
```

```
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "wxiulan@example.com",
        "ExtraParams": []
    },
    "AdminPrivacy": true,
    "RegistrantPrivacy": true,
    "TechPrivacy": true,
    "RegistrarName": "Amazon Registrar, Inc.",
    "WhoIsServer": "whois.registrar.amazon",
    "RegistrarUrl": "http://registrar.amazon.com",
    "AbuseContactEmail": "abuse@registrar.amazon.com",
    "AbuseContactPhone": "+1.2062661000",
    "CreationDate": 1444934889.601,
    "ExpirationDate": 1602787689.0,
    "StatusList": [
        "clientTransferProhibited"
    ]
}
```

For more information, see Registering and managing domains using Amazon Route 53 in the *Amazon Route 53 Developer Guide*.

- For API details, see GetDomainDetail in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void getDomainDetails(Route53DomainsClient
route53DomainsClient, String domainSuggestion) {
       try {
```

```
            GetDomainDetailRequest detailRequest =
GetDomainDetailRequest.builder()
                    .domainName(domainSuggestion)
                    .build();

            GetDomainDetailResponse response =
route53DomainsClient.getDomainDetail(detailRequest);
            System.out.println("The contact first name is " +
response.registrantContact().firstName());
            System.out.println("The contact last name is " +
response.registrantContact().lastName());
            System.out.println("The contact org name is " +
response.registrantContact().organizationName());

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

- For API details, see GetDomainDetail in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun getDomainDetails(domainSuggestion: String?) {
    val detailRequest =
        GetDomainDetailRequest {
            domainName = domainSuggestion
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
{ route53DomainsClient ->
        val response = route53DomainsClient.getDomainDetail(detailRequest)
```

```
        println("The contact first name is
 ${response.registrantContact?.firstName}")
        println("The contact last name is
 ${response.registrantContact?.lastName}")
        println("The contact org name is
 ${response.registrantContact?.organizationName}")
    }
}
```

- For API details, see GetDomainDetail in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `GetDomainSuggestions` with an AWS SDK or CLI

The following code examples show how to use `GetDomainSuggestions`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Get a list of suggestions for a given domain.
    /// </summary>
    /// <param name="domain">The domain to check for suggestions.</param>
```

```
    /// <param name="onlyAvailable">If true, only returns available domains.</
param>
    /// <param name="suggestionCount">The number of suggestions to return.
 Defaults to the max of 50.</param>
    /// <returns>A collection of domain suggestions.</returns>
    public async Task<List<DomainSuggestion>> GetDomainSuggestions(string domain,
 bool onlyAvailable, int suggestionCount = 50)
    {
        var result = await _amazonRoute53Domains.GetDomainSuggestionsAsync(
            new GetDomainSuggestionsRequest
            {
                DomainName = domain,
                OnlyAvailable = onlyAvailable,
                SuggestionCount = suggestionCount
            }
        );
        return result.SuggestionsList;
    }
```

- For API details, see GetDomainSuggestions in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To get a list of suggested domain names

The following `get-domain-suggestions` example displays a list of suggested domain names based on the domain name `example.com`. The response includes only domain names that are available. This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains get-domain-suggestions \
    --region us-east-1 \
    --domain-name example.com \
    --suggestion-count 10 \
    --only-available
```

Output:

```
{
```

```
    "SuggestionsList": [
        {
            "DomainName": "egzaampal.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "examplelaw.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "examplehouse.net",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "homeexample.net",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "examplelist.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "examplenews.net",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "officeexample.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "exampleworld.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "exampleart.com",
            "Availability": "AVAILABLE"
        }
    ]
}
```

For more information, see Registering a new domain in the *Amazon Route 53 Developer Guide*.

- For API details, see GetDomainSuggestions in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void listDomainSuggestions(Route53DomainsClient
route53DomainsClient, String domainSuggestion) {
       try {
           GetDomainSuggestionsRequest suggestionsRequest =
GetDomainSuggestionsRequest.builder()
                   .domainName(domainSuggestion)
                   .suggestionCount(5)
                   .onlyAvailable(true)
                   .build();

           GetDomainSuggestionsResponse response =
route53DomainsClient.getDomainSuggestions(suggestionsRequest);
           List<DomainSuggestion> suggestions = response.suggestionsList();
           for (DomainSuggestion suggestion : suggestions) {
               System.out.println("Suggestion Name: " +
suggestion.domainName());
               System.out.println("Availability: " + suggestion.availability());
               System.out.println(" ");
           }

       } catch (Route53Exception e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
   }
```

- For API details, see GetDomainSuggestions in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun listDomainSuggestions(domainSuggestion: String?) {
    val suggestionsRequest =
        GetDomainSuggestionsRequest {
            domainName = domainSuggestion
            suggestionCount = 5
            onlyAvailable = true
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response =
 route53DomainsClient.getDomainSuggestions(suggestionsRequest)
        response.suggestionsList?.forEach { suggestion ->
            println("Suggestion Name: ${suggestion.domainName}")
            println("Availability: ${suggestion.availability}")
            println(" ")
        }
    }
}
```

- For API details, see GetDomainSuggestions in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `GetOperationDetail` with an AWS SDK or CLI

The following code examples show how to use `GetOperationDetail`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Get details for a domain action operation.
    /// </summary>
    /// <param name="operationId">The operational Id.</param>
    /// <returns>A string describing the operational details.</returns>
    public async Task<string> GetOperationDetail(string? operationId)
    {
        if (operationId == null)
            return "Unable to get operational details because ID is null.";
        try
        {
            var operationDetails =
                await _amazonRoute53Domains.GetOperationDetailAsync(
                    new GetOperationDetailRequest
                    {
                        OperationId = operationId
                    }
                );

            var details = $"\tOperation {operationId}:\n" +
                          $"\tFor domain {operationDetails.DomainName} on
{operationDetails.SubmittedDate.ToShortDateString()}.\n" +
                          $"\tMessage is {operationDetails.Message}.\n" +
                          $"\tStatus is {operationDetails.Status}.\n";
```

```
            return details;
        }
        catch (AmazonRoute53DomainsException ex)
        {
            return $"Unable to get operation details. Here's why: {ex.Message}.";
        }
    }
```

- For API details, see GetOperationDetail in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To get the current status of an operation

The following `get-operation-detail` example returns the status of the specified operation. Some domain registration operations operate asynchronously and return a response before they finish. These operations return an operation ID that you can use to get the current status.

This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains get-operation-detail \
    --region us-east-1 \
    --operation-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Status": "SUCCESSFUL",
    "DomainName": "example.com",
    "Type": "DOMAIN_LOCK",
    "SubmittedDate": 1573749367.864
}
```

For more information, see Viewing the status of a domain registration in the *Amazon Route 53 Developer Guide*.

- For API details, see GetOperationDetail in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void getOperationalDetail(Route53DomainsClient
route53DomainsClient, String operationId) {
       try {
           GetOperationDetailRequest detailRequest =
GetOperationDetailRequest.builder()
                   .operationId(operationId)
                   .build();

           GetOperationDetailResponse response =
route53DomainsClient.getOperationDetail(detailRequest);
           System.out.println("Operation detail message is " +
response.message());

       } catch (Route53Exception e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
   }
```

- For API details, see GetOperationDetail in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun getOperationalDetail(opId: String?) {
    val detailRequest =
        GetOperationDetailRequest {
            operationId = opId
        }
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response = route53DomainsClient.getOperationDetail(detailRequest)
        println("Operation detail message is ${response.message}")
    }
}
```

- For API details, see GetOperationDetail in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `ListDomains` with an AWS SDK or CLI

The following code examples show how to use `ListDomains`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
/// <summary>
/// List the domains for the account.
/// </summary>
/// <returns>A collection of domain summary records.</returns>
public async Task<List<DomainSummary>> ListDomains()
{
    var results = new List<DomainSummary>();
    var paginateDomains = _amazonRoute53Domains.Paginators.ListDomains(
        new ListDomainsRequest());

    // Get the entire list using the paginator.
    await foreach (var domain in paginateDomains.Domains)
    {
        results.Add(domain);
    }
    return results;
}
```

- For API details, see ListDomains in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To list the domains that are registered with the current AWS account

The following `list-domains` example lists summary information about the domains that are registered with the current AWS account.

This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains list-domains \
    --region us-east-1
```

Output:

```
{
    "Domains": [
        {
            "DomainName": "example.com",
            "AutoRenew": true,
            "TransferLock": true,
            "Expiry": 1602712345.0
        },
        {
            "DomainName": "example.net",
            "AutoRenew": true,
            "TransferLock": true,
            "Expiry": 1602723456.0
        },
        {
            "DomainName": "example.org",
            "AutoRenew": true,
            "TransferLock": true,
            "Expiry": 1602734567.0
        }
    ]
}
```

For more information, see Registering and managing domains using Amazon Route 53 in the *Amazon Route 53 Developer Guide*.

- For API details, see ListDomains in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    public static void listDomains(Route53DomainsClient route53DomainsClient) {
        try {
            ListDomainsIterable listRes =
route53DomainsClient.listDomainsPaginator();
            listRes.stream()
                    .flatMap(r -> r.domains().stream())
                    .forEach(content -> System.out.println("The domain name is "
+ content.domainName()));

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

- For API details, see ListDomains in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun listDomains() {
```

```
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .listDomainsPaginated(ListDomainsRequest {})
            .transform { it.domains?.forEach { obj -> emit(obj) } }
            .collect { content ->
                println("The domain name is ${content.domainName}")
            }
    }
}
```

- For API details, see ListDomains in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `ListOperations` with an AWS SDK or CLI

The following code examples show how to use `ListOperations`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// List operations for the account that are submitted after a specified
 date.
```

```
    /// </summary>
    /// <returns>A collection of operation summary records.</returns>
    public async Task<List<OperationSummary>> ListOperations(DateTime
 submittedSince)
    {
        var results = new List<OperationSummary>();
        var paginateOperations = _amazonRoute53Domains.Paginators.ListOperations(
            new ListOperationsRequest()
            {
                SubmittedSince = submittedSince
            });

        // Get the entire list using the paginator.
        await foreach (var operations in paginateOperations.Operations)
        {
            results.Add(operations);
        }
        return results;
    }
```

- For API details, see ListOperations in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To list the status of operations that return an operation ID

The following `list-operations` example lists summary information, including the status, about the current domain-registration operations. Some domain registration operations run asynchronously and return a response before they finish. These operations return an operation ID that you can use to get the current status.

This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains list-operations \
    --region us-east-1
```

Output:

```
{
    "Operations": [
        {
            "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Status": "SUCCESSFUL",
            "Type": "DOMAIN_LOCK",
            "SubmittedDate": 1455321739.986
        },
        {
            "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "Status": "SUCCESSFUL",
            "Type": "UPDATE_NAMESERVER",
            "SubmittedDate": 1468960475.109
        },
        {
            "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "Status": "SUCCESSFUL",
            "Type": "RENEW_DOMAIN",
            "SubmittedDate": 1473561835.943
        },
        {
            "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
            "Status": "SUCCESSFUL",
            "Type": "UPDATE_DOMAIN_CONTACT",
            "SubmittedDate": 1547501003.41
        }
    ]
}
```


The output includes all the operations that return an operation ID and that you have performed on all the domains that you have ever registered using the current AWS account. If you want to get only the operations that you submitted after a specified date, you can include the `submitted-since` parameter and specify a date in Unix format and Coordinated Universal Time (UTC). The following command gets the status of all operations that were submitted after 12:00 am UTC on January 1, 2020.

```
aws route53domains list-operations \
    --submitted-since 1577836800
```

For more information, see Viewing the status of a domain registration in the *Amazon Route 53 Developer Guide*.

- For API details, see ListOperations in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void listOperations(Route53DomainsClient route53DomainsClient)
{
       try {
           Date currentDate = new Date();
           LocalDateTime localDateTime =
currentDate.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime();
           ZoneOffset zoneOffset = ZoneOffset.of("+01:00");
           localDateTime = localDateTime.minusYears(1);
           Instant myTime = localDateTime.toInstant(zoneOffset);

           ListOperationsRequest operationsRequest =
ListOperationsRequest.builder()
                   .submittedSince(myTime)
                   .build();

           ListOperationsIterable listRes =
route53DomainsClient.listOperationsPaginator(operationsRequest);
           listRes.stream()
                   .flatMap(r -> r.operations().stream())
                   .forEach(content -> System.out.println(" Operation Id: " +
content.operationId() +
                           " Status: " + content.statusAsString() +
                           " Date: " + content.submittedDate()));

       } catch (Route53Exception e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
   }
```

- For API details, see ListOperations in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun listOperations() {
    val currentDate = Date()
    var localDateTime =
 currentDate.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime()
    val zoneOffset = ZoneOffset.of("+01:00")
    localDateTime = localDateTime.minusYears(1)
    val myTime: java.time.Instant? = localDateTime.toInstant(zoneOffset)
    val time2: Instant? = myTime?.let { Instant(it) }
    val operationsRequest =
        ListOperationsRequest {
            submittedSince = time2
        }

    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .listOperationsPaginated(operationsRequest)
            .transform { it.operations?.forEach { obj -> emit(obj) } }
            .collect { content ->
                println("Operation Id: ${content.operationId}")
                println("Status: ${content.status}")
                println("Date: ${content.submittedDate}")
            }
    }
}
```

- For API details, see ListOperations in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `ListPrices` with an AWS SDK

The following code examples show how to use `ListPrices`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// List prices for domain type operations.
    /// </summary>
    /// <param name="domainTypes">Domain types to include in the results.</param>
    /// <returns>The list of domain prices.</returns>
    public async Task<List<DomainPrice>> ListPrices(List<string> domainTypes)
    {
        var results = new List<DomainPrice>();
        var paginatePrices = _amazonRoute53Domains.Paginators.ListPrices(new
ListPricesRequest());
        // Get the entire list using the paginator.
        await foreach (var prices in paginatePrices.Prices)
        {
            results.Add(prices);
        }
```

```
        return results.Where(p => domainTypes.Contains(p.Name)).ToList();
    }
```

- For API details, see ListPrices in *AWS SDK for .NET API Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void listPrices(Route53DomainsClient route53DomainsClient,
String domainType) {
       try {
           ListPricesRequest pricesRequest = ListPricesRequest.builder()
                   .tld(domainType)
                   .build();

           ListPricesIterable listRes =
route53DomainsClient.listPricesPaginator(pricesRequest);
           listRes.stream()
                   .flatMap(r -> r.prices().stream())
                   .forEach(content -> System.out.println(" Name: " +
content.name() +
                           " Registration: " +
content.registrationPrice().price() + " "
                           + content.registrationPrice().currency() +
                           " Renewal: " + content.renewalPrice().price() + " " +
content.renewalPrice().currency()));

       } catch (Route53Exception e) {
           System.err.println(e.getMessage());
           System.exit(1);
       }
   }
```

- For API details, see ListPrices in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun listAllPrices(domainType: String?) {
    val pricesRequest =
        ListPricesRequest {
            tld = domainType
        }

    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .listPricesPaginated(pricesRequest)
            .transform { it.prices?.forEach { obj -> emit(obj) } }
            .collect { pr ->
                println("Registration: ${pr.registrationPrice}
 ${pr.registrationPrice?.currency}")
                println("Renewal: ${pr.renewalPrice?.price}
 ${pr.renewalPrice?.currency}")
                println("Transfer: ${pr.transferPrice?.price}
 ${pr.transferPrice?.currency}")
                println("Restoration: ${pr.restorationPrice?.price}
 ${pr.restorationPrice?.currency}")
            }
    }
}
```

- For API details, see ListPrices in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `RegisterDomain` with an AWS SDK or CLI

The following code examples show how to use `RegisterDomain`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// Initiate a domain registration request.
    /// </summary>
    /// <param name="contact">Contact details.</param>
    /// <param name="domainName">The domain name to register.</param>
    /// <param name="autoRenew">True if the domain should automatically renew.</
param>
    /// <param name="duration">The duration in years for the domain
 registration.</param>
    /// <returns>The operation Id.</returns>
    public async Task<string?> RegisterDomain(string domainName, bool autoRenew,
 int duration, ContactDetail contact)
    {
        // This example uses the same contact information for admin, registrant,
 and tech contacts.
        try
        {
```

```
            var result = await _amazonRoute53Domains.RegisterDomainAsync(
                new RegisterDomainRequest()
                {
                    AdminContact = contact,
                    RegistrantContact = contact,
                    TechContact = contact,
                    DomainName = domainName,
                    AutoRenew = autoRenew,
                    DurationInYears = duration,
                    PrivacyProtectAdminContact = false,
                    PrivacyProtectRegistrantContact = false,
                    PrivacyProtectTechContact = false
                }
            );
            return result.OperationId;
        }
        catch (InvalidInputException)
        {
            _logger.LogInformation($"Unable to request registration for domain
{domainName}");
            return null;
        }
    }
```

- For API details, see RegisterDomain in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To register a domain

The following `register-domain` example registers a domain, retrieving all parameter values from a JSON-formatted file.

This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains register-domain \
    --region us-east-1 \
    --cli-input-json file://register-domain.json
```

Contents of `register-domain.json`:

```
{
    "DomainName": "example.com",
    "DurationInYears": 1,
    "AutoRenew": true,
    "AdminContact": {
        "FirstName": "Martha",
        "LastName": "Rivera",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "mrivera@example.com"
    },
    "RegistrantContact": {
        "FirstName": "Li",
        "LastName": "Juan",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "ljuan@example.com"
    },
    "TechContact": {
        "FirstName": "Mateo",
        "LastName": "Jackson",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "mjackson@example.com"
```

```
    },
    "PrivacyProtectAdminContact": true,
    "PrivacyProtectRegistrantContact": true,
    "PrivacyProtectTechContact": true
}
```

Output:

```
{
    "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

To confirm that the operation succeeded, you can run `get-operation-detail`. For more information, see get-operation-detail .

For information about which top-level domains (TLDs) require values for `ExtraParams` and what the valid values are, see Domains that you can register with Amazon Route 53 in the *Amazon Route 53 Developer Guide*.

For more information, see Registering a New Domain in the *Amazon Route 53 Developer Guide*.

- For API details, see RegisterDomain in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static String requestDomainRegistration(Route53DomainsClient
route53DomainsClient,
           String domainSuggestion,
           String phoneNumber,
           String email,
           String firstName,
```

```
            String lastName,
            String city) {

        try {
            ContactDetail contactDetail = ContactDetail.builder()
                    .contactType(ContactType.COMPANY)
                    .state("LA")
                    .countryCode(CountryCode.IN)
                    .email(email)
                    .firstName(firstName)
                    .lastName(lastName)
                    .city(city)
                    .phoneNumber(phoneNumber)
                    .organizationName("My Org")
                    .addressLine1("My Address")
                    .zipCode("123 123")
                    .build();

            RegisterDomainRequest domainRequest = RegisterDomainRequest.builder()
                    .adminContact(contactDetail)
                    .registrantContact(contactDetail)
                    .techContact(contactDetail)
                    .domainName(domainSuggestion)
                    .autoRenew(true)
                    .durationInYears(1)
                    .build();

            RegisterDomainResponse response =
route53DomainsClient.registerDomain(domainRequest);
            System.out.println("Registration requested. Operation Id: " +
response.operationId());
            return response.operationId();

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
        return "";
    }
```

- For API details, see RegisterDomain in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun requestDomainRegistration(
    domainSuggestion: String?,
    phoneNumberVal: String?,
    emailVal: String?,
    firstNameVal: String?,
    lastNameVal: String?,
    cityVal: String?,
): String? {
    val contactDetail =
        ContactDetail {
            contactType = ContactType.Company
            state = "LA"
            countryCode = CountryCode.In
            email = emailVal
            firstName = firstNameVal
            lastName = lastNameVal
            city = cityVal
            phoneNumber = phoneNumberVal
            organizationName = "My Org"
            addressLine1 = "My Address"
            zipCode = "123 123"
        }

    val domainRequest =
        RegisterDomainRequest {
            adminContact = contactDetail
            registrantContact = contactDetail
            techContact = contactDetail
            domainName = domainSuggestion
            autoRenew = true
            durationInYears = 1
        }
```

```
    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        val response = route53DomainsClient.registerDomain(domainRequest)
        println("Registration requested. Operation Id: ${response.operationId}")
        return response.operationId
    }
}
```

- For API details, see RegisterDomain in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Use `ViewBilling` with an AWS SDK or CLI

The following code examples show how to use `ViewBilling`.

Action examples are code excerpts from larger programs and must be run in context. You can see this action in context in the following code example:

- Learn the basics

.NET

SDK for .NET

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
    /// <summary>
    /// View billing records for the account between a start and end date.
    /// </summary>
    /// <param name="startDate">The start date for billing results.</param>
    /// <param name="endDate">The end date for billing results.</param>
```

```
    /// <returns>A collection of billing records.</returns>
    public async Task<List<BillingRecord>> ViewBilling(DateTime startDate,
 DateTime endDate)
    {
        var results = new List<BillingRecord>();
        var paginateBilling = _amazonRoute53Domains.Paginators.ViewBilling(
            new ViewBillingRequest()
            {
                Start = startDate,
                End = endDate
            });

        // Get the entire list using the paginator.
        await foreach (var billingRecords in paginateBilling.BillingRecords)
        {
            results.Add(billingRecords);
        }
        return results;
    }
```

- For API details, see ViewBilling in *AWS SDK for .NET API Reference*.

CLI

AWS CLI

To get billing information for domain registration charges for the current AWS account

The following `view-billing` example returns all the domain-related billing records for the current account for the period from January 1, 2018 (1514764800 in Unix time) and midnight on December 31, 2019 (1577836800 in Unix time).

This command runs only in the `us-east-1` Region. If your default region is set to `us-east-1`, you can omit the `region` parameter.

```
aws route53domains view-billing \
    --region us-east-1 \
    --start-time 1514764800 \
    --end-time 1577836800
```

Output:

```
{
    "BillingRecords": [
        {
            "DomainName": "example.com",
            "Operation": "RENEW_DOMAIN",
            "InvoiceId": "149962827",
            "BillDate": 1536618063.181,
            "Price": 12.0
        },
        {
            "DomainName": "example.com",
            "Operation": "RENEW_DOMAIN",
            "InvoiceId": "290913289",
            "BillDate": 1568162630.884,
            "Price": 12.0
        }
    ]
}
```

For more information, see Downloading a domain billing report in the *Amazon Route 53 Developer Guide*.

- For API details, see ViewBilling in *AWS CLI Command Reference*.

Java

SDK for Java 2.x

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
   public static void listBillingRecords(Route53DomainsClient
route53DomainsClient) {
       try {
           Date currentDate = new Date();
           LocalDateTime localDateTime =
currentDate.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime();
           ZoneOffset zoneOffset = ZoneOffset.of("+01:00");
```

```
            LocalDateTime localDateTime2 = localDateTime.minusYears(1);
            Instant myStartTime = localDateTime2.toInstant(zoneOffset);
            Instant myEndTime = localDateTime.toInstant(zoneOffset);

            ViewBillingRequest viewBillingRequest = ViewBillingRequest.builder()
                    .start(myStartTime)
                    .end(myEndTime)
                    .build();

            ViewBillingIterable listRes =
 route53DomainsClient.viewBillingPaginator(viewBillingRequest);
            listRes.stream()
                    .flatMap(r -> r.billingRecords().stream())
                    .forEach(content -> System.out.println(" Bill Date:: " +
 content.billDate() +
                            " Operation: " + content.operationAsString() +
                            " Price: " + content.price()));

        } catch (Route53Exception e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

- For API details, see ViewBilling in *AWS SDK for Java 2.x API Reference*.

Kotlin

SDK for Kotlin

> **Note**
>
> There's more on GitHub. Find the complete example and learn how to set up and run in the AWS Code Examples Repository.

```
suspend fun listBillingRecords() {
    val currentDate = Date()
    val localDateTime =
 currentDate.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime()
    val zoneOffset = ZoneOffset.of("+01:00")
```

```
    val localDateTime2 = localDateTime.minusYears(1)
    val myStartTime = localDateTime2.toInstant(zoneOffset)
    val myEndTime = localDateTime.toInstant(zoneOffset)
    val timeStart: Instant? = myStartTime?.let { Instant(it) }
    val timeEnd: Instant? = myEndTime?.let { Instant(it) }

    val viewBillingRequest =
        ViewBillingRequest {
            start = timeStart
            end = timeEnd
        }

    Route53DomainsClient.fromEnvironment { region = "us-east-1" }.use
 { route53DomainsClient ->
        route53DomainsClient
            .viewBillingPaginated(viewBillingRequest)
            .transform { it.billingRecords?.forEach { obj -> emit(obj) } }
            .collect { billing ->
                println("Bill Date: ${billing.billDate}")
                println("Operation: ${billing.operation}")
                println("Price: ${billing.price}")
            }
    }
}
```

- For API details, see ViewBilling in *AWS SDK for Kotlin API reference*.

For a complete list of AWS SDK developer guides and code examples, see Using Route 53 with an AWS SDK. This topic also includes information about getting started and details about previous SDK versions.

Security in Amazon Route 53

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security *of* the cloud and security *in* the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS compliance programs. To learn about the compliance programs that apply to Amazon Route 53, see AWS Services in Scope by Compliance Program.
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using Route 53. The following topics show you how to configure Route 53 to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Route 53 resources.

Topics

- Data protection in Route 53
- Identity and access management in Amazon Route 53
- Logging and monitoring in Amazon Route 53
- Compliance validation for Amazon Route 53
- Resilience in Amazon Route 53
- Infrastructure security in Amazon Route 53

Data protection in Route 53

The AWS shared responsibility model applies to data protection in Amazon Route 53. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the

AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. You are also responsible for the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see Data Privacy FAQ. For information about data protection in Europe, see the General Data Protection Regulation (GDPR) Center.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual users with AWS IAM Identity Center or AWS Identity and Access Management (IAM). That way, each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail. For information about using CloudTrail trails to capture AWS activities, see Working with CloudTrail trails in the *AWS CloudTrail User Guide*.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3.
- If you require FIPS 140-3 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-3.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form text fields such as a **Name** field. This includes when you work with Route 53 or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form text fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

Protection from dangling delegation records in Route 53

With Route 53, a customer can create a hosted zone, such as `example.com`, to host their DNS records. Each hosted zone comes with a "delegation set", which is a set of four name servers that a customer can use to configure NS records in the parent domain. These NS records can be called "delegation NS records", or "delegation records".

In order for the `example.com` Route 53 hosted zone to become authoritative, the rightful owner of the `example.com` domain needs to configure delegation records in their ".com" parent domain through the domain registrar. In cases where a customer loses access to the four name servers configured in the parent domain, for example because the associated hosted zone is deleted, it can create a risk that an attacker can exploit. This is referred to as a "dangling delegation records" risk.

Route 53 protects against the dangling delegation record risk in the case where a hosted zone is deleted. After deletion, if a new hosted zone is being created with the same domain name, Route 53 will check if the delegation records pointing to the deleted hosted zone are still present in the parent domain. If they are, Route 53 will prevent any overlapping name servers from being assigned. This is scenario 1 in the following examples.

However, there are other dangling delegation record risks, which Route 53 can't protect against, as detailed in scenarios 2 through 5 in the following examples. To protect yourself against this broader set of risks, make sure the parent NS records match the delegation set for the Route 53 hosted zone. You can find the delegation set of a hosted zone through the Route 53 console or AWS CLI. For more information, see Listing records or get-hosted-zone.

Additionally, enabling DNSSEC signing for a Route 53 hosted zone can serve as another layer of protection beyond the best practices mentioned above. DNSSEC authenticates that DNS answers come from the authoritative source, effectively protecting against this risk. For more information see Configuring DNSSEC signing in Amazon Route 53.

Examples

In the following examples, we assume you have a domain `example.com`, and its child domain `child.example.com`. We will explain how in various scenarios dangling delegation records can get created, how Route 53 protects your domain against abuse and how to effectively mitigate the risks associated with dangling delegation records.

Scenario 1:

You create a hosted zone `child.example.com` with four name servers: <ns1>, <ns2>, <ns3>, and <ns4>. You properly setup the delegation in hosted zone `example.com`, creating delegation NS records for `child.example.com` with four name servers <ns1>, <ns2>, <ns3>, and <ns4>. When `child.example.com` hosted zone gets deleted without removing the delegation NS records in `example.com`, Route 53 protects `child.example.com` from dangling delegation records risk by preventing <ns1>, <ns2>, <ns3>, and <ns4> from being assigned to newly created hosted zones with the same domain name.

Scenario 2:

Similar to scenario 1, but this time you delete child hosted zone AND the delegation NS records in hosted zone `example.com`. However, you add back delegation NS records <ns1>, <ns2>, <ns3>, and <ns4> without creating a child hosted zone. Here, <ns1>, <ns2>, <ns3>, and <ns4> are dangling delegation records, because Route 53 removes the hold, which was preventing <ns1>, <ns2>, <ns3>, and <ns4> from being assigned and will now allow newly created hosted zones to use above name servers. To mitigate the risk, remove <ns1>, <ns2>, <ns3>, and <ns4> from the delegation records and only add them back after the child hosted zone has been created.

Scenario 3:

In this scenario, you create a Route 53 reusable delegation set with name servers <ns1>, <ns2>, <ns3>, and <ns4>. Then, you delegate the domain `example.com` to these name servers in the parent domain `.com`. However, you haven't created the hosted zone for `example.com` on the reusable delegation set yet. Here, <ns1>, <ns2>, <ns3>, and <ns4> are dangling delegation records. To mitigate the risk, create the hosted zone using the reusable delegation set with name servers <ns1>, <ns2>, <ns3>, and <ns4>.

Scenario 4:

You create hosted zones for both `child.example.com` with name servers <ns1>, <ns2>, <ns3>, and <ns4>, and `grandchild.child.example.com` with name servers <ns5>, <ns6>, <ns7>, and <ns8>. However, you delegate both directly in the `example.com` zone, which creates a dangling delegation risk. To make sure that delegations follow proper DNS hierarchy, only delegate subdomains through their immediate parent zones. For example, if you want to delegate `grandchild.child.example.com`: first delegate `child.example.com` with name servers <ns1>, <ns2>, <ns3>, and <ns4> in the `example.com` zone, then delegate `grandchild.child.example.com` with name servers <ns5>, <ns6>, <ns7>, and <ns8> in the `child.example.com` zone, and remove any direct delegations for `grandchild.child.example.com` from the `example.com` zone.

Scenario 5:

You delegate a domain or subdomain to Route 53 name servers before creating a corresponding hosted zone, this creates dangling delegation records. This is similar to the case in Scenario 3, but the risk also applies when no reusable delegation set is created. For example, you delegate the domain `example.com` to name servers <ns1>, <ns2>, <ns3>, and <ns4> in the parent domain `.com`, but none of these name servers has ever hosted `example.com`. Route 53 cannot

protect against this because no hosted zone has ever existed to establish a hold on those name servers for that domain name. To mitigate the risk, only delegate to Route 53 name servers that belong to a public hosted zone that you control.

Identity and access management in Amazon Route 53

To perform any operation on Amazon Route 53 resources, such as registering a domain or updating a record, AWS Identity and Access Management (IAM) requires you to authenticate that you're an approved AWS user. If you're using the Route 53 console, you authenticate your identity by providing your AWS user name and a password.

After you authenticate your identity, IAM controls your access to AWS by verifying that you have permissions to perform operations and to access resources. If you are an account administrator, you can use IAM to control the access of other users to the resources that are associated with your account.

This chapter explains how to use IAM and Route 53 to help secure your resources.

Topics

- Authenticating with identities
- Access control
- Using Service-Linked Roles for Amazon Route 53 Resolver
- Using service-linked roles for Amazon Route 53 Private endpoint health checks
- AWS managed policies for Amazon Route 53
- Using IAM policy conditions for fine-grained access control
- Amazon Route 53 API permissions: Actions, resources, and conditions reference

Authenticating with identities

Authentication is how you sign in to AWS using your identity credentials. You must be authenticated as the AWS account root user, an IAM user, or by assuming an IAM role.

You can sign in as a federated identity using credentials from an identity source like AWS IAM Identity Center (IAM Identity Center), single sign-on authentication, or Google/Facebook credentials. For more information about signing in, see How to sign in to your AWS account in the *AWS Sign-In User Guide*.

For programmatic access, AWS provides an SDK and CLI to cryptographically sign requests. For more information, see AWS Signature Version 4 for API requests in the *IAM User Guide*.

AWS account root user

When you create an AWS account, you begin with one sign-in identity called the AWS account *root user* that has complete access to all AWS services and resources. We strongly recommend that you don't use the root user for everyday tasks. For tasks that require root user credentials, see Tasks that require root user credentials in the *IAM User Guide*.

Federated identity

As a best practice, require human users to use federation with an identity provider to access AWS services using temporary credentials.

A *federated identity* is a user from your enterprise directory, web identity provider, or Directory Service that accesses AWS services using credentials from an identity source. Federated identities assume roles that provide temporary credentials.

For centralized access management, we recommend AWS IAM Identity Center. For more information, see What is IAM Identity Center? in the *AWS IAM Identity Center User Guide*.

IAM users and groups

An *IAM user* is an identity with specific permissions for a single person or application. We recommend using temporary credentials instead of IAM users with long-term credentials. For more information, see Require human users to use federation with an identity provider to access AWS using temporary credentials in the *IAM User Guide*.

An *IAM group* specifies a collection of IAM users and makes permissions easier to manage for large sets of users. For more information, see Use cases for IAM users in the *IAM User Guide*.

IAM roles

An *IAM role* is an identity with specific permissions that provides temporary credentials. You can assume a role by switching from a user to an IAM role (console) or by calling an AWS CLI or AWS API operation. For more information, see Methods to assume a role in the *IAM User Guide*.

IAM roles are useful for federated user access, temporary IAM user permissions, cross-account access, cross-service access, and applications running on Amazon EC2. For more information, see Cross account resource access in IAM in the *IAM User Guide*.

Access control

To create, update, delete, or list Amazon Route 53 resources, you need permissions to perform the operation, and you need permission to access the corresponding resources.

The following sections describe how to manage permissions for Route 53. We recommend that you read the overview first.

Topics

- Overview of managing access permissions to your Amazon Route 53 resources
- Using identity-based policies (IAM policies) for Amazon Route 53

Overview of managing access permissions to your Amazon Route 53 resources

Every AWS resource is owned by an AWS account, and permissions to create or access a resource are governed by permissions policies.

> **Note**
>
> An *account administrator* (or administrator user) is a user that has administrator privileges. For more information about administrators, see IAM best practices in the *IAM User Guide*.

When you grant permissions, you decide who gets the permissions, the resources they get permissions for, and the actions that they get permissions to perform.

Users need programmatic access if they want to interact with AWS outside of the AWS Management Console. The way to grant programmatic access depends on the type of user that's accessing AWS.

To grant users programmatic access, choose one of the following options.

Which user needs programmatic access?	To	By
IAM	(Recommended) Use console credentials as temporary credentials to sign	Following the instructions for the interface that you want to use.

Which user needs programmatic access?	To	By
	programmatic requests to the AWS CLI, AWS SDKs, or AWS APIs.	• For the AWS CLI, see Login for AWS local development in the *AWS Command Line Interface User Guide*. • For AWS SDKs, see Login for AWS local development in the *AWS SDKs and Tools Reference Guide*.
Workforce identity (Users managed in IAM Identity Center)	Use temporary credentials to sign programmatic requests to the AWS CLI, AWS SDKs, or AWS APIs.	Following the instructions for the interface that you want to use. • For the AWS CLI, see Configuring the AWS CLI to use AWS IAM Identity Center in the *AWS Command Line Interface User Guide*. • For AWS SDKs, tools, and AWS APIs, see IAM Identity Center authentication in the *AWS SDKs and Tools Reference Guide*.
IAM	Use temporary credentials to sign programmatic requests to the AWS CLI, AWS SDKs, or AWS APIs.	Following the instructions in Using temporary credentials with AWS resources in the *IAM User Guide*.

Which user needs programmatic access?	To	By
IAM	(Not recommended) Use long-term credentials to sign programmatic requests to the AWS CLI, AWS SDKs, or AWS APIs.	Following the instructions for the interface that you want to use. • For the AWS CLI, see Authenticating using IAM user credentials in the *AWS Command Line Interface User Guide*. • For AWS SDKs and tools, see Authenticate using long-term credentials in the *AWS SDKs and Tools Reference Guide*. • For AWS APIs, see Managing access keys for IAM users in the *IAM User Guide*.

Topics

- ARNs for Amazon Route 53 resources
- Understanding resource ownership
- Managing access to resources
- Specifying policy elements: Resources, actions, effects, and principals
- Specifying conditions in a policy

ARNs for Amazon Route 53 resources

Amazon Route 53 supports a variety of resource types for DNS, health checking, and domain registration. In a policy, you can grant or deny access to the following resources by using * for the ARN:

- Health checks
- Hosted zones
- Reusable delegation sets
- Status of a resource record set change batch (API only)
- Traffic policies (traffic flow)
- Traffic policy instances (traffic flow)

Not all Route 53 resources support permissions. You can't grant or deny access to the following resources:

- Domains
- Individual records
- Tags for domains
- Tags for health checks
- Tags for hosted zones

Route 53 provides API actions to work with each of these types of resources. For more information, see the Amazon Route 53 API Reference. For a list of actions and the ARN that you specify to grant or deny permission to use each action, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

Understanding resource ownership

An AWS account owns the resources that are created in the account, regardless of who created the resources. Specifically, the resource owner is the AWS account of the principal entity (that is, the root account, or an IAM role) that authenticates the resource creation request.

The following examples illustrate how this works:

- If you use the root account credentials of your AWS account to create a hosted zone, your AWS account is the owner of the resource.
- If you create a user in your AWS account and grant permissions to create a hosted zone to that user, the user can create a hosted zone. However, your AWS account, to which the user belongs, owns the hosted zone resource.

- If you create an IAM role in your AWS account with permissions to create a hosted zone, anyone who can assume the role can create a hosted zone. Your AWS account, to which the role belongs, owns the hosted zone resource.

Managing access to resources

A *permissions policy* specifies who has access to what. This section explains the options for creating permissions policies for Amazon Route 53. For general information about IAM policy syntax and descriptions, see the AWS IAM Policy Reference in the *IAM User Guide*.

Policies attached to an IAM identity are referred to as *identity-based* policies (IAM policies), and policies attached to a resource are referred to as *resource-based* policies. Route 53 supports only identity-based policies (IAM policies).

Topics

- Identity-based policies (IAM policies)
- Resource-based policies

Identity-based policies (IAM policies)

You can attach policies to IAM identities. For example, you can do the following:

- **Attach a permissions policy to a user or a group in your account** – An account administrator can use a permissions policy that is associated with a particular user to grant permissions for that user to create Amazon Route 53 resources.
- **Attach a permissions policy to a role (grant cross-account permissions)** – You can grant permission to perform Route 53 actions to a user that was created by another AWS account. To do so, you attach a permissions policy to an IAM role, and then you allow the user in the other account to assume the role. The following example explains how this works for two AWS accounts, account A and account B:
  1. Account A administrator creates an IAM role and attaches to the role a permissions policy that grants permissions to create or access resources that are owned by account A.
  2. Account A administrator attaches a trust policy to the role. The trust policy identifies account B as the principal that can assume the role.
  3. Account B administrator can then delegate permissions to assume the role to users or groups in Account B. This allows users in account B to create or access resources in account A.

For more information about how to delegate permissions to users in another AWS account, see Access management in the *IAM User Guide*.

The following example policy allows a user to perform the `CreateHostedZone` action to create a public hosted zone for any AWS account:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "route53:CreateHostedZone"
            ],
            "Resource":"*"
        }
    ]
}
```

If you want the policy to also apply to private hosted zones, you need to grant permissions to use the Route 53 `AssociateVPCWithHostedZone` action and two Amazon EC2 actions, `DescribeVpcs` and `DescribeRegion`, as shown in the following example:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "route53:CreateHostedZone",
                "route53:AssociateVPCWithHostedZone"
            ],
            "Resource": "*"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeVpcs",
                "ec2:DescribeRegions"
            ],
            "Resource": "*"
        }
    ]
}
```

For more information about attaching policies to identities for Route 53, see Using identity-based policies (IAM policies) for Amazon Route 53. For more information about users, groups, roles, and permissions, see Identities (users, groups, and roles) in the *IAM User Guide*.

Resource-based policies

Other services, such as Amazon S3, also support attaching permissions policies to resources. For example, you can attach a policy to an S3 bucket to manage access permissions to that bucket. Amazon Route 53 doesn't support attaching policies to resources.

Specifying policy elements: Resources, actions, effects, and principals

Amazon Route 53 includes API actions (see the Amazon Route 53 API Reference) that you can use on each Route 53 resource (see ARNs for Amazon Route 53 resources). You can grant a user or a federated user permissions to perform any or all of these actions. Note that some API actions, such as registering a domain, require permissions to perform more than one action.

The following are the basic policy elements:

- **Resource** – You use an Amazon Resource Name (ARN) to identify the resource that the policy applies to. For more information, see ARNs for Amazon Route 53 resources.
- **Action** – You use action keywords to identify resource operations that you want to allow or deny. For example, depending on the specified `Effect`, the `route53:CreateHostedZone` permission allows or denies a user the ability to perform the Route 53 `CreateHostedZone` action.
- **Effect** – You specify the effect, either allow or deny, when a user tries to perform the action on the specified resource. If you don't explicitly grant access to an action, access is implicitly denied. You can also explicitly deny access to a resource, which you might do to make sure that a user cannot access it, even if a different policy grants access.

- **Principal** – In identity-based policies (IAM policies), the user that the policy is attached to is the implicit principal. For resource-based policies, you specify the user, account, service, or other entity that you want to receive permissions (applies to resource-based policies only). Route 53 doesn't support resource-based policies.

For more information about IAM policy syntax and descriptions, see the AWS IAM Policy Reference in the *IAM User Guide*.

For a list showing all of the Route 53 API operations and the resources that they apply to, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

Specifying conditions in a policy

When you grant permissions, you can use the IAM policy language to specify when a policy should take effect. For example, you might want a policy to be applied only after a specific date. For more information about specifying conditions in a policy language, see IAM JSON policy elements: Condition in the *IAM User Guide*.

To express conditions, you use predefined condition keys. Route 53 supports service-specific condition keys for some actions. For more information, see Using IAM policy conditions for fine-grained access control. You can also use AWS wide condition keys as needed. For a complete list of AWS wide keys, see Available keys for conditions in the *IAM User Guide*.

The record set protection actions (`route53:CreateResourceRecordSetProtections`, `route53:DeleteResourceRecordSetProtections`, and `route53:ListResourceRecordSetProtections`) support only the hosted zone resource and don't support service-specific condition keys. For more information about permissions for record set protections, see Permissions for record set protections.

The Change History actions (`route53:ListHostedZoneChangeBatches` and `route53:GetHostedZoneChangeBatch`) support only the hosted zone resource and don't support service-specific condition keys. `GetHostedZoneChangeBatch` and `ListResourceRecordSetChanges` also require `route53:ListResourceRecordSets`. For more information about permissions for Change History, see Permissions for Change History.

Using identity-based policies (IAM policies) for Amazon Route 53

This topic provides examples of identity-based policies that demonstrate how an account administrator can attach permissions policies to IAM identities and thereby grant permissions to perform operations on Amazon Route 53 resources.

Important

We recommend that you first review the introductory topics that explain the basic concepts and options to manage access to your Route 53 resources. For more information, see Overview of managing access permissions to your Amazon Route 53 resources.

Note

When granting access, the hosted zone and the Amazon VPC must belong to the same partition. A partition is a group of AWS Regions. Each AWS account is scoped to one partition.
The following are the supported partitions:

- `aws` - AWS Regions
- `aws-cn` - China Regions
- `aws-us-gov` - AWS GovCloud (US) Region

For more information, see Access Management and Amazon Route 53 endpoints and quotas in the *AWS General Reference*.

Topics

- Permissions required to use the Amazon Route 53 console
- Example permissions for a domain record owner
- Route 53 customer managed key permissions required for DNSSEC signing
- Customer managed policy examples

The following example shows a permissions policy. The `Sid`, or statement ID, is optional:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Sid" : "AllowPublicHostedZonePermissions",
            "Effect": "Allow",
            "Action": [
                "route53:CreateHostedZone",
                "route53:UpdateHostedZoneComment",
                "route53:GetHostedZone",
                "route53:ListHostedZones",
                "route53:DeleteHostedZone",
                "route53:ChangeResourceRecordSets",
                "route53:ListResourceRecordSets",
                "route53:GetHostedZoneCount",
                "route53:ListHostedZonesByName"
            ],
            "Resource": "*"
        },
        {
         "Sid" : "AllowHealthCheckPermissions",
            "Effect": "Allow",
            "Action": [
                "route53:CreateHealthCheck",
                "route53:UpdateHealthCheck",
                "route53:GetHealthCheck",
                "route53:ListHealthChecks",
                "route53:DeleteHealthCheck",
                "route53:GetCheckerIpRanges",
                "route53:GetHealthCheckCount",
                "route53:GetHealthCheckStatus",
                "route53:GetHealthCheckLastFailureReason"
            ],
            "Resource": "*"
        }
    ]
}
```

The policy includes two statements:

- The first statement grants permissions to the actions that are required to create and manage public hosted zones and their records. The wildcard character (*) in the Amazon Resource Name (ARN) grants access to all the hosted zones that are owned by the current AWS account.
- The second statement grants permissions to all the actions that are required to create and manage health checks.

For a list of actions and the ARN that you specify to grant or deny permission to use each action, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

Permissions required to use the Amazon Route 53 console

To grant full access to the Amazon Route 53 console, you grant the permissions in the following permissions policy:

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Effect":"Allow",
            "Action":[
                "route53:*",
                "route53domains:*",
                "tag:*",
                "ssm:GetParametersByPath",
                "cloudfront:ListDistributions",
                "elasticloadbalancing:DescribeLoadBalancers",
                "elasticbeanstalk:DescribeEnvironments",
                "s3:ListAllMyBuckets",
                "s3:GetBucketLocation",
                "s3:GetBucketWebsite",
                "ec2:DescribeRegions",
                "ec2:DescribeVpcs",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:ModifyNetworkInterfaceAttribute",
                "sns:ListTopics",
                "sns:ListSubscriptionsByTopic",
                "sns:CreateTopic",
                "kms:ListAliases",
                "kms:DescribeKey",
                "kms:CreateKey",
                "kms:CreateAlias",
```

```
                "kms:Sign",
                "cloudwatch:DescribeAlarms",
                "cloudwatch:PutMetricAlarm",
                "cloudwatch:DeleteAlarms",
                "cloudwatch:GetMetricStatistics"
            ],
            "Resource":"*"
        },
        {
            "Effect": "Allow",
            "Action": "apigateway:GET",
            "Resource": "arn:aws:apigateway:*::/domainnames"
        }
    ]
}
```

Here's why the permissions are required:

`route53:*`

Lets you perform all Route 53 actions *except* the following:

- Create and update alias records for which the value of **Alias Target** is a CloudFront distribution, an Elastic Load Balancing load balancer, an Elastic Beanstalk environment, or an Amazon S3 bucket. (With these permissions, you can create alias records for which the value of **Alias Target** is another record in the same hosted zone.)
- Work with private hosted zones.
- Work with domains.
- Create, delete, and view CloudWatch alarms.
- Render CloudWatch metrics in the Route 53 console.

`route53domains:*`

Lets you work with domains.

> **Important**
>
> If you list `route53` actions individually, you must include `route53:CreateHostedZone` to work with domains. When you register a domain, a

hosted zone is created at the same time, so a policy that includes permissions to register domains also requires permission to create hosted zones.

For domain registration, Route 53 doesn't support granting or denying permissions to individual resources.

`route53resolver:*`

Lets you work with Route 53 VPC Resolver.

`ssm:GetParametersByPath`

Lets you fetch publicly available Regions when you create new alias records, private hosted zones, and health checks.

`cloudfront:ListDistributions`

Lets you create and update alias records for which the value of **Alias Target** is a CloudFront distribution.

These permissions aren't required if you aren't using the Route 53 console. Route 53 uses it only to get a list of distributions to display in the console.

`elasticloadbalancing:DescribeLoadBalancers`

Lets you create and update alias records for which the value of **Alias Target** is an ELB load balancer.

These permissions aren't required if you aren't using the Route 53 console. Route 53 uses it only to get a list of load balancers to display in the console.

`elasticbeanstalk:DescribeEnvironments`

Lets you create and update alias records for which the value of **Alias Target** is an Elastic Beanstalk environment.

These permissions aren't required if you aren't using the Route 53 console. Route 53 uses it only to get a list of environments to display in the console.

`s3:ListAllMyBuckets`, `s3:GetBucketLocation`, and `s3:GetBucketWebsite`

Let you create and update alias records for which the value of **Alias Target** is an Amazon S3 bucket. (You can create an alias to an Amazon S3 bucket only if the bucket is configured as a website endpoint; `s3:GetBucketWebsite` gets the required configuration information.)

These permissions aren't required if you aren't using the Route 53 console. Route 53 uses it only to get a list of buckets to display in the console.

`ec2:DescribeVpcs` and `ec2:DescribeRegions`

Let you work with private hosted zones.

All listed `ec2` permissions

Let you work with Route 53 VPC Resolver.

`sns:ListTopics`, `sns:ListSubscriptionsByTopic`, `sns:CreateTopic`, `cloudwatch:DescribeAlarms`, `cloudwatch:PutMetricAlarm`, `cloudwatch:DeleteAlarms`

Let you create, delete, and view CloudWatch alarms.

`cloudwatch:GetMetricStatistics`

Lets you create CloudWatch metric health checks.

These permissions aren't required if you aren't using the Route 53 console. Route 53 uses it only to get statistics to display in the console.

`apigateway:GET`

Lets you create and update alias records for which the value of **Alias Target** is an Amazon API Gateway API.

This permission isn't required if you aren't using the Route 53 console. Route 53 uses it only to get a list of APIs to display in the console.

`kms:*`

Lets you work with AWS KMS to enable DNSSEC signing.

Example permissions for a domain record owner

With resource record set permissions you can set granular permissions that limit what the AWS user can update or modify. For more information, see Using IAM policy conditions for fine-grained access control.

In some scenarios, a hosted zone owner might be responsible for the overall management of the hosted zone, while another person in the organization is responsible for a subset of those tasks. A hosted zone owner who has enabled DNSSEC signing, for example, might want to create an IAM

policy that includes the permission for someone else to add and delete Resource Set Records (RRs) in the hosted zone, among other tasks. The specific permissions that a hosted zone owner chooses to enable for a record owner or other people will depend on their organization's policy.

The following is an example IAM policy that allows a record owner to make modifications to RRs, traffic policies, and health checks. A record owner with this policy is not allowed to do zone-level operations, such as creating or deleting a zone, enabling or disabling query logging, creating or deleting a reusable delegation set, or changing DNSSEC settings.

```
{
      "Sid": "Do not allow zone-level modification ",
      "Effect": "Allow",
      "Action": [
        "route53:ChangeResourceRecordSets",
        "route53:CreateTrafficPolicy",
        "route53:DeleteTrafficPolicy",
        "route53:CreateTrafficPolicyInstance",
        "route53:CreateTrafficPolicyVersion",
        "route53:UpdateTrafficPolicyInstance",
        "route53:UpdateTrafficPolicyComment",
        "route53:DeleteTrafficPolicyInstance",
        "route53:CreateHealthCheck",
        "route53:UpdateHealthCheck",
        "route53:DeleteHealthCheck",
        "route53:List*",
        "route53:Get*"
      ],
      "Resource": [
        "*"
      ]
}
```

Route 53 customer managed key permissions required for DNSSEC signing

When you enable DNSSEC signing for Route 53, Route 53 creates a key-signing key (KSK) based on a customer managed key in AWS Key Management Service (AWS KMS). You can use an existing customer managed key that supports DNSSEC signing or create a new one. Route 53 must have permission to access your customer managed key so that it can create the KSK for you.

To enable Route 53 to access your customer managed key, make sure that your customer managed key policy contains the following statements:

```
{
            "Sid": "Allow Route 53 DNSSEC Service",
            "Effect": "Allow",
            "Principal": {
                "Service": "dnssec-route53.amazonaws.com"
            },
            "Action": ["kms:DescribeKey",
                       "kms:GetPublicKey",
                       "kms:Sign"],
            "Resource": "*"
        },
        {
            "Sid": "Allow Route 53 DNSSEC to CreateGrant",
            "Effect": "Allow",
            "Principal": {
                "Service": "dnssec-route53.amazonaws.com"
            },
            "Action": ["kms:CreateGrant"],
            "Resource": "*",
            "Condition": {
                "Bool": {
                    "kms:GrantIsForAWSResource": true
                }
            }
        }
```

The confused deputy problem is a security issue where an entity without a permission for an action can coerce a more-privileged entity to perform it. To protect your AWS KMS from it, you can optionally limit the permissions that a service has to a resource in a resource-based policy by supplying a combination of `aws:SourceAccount` and `aws:SourceArn` conditions (both or one). `aws:SourceAccount` is an AWS account ID of an owner of a hosted zone. `aws:SourceArn` is an ARN of a hosted zone.

The following are two examples of permissions you can add:

```
{
    "Sid": "Allow Route 53 DNSSEC Service",
    …
    "Resource": "*",
    "Condition": {
        "StringEquals": {
```

```
            "aws:SourceAccount": "111122223333"
        },
        "ArnEquals": {
            "aws:SourceArn": "arn:aws:route53:::hostedzone/HOSTED_ZONE_ID"
        }
    }
},
```

- Or -

```
{
    "Sid": "Allow Route 53 DNSSEC Service",
    …
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": ["1111-2222-3333","4444-5555-6666"]
        },
        "ArnLike": {
            "aws:SourceArn": "arn:aws:route53:::hostedzone/*"
        }
    }
},
```

For more information, see The confused deputy problem in the *IAM User Guide*.

Customer managed policy examples

You can create your own custom IAM policies to allow permissions for Route 53 actions. You can attach these custom policies to the IAM groups that require the specified permissions. These policies work when you are using the Route 53 API, the AWS SDKs, or the AWS CLI. The following examples show permissions for several common use cases. For the policy that grants a user full access to Route 53, see Permissions required to use the Amazon Route 53 console.

Examples

- Example 1: Allow read access to all hosted zones
- Example 2: Allow creation and deletion of hosted zones
- Example 3: Allow full access to all domains (public hosted zones only)
- Example 4: Allow creation of inbound and outbound Route 53 VPC Resolver endpoints
- Example 5: Deny removal of record set protections

- Example 6: Allow reading the change history of a hosted zone

Example 1: Allow read access to all hosted zones

The following permissions policy grants the user permissions to list all hosted zones and view all the records in a hosted zone.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Effect":"Allow",
            "Action":[
                "route53:GetHostedZone",
                "route53:ListResourceRecordSets"
            ],
            "Resource":"*"
        },
        {
            "Effect":"Allow",
            "Action":["route53:ListHostedZones"],
            "Resource":"*"
        }
    ]
}
```

Example 2: Allow creation and deletion of hosted zones

The following permissions policy allows users to create and delete hosted zones, and to track the progress of the change.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Effect":"Allow",
```

```
            "Action":["route53:CreateHostedZone"],
            "Resource":"*"
        },
        {
            "Effect":"Allow",
            "Action":["route53:DeleteHostedZone"],
            "Resource":"*"
        },
        {
            "Effect":"Allow",
            "Action":["route53:GetChange"],
            "Resource":"*"
        }
    ]
}
```

Example 3: Allow full access to all domains (public hosted zones only)

The following permissions policy allows users to perform all actions on domain registrations, including permissions to register domains and create hosted zones.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Effect":"Allow",
            "Action":[
                "route53domains:*",
                "route53:CreateHostedZone"
            ],
            "Resource":"*"
        }
    ]
}
```

When you register a domain, a hosted zone is created at the same time, so a policy that includes permissions to register domains also requires permissions to create hosted zones. (For domain registration, Route 53 doesn't support granting permissions to individual resources.)

For information about permissions that are required to work with private hosted zones, see Permissions required to use the Amazon Route 53 console.

Example 4: Allow creation of inbound and outbound Route 53 VPC Resolver endpoints

The following permissions policy allows users to use the Route 53 console to create Resolver inbound and outbound endpoints.

Some of these permissions are required only to create endpoints in the console. You can omit these permissions if you want to grant permissions only to create inbound and outbound endpoints programmatically:

- `route53resolver:ListResolverEndpoints` lets users see the list of inbound or outbound endpoints so they can verify that an endpoint was created.
- `DescribeAvailabilityZones` is required to display a list of Availability Zones.
- `DescribeVpcs` is required to display a list of VPCs.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "route53resolver:CreateResolverEndpoint",
                "route53resolver:ListResolverEndpoints",
                "ec2:CreateNetworkInterface",
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs"
            ],
            "Resource": "*"
        }
    ]
}
```

Example 5: Deny removal of record set protections

The following permissions policy denies permission to delete record set protections in a hosted zone. Because you can't change a protected record until you delete its protection, a user with this policy can't change protected records in that hosted zone. For more information about record set protections, see Protecting records from changes.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "DenyRemovingRecordProtections",
            "Effect": "Deny",
            "Action": "route53:DeleteResourceRecordSetProtections",
            "Resource": "arn:aws:route53:::hostedzone/Z1D633PJN98FT9"
        }
    ]
}
```

Example 6: Allow reading the change history of a hosted zone

The following permissions policy grants permission to read the change history of one hosted zone, including the record values in each change. `GetHostedZoneChangeBatch` and `ListResourceRecordSetChanges` also require `route53:ListResourceRecordSets`. For more information about Change History, see Viewing the change history of a hosted zone.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowReadingChangeHistory",
            "Effect": "Allow",
            "Action": [
                "route53:ListHostedZoneChangeBatches",
                "route53:GetHostedZoneChangeBatch",
                "route53:ListResourceRecordSets"
            ],
            "Resource": "arn:aws:route53:::hostedzone/Z11111112222222333333"
        }
    ]
}
```

Using Service-Linked Roles for Amazon Route 53 Resolver

Route 53 VPC Resolver uses AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique type of IAM role that is linked directly to VPC Resolver. Service-linked roles are predefined by VPC Resolver and include all the permissions that the service requires to call other AWS services on your behalf.

A service-linked role makes setting up VPC Resolver easier because you don't have to manually add the necessary permissions. VPC Resolver defines the permissions of its service-linked roles, and unless defined otherwise, only VPC Resolver can assume its roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other IAM entity.

You can delete a service-linked role only after first deleting the related resources. This protects your VPC Resolver resources because you can't inadvertently remove permission to access the resources.

For information about other services that support service-linked roles, see AWS Services that Work with IAM and look for the services that have **Yes** in the **Service-Linked Role** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

Topics

- Service-Linked Role Permissions for VPC Resolver
- Creating a Service-Linked Role for VPC Resolver
- Editing a Service-Linked Role for VPC Resolver
- Deleting a Service-Linked Role for VPC Resolver
- Supported Regions for VPC Resolver Service-Linked Roles

Service-Linked Role Permissions for VPC Resolver

VPC Resolver uses the **`AWSServiceRoleForRoute53Resolver`** service-linked role to deliver query logs on your behalf.

The role permissions policy allows VPC Resolver to complete the following actions on your resources:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Action": [
        "logs:CreateLogDelivery",
        "logs:GetLogDelivery",
        "logs:UpdateLogDelivery",
        "logs:DeleteLogDelivery",
        "logs:ListLogDeliveries",
        "logs:DescribeResourcePolicies",
        "logs:DescribeLogGroups",
        "s3:GetBucketPolicy"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

You must configure permissions to allow an IAM entity (such as a user, group, or role) to create, edit, or delete a service-linked role. For more information, see Service-Linked Role Permissions in the *IAM User Guide*.

Creating a Service-Linked Role for VPC Resolver

You don't need to manually create a service-linked role. When you create a resolver query log configuration in the Amazon Route 53 console, the AWS CLI, or the AWS API, VPC Resolver creates the service-linked role for you.

> **Important**
>
> This service-linked role can appear in your account if you completed an action in another service that uses the features supported by this role. Also, if you were using the VPC Resolver service before August 12, 2020, when it began supporting service-linked roles, then VPC Resolver created the `AWSServiceRoleForRoute53Resolver` role in your account. To learn more, see A New Role Appeared in My IAM Account.

If you delete this service-linked role, and then need to create it again, you can use the same process to recreate the role in your account. When you create a new Resolver query log configuration, the `AWSServiceRoleForRoute53Resolver` service-linked role is created for you again.

Editing a Service-Linked Role for VPC Resolver

VPC Resolver does not allow you to edit the `AWSServiceRoleForRoute53Resolver` service-linked role. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a Service-Linked Role in the *IAM User Guide*.

Deleting a Service-Linked Role for VPC Resolver

If you no longer need to use a feature or service that requires a service-linked role, we recommend that you delete that role. That way you don't have an unused entity that is not actively monitored or maintained. However, you must clean up the resources for your service-linked role before you can manually delete it.

> **Note**
>
> If the VPC Resolver service is using the role when you try to delete the resources, then the deletion might fail. If that happens, wait for a few minutes and try the operation again.

To delete VPC Resolver resources used by the `AWSServiceRoleForRoute53Resolver`

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. Expand the Route 53 console menu. In the upper left corner of the console, choose the three horizontal bars (☰) icon.
3. Within the **Resolver** menu, choose **Query logging**.
4. Select the check box next to the name of your query logging configuration, and then choose **Delete**.
5. In the **Delete query logging configuration** text box, select **Stop logging queries**.

This will disassociate the configuration from the VPC. You can also disassociate the query logging configuration programmatically. For more information, see disassociate-resolver-query-log-config.

6. After logging queries has stopped, type **`delete`** in the field and choose **Delete** to delete the query logging configuration.

To manually delete the service-linked role using IAM

Use the IAM console, the AWS CLI, or the AWS API to delete the `AWSServiceRoleForRoute53Resolver` service-linked role. For more information, see Deleting a Service-Linked Role in the *IAM User Guide*.

Supported Regions for VPC Resolver Service-Linked Roles

VPC Resolver does not support using service-linked roles in every Region where the service is available. You can use the `AWSServiceRoleForRoute53Resolver` role in the following Regions.

Region name	Region identity	Support in VPC Resolver
US East (N. Virginia)	us-east-1	Yes
US East (Ohio)	us-east-2	Yes
US West (N. California)	us-west-1	Yes
US West (Oregon)	us-west-2	Yes
Asia Pacific (Mumbai)	ap-south-1	Yes
Asia Pacific (Osaka)	ap-northeast-3	Yes
Asia Pacific (Seoul)	ap-northeast-2	Yes
Asia Pacific (Singapore)	ap-southeast-1	Yes
Asia Pacific (Sydney)	ap-southeast-2	Yes
Asia Pacific (Tokyo)	ap-northeast-1	Yes

Region name	Region identity	Support in VPC Resolver
Canada (Central)	ca-central-1	Yes
Europe (Frankfurt)	eu-central-1	Yes
Europe (Ireland)	eu-west-1	Yes
Europe (London)	eu-west-2	Yes
Europe (Paris)	eu-west-3	Yes
South America (São Paulo)	sa-east-1	Yes
China (Beijing)	cn-north-1	Yes
China (Ningxia)	cn-northwest-1	Yes
AWS GovCloud (US)	us-gov-east-1	Yes
AWS GovCloud (US)	us-gov-west-1	Yes

Using service-linked roles for Amazon Route 53 Private endpoint health checks

Amazon Route 53 Private endpoint health checks uses an AWS Identity and Access Management (IAM) service-linked role. A service-linked role is a unique type of IAM role that is linked directly to Route 53. Service-linked roles are predefined by Route 53 and include all the permissions that the service requires to call other AWS services on your behalf.

With a service-linked role, you can set up private endpoint health checks without manually adding the necessary permissions. Route 53 defines the permissions of its service-linked roles, and unless defined otherwise, only Route 53 can assume its roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other IAM entity.

You can delete a service-linked role only after first deleting the related resources. This protects your Route 53 resources because you can't inadvertently remove permission to access the resources.

For information about other services that support service-linked roles, see AWS Services that Work with IAM and look for the services that have **Yes** in the **Service-Linked Role** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

Topics

- Understanding service-linked role permissions for Private endpoint health checks
- Creating a service-linked role for Private endpoint health checks
- Editing a service-linked role for Private endpoint health checks
- Deleting a service-linked role for Private endpoint health checks
- Identifying supported Regions for Private endpoint health checks service-linked roles

Understanding service-linked role permissions for Private endpoint health checks

Route 53 uses the `AWSServiceRoleForRoute53PrivateHealthChecks` service-linked role to create and manage the network interfaces that connect Route 53 health checkers to the private endpoints in your VPCs. The role trusts the `route53healthchecks.amazonaws.com` service to assume the role.

The role uses the AWS managed policy `Route53PrivateHealthChecksServiceRolePolicy`. For information about the policy's permissions, see Route53PrivateHealthChecksServiceRolePolicy in the *AWS Managed Policy Reference*. Route 53 uses this policy to complete the following actions on your resources:

Action	Description
`ec2:CreateNetworkInterface`	Create network interfaces in your subnets and security groups. Route 53 gives each new network interface a `Name` tag that starts with `healthchecksource-` , and no other tags.
`ec2:CreateTags`	Add the `Name` tag to a network interface only while Route 53 creates it. Route 53 sets the tag value to start with `healthchecksource-` .

Action	Description
ec2:DeleteNetworkInterface	Delete network interfaces that have a Name tag that starts with healthchecksource- .
ec2:DescribeNetworkInterfaces	Read the network interfaces in your account.
ec2:DescribeSecurityGroups	Read the security groups in your account.
ec2:DescribeVpcs	Read the VPCs in your account.
ec2:DescribeSubnets	Read the subnets in your account.

You must configure permissions to allow an IAM entity (such as a user, group, or role) to create, edit, or delete a service-linked role. For more information, see Service-Linked Role Permissions in the *IAM User Guide*.

Creating a service-linked role for Private endpoint health checks

You don't need to manually create a service-linked role. When you create a health check source in the Amazon Route 53 console, the AWS CLI, or the AWS API (CreateHealthCheckSource), Route 53 creates the AWSServiceRoleForRoute53PrivateHealthChecks service-linked role for you. Route 53 also makes sure that the role exists when you delete a health check source (DeleteHealthCheckSource), because it uses the role to delete the network interfaces. The IAM entity that creates or deletes a health check source must have permission for iam:CreateServiceLinkedRole for the route53healthchecks.amazonaws.com service. The AWS managed policy: AmazonRoute53FullAccess includes this permission.

If you delete this service-linked role, and then need to create it again, you can use the same process to recreate the role in your account. When you create a new health check source, Route 53 creates the AWSServiceRoleForRoute53PrivateHealthChecks service-linked role for you again.

Editing a service-linked role for Private endpoint health checks

Route 53 does not allow you to edit the AWSServiceRoleForRoute53PrivateHealthChecks service-linked role. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a Service-Linked Role in the *IAM User Guide*.

Deleting a service-linked role for Private endpoint health checks

If you no longer need private endpoint health checks, delete the `AWSServiceRoleForRoute53PrivateHealthChecks` role. That way you don't have an unused role that nobody actively monitors or maintains. However, you must clean up the resources for your service-linked role before you can manually delete it. IAM doesn't delete the role while your account has any health check sources.

To delete the role, you need the `iam:DeleteServiceLinkedRole` permission. If you don't have this permission, ask an administrator to delete the role for you.

> **Note**
>
> If Route 53 is using the role when you try to delete the resources, then the deletion might fail. If that happens, wait for a few minutes and try the operation again. If the deletion still fails with an access denied error, make sure that you have the required IAM permissions, or ask an administrator to complete the steps.

To delete Route 53 resources used by the service-linked role

1. Update or delete the records that use the health checks. Route 53 doesn't prevent you from deleting a health check that records still reference, and routing for those records might be unpredictable. For more information, see Updating or deleting health checks when DNS failover is configured.
2. Delete the health checks that use your health check sources.
3. Delete your health check sources by using the Amazon Route 53 console, the AWS CLI, or the AWS API (`DeleteHealthCheckSource`). Route 53 deletes the network interfaces that it created for each health check source.

After you delete the resources, use the IAM console, the AWS CLI, or the AWS API to delete the `AWSServiceRoleForRoute53PrivateHealthChecks` service-linked role. For more information, see Deleting a service-linked role in the *IAM User Guide*.

Identifying supported Regions for Private endpoint health checks service-linked roles

Route 53 supports using the `AWSServiceRoleForRoute53PrivateHealthChecks` service-linked role in all of the AWS Regions where private endpoint health checks are available.

AWS managed policies for Amazon Route 53

An AWS managed policy is a standalone policy that is created and administered by AWS. AWS managed policies are designed to provide permissions for many common use cases so that you can start assigning permissions to users, groups, and roles.

Keep in mind that AWS managed policies might not grant least-privilege permissions for your specific use cases because they're available for all AWS customers to use. We recommend that you reduce permissions further by defining customer managed policies that are specific to your use cases.

You cannot change the permissions defined in AWS managed policies. If AWS updates the permissions defined in an AWS managed policy, the update affects all principal identities (users, groups, and roles) that the policy is attached to. AWS is most likely to update an AWS managed policy when a new AWS service is launched or new API operations become available for existing services.

For more information, see AWS managed policies in the *IAM User Guide*.

AWS managed policy: AmazonRoute53FullAccess

You can attach the `AmazonRoute53FullAccess` policy to your IAM identities.

This policy grants full access to Route 53 resources, including domain registration and health checking, but excluding VPC Resolver.

Permissions details

This policy includes the following permissions.

- `route53:*` – Lets you perform all Route 53 actions *except* the following:
  - Create and update alias records for which the value of **Alias Target** is a CloudFront distribution, an Elastic Load Balancing load balancer, an Elastic Beanstalk environment, or an Amazon S3 bucket. (With these permissions, you can create alias records for which the value of **Alias Target** is another record in the same hosted zone.)

  - Work with private hosted zones.
  - Work with domains.
  - Create, delete, and view CloudWatch alarms.
  - Render CloudWatch metrics in the Route 53 console.
- `route53domains:*`– Lets you work with domains.
- `cloudfront:ListDistributions` – Lets you create and update alias records for which the value of **Alias Target** is a CloudFront distribution.

  This permission isn't required if you aren't using the Route 53 console. Route 53 uses it only to get a list of distributions to display in the console.
- `cloudfront:GetDistributionTenantByDomain` – Used to fetch the CloudFront multi-tenant distributions to let you create and update alias records for which the value of **Alias Target** is a CloudFront distribution tenant.
- `cloudfront:GetConnectionGroup` – Used to fetch the CloudFront multi-tenant distributions to let you create and update alias records for which the value of **Alias Target** is a CloudFront distribution tenant.
- `cloudwatch:DescribeAlarms` – Together with `sns:ListTopics` and `sns:ListSubscriptionsByTopic`, lets you create, delete, and view CloudWatch alarms.
- `cloudwatch:GetMetricStatistics` – Lets you create CloudWatch metric health checks.

  These permissions aren't required if you aren't using the Route 53 console. Route 53 uses it only to get statistics to display in the console.
- `cloudwatch:GetMetricData` – Lets you display the status of your CloudWatch health check metrics.
- `ec2:DescribeVpcs` – Lets you display a list of VPCs.
- `ec2:DescribeVpcEndpoints` – Lets you display a list of VPC endpoints.
- `ec2:DescribeRegions` – Lets you display a list of Availability Zones.
- `elasticloadbalancing:DescribeLoadBalancers` – Lets you create and update alias records for which the value of **Alias Target** is an Elastic Load Balancing load balancer.

  These permissions aren't required if you aren't using the Route 53 console. Route 53 uses it only to get a list of load balancers to display in the console.
- `elasticbeanstalk:DescribeEnvironments` – Lets you create and update alias records for which the value of **Alias Target** is an Elastic Beanstalk environment.

These permissions aren't required if you aren't using the Route 53 console. Route 53 uses it only to get a list of environments to display in the console.

- `es:ListDomainNames` – Lets you display the names of all Amazon OpenSearch Service domains owned by the current user in the active Region.
- `es:DescribeDomains` – Lets you get the domain configuration for the specified Amazon OpenSearch Service domains.
- `lightsail:GetContainerServices` – Lets you the Lightsail container services to let you create and update alias records for which the value of **Alias Target** is a Lightsail domain.
- `s3:ListBucket`, `s3:GetBucketLocation`, and `s3:GetBucketWebsite` – Let you create and update alias records for which the value of **Alias Target** is an Amazon S3 bucket. (You can create an alias to an Amazon S3 bucket only if the bucket is configured as a website endpoint; `s3:GetBucketWebsite` gets the required configuration information.)

  These permissions aren't required if you aren't using the Route 53 console. Route 53 uses these only to get a list of buckets to display in the console.
- `sns:ListTopics`, `sns:ListSubscriptionsByTopic`, `cloudwatch:DescribeAlarms` – Let you create, delete, and view CloudWatch alarms.
- `tag:GetResources` – Lets you display the tags in your resources. For example, names of your health checks.
- `apigateway:GET` – Lets you create and update alias records for which the value of **Alias Target** is an Amazon API Gateway API.
- `iam:CreateServiceLinkedRole` – Allows principals to create the service-linked role for health checking. This permission is needed when the service-linked role doesn't exist in the account yet.

For more information about the permissions, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
```

```
   "Action": [
    "route53:*",
    "route53domains:*",
    "cloudfront:ListDistributions",
    "cloudfront:GetDistributionTenantByDomain",
    "cloudfront:GetConnectionGroup",
    "cloudwatch:DescribeAlarms",
    "cloudwatch:GetMetricStatistics",
    "cloudwatch:GetMetricData",
    "ec2:DescribeVpcs",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeRegions",
    "elasticloadbalancing:DescribeLoadBalancers",
    "elasticbeanstalk:DescribeEnvironments",
    "es:ListDomainNames",
    "es:DescribeDomains",
    "lightsail:GetContainerServices",
    "s3:ListBucket",
    "s3:GetBucketLocation",
    "s3:GetBucketWebsite",
    "sns:ListTopics",
    "sns:ListSubscriptionsByTopic",
    "tag:GetResources"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": "apigateway:GET",
   "Resource": "arn:aws:apigateway:*::/domainnames"
  },
  {
   "Effect": "Allow",
   "Action": [
   "iam:CreateServiceLinkedRole"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "iam:AWSServiceName": "route53healthchecks.amazonaws.com"
    }
   }
  }
 ]
```

```
}
```

AWS managed policy: AmazonRoute53ReadOnlyAccess

You can attach the `AmazonRoute53ReadOnlyAccess` policy to your IAM identities.

This policy grants read-only access to Route 53 resources, including domain registration and health checking, but excluding VPC Resolver.

Permissions details

This policy includes the following permissions.

- `route53:Get*` – Gets the Route 53 resources.
- `route53:List*` – Lists the Route 53 resources.
- `route53:TestDNSAnswer` – Gets the value that Route 53 returns in response to a DNS request.

For more information about the permissions, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "route53:Get*",
                "route53:List*",
                "route53:TestDNSAnswer"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

AWS managed policy: AmazonRoute53DomainsFullAccess

You can attach the `AmazonRoute53DomainsFullAccess` policy to your IAM identities.

This policy grants full access to Route 53 domain registration resources.

Permissions details

This policy includes the following permissions.

- `route53:CreateHostedZone` – Lets you create a Route 53 hosted zone.
- `route53domains:*` – Lets you register domain names and perform related operations.

For more information about the permissions, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "route53:CreateHostedZone",
                "route53domains:*"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

AWS managed policy: AmazonRoute53DomainsReadOnlyAccess

You can attach the `AmazonRoute53DomainsReadOnlyAccess` policy to your IAM identities.

This policy grants read-only access to Route 53 domain registration resources.

Permissions details

This policy includes the following permissions.

- `route53domains:Get*` – Lets you retrieve a list of domains from Route 53.
- `route53domains:List*` – Lets you display a list of Route 53 domains.

For more information about the permissions, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "route53domains:Get*",
                "route53domains:List*"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

AWS managed policy: AmazonRoute53ResolverFullAccess

You can attach the `AmazonRoute53ResolverFullAccess` policy to your IAM identities.

This policy grants full access to Route 53 VPC Resolver resources.

Permissions details

This policy includes the following permissions.

- `route53resolver:*` – Lets you create and manage VPC Resolver resources on the Route 53 console.

- `ec2:DescribeSubnets` – Lets you list your Amazon VPC subnets.
- `ec2:CreateNetworkInterface`, `ec2:DeleteNetworkInterface`, and `ec2:ModifyNetworkInterfaceAttribute` – Let you create, modify, and delete network interfaces.
- `ec2:DescribeNetworkInterfaces` – Lets you display a list of network interfaces.
- `ec2:DescribeSecurityGroups` – Lets you display a list of all of your security groups.
- `ec2:DescribeVpcs` – Lets you display a list of VPCs.
- `ec2:DescribeAvailabilityZones` – Lets you list the zones that are available to you.

For more information about the permissions, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonRoute53ResolverFullAccess",
            "Effect": "Allow",
            "Action": [
                "route53resolver:*",
                "ec2:DescribeSubnets",
                "ec2:CreateNetworkInterface",
                "ec2:DeleteNetworkInterface",
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:DescribeNetworkInterfaces",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeVpcs",
                "ec2:DescribeAvailabilityZones"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

AWS managed policy: AmazonRoute53ResolverReadOnlyAccess

You can attach the `AmazonRoute53ResolverReadOnlyAccess` policy to your IAM identities.

This policy grants read-only access to Route 53 VPC Resolver resources.

Permissions details

This policy includes the following permissions.

- `route53resolver:Get*` – Gets VPC Resolver resources.
- `route53resolver:List*` – Lets you display a list of VPC Resolver resources.
- `ec2:DescribeNetworkInterfaces` – Lets you display a list of network interfaces.
- `ec2:DescribeSecurityGroups` – Lets you display a list of all of your security groups.

For more information about the permissions, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonRoute53ResolverReadOnlyAccess",
            "Effect": "Allow",
            "Action": [
                "route53resolver:Get*",
                "route53resolver:List*",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

AWS managed policy: Route53ResolverServiceRolePolicy

You can't attach `Route53ResolverServiceRolePolicy` to your IAM entities. This policy is attached to a service-linked role that allows Route 53 VPC Resolver to access AWS services and resources that are used or managed by VPC Resolver. For more information, see Using Service-Linked Roles for Amazon Route 53 Resolver.

AWS managed policy: Route53PrivateHealthChecksServiceRolePolicy

You can't attach `Route53PrivateHealthChecksServiceRolePolicy` to your IAM entities. Route 53 attaches this policy to the `AWSServiceRoleForRoute53PrivateHealthChecks` service-linked role, which Route 53 uses to create and manage the network interfaces that connect Route 53 health checkers to the private endpoints in your VPCs. Route 53 can create, tag, and delete only network interfaces that have a `Name` tag that starts with `healthchecksource-`, and can read your network interfaces, security groups, VPCs, and subnets. For more information, see Using service-linked roles for Amazon Route 53 Private endpoint health checks.

To view the permissions for this policy, see Route53PrivateHealthChecksServiceRolePolicy in the *AWS Managed Policy Reference*.

AWS managed policy: AmazonRoute53ProfilesFullAccess

You can attach the `AmazonRoute53ProfilesReadOnlyAccess` policy to your IAM identities.

This policy grants full access to Amazon Route 53 Profile resources.

Permissions details

This policy includes the following permissions.

- `route53profiles` – Lets you create and manage Profile resources on the Route 53 console.
- `ec2` – Allows principals to get information about VPCs.

For more information about the permissions, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

JSON

```
{
    "Version":"2012-10-17",
```

```
    "Statement": [
        {
            "Sid": "AmazonRoute53ProfilesFullAccess",
            "Effect": "Allow",
            "Action": [
                "route53profiles:AssociateProfile",
                "route53profiles:AssociateResourceToProfile",
                "route53profiles:CreateProfile",
                "route53profiles:DeleteProfile",
                "route53profiles:DisassociateProfile",
                "route53profiles:DisassociateResourceFromProfile",
                "route53profiles:UpdateProfileResourceAssociation",
                "route53profiles:GetProfile",
                "route53profiles:GetProfileAssociation",
                "route53profiles:GetProfileResourceAssociation",
                "route53profiles:GetProfilePolicy",
                "route53profiles:ListProfileAssociations",
                "route53profiles:ListProfileResourceAssociations",
                "route53profiles:ListProfiles",
                "route53profiles:PutProfilePolicy",
                "route53profiles:ListTagsForResource",
                "route53profiles:TagResource",
                "route53profiles:UntagResource",
                "route53resolver:GetFirewallConfig",
                "route53resolver:GetFirewallRuleGroup",
                "route53resolver:GetResolverConfig",
                "route53resolver:GetResolverDnssecConfig",
                "route53resolver:GetResolverQueryLogConfig",
                "route53resolver:GetResolverRule",
                "ec2:DescribeVpcs",
                "route53:GetHostedZone"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

AWS managed policy: AmazonRoute53ProfilesReadOnlyAccess

You can attach the `AmazonRoute53ProfilesReadOnlyAccess` policy to your IAM identities.

This policy grants read-only access to Amazon Route 53 Profile resources.

Permissions details

For more information about the permissions, see Amazon Route 53 API permissions: Actions, resources, and conditions reference.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonRoute53ProfilesReadOnlyAccess",
            "Effect": "Allow",
            "Action": [
                "route53profiles:GetProfile",
                "route53profiles:GetProfileAssociation",
                "route53profiles:GetProfileResourceAssociation",
                "route53profiles:GetProfilePolicy",
                "route53profiles:ListProfileAssociations",
                "route53profiles:ListProfileResourceAssociations",
                "route53profiles:ListProfiles",
                "route53profiles:ListTagsForResource",
                "route53resolver:GetFirewallConfig",
                "route53resolver:GetResolverConfig",
                "route53resolver:GetResolverDnssecConfig",
                "route53resolver:GetResolverQueryLogConfig"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Route 53 updates to AWS managed policies

View details about updates to AWS managed policies for Route 53 since this service began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the Route 53 Document history page.

Managed policy change history

Change	Description	Date
AmazonRoute53ProfilesFullAccess – Updated policy	Adds permission for `UpdateProfileAssociation` . With this permission, principals can change the name or priority of a Profile association with a VPC.	November 09, 2026
Route53PrivateHealthChecksServiceRolePolicy – New policy	Amazon Route 53 added a new policy for the `AWSServiceRoleForRoute53PrivateHealthChecks` service-linked role. Route 53 uses the role to create and manage the network interfaces that private endpoint health checks use to reach endpoints in your VPCs.	October 30, 2026
AmazonRoute53FullAccess – Updated policy	Adds permission for `iam:CreateServiceLinkedRole` , limited to the `route53healthchecks.amazonaws.com` service. With this permission, principals can create the service-linked role for health checking when the role doesn't exist in the account yet.	October 08, 2026
AmazonRoute53FullAccess – Updated policy	Adds permissions for `cloudwatch:GetMetr`	June 01, 2025

Change	Description	Date
	`icData` ,`tag:GetResources` ,`es:ListDomainNames` ,`es:DescribeDomains` ,`cloudfront:GetDistributionTenantByDomain` , `cloudfront:GetConnectionGroup` and `lightsail:GetContainerServices` . With these permissions, you can fetch up to 500 CloudWatch health check metrics, up to 100 names of health checks, get the domain configuration for the specified Amazon OpenSearch Service domains, and list the names of all Amazon OpenSearch Service domains owned by the current user in the active Region, fetch the CloudFront multi-tenant distributions and get the Lightsail container services.	

Change	Description	Date
AmazonRoute53ProfilesFullAccess – Updated policy	Adds permissions for `GetProfilePolicy` and `PutProfilePolicy` . These are permission-only IAM actions. If an IAM principal doesn't have these permissions granted, an error occurs when attempting to share the Profile using the AWS RAM service.	August 27, 2024
AmazonRoute53ProfilesReadOnlyAccess – Updated policy	Adds permissions for `GetProfilePolicy` . This is a permission-only IAM action. If an IAM principal doesn't have this permission granted, an error occurs when attempting to access the Profile's policy using the AWS RAM service.	August 27, 2024
AmazonRoute53ResolverFullAccess– Updated policy	Added a statement id (Sid) to uniquely identity the policy.	August 5, 2024
AmazonRoute53ResolverReadOnlyAccess– Updated policy	Added a statement id (Sid) to uniquely identity the policy.	August 5, 2024
AmazonRoute53ProfilesFullAccess – New policy	Amazon Route 53 added a new policy to allow full access to Amazon Route 53 Profile resources.	April 22, 2024

Change	Description	Date
AmazonRoute53ProfilesReadOnlyAccess – New policy	Amazon Route 53 added a new policy to allow read-only access to Amazon Route 53 Profile resources.	April 22, 2024
Route53ResolverServiceRolePolicy– New policy	Amazon Route 53 added a new policy that is attached to a service-linked role that allows VPC Resolver to access AWS services and resources that are used or managed by Resolver.	July 14, 2021
AmazonRoute53ResolverReadOnlyAccess– New policy	Amazon Route 53 added a new policy to allow read-only access to VPC Resolver resources.	July 14, 2021
AmazonRoute53ResolverFullAccess– New policy	Amazon Route 53 added a new policy to allow full access to VPC Resolver resources.	July 14, 2021
AmazonRoute53DomainsReadOnlyAccess– New policy	Amazon Route 53 added a new policy to allow read-only access to Route 53 domains resources.	July 14, 2021
AmazonRoute53DomainsFullAccess– New policy	Amazon Route 53 added a new policy to allow full access to Route 53 domains resources.	July 14, 2021
AmazonRoute53ReadOnlyAccess– New policy	Amazon Route 53 added a new policy to allow read-only access to Route 53 resources.	July 14, 2021

Change	Description	Date
AmazonRoute53FullAccess– New policy	Amazon Route 53 added a new policy to allow full access to Route 53 resources.	July 14, 2021
Route 53 started tracking changes	Route 53 started tracking changes for its AWS managed policies.	July 14, 2021

Using IAM policy conditions for fine-grained access control

In Route 53, you can specify conditions when granting permissions using an IAM policy (see Access control). For example, you can:

- Grant permissions to allow access to a single resource record set.
- Grant permissions to allow users access to all resource record sets of a specific DNS record type in a hosted zone, for example A and AAAA records.
- Grant permissions to allow users access to a resource record set where its name contains a specific string.
- Grant permissions to allow users to perform only a subset of the `CREATE | UPSERT | DELETE` actions on the Route 53 console, or when using the ChangeResourceRecordSets API.
- Grant permissions to allow users to associate or dissociate private hosted zones from a particular VPC.
- Grant permissions to allow users to list hosted zones associated to a particular VPC.
- Grant permissions to allow users access to create a new private hosted zone and associate it to a particular VPC.
- Grant permissions to allow users to create or delete a VPC association authorization.
- Grant permissions to allow users to manage (associate/disassociate/update) only specific resource types with a Route 53 Profile.
- Grant permissions to allow users to manage (associate/disassociate/update) only specific resource ARNs with a Route 53 Profile.
- Grant permissions to allow users to manage (associate/disassociate/update) only specific hosted zone domains with a Route 53 Profile.

- Grant permissions to allow users to manage (associate/disassociate/update) only specific Resolver Rule domains with a Route 53 Profile.
- Grant permissions to allow users to manage (associate/disassociate/update) Firewall Rule Groups with a specific priority range in a Route 53 Profile.
- Grant permissions to allow users to manage (associate/disassociate) a Route 53 Profile with specific VPCs.

You can also create permissions that combine any of the granular permissions.

Normalizing the Route 53 condition key values

The values you enter for the policy conditions must be formatted, or normalized, as follows:

For `route53:ChangeResourceRecordSetsNormalizedRecordNames`:

- All letters must be lowercase.
- The DNS name must be without the trailing dot.
- Characters other than a–z, 0–9, - (hyphen), _ (underscore), and . (period, as a delimiter between labels) must use escape codes in the format \three-digit octal code. For example, `\052` is the octal code for character *.

For `route53:ChangeResourceRecordSetsActions`, the value can be any of the following and must be uppercase:

- CREATE
- UPSERT
- DELETE

For `route53:ChangeResourceRecordSetsRecordTypes`:

- The value must be in uppercase, and can be any of the Route 53 supported DNS record types. For more information, see Supported DNS record types.

For `route53:VPCs`:

- The value must be in the format of `VPCId=<vpc-id>,VPCRegion=<region>`.

- The value of <vpc-id> and <region>must be in lowercase, such as VPCId=vpc-123abc and VPCRegion=us-east-1.
- The context keys and values are case sensitive.

> **Important**
>
> For your permissions to allow or restrict actions as you intend, you must follow these conventions. Only VPCId and VPCRegion elements are accepted by this condition key, any other AWS resources, such as AWS account, are not supported.

For route53profiles:ResourceTypes, the value can be any of the following and is case sensitive:

- HostedZone
- FirewallRuleGroup
- ResolverQueryLoggingConfig
- ResolverRule
- VPCEndpoint

For route53profiles:ResourceArns:

- The value must be a valid AWS resource ARN, such as arn:aws:route53:::hostedzone/Z12345.
- Use the ArnEquals or ArnLike condition operator when comparing ARN values.

For route53profiles:HostedZoneDomains:

- The value must be a valid domain name, such as example.com.
- The domain name must be without the trailing dot.
- The values are case sensitive.

For route53profiles:ResolverRuleDomains:

- The value must be a valid domain name, such as example.com.

- The domain name must be without the trailing dot.
- The values are case sensitive.

For `route53profiles:FirewallRuleGroupPriority`:

- The value must be a numeric value representing the priority of the Firewall Rule Group.
- Use numeric condition operators such as `NumericEquals`, `NumericGreaterThanEquals`, or `NumericLessThanEquals` to compare priority values or define a priority range.

For `route53profiles:ResourceIds`:

- The value must be a valid VPC ID, such as `vpc-1a2b3c4d5e6f`.
- The values are case sensitive.

You can use the Access Analyzer or Policy Simulator in the *IAM User Guide* to validate that your policy grants or restricts the permissions as expected. You can also validate the permissions by applying an IAM policy to a test user or role to carry out Route 53 operations.

Specifying conditions: using condition keys

AWS provides a set of predefined condition keys (AWS-wide condition keys) for all AWS services that support IAM for access control. For example, you can use the `aws:SourceIp` condition key to check the requester's IP address before allowing an action to be performed. For more information and a list of the AWS-wide keys, see Available Keys for Conditions in the *IAM User Guide*.

> **Note**
>
> Route 53 doesn't support tag-based condition keys.

The following table shows the Route 53 service-specific condition keys that apply to Route 53.

Route 53 Condition Key	API operations	Value type	Description
route53:ChangeResourceRecordSetsNormalizedRecordNames	ChangeResourceRecordSets	Multi-valued	Represents a list of DNS record names in the request of ChangeResourceRecordSets. To get the expected behavior, DNS names in the IAM policy must be normalized as follows: • All letters must be lowercase. • The DNS name must be without the trailing dot. • Characters other than a to z, 0 to 9, - (hyphen), _ (underscore), and . (period, as a delimiter between labels) must use escape codes in the format \three-digit octal code.
route53:ChangeResourceRecordSetsRecordTypes	ChangeResourceRecordSets	Multi-valued	Represents a list of DNS record types in the request of ChangeResourceRecordSets. ChangeResourceRecordSetsRecordTypes can be any of the Route 53 supported DNS record types. For more information, see Supported DNS record types. All must be entered in uppercase in the policy.
route53:ChangeResourceRecordSetsActions	ChangeResourceRecordSets	Multi-valued	Represents a list of actions in the request of ChangeResourceRecordSets. ChangeResourceRecordSetsActions can be any of the following values (must be uppercase): • CREATE •

Route 53 Condition Key	API operations	Value type	Description
			UPSERT • DELETE
route53:VPCs	AssociateVPCWithHostedZone DisassociateVPCFromHostedZone ListHostedZonesByVPC CreateHostedZone CreateVPCAssociationAuthorization DeleteVPCAssociationAuthorization	Multi-valued	Represents a list of VPCs in the request of AssociateVPCWithHostedZone , DisassociateVPCFromHostedZone , ListHostedZonesByVPC , CreateHostedZone , CreateVPCAssociationAuthorization , and DeleteVPCAssociationAuthorization , in the format of "VPCId=<vpc-id>,VPCRegion=<region>

Route 53 Condition Key	API operations	Value type	Description
`route53profiles:ResourceTypes`	AssociateResourceToProfile DisassociateResourceFromProfile UpdateProfileResourceAssociation	String	Filters access by specific resource type. `route53profiles:ResourceTypes` can be any of the following values (case sensitive): • HostedZone • FirewallRuleGroup • ResolverQueryLoggingConfig • ResolverRule • VPCEndpoint
`route53profiles:ResourceArns`	AssociateResourceToProfile DisassociateResourceFromProfile	ARN	Filters access by specific resource ARNs.

Route 53 Condition Key	API operations	Value type	Description
route53profiles:HostedZoneDomains	AssociateResourceToProfile DisassociateResourceFromProfile UpdateProfileResourceAssociation	String	Filters access by Hosted Zone domains. To get the expected behavior, domain names in the IAM policy must be normalized as follows: • The domain name must be without the trailing dot. • The values are case sensitive.
route53profiles:ResolverRuleDomains	AssociateResourceToProfile DisassociateResourceFromProfile UpdateProfileResourceAssociation	String	Filters access by Resolver Rule domains. To get the expected behavior, domain names in the IAM policy must be normalized as follows: • The domain name must be without the trailing dot. • The values are case sensitive.

Route 53 Condition Key	API operations	Value type	Description
route53profiles:FirewallRuleGroupPriority	AssociateResourceToProfile DisassociateResourceFromProfile UpdateProfileResourceAssociation	Numeric	Filters access by priority range of a Firewall Rule Group.
route53profiles:ResourceIds	AssociateProfile DisassociateProfile UpdateProfileAssociation	String	Filters access by given VPCs.

Example policies: Using conditions for fine-grained access

Each of the examples in this section sets the Effect clause to Allow and specifies only the actions, resources, and parameters that are allowed. Access is permitted only to what is explicitly listed in the IAM policy.

In some cases, it is possible to rewrite these policies so that they are deny-based (that is, setting the Effect clause to Deny and inverting all of the logic in the policy). However, we recommend

that you avoid using deny-based policies because they are difficult to write correctly, compared to allow-based policies. This is especially true for Route 53 due to text normalization that is required.

Grant permissions that limit access to DNS records with specific names

The following permissions policy grants permissions that allow `ChangeResourceRecordSets` actions on the Hosted Zone Z12345 for example.com and marketing.example.com. It uses the `route53:ChangeResourceRecordSetsNormalizedRecordNames` condition key to limit user actions only on the records that match the specified names.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "route53:ChangeResourceRecordSets",
            "Resource": "arn:aws:route53:::hostedzone/Z11111112222222333333",
            "Condition": {
                "ForAllValues:StringEquals":{
                    "route53:ChangeResourceRecordSetsNormalizedRecordNames":
 ["example.com", "marketing.example.com"]
                }
            }
          }
        ]
}
```

`ForAllValues:StringEquals` is an IAM condition operator that applies to multi-valued keys. The condition in the policy above will allow the operation only when all changes in `ChangeResourceRecordSets` have the DNS name of example.com. For more information, see IAM condition operators and IAM condition with multiple keys or values in the IAM User Guide.

To implement the permission that matches names with certain suffixes, you can use the IAM wildcard (*) in the policy with condition operator `StringLike` or `StringNotLike`. The following policy will allow the operation when all changes in the `ChangeResourceRecordSets` operation have DNS names that end with “-beta.example.com”.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "route53:ChangeResourceRecordSets",
            "Resource": "arn:aws:route53:::hostedzone/Z11111112222222333333",
            "Condition": {
                "ForAllValues:StringLike":{
                    "route53:ChangeResourceRecordSetsNormalizedRecordNames":
 ["*-beta.example.com"]
                }
            }
          }
        ]
}
```

Note

The IAM wildcard isn't the same as the domain name wildcard. See the following example for how to use the wildcard with a domain name.

Grant permissions that limit access to DNS records that match a domain name containing a wildcard

The following permissions policy grants permissions that allow `ChangeResourceRecordSets` actions on the Hosted Zone Z12345 for example.com. It uses the `route53:ChangeResourceRecordSetsNormalizedRecordNames` condition key to limit user actions only to the records that match *.example.com.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": "route53:ChangeResourceRecordSets",
            "Resource": "arn:aws:route53:::hostedzone/Z11111112222222333333",
            "Condition": {
                "ForAllValues:StringEquals":{
                    "route53:ChangeResourceRecordSetsNormalizedRecordNames": ["\
\052.example.com"]
                }
            }
          }
        ]
}
```

\052 is the octal code for character * in the DNS name, and \ in \052 is escaped to be \\ to follow JSON syntax.

Grant permissions that limit access to specific DNS records

The following permissions policy grants permissions that allow ChangeResourceRecordSets actions on the Hosted Zone Z12345 for example.com. It uses the combination of three condition keys to limit user actions to allow only creating or editing DNS records with certain DNS name and type.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "route53:ChangeResourceRecordSets",
            "Resource": "arn:aws:route53:::hostedzone/Z11111112222222333333",
            "Condition": {
                "ForAllValues:StringEquals":{
                    "route53:ChangeResourceRecordSetsNormalizedRecordNames":
 ["example.com"],
                    "route53:ChangeResourceRecordSetsRecordTypes": ["MX"],
                    "route53:ChangeResourceRecordSetsActions": ["CREATE",
 "UPSERT"]
                }
            }
```

```
          }
        ]
}
```

Grant permissions that limit access to creating and editing only the specified types of DNS records

The following permissions policy grants permissions that allow `ChangeResourceRecordSets` actions on the Hosted Zone Z12345 for example.com. It uses the `route53:ChangeResourceRecordSetsRecordTypes` condition key to limit user actions only on the records which match the specified types (A and AAAA).

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "route53:ChangeResourceRecordSets",
            "Resource": "arn:aws:route53:::hostedzone/Z11111112222222333333",
            "Condition": {
                "ForAllValues:StringEquals":{
                     "route53:ChangeResourceRecordSetsRecordTypes": ["A", "AAAA"]
                }
            }
          }
        ]
}
```

Grant permissions that specifies the VPC that the IAM principal can operate in

The following permissions policy grants permissions that allow `AssociateVPCWithHostedZone`, `DisassociateVPCFromHostedZone`, `ListHostedZonesByVPC`, `CreateHostedZone`, `CreateVPCAssociationAuthorization`, and `DeleteVPCAssociationAuthorization` actions on the VPC specified by the vpc-id.

Important

The condition value must be in the format of `VPCId=<vpc-id>,VPCRegion=<region>`. If you specify a VPC ARN in the condition value, the condition key will not take effect.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Allow",
            "Action": [
                "route53:*"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringLike": {
                    "route53:VPCs": [
                        "VPCId=<vpc-id>,VPCRegion=<region>"
                    ]
                }
            }
        },
        {
            "Sid": "Statement2",
            "Effect": "Allow",
            "Action": "ec2:DescribeVpcs",
            "Resource": "*"
        }
    ]
}
```

Important

The `route53profiles` condition keys are available in all AWS Regions where Route 53 Route53Profiles is available, except for me-central-1 and me-south-1.

Grant permissions that limit resource association to specific resource types in Route 53 Profiles

The following permissions policy grants permissions that allow `AssociateResourceToProfile` and `DisassociateResourceFromProfile` actions only when the resource type is a hosted zone. It uses the `route53profiles:ResourceTypes` condition key to restrict the resource types that can be associated with a profile.

```
{
    "Effect": "Allow",
    "Action": [
        "route53profiles:AssociateResourceToProfile",
        "route53profiles:DisassociateResourceFromProfile"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "route53profiles:ResourceTypes": "HostedZone"
        }
    }
}
```

Grant permissions that limit resource association to specific resource ARNs in Route 53 Profiles

The following permissions policy grants permissions that allow `AssociateResourceToProfile` and `DisassociateResourceFromProfile` actions only for the specified resource ARN. It uses the `route53profiles:ResourceArns` condition key to restrict which resources can be associated with a profile.

```
{
    "Effect": "Allow",
    "Action": [
        "route53profiles:AssociateResourceToProfile",
        "route53profiles:DisassociateResourceFromProfile"
    ],
    "Resource": "*",
```

```
    "Condition": {
        "ArnEquals": {
            "route53profiles:ResourceArns": "arn:aws:route53:::hostedzone/Z12345"
        }
    }
}
```

Grant permissions that limit resource association to specific hosted zone domains in Route 53 Profiles

The following permissions policy grants permissions that allow `AssociateResourceToProfile`, `DisassociateResourceFromProfile`, and `UpdateProfileResourceAssociation` actions only when the hosted zone domain matches the specified value. It uses the `route53profiles:HostedZoneDomains` condition key to restrict which hosted zone domains can be associated with a profile.

```
{
    "Effect": "Allow",
    "Action": [
        "route53profiles:AssociateResourceToProfile",
        "route53profiles:DisassociateResourceFromProfile",
        "route53profiles:UpdateProfileResourceAssociation"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "route53profiles:HostedZoneDomains": "example.com"
        }
    }
}
```

Grant permissions that limit resource association to specific Resolver Rule domains in Route 53 Profiles

The following permissions policy grants permissions that allow `AssociateResourceToProfile`, `DisassociateResourceFromProfile`, and `UpdateProfileResourceAssociation` actions only when the Resolver Rule domain matches the specified value. It uses the `route53profiles:ResolverRuleDomains` condition key to restrict which Resolver Rule domains can be associated with a profile.

```
{
```

```
    "Effect": "Allow",
    "Action": [
        "route53profiles:AssociateResourceToProfile",
        "route53profiles:DisassociateResourceFromProfile",
        "route53profiles:UpdateProfileResourceAssociation"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "route53profiles:ResolverRuleDomains": "example.com"
        }
    }
}
```

Grant permissions that limit Firewall Rule Group association to a specific priority range in Route 53 Profiles

The following permissions policy grants permissions that allow `AssociateResourceToProfile`, `DisassociateResourceFromProfile`, and `UpdateProfileResourceAssociation` actions only when the Firewall Rule Group priority is within the specified range. It uses the `route53profiles:FirewallRuleGroupPriority` condition key to restrict the priority values that can be used.

```
{
    "Effect": "Allow",
    "Action": [
        "route53profiles:AssociateResourceToProfile",
        "route53profiles:DisassociateResourceFromProfile",
        "route53profiles:UpdateProfileResourceAssociation"
    ],
    "Resource": "*",
    "Condition": {
        "NumericGreaterThanEquals": {
            "route53profiles:FirewallRuleGroupPriority": "100"
        }
    }
}
```

Grant permissions that limit profile association to specific VPCs in Route 53 Profiles

The following permissions policy grants permissions that allow `AssociateProfile` and `DisassociateProfile` actions only for the specified VPC. It uses the

`route53profiles:ResourceIds` condition key to restrict which VPCs a profile can be associated with.

```
{
    "Effect": "Allow",
    "Action": [
        "route53profiles:AssociateProfile",
        "route53profiles:DisassociateProfile"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "route53profiles:ResourceIds": "vpc-1a2b3c4d5e6f"
        }
    }
}
```

Amazon Route 53 API permissions: Actions, resources, and conditions reference

When you set up Access control and write a permissions policy that you can attach to an IAM identity (identity-based policies), you can use the lists of Actions, resources, and condition keys for Route 53, Actions, resources, and condition keys for Route 53 Domains, Actions, resources, and condition keys for VPC Resolver, and Actions, resources, and condition keys for Amazon Route 53 Profiles enables sharing DNS settings with VPCs in the *Service Authorization Reference*. The pages include each Amazon Route 53 API action, the actions that you must grant permissions access to, and the AWS resource that you must grant access to. You specify the actions in the policy's `Action` field, and you specify the resource value in the policy's `Resource` field.

You can use AWS-wide condition keys in your Route 53 policies to express conditions. For a complete list of AWS-wide keys, see Available keys in the *IAM User Guide*.

Note

When granting access, the hosted zone and the Amazon VPC must belong to the same partition. A partition is a group of AWS Regions. Each AWS account is scoped to one partition.
The following are the supported partitions:

- `aws` - AWS Regions

- `aws-cn` - China Regions
- `aws-us-gov` - AWS GovCloud (US) Region

For more information, see Access Management in the *AWS General Reference*.

Note

To specify an action, use the applicable prefix (`route53`, `route53domains`, or `route53resolver`) followed by the API operation name, for example:

- `route53:CreateHostedZone`
- `route53domains:RegisterDomain`
- `route53resolver:CreateResolverEndpoint`

Logging and monitoring in Amazon Route 53

Amazon Route 53 provides DNS query logging and the ability to monitor your resources using health checks. In addition, Route 53 integrates with other AWS services to provide additional logging and monitoring:

Logging DNS queries

You can configure Route 53 to log information about the queries that Route 53 receives, such as the domain or subdomain that was requested, the date and time of the request, and the DNS record type, such as A or AAAA.

For more information, see Public DNS query logging.

Using AWS CloudTrail to log console and programmatic actions

CloudTrail provides a record of Route 53 actions taken by a user, a role, or an AWS service. Using the information collected by CloudTrail, you can track the requests that are made, the IP addresses that requests originate from, who made the request, when it was made, and additional details. For more information, see Logging Amazon Route 53 API calls with AWS CloudTrail.

Monitoring domain registrations

The Route 53 dashboard provides detailed information about the status of your domain registrations, such as the status of domain transfers and domains that are approaching the expiration date.

For more information, see Monitoring domain registrations.

Using Route 53 health checks and Amazon CloudWatch to monitor your resources

You can monitor your resources by creating Route 53 health checks, which use CloudWatch to collect and process raw data into readable, near real-time metrics.

For more information, see Monitoring your resources with Amazon Route 53 health checks and Amazon CloudWatch.

Using Amazon CloudWatch to monitor Route 53 VPC Resolver endpoints

You can use CloudWatch to monitor the number of DNS queries that are forwarded by Resolver endpoints.

For more information, see Monitoring Route 53 VPC Resolver endpoints with Amazon CloudWatch.

Using AWS Trusted Advisor

Trusted Advisor draws upon best practices learned from serving AWS customers. Trusted Advisor inspects your AWS environment and then makes recommendations when opportunities exist to save money, improve system availability and performance, or help close security gaps. All AWS customers have access to five Trusted Advisor checks. Customers with a Business or Enterprise support plan can view all Trusted Advisor checks.

For more information, see Trusted Advisor.

Compliance validation for Amazon Route 53

Third-party auditors assess the security and compliance of Amazon Route 53 as part of multiple AWS compliance programs. These include SOC, PCI, FedRAMP, HIPAA, and others.

For a list of AWS services in scope of specific compliance programs, see AWS Services in Scope by Compliance Program. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading reports in AWS Artifact.

Your compliance responsibility when using Route 53 is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. If your use of Route 53 is subject to compliance with standards such as HIPAA, PCI, or FedRAMP, AWS provides resources to help:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying security- and compliance-focused baseline environments on AWS.
- Architecting for HIPAA Security and Compliance Whitepaper – This whitepaper describes how companies can use AWS to create HIPAA-compliant applications.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- AWS Config – This AWS service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub CSPM – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.

Resilience in Amazon Route 53

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between Availability Zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

Route 53 divides its functionality into a control and a data plane. Route 53 service, like most AWS services, includes a control plane that enables you to perform management operations such as creating, updating, and deleting resources, and a data plane that provides the service's core functionality. For more information about control and data planes in Route 53, see Control and data plane concepts.

Route 53 is primarily a global service, but the following features support AWS Regions:

- If you're using Route 53 VPC Resolver to set up hybrid configurations, you create endpoints in AWS Regions that you choose, and you specify IP addresses in multiple Availability Zones. For outbound endpoints, you create rules in the same Region where you created the endpoint. For more information, see What is Route 53 VPC Resolver?.
- You can configure Route 53 health checks to check the health of resources that you create in specific Regions, such as Amazon EC2 instances and Elastic Load Balancing load balancers.
- When you create a health check that monitors an endpoint, you can optionally specify the Regions that you want Route 53 to perform health checks from.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

Infrastructure security in Amazon Route 53

As a managed service, Amazon Route 53 is protected by AWS global network security. For information about AWS security services and how AWS protects infrastructure, see AWS Cloud Security. To design your AWS environment using the best practices for infrastructure security, see Infrastructure Protection in *Security Pillar AWS Well-Architected Framework*.

You use AWS published API calls to access Route 53 through the network. Clients must support the following:

- Transport Layer Security (TLS). We require TLS 1.2 and recommend TLS 1.3.
- Cipher suites with perfect forward secrecy (PFS) such as DHE (Ephemeral Diffie-Hellman) or ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Most modern systems such as Java 7 and later support these modes.

Sending findings from Resolver DNS Firewall to Security Hub CSPM

AWS Security Hub CSPM provides you with a comprehensive view of your security state in AWS and helps you to check your environment against security industry standards and best practices. Security Hub CSPM collects security data from across AWS accounts, AWS services, and supported third-party partner products, and helps you to analyze security trends and identify the highest priority security issues.

By integrating Resolver DNS Firewall with Security Hub CSPM, you can send findings from DNS Firewall to Security Hub CSPM. Security Hub CSPM then includes those findings in its analysis of your security posture.

Contents

- How findings work in Security Hub CSPM
  - Types of findings that DNS Firewall sends
  - Retrying when Security Hub CSPM is unavailable
  - Updating existing findings in Security Hub CSPM
- Typical finding from DNS Firewall
- Enabling and configuring the integration
- Stopping the delivery of findings to Security Hub CSPM

How findings work in Security Hub CSPM

In Security Hub CSPM, a finding is an observable record of a security check or security-related detection. Some findings come from issues that are detected by other AWS services or by third-party partners. Security Hub CSPM also has its own security controls that it uses to detect security issues and generate findings.

Security Hub CSPM provides tools to manage findings from across all of these sources. You can view and filter lists of findings and view details of a finding. For information, see Reviewing finding details and finding history in Security Hub CSPM in the *AWS Security Hub User Guide*. You can also automatically update findings or send them to a custom action. For more information, see Automatically modifying and taking action on Security Hub CSPM findings in the *AWS Security Hub User Guide*.

All findings in Security Hub CSPM use a standard JSON format called the AWS Security Finding Format (ASFF). The ASFF includes details about the source of the security issue, the affected resources, and the current status of the finding. For more information, see AWS Security Finding Format (ASFF) in the *AWS Security Hub User Guide*.

DNS Firewall is one of the AWS services that sends findings to Security Hub CSPM.

Types of findings that DNS Firewall sends

DNS Firewall has the following integrations:

- **Managed Domain Lists**: security findings related to queries blocked or alerted on for domains associated with AWS Managed Domain Lists.
- **Custom domain lists**: security findings related to queries blocked or alerted on for domains associated with the customer's domain list.
- **DNS Firewall Advanced**: security findings related to queries blocked or alerted on by DNS Firewall Advanced.

Security Hub CSPM ingests findings from DNS Firewall in the AWS Security Finding Format (ASFF). In ASFF, the `Types` field provides the finding type. Findings from DNS Firewall can have the following values for `Types`.

- `TTPs/Impact/Impact:Runtime-MaliciousDomainRequest.Reputation`

Retrying when Security Hub CSPM is unavailable

If Security Hub CSPM is unavailable, DNS Firewall retries sending the findings until they are received.

Updating existing findings in Security Hub CSPM

DNS Firewall will update the existing findings if the same finding is observed again.

Typical finding from DNS Firewall

Security Hub CSPM ingests DNS Firewall findings in the AWS Security Finding Format (ASFF).

Here is an example of a typical finding from DNS Firewall in ASFF.


```
{
            "SchemaVersion": "2018-10-08",
            "Id": "00000000-0000-0000-0000-example1",
            "ProductArn": "arn:aws:securityhub:us-east-1::product/amazon/route-53-
resolver-dns-firewall-aws-list",
            "ProductName": "Route 53 Resolver DNS Firewall - AWS List",
            "CompanyName": "Amazon",
            "Region": "us-east-1",
            "GeneratorId": "arn:aws:route53resolver:us-east-1:000000000000:firewall-
rule-group/rslvr-frg-example1",
            "AwsAccountId": "000000000000",
            "Types": [
                "TTPs/Impact/Impact:Runtime-MaliciousDomainRequest.Reputation"
            ],
            "FirstObservedAt": "2024-12-06T19:58:49.000Z",
            "LastObservedAt": "2024-12-06T19:58:49.000Z",
            "CreatedAt": "2024-12-06T19:58:49.000Z",
            "UpdatedAt": "2024-12-06T19:58:49.000Z",
            "Severity": {
                "Label": "HIGH",
                "Normalized": 70
            },
            "Title": "DNS Firewall ALERT generated for domain example1.com. from VPC
 vpc-example1",
            "Description": "DNS Firewall ALERT",
            "ProductFields": {
                "aws/route53resolver/dnsfirewall/queryName": "example1.com.",
                "aws/route53resolver/dnsfirewall/firewallRuleGroupId": "rslvr-frg-
example1",
                "aws/route53resolver/dnsfirewall/queryType": "A",
                "aws/route53resolver/dnsfirewall/queryClass": "IN",
                "aws/route53resolver/dnsfirewall/firewallDomainListId": "rslvr-fdl-
example1",
                "aws/route53resolver/dnsfirewall/transport": "UDP",
                "aws/route53resolver/dnsfirewall/firewallRuleAction": "ALERT",
                "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-1::product/
amazon/route-53-resolver-dns-firewall-aws-list/00000000-0000-0000-0000-example1",
                "aws/securityhub/ProductName": "Route 53 Resolver DNS Firewall - AWS
 List",
                "aws/securityhub/CompanyName": "Amazon"
            },
            "Resources": [
```

```
            {
                "Type": "Other",
                "Id": "rslvr-in-example1",
                "Partition": "aws",
                "Region": "us-east-1",
                "Details": {
                    "Other": {
                        "ResourceType": "ResolverEndpoint",
                        "EndpointId": "rslvr-in-example1"
                    }
                }
            },
            {
                "Type": "Other",
                "Id": "rni-example1",
                "Partition": "aws",
                "Region": "us-east-1",
                "Details": {
                    "Other": {
                        "NetworkInterfaceId": "rni-example1",
                        "ResourceType": "ResolverNetworkInterface"
                    }
                }
            }
        ],
        "WorkflowState": "NEW",
        "Workflow": {
            "Status": "NEW"
        },
        "RecordState": "ACTIVE",
        "FindingProviderFields": {
            "Severity": {
                "Label": "HIGH"
            },
            "Types": [
                "TTPs/Impact/Impact:Runtime-MaliciousDomainRequest.Reputation"
            ]
        },
        "ProcessedAt": "2024-12-11T19:33:35.494Z"
    }
```

Enabling and configuring the integration

To integrate DNS Firewall with Security Hub CSPM, you must first enable Security Hub CSPM. For information about enabling Security Hub CSPM, see Enabling Security Hub CSPM in the *AWS Security Hub User Guide*.

Stopping the delivery of findings to Security Hub CSPM

To stop sending DNS Firewall findings to Security Hub CSPM, you can use the Security Hub CSPM console or the Security Hub CSPM API.

For instructions, see Disabling the flow of findings from an integration in the *AWS Security Hub User Guide*.

Monitoring Amazon Route 53

Monitoring is an important part of maintaining the reliability, availability, and performance of your AWS solutions. You should collect monitoring data from all of the parts of your AWS solution so that you can more easily debug a multi-point failure if one occurs. However, before you start monitoring, you should create a monitoring plan that includes answers to the following questions:

- What are your monitoring goals?
- What resources will you monitor?
- How often will you monitor these resources?
- What monitoring tools will you use?
- Who will perform the monitoring tasks?
- Who should be notified when something goes wrong?

Topics

- Public DNS query logging
- Resolver query logging
- Monitoring domain registrations
- Monitoring your resources with Amazon Route 53 health checks and Amazon CloudWatch
- Monitoring hosted zones using Amazon CloudWatch
- Monitoring Route 53 VPC Resolver endpoints with Amazon CloudWatch
- Monitoring Resolver DNS Firewall rule groups with Amazon CloudWatch
- Managing Resolver DNS Firewall events using Amazon EventBridge
- Logging Amazon Route 53 API calls with AWS CloudTrail

Public DNS query logging

You can configure Amazon Route 53 to log information about the public DNS queries that Route 53 receives, such as the following:

- Domain or subdomain that was requested

- Date and time of the request
- DNS record type (such as A or AAAA)
- Route 53 edge location that responded to the DNS query
- DNS response code, such as `NoError` or `ServFail`

After you configure query logging, Route 53 will send logs to CloudWatch Logs. You use CloudWatch Logs tools to access the query logs.

Query logs contain only the queries that DNS resolvers send to Route 53. If a DNS resolver has already cached the response to a query (such as the IP address for a load balancer for example.com), the resolver will continue to return the cached response without sending the query to Route 53 until the TTL for the corresponding record expires.

Query logs might not capture every DNS query. The number of logged queries depends on how many queries are submitted for a domain name (example.com) or subdomain name (www.example.com), which resolvers your users use, and the TTL for the record. In some cases, logs contain only one query out of every several thousand. For more information about how DNS works, see How internet traffic is routed to your website or web application.

If you don't need detailed logging information, you can use Amazon CloudWatch metrics to see the total number of DNS queries that Route 53 responds to for a hosted zone. For more information, see Viewing DNS query metrics for a public hosted zone.

Topics

- Configuring logging for DNS queries
- Using Amazon CloudWatch to access DNS query logs
- Changing the retention period for logs and exporting logs to Amazon S3
- Stopping query logging
- Values that appear in DNS query logs
- Query log example

Configuring logging for DNS queries

To start logging DNS queries for a specified hosted zone, you perform the following tasks in the Amazon Route 53 console:

- Choose the CloudWatch Logs log group that you want Route 53 to publish logs to, or create a new log group.

  Note

  The log group must be in the US East (N. Virginia) Region.

- Choose **Create** to finish.

Note

If users are submitting DNS queries for your domain, you should start to see queries in the logs within several minutes after you create the query logging configuration.

To configure logging for DNS queries

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the hosted zone that you want to configure query logging for.
4. In the **Hosted zone details** pane, choose **Configure query logging**.
5. Choose an existing log group or create a new log group.
6. If you receive an alert about permissions (this happens if you haven't configured query logging with the new console before), do one of the following:

   - If you have 10 resource policies already, you can't create any more. Select any of your resource policies, and select **Edit** . Editing will give Route 53 permissions to write logs to your log groups. Choose **Save**. The alert goes away and you can continue to the next step.
   - If you have never configured query logging before (or if you haven't created 10 resource policies already), you need to grant permissions to Route 53 to write logs to your CloudWatch Logs groups. Choose **Grant permissions**. The alert goes away and you can continue to the next step.
7. Choose **Permissions - optional** to see a table that shows whether the resource policy matches the CloudWatch log group, and whether the Route 53 has the permission to publish logs to CloudWatch.

8. Choose **Create**.

Using Amazon CloudWatch to access DNS query logs

Amazon Route 53 sends query logs directly to CloudWatch Logs; the logs are never accessible through Route 53. Instead, you use CloudWatch Logs to view logs in near real-time, search and filter data, and export logs to Amazon S3.

Route 53 creates one CloudWatch Logs log stream for each Route 53 edge location that responds to DNS queries for the specified hosted zone and sends query logs to the applicable log stream. The format for the name of each log stream is *hosted-zone-id*/*edge-location-ID*, for example, `Z1D633PJN98FT9/DFW3`.

Each edge location is identified by a three-letter code and an arbitrarily assigned number, for example, DFW3. The three-letter code typically corresponds with the International Air Transport Association airport code for an airport near the edge location. (These abbreviations might change in the future.) For a list of edge locations, see "The Route 53 Global Network" on the Route 53 Product Details page.

> **Note**
>
> You might see some prefixes or suffixes that don't follow the above convention. Those encode attributes that are for internal use only.

For more information, see the applicable documentation:

- Amazon CloudWatch Logs User Guide
- Amazon CloudWatch Logs API Reference
- CloudWatch Logs section of the AWS CLI Command Reference
- Values that appear in DNS query logs

Changing the retention period for logs and exporting logs to Amazon S3

By default, CloudWatch Logs stores query logs indefinitely. You can optionally specify a retention period so that CloudWatch Logs deletes logs that are older than the retention period. For more

information, see Change log data retention in CloudWatch Logs in the *Amazon CloudWatch User Guide*.

If you want to retain log data but you don't need CloudWatch Logs tools to view and analyze the data, you can export logs to Amazon S3, which can reduce your storage costs. For more information, see Exporting log data to Amazon S3.

For information about pricing, see the applicable pricing page:

- "Amazon CloudWatch Logs" on the CloudWatch Pricing page
- Amazon S3 Pricing

> **Note**
>
> When you configure Route 53 to log DNS queries, you don't incur any Route 53 charges.

Stopping query logging

If you want Amazon Route 53 to stop sending query logs to CloudWatch Logs, perform the following procedure to delete the query logging configuration.

To delete a query logging configuration

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted zones**.
3. Choose the name for the hosted zone that you want to delete the query logging configuration for.
4. In the **Hosted zone details** pane, choose **Delete query logging configuration**.
5. Choose **Delete** to confirm.

Values that appear in DNS query logs

Each log file contains one log entry for each DNS query that Amazon Route 53 received from DNS resolvers in the corresponding edge location. Each log entry includes the following values:

Log format version

The version number of this query log. If we add fields to the log or change the format of existing fields, we'll increment this value.

Query timestamp

The date and time that Route 53 responded to the request, in ISO 8601 format and Coordinated Universal Time (UTC), for example, `2017-03-16T19:20:25.177Z`.

For information about ISO 8601 format, see the Wikipedia article ISO 8601. For information about UTC, see the Wikipedia article Coordinated Universal Time.

Hosted zone ID

The ID of the hosted zone that is associated with all the DNS queries in this log.

Query name

The domain or subdomain that was specified in the request.

Query type

Either the DNS record type that was specified in the request, or `ANY`. For information about the types that Route 53 supports, see Supported DNS record types.

Response code

The DNS response code that Route 53 returned in response to the DNS query.

Layer 4 protocol

The protocol that was used to submit the query, either `TCP` or `UDP`.

Route 53 edge location

The Route 53 edge location that responded to the query. Each edge location is identified by a three-letter code and an arbitrary number, for example, DFW3. The three-letter code typically corresponds with the International Air Transport Association airport code for an airport near the edge location. (These abbreviations might change in the future.)

For a list of edge locations, see "The Route 53 Global Network" on the Route 53 Product Detail page.

Resolver IP address

The IP address of the DNS resolver that submitted the request to Route 53.

EDNS client subnet

A partial IP address for the client that the request originated from, if available from the DNS resolver.

For more information, see the IETF draft Client Subnet in DNS Requests.

Query log example

Here's an example query log (Region is a placeholder):

```
1.0 2017-12-13T08:16:02.130Z Z123412341234 example.com A NOERROR UDP Region 192.168.1.1
 -
1.0 2017-12-13T08:15:50.235Z Z123412341234 example.com AAAA NOERROR TCP Region
 192.168.3.1 192.168.222.0/24
1.0 2017-12-13T08:16:03.983Z Z123412341234 example.com ANY NOERROR UDP Region
 2001:db8::1234 2001:db8:abcd::/48
1.0 2017-12-13T08:15:50.342Z Z123412341234 bad.example.com A NXDOMAIN UDP Region
 192.168.3.1 192.168.111.0/24
1.0 2017-12-13T08:16:05.744Z Z123412341234 txt.example.com TXT NOERROR UDP Region
 192.168.1.2 -
```

Resolver query logging

You can log the following DNS queries:

- Queries that originate in Amazon Virtual Private Cloud VPCs that you specify, as well as the responses to those DNS queries.
- Queries from on-premises resources that use an inbound Resolver endpoint.
- Queries that use an outbound Resolver endpoint for recursive DNS resolution.
- Queries that use Resolver DNS Firewall rules to block, allow, or monitor domain lists.

VPC Resolver query logs include values such as the following:

- The AWS Region where the VPC was created
- The ID of the VPC that the query originated from
- The IP address of the instance that the query originated from
- The instance ID of the resource that the query originated from

- The date and time that the query was first made
- The DNS name requested (such as prod.example.com)
- The DNS record type (such as A or AAAA)
- The DNS response code, such as `NoError` or `ServFail`
- The DNS response data, such as the IP address that is returned in response to the DNS query
- A response to a DNS Firewall rule action

For a detailed list of all of the values logged and an example, see Values that appear in VPC Resolver query logs.

Note

As is standard for DNS resolvers, resolvers cache DNS queries for a length of time determined by the time-to-live (TTL) for the resolver. The Route 53 VPC Resolver caches queries that originate in your VPCs, and responds from the cache whenever possible to speed up responses. VPC Resolver query logging logs only unique queries, not queries that VPC Resolver is able to respond to from the cache.
For example, suppose that an EC2 instance in one of the VPCs that a query logging configuration is logging queries for, submits a request for accounting.example.com. VPC Resolver caches the response to that query, and logs the query. If the same instance's elastic network interface makes a query for accounting.example.com within the TTL of the VPC Resolver's cache, VPC Resolver responds to the query from the cache. The second query is not logged.

You can send the logs to one of the following AWS resources:

- Amazon CloudWatch Logs (CloudWatch Logs) log group
- Amazon S3 (S3) bucket
- Firehose delivery stream

For more information, see AWS resources that you can send VPC Resolver query logs to.

Topics

- AWS resources that you can send VPC Resolver query logs to

- Managing Resolver query logging configurations

AWS resources that you can send VPC Resolver query logs to

> **Note**
>
> If you expect to log queries for workloads with high queries per second (QPS), you should use Amazon S3 to make sure your query logs are not throttled when written to your destination. If you use Amazon CloudWatch, you can increase your requests per second limit for the `PutLogEvents` operation. To learn more about increasing your CloudWatch limits, see CloudWatch Logs quotas in the *Amazon CloudWatch User Guide*.

You can send VPC Resolver query logs to the following AWS resources:

Amazon CloudWatch Logs (Amazon CloudWatch Logs) log group

You can analyze logs with Logs Insights and create metrics and alarms.

For more information, see the Amazon CloudWatch Logs User Guide.

Amazon S3 (S3) bucket

An S3 bucket is economical for long-term log archiving. Latency is typically higher.

All S3 server-side encryption options are supported. For more information, see Protecting data with server-side encryption in the *Amazon S3 User Guide*.

If you choose Server-Side Encryption with AWS KMS Keys (SSE-KMS), you must update the key policy for your customer managed key so that the log delivery account can write to your Amazon S3 bucket. For more information about the required key policy for use with SSE-KMS, see Amazon S3 bucket server-side encryption in the *Amazon CloudWatch User Guide*.

If the S3 bucket is in an account that you own, the required permissions are automatically added to your bucket policy. If you want to send logs to an S3 bucket in an account that you don't own, the owner of the S3 bucket must add permissions for your account in their bucket policy. For example:

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Id": "CrossAccountAccess",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::your_bucket_name/
AWSLogs/your_caller_account/*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "your_caller_account"
                }
            }
        },
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::your_bucket_name",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "your_caller_account"
                }
            }
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "iam_user_arn_or_account_number_for_root"
            },
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::your_bucket_name"
        }
    ]
}
```

> **Note**
>
> If you want to store logs in a central S3 bucket for your organization, we recommend that you set up your query logging configuration from a centralized account (with the necessary permissions to write to a central bucket) and use RAM to share the configuration across accounts.

For more information, see the Amazon Simple Storage Service User Guide.

Firehose delivery stream

You can stream logs in real time to Amazon OpenSearch Service, Amazon Redshift, or other applications.

For more information, see the Amazon Data Firehose Developer Guide.

For information about the pricing for Resolver query logging, see Amazon CloudWatch pricing.

CloudWatch Vended Logs charges apply when using VPC Resolver logs, even when logs are published directly to Amazon S3. For more information, see *Logs pricing* at Amazon CloudWatch pricing.

Managing Resolver query logging configurations

Configuring (VPC Resolver query logging)

You can configure VPC Resolver query logging in two ways:

- **Direct VPC association** - Associate VPCs directly to a query logging configuration.
- **Profile association** - Associate a query logging configuration to a Route 53 Profile, which applies the logging to all VPCs associated with that Profile. For more information, see Associate VPC Resolver query logging configurations to a Route 53 Profile.

To start logging DNS queries that originate in your VPCs, you perform the following tasks in the Amazon Route 53 console:

To configure Resolver query logging

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. Expand the Route 53 console menu. In the upper left corner of the console, choose the three horizontal bars

   ( )

   icon.
3. Within the Resolver menu, choose **Query logging**.
4. In the Region selector, choose the AWS Region where you want to create the query logging configuration. This must be the same Region where you created the VPCs that you want to log DNS queries for. If you have VPCs in multiple Regions, you must create at least one query logging configuration for each Region.
5. Choose **Configure query logging**.
6. Specify the following values:

   Query logging configuration name

   Enter a name for your query logging configuration. The name appears in the console in the list of query logging configurations. Enter a name that will help you find this configuration later.

   Query logs destination

   Choose the type of AWS resource that you want VPC Resolver to send query logs to. For information about how to choose among the options (CloudWatch Logs log group, S3 bucket, and Firehose delivery stream), see AWS resources that you can send VPC Resolver query logs to.

   After you choose the type of resource, you can either create another resource of that type or choose an existing resource that was created by the current AWS account.

   > **Note**
   >
   > You can choose only resources that were created in the AWS Region that you chose in step 4, the Region where you're creating the query logging configuration. If you choose to create a new resource, that resource will be created in the same Region.

VPCs to log queries for

This query logging configuration will log DNS queries that originate in the VPCs that you choose. Check the check box for each VPC in the current Region that you want VPC Resolver to log queries for, then choose **Choose**.

Alternative: Instead of associating VPCs directly, you can associate this query logging configuration to a Route 53 Profile, which will apply logging to all VPCs associated with that Profile. For more information, see Associate VPC Resolver query logging configurations to a Route 53 Profile.

> **Note**
>
> VPC log delivery can be enabled only once for a specific destination type. The logs can't be delivered to multiple destinations of the same type, for example, VPC logs can't be delivered to 2 Amazon S3 destinations.

7. Choose **Configure query logging**.

> **Note**
>
> You should start to see DNS queries made by resources in your VPC in the logs within a few minutes of successfully creating the query logging configuration.

Values that appear in VPC Resolver query logs

Each log file contains one log entry for each DNS query that Amazon Route 53 received from DNS resolvers in the corresponding edge location. Each log entry includes the following values:

version

The version number of the query log format. The current version is `1.1`.

The version value is a major and minor version in the form **`major_version.minor_version`**. For example, you can have a `version` value of `1.7`, where `1` is the major version, and 7 is the minor version.

Route 53 increments the major version if a change is made to the log structure that is not backward-compatible. This includes removing a JSON field that already exists, or changing how the contents of a field are represented (for example, a date format).

Route 53 increments the minor version if a change adds new fields to the log file. This can occur if new information is available for some or all existing DNS queries within a VPC.

account_id

The ID of the AWS account that created the VPC.

region

The AWS Region that you created the VPC in.

vpc_id | resource_id

- `vpc_id` – The ID of the VPC that the query originated in. VPC ID is not included when the query logging configuration is applied through Route 53 Profiles.
- `resource_id` – The ID of the Route 53 Profile that the query logging configuration is associated with. Included only when the configuration is associated with a Route 53 Profile.

query_timestamp

The date and time that the query was submitted, in ISO 8601 format and Coordinated Universal Time (UTC), for example, `2017-03-16T19:20:177Z`.

For information about ISO 8601 format, see the Wikipedia article ISO 8601. For information about UTC, see the Wikipedia article Coordinated Universal Time.

query_name

The domain name (example.com) or subdomain name (www.example.com) that was specified in the query.

query_type

Either the DNS record type that was specified in the request, or `ANY`. For information about the types that Route 53 supports, see Supported DNS record types.

query_class

The class of the query.

rcode

The DNS response code that VPC Resolver returned in response to the DNS query. The response code indicates whether the query was valid or not. The most common response code is

NOERROR, meaning that the query was valid. If the response is not valid, Resolver returns a response code that explains why not. For a list of possible response codes, see DNS RCODEs on the IANA website.

answer_type

The DNS record type (such as A, MX, or CNAME) of the value that VPC Resolver is returning in response to the query. For information about the types that Route 53 supports, see Supported DNS record types.

rdata

The value that VPC Resolver returned in response to the query. For example, for an A record, this is an IP address in IPv4 format. For a CNAME record, this is the domain name in the CNAME record.

answer_class

The class of the VPC Resolver response to the query.

srcaddr

IP address of the host that originated the query.

srcport

The port on the instance that the query originated from.

transport

The protocol used to submit the DNS query.

srcids

IDs of the `instance`, `resolver_endpoint`, and the `resolver_network_interface` that the DNS query originated from or passed through.

instance

The ID of the instance that the query originated from.

> **Note**
>
> If you see an instance ID in Route 53 VPC Resolver query logs which is not visible in your account, it might be because the DNS query originated from either AWS CloudShell, AWS Lambda, Amazon EKS, or Fargate console, which was used by you.

resolver_endpoint

The ID of the resolver endpoint that passes the DNS query to on-premises DNS servers.

> **Note**
>
> If you have CNAME records that chain across different forwarding rules using different resolver endpoints, query logs show only the ID of the last resolver endpoint used in the chain. To trace the complete resolution path through multiple endpoints, you can correlate logs across different query logging configurations.

firewall_rule_group_id

The ID of the DNS Firewall rule group that matched the domain name in the query. This is populated only if DNS Firewall found a match for a rule with action set to alert or block.

For more information about the firewall rule groups, see DNS Firewall rule groups and rules.

firewall_rule_action

The action specified by the rule that matched the domain name in the query. This is populated only if DNS Firewall found a match for a rule with action set to alert or block.

firewall_domain_list_id

The domain list used by the rule that matched the domain name in the query. This is populated only if DNS Firewall found a match for a rule with action set to alert or block.

additional_properties

Additional information of the log delivery events. **is_delayed**: If there is a delay in delivering the logs.

Route 53 VPC Resolver query log example

Here's a resolver query log example:

```
{
  "srcaddr": "4.5.64.102",
```

```
        "vpc_id": "vpc-7example",
        "answers": [
            {
                "Rdata": "203.0.113.9",
                "Type": "PTR",
                "Class": "IN"
            }
        ],
        "firewall_rule_group_id": "rslvr-frg-01234567890abcdef",
        "firewall_rule_action": "BLOCK",
        "query_name": "15.3.4.32.in-addr.arpa.",
        "firewall_domain_list_id": "rslvr-fdl-01234567890abcdef",
        "query_class": "IN",
        "srcids": {
            "instance": "i-0d15cd0d3example"
        },
        "rcode": "NOERROR",
        "query_type": "PTR",
        "transport": "UDP",
        "version": "1.100000",
        "account_id": "111122223333",
        "srcport": "56067",
        "query_timestamp": "2021-02-04T17:51:55Z",
        "region": "us-east-1"
    }
```


Sharing Resolver query logging configurations with other AWS accounts

You can share the query logging configurations that you created using one AWS account with other AWS accounts. To share configurations, the Route 53 VPC Resolver console integrates with AWS Resource Access Manager. For more information about Resource Access Manager, see the Resource Access Manager User Guide.

Note the following:

Associating VPCs with shared query logging configurations

If another AWS account has shared one or more configurations with your account, you can associate VPCs with the configuration the same way that you associate VPCs with configurations that you created.

Deleting or unsharing a configuration

If you share a configuration with other accounts and then either delete the configuration or stop sharing it, and if one or more VPCs were associated with the configuration, Route 53 VPC Resolver stops logging DNS queries that originate in those VPCs.

Maximum number of query logging configurations and VPCs that can be associated with a config

When an account creates a configuration and shares it with one or more other accounts, the maximum number of VPCs that can be associated with the configuration are applied per account. For example, if you have 10,000 accounts in your organization, you can create the query logging configuration in the central account and share it through AWS RAM to share it to the organization accounts. The organization accounts will then associate the configuration with their VPCs counting them against their account's query log configuration VPC associations per AWS Region limit of 100. However, if all the VPCs are in a single account, then the account's service limits might be needed to increased.

For current VPC Resolver quotas, see Quotas on Route 53 VPC Resolver.

Permissions

To share a rule with another AWS account, you must have permission to use the PutResolverQueryLogConfigPolicy action.

Restrictions on the AWS account that a rule is shared with

The account that a rule is shared with can't change or delete the rule.

Tagging

Only the account that created a rule can add, delete, or see tags on the rule.

To view the current sharing status of a rule (including the account that shared the rule or the account that a rule is shared with), and to share rules with another account, perform the following procedure.

To view sharing status and share query logging configurations with another AWS account

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Query Logging**.

3. On the navigation bar, choose the Region where you created the rule.

   The **Sharing status** column shows the current sharing status of rules that were created by the current account or that are shared with the current account:

   - **Not shared**: The current AWS account created the rule, and the rule is not shared with any other accounts.
   - **Shared by me**: The current account created the rule and shared it with one or more accounts.
   - **Shared with me**: Another account created the rule and shared it with the current account.
4. Choose the name of the rule that you want to display sharing information for or that you want to share with another account.

   On the **Rule: *`rule name`*** page, the value under **Owner** displays ID of the account that created the rule. That's the current account unless the value of **Sharing status** is **Shared with me**. In that case, **Owner** is the account that created the rule and shared it with the current account.

   The sharing status is also displayed.
5. Choose **Share configuration** to open the AWS RAM console
6. To create a resource share, follow the steps in Creating a resource share in AWS RAM in the *AWS RAM user guide*.

   > **Note**
   >
   > You can't update sharing settings. If you want to change any of the following settings, you must reshare a rule with the new settings and then remove the old sharing settings.

Monitoring domain registrations

The Amazon Route 53 dashboard provides detailed information about the status of your domain registrations, including the following:

- Status of new domain registrations
- Status of domain transfers to Route 53
- List of domains that are approaching the expiration date

We recommend that you periodically check the dashboard in the Route 53 console, especially after you register a new domain or transfer a domain to Route 53, to confirm that there are no issues for you to address.

We also recommend that you confirm that the contact information for your domains is up to date. As the expiration date for a domain approaches, Route 53 emails the registrant contact for the domain with information about when the domain expires and how to renew.

Monitoring your resources with Amazon Route 53 health checks and Amazon CloudWatch

You can monitor your resources by creating Amazon Route 53 health checks, which use CloudWatch to collect and process raw data into readable, near real-time metrics. These statistics are recorded for a period of two weeks, so that you can access historical information and gain a better perspective on how your resources are performing. By default, metric data for Route 53 health checks is automatically sent to CloudWatch at one-minute intervals.

For more information about Route 53 health checks, see Monitoring health checks using CloudWatch. For more information about CloudWatch, see What is Amazon CloudWatch? in the *Amazon CloudWatch User Guide*.

Metrics and dimensions for Route 53 health checks

When you create a health check, Amazon Route 53 starts to send metrics and dimensions once a minute to CloudWatch about the resource that you specify. With the Route 53 console, you can view the status of your health checks. You can also use the following procedures to view the metrics in the CloudWatch console or view them by using the AWS Command Line Interface (AWS CLI).

To view metrics using the CloudWatch console

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. In the navigation pane, choose **Metrics**.
3. On the **All Metrics** tab, choose **Route 53**.
4. Choose **Health Check Metrics**.

To view metrics using the AWS CLI

- At a command prompt, use the following command:

```
aws cloudwatch list-metrics --namespace "AWS/Route53"
```

Topics

- CloudWatch metrics for Route 53 health checks
- Dimensions for Route 53 health check metrics

CloudWatch metrics for Route 53 health checks

The `AWS/Route53` namespace includes the following metrics for Route 53 health checks.

ChildHealthCheckHealthyCount

For a calculated health check, the number of health checks that are healthy.

Valid statistics: Average (recommended), Minimum, Maximum

Units: Count

ConnectionTime

The average time, in milliseconds, that it took Route 53 health checkers to establish a TCP connection with the endpoint. You can view `ConnectionTime` for a health check either across all regions or for a selected geographic region.

Valid statistics: Average (recommended), Minimum, Maximum

Units: Milliseconds

HealthCheckPercentageHealthy

The percentage of Route 53 health checkers that consider the selected endpoint to be healthy.

Valid statistics: Average, Minimum, Maximum

Units: Percent

HealthCheckStatus

The status of the health check endpoint that CloudWatch is checking. **1** indicates healthy, and **0** indicates unhealthy.

Valid statistics: Minimum, Average, and Maximum

Units: none

SSLHandshakeTime

The average time, in milliseconds, that it took Route 53 health checkers to complete the SSL handshake. You can view `SSLHandshakeTime` for a health check either across all regions or for a selected geographic region.

Valid statistics: Average (recommended), Minimum, Maximum

Units: Milliseconds

TimeToFirstByte

The average time, in milliseconds, that it took Route 53 health checkers to receive the first byte of the response to an HTTP or HTTPS request. You can view `TimeToFirstByte` for a health check either across all regions or for a selected geographic region.

Valid statistics: Average (recommended), Minimum, Maximum

Units: Milliseconds

Dimensions for Route 53 health check metrics

Route 53 metrics for health checks use the `AWS/Route53` namespace and provide metrics for `HealthCheckId`. When retrieving metrics, you must supply the `HealthCheckId` dimension.

In addition, for `ConnectionTime`, `SSLHandshakeTime`, and `TimeToFirstByte`, you can optionally specify `Region`. If you omit `Region`, CloudWatch returns metrics across all regions. If you include `Region`, CloudWatch returns metrics only for the specified region.

For more information, see Monitoring health checks using CloudWatch.

Monitoring hosted zones using Amazon CloudWatch

You can monitor your public hosted zones by using Amazon CloudWatch to collect and process raw data into readable, near real-time metrics. Metrics are available shortly after Route 53 receives the DNS queries that the metrics are based on. CloudWatch metric data for Route 53 hosted zones has a granularity of one minute.

For more information, see the following documentation

- For an overview and information about how to view metrics in the Amazon CloudWatch console and how to retrieve metrics using the AWS Command Line Interface (AWS CLI), see Viewing DNS query metrics for a public hosted zone
- For information about the retention period for metrics, see GetMetricStatistics in the *Amazon CloudWatch API Reference*.
- For more information about CloudWatch, see What is Amazon CloudWatch? in the *Amazon CloudWatch User Guide*.
- For more information about CloudWatch metrics, see Using Amazon CloudWatch metrics in the *Amazon CloudWatch User Guide*.

Topics

- CloudWatch metrics for Route 53 public hosted zones
- CloudWatch dimension for Route 53 public hosted zone metrics

CloudWatch metrics for Route 53 public hosted zones

The `AWS/Route53` namespace includes the following metrics for Route 53 hosted zones:

DNSQueries

For a hosted zone, the number of DNS queries that Route 53 responds to in a specified time period.

Valid statistics: Sum, SampleCount

Units: Count

Region: Route 53 is a global service. To get hosted zone metrics, you must specify US East (N. Virginia) for the Region.

DNSSECInternalFailure

Value is 1 if any object in the hosted zone is in an INTERNAL_FAILURE state. Otherwise, value is 0.

Valid statistics: Sum

Units: Count

Volume: 1 per 4 hours per hosted zone

Region: Route 53 is a global service. To get hosted zone metrics, you must specify US East (N. Virginia) for the Region.

DNSSECKeySigningKeysNeedingAction

Number of key signing keys (KSKs) that have an ACTION_NEEDED state (due to KMS failure).

Valid statistics: Sum, SampleCount

Units: Count

Volume: 1 per 4 hours per hosted zone

Region: Route 53 is a global service. To get hosted zone metrics, you must specify US East (N. Virginia) for the Region.

DNSSECKeySigningKeyMaxNeedingActionAge

Time elapsed since the key signing key (KSK) was set to the ACTION_NEEDED state.

Valid statistics: Maximum

Units: Seconds

Volume: 1 per 4 hours per hosted zone

Region: Route 53 is a global service. To get hosted zone metrics, you must specify US East (N. Virginia) for the Region.

DNSSECKeySigningKeyAge

The time elapsed since the key signing key (KSK) was created (not since it was activated).

Valid statistics: Maximum

Units: Seconds

Volume: 1 per 4 hours per hosted zone

Region: Route 53 is a global service. To get hosted zone metrics, you must specify US East (N. Virginia) for the Region.

CloudWatch dimension for Route 53 public hosted zone metrics

Route 53 metrics for hosted zones use the `AWS/Route53` namespace and provide metrics for `HostedZoneId`. To get the number of DNS queries, you must specify the ID of the hosted zone in the `HostedZoneId` dimension.

Monitoring Route 53 VPC Resolver endpoints with Amazon CloudWatch

You can use Amazon CloudWatch to monitor the number of DNS queries that are forwarded by Route 53 VPC Resolver endpoints. Amazon CloudWatch collects and processes raw data into readable, near real-time metrics. These statistics are recorded for a period of two weeks, so that you can access historical information and gain a better perspective on how your resources are performing. By default, metric data for Resolver endpoints is automatically sent to CloudWatch at five-minute intervals. The five-minute interval is also the smallest interval at which the metric data can be sent.

For more information about VPC Resolver, see What is Route 53 VPC Resolver?. For more information about CloudWatch, see What is Amazon CloudWatch? in the *Amazon CloudWatch User Guide*.

Metrics and dimensions for Route 53 VPC Resolver

When you configure VPC Resolver to forward DNS queries to your network or vice versa, VPC Resolver starts to send metrics and dimensions once every five minutes to CloudWatch about the number of queries that are forwarded. You can use the following procedures to view the metrics in the CloudWatch console or view them by using the AWS Command Line Interface (AWS CLI).

To view VPC Resolver metrics using the CloudWatch console

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.

2. On the navigation bar, choose the Region where you created the endpoint.
3. In the navigation pane, choose **Metrics**.
4. On the **All metrics** tab, choose **Route 53 Resolver**.
5. Choose **By Endpoint** to view query counts for a specified endpoint. Then choose the endpoints that you want to view the number of queries for.

   Choose **Across All Endpoints** to view query counts for all inbound endpoints or for all outbound endpoints that were created by the current AWS account. Then choose **InboundQueryVolume** or **OutboundQueryVolume** to view the desired counts.

To view metrics using the AWS CLI

- At a command prompt, use the following command:

```
aws cloudwatch list-metrics --namespace "AWS/Route53Resolver"
```

Topics

- CloudWatch Basic Metrics for Route 53 VPC Resolver
- CloudWatch Detailed Metrics for Route 53 VPC Resolver
- Dimensions for Route 53 VPC Resolver metrics

CloudWatch Basic Metrics for Route 53 VPC Resolver

`AWS/Route53Resolver` namespace includes basic metrics for Route 53 VPC Resolver endpoints and for IP addresses at free of cost.

Topics

- Metrics for Route 53 VPC Resolver endpoints
- Metrics for Route 53 VPC Resolver IP addresses

Metrics for Route 53 VPC Resolver endpoints

The `AWS/Route53Resolver` namespace includes the following metrics for Route 53 VPC Resolver endpoints.

EndpointHealthyENICount

The number of elastic network interfaces in the `OPERATIONAL` status. This means that the Amazon VPC network interfaces for the endpoint (specified by `EndpointId`) are correctly configured and able to pass inbound or outbound DNS queries between your network and Resolver.

Valid Statistics: Minimum, Maximum, Average

Units: Count

EndpointUnhealthyENICount

The number of elastic network interfaces in the `AUTO_RECOVERING` status.

This means that the resolver is trying to recover one or more of the Amazon VPC network interfaces that are associated with the endpoint (specified by `EndpointId`). During the recovery process, the endpoint functions with limited capacity and is unable to process DNS queries until it's fully recovered.

Valid Statistics: Minimum, Maximum, Average

Units: Count

InboundQueryVolume

For inbound endpoints, the number of DNS queries forwarded from your network to your VPCs through the endpoint specified by `EndpointId`.

Valid Statistics: Sum

Units: Count

OutboundQueryVolume

For outbound endpoints, the number of DNS queries forwarded from your VPCs to your network through the endpoint specified by `EndpointId`.

Valid Statistics: Sum

Units: Count

OutboundQueryAggregateVolume

For outbound endpoints, the total number of DNS queries forwarded from Amazon VPCs to your network, including the following:

- The number of DNS queries forwarded from your VPCs to your network through the endpoint that is specified by `EndpointId`.
- When the current account shares Resolver rules with other accounts, queries from VPCs that are created by other accounts that are forwarded to your network through the endpoint that is specified by `EndpointId`.

Valid Statistics: Sum

Units: Count

ResolverEndpointCapacityStatus

The capacity status of the Resolver endpoint. The metric indicates the current capacity utilization state where: 0 = OK (Normal operating capacity), 1 = Warning (At least one elastic network interface exceeds 50% capacity utilization), and 2 = Critical (At least one elastic network interface exceeds 75% capacity utilization).

The capacity status is determined by multiple factors including query volume, query latency, DNS protocols, DNS packet size, and connection tracking status.

Valid Statistics: Maximum

Units: None

Best practices for VPC Resolver Endpoints Capacity Management

To address capacity issues, we generally recommend increasing the number of elastic network interfaces for your Resolver endpoint. However, there are important considerations for specific endpoint types:

For **inbound endpoints** the traffic load balancing is customer-dependent. Therefore capacity warnings or critical alerts might indicate a "hot spot" where a subset of elastic network interfaces is disproportionately utilized.

- To identify potential load balancing issues, examine the InboundQueryVolume metrics for each elastic network interface individually.

For **outbound endpoints** the traffic is automatically balanced across elastic network interfaces. Capacity issues might be due to problems with the target name server, or because high-latency queries of timeouts overwhelm the Resolver network interfaces.

- In these cases, simply increasing the elastic network interfaces might not be effective, and we recommend fixing the target name server.

Metrics for Route 53 VPC Resolver IP addresses

The `AWS/Route53Resolver` namespace includes the following metrics for each IP address that's associated with a Resolver inbound or outbound endpoint. (When you specify an endpoint, VPC Resolver creates an Amazon VPC elastic network interface.)

InboundQueryVolume

For each IP address for your inbound endpoints, the number of DNS queries forwarded from your network to the specified IP address. Each IP address is identified by the IP address ID. You can get this value using the Route 53 console. On the page for the applicable endpoint, in the IP addresses section, see the **IP address ID** column. You can also get the value programmatically using ListResolverEndpointIpAddresses.

Valid Statistics: Sum

Units: Count

OutboundQueryAggregateVolume

For each IP address for your outbound endpoints, the total number of DNS queries forwarded from Amazon VPCs to your network, including the following:

- The number of DNS queries forwarded from your VPCs to your network using the specified IP address.
- When the current account shares Resolver rules with other accounts, queries from VPCs that are created by other accounts that are forwarded to your network through using the specified IP address.

Each IP address is identified by the IP address ID. You can get this value using the Route 53 console. On the page for the applicable endpoint, in the IP addresses section, see the **IP address ID** column. You can also get the value programmatically using ListResolverEndpointIpAddresses.

Valid Statistics: Sum

Units: Count

CloudWatch Detailed Metrics for Route 53 VPC Resolver

Route 53 VPC Resolver provides RNI Enhanced and Target Name Server Metrics as opt in features for endpoints. These metrics are sent to CloudWatch at 1-minute intervals.

> **Note**
>
> - Detailed metrics are not enabled by default, but can be enabled at the endpoint level. These metrics can be enabled programmatically while creating or updating endpoints using the RniEnhancedMetricsEnabled and TargetNameServerMetricsEnabled flags. For more information, see CreateResolverEndpoint and UpdateResolverEndpoint.
> - Standard CloudWatch pricing and charges are applied for using the Route 53 Resolver endpoint detailed metrics. For more information, see Amazon CloudWatch Pricing.

Topics

- RNI Enhanced Metrics
- Target Name Server Metrics

RNI Enhanced Metrics

Route 53 Resolver publishes RNI enhanced metrics to Amazon CloudWatch for monitoring the performance and health of Resolver endpoints and Resolver IP addresses. The `AWS/Route53Resolver` namespace includes the following RNI enhanced metrics for Route 53 Resolver inbound and outbound endpoints in `EndpointId`, `RniId` dimension:

P90ResponseTime

The 90th percentile response latency of DNS queries received by the Resolver IP (`RniId`) associated with the Resolver endpoint (`EndpointId`)

Valid Statistics: Maximum

Units: Microseconds

ServFailQueries

Number of SERVFAIL responses for DNS queries sent to the Resolver IP (`RniId`) associated with the Resolver endpoint (`EndpointId`)

Valid Statistics: Sum

Units: Count

NxDomainQueries

Number of NXDOMAIN responses for DNS queries sent to the Resolver IP (`RniId`) associated with the Resolver endpoint (`EndpointId`)

Valid Statistics: Sum

Units: Count

RefusedQueries

Number of REFUSED responses for DNS queries sent to the Resolver IP (`RniId`) associated with the Resolver endpoint (`EndpointId`)

Valid Statistics: Sum

Units: Count

FormErrorQueries

Number of FORMERR responses for DNS queries sent to the Resolver IP (`RniId`) associated with the Resolver endpoint (`EndpointId`)

Valid Statistics: Sum

Units: Count

TimeoutQueries

Number of timeouts for DNS queries sent to the Resolver IP (`RniId`) associated with the Resolver endpoint (`EndpointId`)

Valid Statistics: Sum

Units: Count

Target Name Server Metrics

Route 53 Resolver publishes target name server metrics to Amazon CloudWatch for monitoring the performance and availability of target name servers associated with Resolver endpoints. The

`AWS/Route53Resolver` namespace includes the following detailed metrics for Route 53 Resolver outbound endpoints in `EndpointID`, `TargetNameServerIP` dimensions:

P90ResponseTime

The 90th percentile response latency of the Target Name Server IP (`TargetNameServerIP`) for DNS queries sent through the Resolver endpoint (`EndpointID`)

Valid Statistics: Maximum

Units: Microseconds

RequestQueries

Number of DNS queries sent to the Target Name Server IP (`TargetNameServerIP`) through the Resolver endpoint (`EndpointID`).

Valid Statistics: Sum

Units: Count

TimeoutQueries

Number of DNS queries sent through the Resolver endpoint (`EndpointID`) that timed out at the Target Name Server IP (`TargetNameServerIP`).

Valid Statistics: Sum

Units: Count

Note

In some cases, gaps might be observed in VPC Resolver metrics (ResolverEndpointCapacityStatus) and RNI enhanced metrics. These gaps can occur when your network interfaces undergo consecutive scheduled maintenance or updates. After Route 53 returns a network interface to service, it takes at least 1 minute for the service to collect operational data and publish these metrics. These gaps do not indicate that your VPC Resolver endpoint is experiencing an outage. If you're configuring a CloudWatch alarm for these metrics, we recommend the following:

- Set the alarm to "Treat missing data as ignore", or
- Configure an evaluation period of more than five minutes for the alarm threshold.

These settings help reduce false alarms during normal maintenance activities.

Dimensions for Route 53 VPC Resolver metrics

Route 53 VPC Resolver metrics for inbound and outbound endpoints use the `AWS/Route53Resolver` namespace and provide metrics for the following dimensions:

- `EndpointId`: If you specify a value for the `EndpointId` dimension, CloudWatch returns the number of DNS queries for the specified endpoint. If you don't specify `EndpointId`, CloudWatch returns the number of DNS queries for all endpoints that were created by the current AWS account.
- `RniId` dimension is supported for `OutboundQueryAggregateVolume` and `InboundQueryVolume` metrics.
- `EndpointId`, `RniId` dimension is supported for `P90ResponseTime`, `ServFailQueries`, `NxDomainQueries`, `RefusedQueries`, `FormErrorQueries`, and `TimeoutQueries` for the Resolver IP Address associated with the resolver endpoint.
- `EndpointID`, `TargetNameServerIP` dimension is supported for `P90ResponseTime`, `RequestQueries`, and `TimeoutQueries` for the target name server associated with the resolver endpoint.

Monitoring Resolver DNS Firewall rule groups with Amazon CloudWatch

You can use Amazon CloudWatch to monitor the number of DNS queries that are filtered by Resolver DNS Firewall rule groups. Amazon CloudWatch collects and processes raw data into readable, near real-time metrics. These statistics are recorded for a period of two weeks, so that you can access historical information and gain a better perspective on how your resources are performing. By default, metric data for DNS Firewall rule groups is automatically sent to CloudWatch at five-minute intervals.

For more information about DNS Firewall, see Using DNS Firewall to filter outbound DNS traffic. For more information about CloudWatch, see What is Amazon CloudWatch? in the *Amazon CloudWatch User Guide*.

Metrics and dimensions for Resolver DNS Firewall

When you associate a Resolver DNS Firewall rule group with a VPC to filter DNS queries, DNS Firewall starts to send metrics and dimensions once every 5 minutes to CloudWatch about the queries that it filters. For information about the metrics and dimensions for DNS Firewall, see CloudWatch metrics for Resolver DNS Firewall.

You can use the following procedures to view the metrics in the CloudWatch console or view them by using the AWS Command Line Interface (AWS CLI).

To view DNS Firewall metrics using the CloudWatch console

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. On the navigation bar, choose the Region that you want to view.
3. In the navigation pane, choose **Metrics**.
4. On the **All metrics** tab, choose **Route 53 VPC Resolver**.
5. Choose a metric that you're interested in.

To view metrics using the AWS CLI

- At a command prompt, use the following command:

```
aws cloudwatch list-metrics --namespace "AWS/Route53Resolver"
```

Topics

- CloudWatch metrics for Resolver DNS Firewall

CloudWatch metrics for Resolver DNS Firewall

The `AWS/Route53Resolver` namespace includes metrics for Resolver DNS Firewall rule groups.

Topics

- Metrics for Resolver DNS Firewall rule groups
- Metrics for VPCs
- Metrics for firewall rule group and VPC association

- Metrics for a domain list in a firewall rule group

Metrics for Resolver DNS Firewall rule groups

FirewallRuleGroupQueryVolume

The number of DNS Firewall queries that match a firewall rule group (specified by `FirewallRuleGroupId`).

Dimensions: `FirewallRuleGroupId`

Valid statistics: Sum

Units: Count

Metrics for VPCs

VpcFirewallQueryVolume

The number of DNS Firewall queries from a VPC (specified by `VpcId`).

Dimensions: `VpcId`

Valid statistics: Sum

Units: Count

Metrics for firewall rule group and VPC association

FirewallRuleGroupVpcQueryVolume

The number of DNS Firewall queries from a VPC (specified by `VpcId`) that match a firewall rule group (specified by `FirewallRuleGroupId`).

Dimensions: `FirewallRuleGroupId, VpcId`

Valid statistics: Sum

Units: Count

Metrics for a domain list in a firewall rule group

FirewallRuleQueryVolume

The number of DNS firewall queries that match a firewall domain list (specified by `FirewallDomainListId`) within a firewall rule group (specified by `FirewallRuleGroupId`).

Dimensions: `FirewallRuleGroupId, FirewallDomainListId`

Valid statistics: Sum

Units: Count

Topics

- Using Amazon CloudWatch Contributor Insights with DNS Firewall data
- Analyzing DNS Firewall logs with Amazon CloudWatch Logs Insights

Using Amazon CloudWatch Contributor Insights with DNS Firewall data

Amazon CloudWatch Contributor Insights analyzes log data to create time series that display contributor data. For example, you can view the rule groups and VPCs that produce the most blocked or allowed DNS queries. You can use this data to identify which rule groups, domains, or VPCs generate the most filtered traffic. For more information, see Using Contributor Insights to analyze high-cardinality data.

To use Amazon CloudWatch Contributor Insights with DNS Firewall data

1. Contributor Insights requires Route 53 VPC Resolver query logging for the VPC that you want to analyze. If you haven't enabled query logging for your target VPC, complete the following step to enable it.
2. In the Amazon VPC console, open the Route 53 VPC Resolver query logging page and create a query logging configuration. For the destination, choose an existing CloudWatch Logs log group, or create one (you can create a log group directly from the query logging console). Then associate the configuration with your target VPC. For more information, see the section called “ Managing configurations”.
3. In the Amazon VPC DNS Firewall console, open the **Analytics** tab, or open the Amazon CloudWatch Contributor Insights console.

> **Note**
>
> You can access both Contributor Insights and Logs Insights from the **Analytics** tab. You can also select an individual rule group that you're interested in and view its analytics.

4. Choose **Contributor Insights**, then **Create rule**. For more information, see Create a Contributor Insights rule.
5. For **Rule type**, choose **Sample rule**. Then choose **Select sample rule** and choose one of the Route 53 DNS Firewall Logs sample rules.

> **Note**
>
> Include the resource ID in the rule name so that you can identify which resource the rule applies to. For example, use a Amazon VPC ID or an account ID.

6. After you select a sample rule, CloudWatch populates all the required fields on the rule page.
7. To view the resulting metrics, open the CloudWatch console. Choose **Metrics**, then **All metrics** (classic metrics), and look for the Route 53 VPC Resolver namespace, which already includes DNS Firewall metrics.

Analyzing DNS Firewall logs with Amazon CloudWatch Logs Insights

With Amazon CloudWatch Logs Insights, you can interactively search and analyze your DNS Firewall query log data. For more information, see Analyzing log data with CloudWatch Logs Insights.

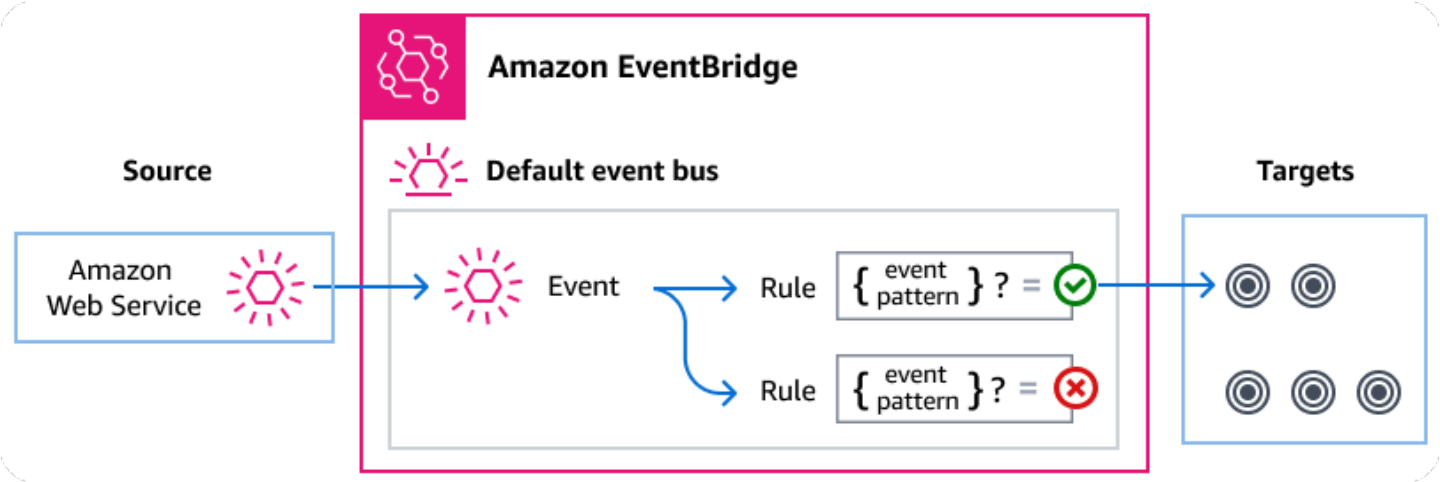
To analyze DNS Firewall logs with Amazon CloudWatch Logs Insights

1. From the **Analytics** tab, choose **Logs Insights**.
2. Select one or more log groups to analyze. These are the log groups that you created for Route 53 VPC Resolver query logging. For more information about selecting fields to index, see Create field indexes.
3. Choose the dimensions to analyze. You can choose from three dimensions: **Resource**, **Metric**, and **Action**.
4. Run the query.

Managing Resolver DNS Firewall events using Amazon EventBridge

Amazon EventBridge is a serverless service that uses events to connect application components together, making it easier for you to build scalable event-driven applications. Event-driven architecture is a style of building loosely-coupled software systems that work together by emitting and responding to events. Events represent a change in a resource or environment.

As with many AWS services, DNS Firewall generates and sends events to the EventBridge default event bus. (The default event bus is automatically provisioned in every AWS account.) An event bus is a router that receives events and delivers them to zero or more destinations, or *targets*. Rules you specify for the event bus evaluate events as they arrive. Each rule checks whether an event matches the rule's *event pattern*. If the event does match, the event bus sends the event to the specified target(s).



Topics

- Resolver DNS Firewall events
- Sending Resolver DNS Firewall events using EventBridge rules
- Amazon EventBridge permissions
- Additional EventBridge resources
- Resolver DNS Firewall events detail reference

Resolver DNS Firewall events

VPC Resolver sends DNS Firewall events to the default EventBridge event bus automatically. You can create rules on the event bus; each rule includes an event pattern and one or more targets. Events that match a rule's event pattern are delivered to the specified targets on a best-effort basis. Events might be delivered out of order.

The following events are generated by DNS Firewall. For more information, see EventBridge in the *Amazon EventBridge User Guide*.

DNS Firewall events

Event detail type	Description
DNS Firewall Block	Any block action performed on a domain.
DNS Firewall Alert	Any alert action performed on a domain.

Sending Resolver DNS Firewall events using EventBridge rules

To have the EventBridge default event bus send DNS Firewall events to a target, you must create a rule that contains an event pattern that matches the data in the desired DNS Firewall events.

Creating a rule consists of the following general steps:

1. Creating an event pattern for the rule that specifies:
   - VPC Resolver is the source of events being evaluated by the rule.
   - (Optional): Any other event data to match against.

   For more information, see ???
2. (Optional): Creating an *input transformer* that customizes the data from the event before EventBridge passes the information to the target of the rule.

   For more information, see Input transformation in the *EventBridge User Guide*.
3. Specifying the target(s) to which you want EventBridge to deliver events that match the event pattern.

   Targets can be other AWS services, software-as-a-service (SaaS) applications, API destinations, or other custom endpoints. For more information, see Targets in the *EventBridge User Guide*.

For comprehensive instructions on creating event bus rules, see Creating rules that react to events in the *EventBridge User Guide*.

Creating event patterns for Resolver DNS Firewall events

When DNS Firewall delivers an event to the default event bus, EventBridge uses the event pattern defined for each rule to determine if the event should be delivered to the rule's target(s). An event pattern matches the data in the desired DNS Firewall events. Each event pattern is a JSON object that contains:

- A `source` attribute that identifies the service sending the event. For DNS Firewall events, the source is `aws.route53resolver`.
- (Optional): A `detail-type` attribute that contains an array of the event types to match.
- (Optional): A `detail` attribute containing any other event data on which to match.

For example, the following event pattern matches against both alert and block events from DNS Firewall:

```
{
  "source": ["aws.route53resolver"],
  "detail-type": ["DNS Firewall Block", "DNS Firewall Alert"]
}
```

While the following event pattern matches against a BLOCK action:

```
{
  "source": ["aws.route53resolver"],
  "detail-type": ["DNS Firewall Block"]
}
```

DNS Firewall sends the same event for the same domain only once within a 6-hour window. For example:

1. Instance i-123 sent a DNS query exampledomain.com at time T1. DNS Firewall sends an alert or block event as this is the first occurrence.
2. Instance i-123 sent a DNSquery exampledomain.com at time T1+30 minutes. DNS Firewall doesn't send an alert or block event as this is a repeat occurrence within the 6-hour window.

3. Instance i-123 sent a DNS query exampledomain.com at time T1+7 hours. DNS Firewall sends an alert or block event as this is occurred outside the 6-hour window.

For more information on writing event patterns, see Event patterns in the *EventBridge User Guide*.

Testing event patterns for DNS Firewall events in EventBridge

You can use the EventBridge Sandbox to quickly define and test an event pattern, without having to complete the larger process of creating or editing a rule. Using the Sandbox, you can define an event pattern and use a sample event to confirm the pattern matches the desired events. EventBridge give you the option of creating a new rule using that event pattern, directly from the sandbox.

For more information, see Testing an event pattern using the EventBridge Sandbox in the *EventBridge User Guide*.

Creating an EventBridge rule and target for DNS Firewall

The following procedure shows you how to create a rule that enables EventBridge to send events for all the DNS Firewall alert and block actions, and add an AWS Lambda function as a target for the rule.

1. Use AWS CLI to create an EventBridge rule:

```
aws events put-rule \
--event-pattern "{\"source\":
[\"aws.route53resolver\"],\"detail-type\":
[\"DNS Firewall Block\", \"DNS Firewall Alert\"]}" \
--name dns-firewall-rule
```

2. Attach a Lambda function as a target for the rule:

```
AWS events put-targets --rule dns-firewall-rule --targets
Id=1,Arn=arn:aws:lambda:us-east-1:111122223333:function:<your_function>
```

3. To add the permissions required to invoke the target, run the following Lambda AWS CLI command:

```
AWS lambda add-permission --function-name <your_function> --statement-
id 1 --action 'lambda:InvokeFunction' --principal events.amazonaws.com
```

Amazon EventBridge permissions

DNS Firewall doesn't require any additional permissions to deliver events to Amazon EventBridge.

The targets you specify might need specific permissions or configuration. For more details on using specific services for targets, see Amazon EventBridge targets in the *Amazon EventBridge User Guide*.

Additional EventBridge resources

Refer to the following topics in the *Amazon EventBridge User Guide* for more information on how to use EventBridge to process and manage events.

- For detailed information on how event buses work, see Amazon EventBridge event bus.
- For information on event structure, see Events.
- For information on constructing event patterns for EventBridge to use when matching events against rules, see Event patterns.
- For information on creating rules to specify which events EventBridge processes, see Rules.
- For information on to specify what services or other destinations EventBridge sends matched events to, see Targets.

Resolver DNS Firewall events detail reference

All events from AWS services have a common set of fields containing metadata about the event. These fields include the source service, the time the event was generated, and the account and Region where the event took place. For definitions of these general fields, see Event structure reference in the *Amazon EventBridge User Guide*.

In addition, each event has a `detail` field that contains data specific to that particular event. The reference below defines the detail fields for the various DNS Firewall events.

When using EventBridge to select and manage DNS Firewall events, it's useful to keep the following in mind:

- The `source` field for all events from DNS Firewall is set to `aws.route53resolver`.
- The `detail-type` field specifies the event type.

  For example, `DNS Firewall Block` or `DNS Firewall Alert`.
- The `detail` field contains the data that is specific to that particular event.

For information on constructing event patterns that enable rules to match DNS Firewall events, see Event patterns in the *Amazon EventBridge User Guide*.

For more information on events and how EventBridge processes them, see Amazon EventBridge events in the *Amazon EventBridge User Guide*.

Topics

- DNS Firewall alert event detail
- DNS Firewall block event detail

DNS Firewall alert event detail

Below are the detail fields for Alert status event detail .

The `source` and `detail-type` fields are included because they contain specific values for Route 53 events.

```
{...,
 "detail-type": "DNS Firewall Alert",
  "source": "aws.route53resolver",
 ...,
 "detail": {
      "account-id": "string",
      "last-observed-at": "string",
      "query-name": "string",
      "query-type": "string",
      "query-class": "string",
      "transport": "string",
      "firewall-rule-action": "string",
      "firewall-rule-group-id": "string",
      "firewall-domain-list-id": "string",
      "firewall-protection": "string",
      "resources": [{
         "resource-type": "string",
         "instance-details": {
             "id": "string",
       }
     },
     {
         "resource-type": "string",
         "resolver-endpoint-details": {
```

```
        "id": "string"
      }
    }
]
```

`detail-type`

Identifies the type of event.

For this event, this value is `DNS Firewall Alert`.

`source`

Identifies the service that generated the event. For DNS Firewall events, this value is `aws.route53resolver`.

`detail`

A JSON object that contains information about the event. The service generating the event determines the content of this field.

For this event, this data includes:

`account-id`

The ID of the AWS account that created the VPC.

`last-observed-at`

The timestamp of when the Alert/Block query was made in the VPC.

`query-name`

The domain name (example.com) or subdomain name (www.example.com) that was specified in the query.

`query-type`

Either the DNS record type that was specified in the request, or ANY. For information about the types that Route 53 supports, see Supported DNS record types.

`query-class`

The class of the query.

`transport`

The protocol used to submit the DNS query.

`firewall-rule-action`

The action specified by the rule that matched the domain name in the query. Either `ALERT` or `BLOCK`.

`firewall-rule-group-id`

The ID of the DNS Firewall rule group that matched the domain name in the query. For more information about the firewall rule groups, see DNS Firewall DNS Firewall rule groups and rules.

`firewall-domain-list-id`

The domain list used by the rule that matched the domain name in the query.

`firewall-protection`

The DNS Firewall Advanced protection: DGA, DICTIONARY_DGA, or DNS_TUNNELING. For more information, see DNS Firewall DNS Firewall Advanced Rules.

`resourcese`

Contains resource types and additional details about them.

`resource-type`

Specifies the resource type, such as resolver endpoint or a VPC instance.

`resource-type``-detail`

Additional details about the resource.

Example DNS Firewall alert event

The following is an example alert event.

```
{
 "version": "1.0",
 "id": "8e5622f9-d81c-4d81-612a-9319e7ee2506",
 "detail-type": "DNS Firewall Alert",
 "source": "aws.route53resolver",
 "account": "123456789012",
 "time": "2023-05-30T21:52:17Z",
 "region": "us-west-2",
```

```
 "resources": [],
 "detail": {
 "account-id": "123456789012",
 "last-observed-at": "2023-05-30T20:15:15.900Z",
 "query-name": "15.3.4.32.in-addr.arpa.",
 "query-type": "A",
 "query-class": "IN",
 "transport": "UDP",
 "firewall-rule-action": "ALERT",
 "firewall-rule-group-id": "rslvr-frg-01234567890abcdef",
 "firewall-domain-list-id": "rslvr-fdl-01234567890abcdef",
 "firewall-protection": "DGA",
 "resources": [{
      "resource-type": "instance",
      "instance-details": {
         "id": "i-05746eb48123455e0",
       }
     },
     {
      "resource-type": "resolver-endpoint",
      "resolver-endpoint-details": {
         "id": "i-05746eb48123455e0"
       }
     }
 ],
"src-addr": "4.5.64.102",
"src-port": "56067",
"vpc-id": "vpc-7example"
 }
}
```

DNS Firewall block event detail

Below are the detail fields for *event name*.

The `source` and `detail-type` fields are included because they contain specific values for Route 53 events.

```
{...,
 "detail-type": "DNS Firewall Block",
  "source": "aws.route53resolver",
 ...,
 "detail": {
```

```
      "account-id": "string",
      "last-observed-at": "string",
      "query-name": "string",
      "query-type": "string",
      "query-class": "string",
      "transport": "string",
      "firewall-rule-action": "string",
      "firewall-rule-group-id": "string",
      "firewall-domain-list-id": "string",
      "firewall-protection": "string",
      "resources": [{
         "resource-type": "string",
         "instance-details": {
             "id": "string",
       }
     },
     {
         "resource-type": "string",
         "resolver-endpoint-details": {
         "id": "string"
       }
     }
 ]
```

`detail-type`

Identifies the type of event.

For this event, this value is `DNS Firewall Alert`.

`source`

Identifies the service that generated the event. For DNS Firewall events, this value is `aws.route53resolver`.

`detail`

A JSON object that contains information about the event. The service generating the event determines the content of this field.

For this event, this data includes:

`account-id`

The ID of the AWS account that created the VPC.

`last-observed-at`

The timestamp of when the Alert/Block query was made in the VPC.

`query-name`

The domain name (example.com) or subdomain name (www.example.com) that was specified in the query.

`query-type`

Either the DNS record type that was specified in the request, or ANY. For information about the types that Route 53 supports, see Supported DNS record types.

`query-class`

The class of the query.

`transport`

The protocol used to submit the DNS query.

`firewall-rule-action`

The action specified by the rule that matched the domain name in the query. Either `ALERT` or `BLOCK`.

`firewall-rule-group-id`

The ID of the DNS Firewall rule group that matched the domain name in the query. For more information about the firewall rule groups, see DNS Firewall DNS Firewall rule groups and rules.

`firewall-domain-list-id`

The domain list used by the rule that matched the domain name in the query.

`firewall-protection`

The DNS Firewall Advanced protection: DGA, DICTIONARY_DGA, or DNS_TUNNELING. For more information, see DNS Firewall DNS Firewall Advanced Rules.

`resourcese`

Contains resource types and additional details about them.

`resource-type`

Specifies the resource type, such as resolver endpoint or a VPC instance.

resource-type-detail

Additional details about the resource.

Example event

The following is an example block event.


```
{
 "version": "1.0",
 "id": "8e5622f9-d81c-4d81-612a-9319e7ee2506",
 "detail-type": "DNS Firewall Block",
 "source": "aws.route53resolver",
 "account": "123456789012",
 "time": "2023-05-30T21:52:17Z",
 "region": "us-west-2",
 "resources": [],
 "detail": {
 "account-id": "123456789012",
 "last-observed-at": "2023-05-30T20:15:15.900Z",
 "query-name": "15.3.4.32.in-addr.arpa.",
 "query-type": "A",
 "query-class": "IN",
 "transport": "UDP",
 "firewall-rule-action": "BLOCK",
 "firewall-rule-group-id": "rslvr-frg-01234567890abcdef",
 "firewall-domain-list-id": "rslvr-fdl-01234567890abcdef",
 "firewall-protection": "DNS_TUNNELING",
 "resources": [{
      "resource-type": "instance",
      "instance-details": {
         "id": "i-05746eb48123455e0"
       }
     },
     {
      "resource-type": "resolver-endpoint",
      "resolver-endpoint-details": {
         "id": "i-05746eb48123455e0",
       }
     }
 ],
"src-addr": "4.5.64.102",
"src-port": "56067",
```

```
"vpc-id": "vpc-7example"
 }
}
```

Logging Amazon Route 53 API calls with AWS CloudTrail

Route 53 is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Route 53. CloudTrail captures all API calls for Route 53 as events, including calls from the Route 53 console and from code calls to the Route 53 APIs. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Route 53. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to Route 53, the IP address that the request was made from, who made the request, when it was made, and additional details.

Topics

- Route 53 information in CloudTrail
- Viewing Route 53 events in event history
- Understanding Route 53 log file entries

Route 53 information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in Route 53, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing events with CloudTrail event history.

For an ongoing record of events in your AWS account, including events for Route 53, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all regions. The trail logs events from all regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see:

- Overview for creating a trail
- CloudTrail supported services and integrations

- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple Regions and Receiving CloudTrail log files from multiple accounts

All Route 53 actions are logged by CloudTrail and are documented in the Amazon Route 53 API Reference. For example, calls to the `CreateHostedZone`, `CreateHealthCheck`, and `RegisterDomain` actions generate entries in the CloudTrail log files.

CloudTrail logs `CreateResourceRecordSetProtections` and `DeleteResourceRecordSetProtections` as management write events, and `ListResourceRecordSetProtections` as a management read event.

CloudTrail logs the Change History actions `ListHostedZoneChangeBatches`, `GetHostedZoneChangeBatch`, and `ListResourceRecordSetChanges` as management read events. For more information about Change History, see Viewing the change history of a hosted zone.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or IAM user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see the CloudTrail userIdentity element.

Viewing Route 53 events in event history

CloudTrail lets you view recent events in **Event history**. To view events for Route 53 API requests, you must choose **US East (N. Virginia)** in the Region selector at the top of the console. For more information, see Viewing events with CloudTrail event history in the *AWS CloudTrail User Guide*.

Understanding Route 53 log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of

the action, request parameters, and so on. CloudTrail log files are not an ordered stack trace of the public API calls, so they do not appear in any specific order.

The `eventName` element identifies the action that occurred. (In CloudTrail logs, the first letter is lowercase for domain registration actions even though it's uppercase in the names of the actions. For example, `UpdateDomainContact` appears as `updateDomainContact` in the logs). CloudTrail supports all Route 53 API actions. The following example shows a CloudTrail log entry that demonstrates the following actions:

- List the hosted zones that are associated with an AWS account
- Create a health check
- Create two records
- Delete a hosted zone
- Update information for a registered domain
- Create a Route 53 VPC Resolver outbound endpoint


```
{
    "Records": [
        {
            "apiVersion": "2013-04-01",
            "awsRegion": "us-east-1",
            "eventID": "1cdbea14-e162-43bb-8853-f9f86d4739ca",
            "eventName": "ListHostedZones",
            "eventSource": "route53.amazonaws.com",
            "eventTime": "2015-01-16T00:41:48Z",
            "eventType": "AwsApiCall",
            "eventVersion": "1.02",
            "recipientAccountId": "444455556666",
            "requestID": "741e0df7-9d18-11e4-b752-f9c6311f3510",
            "requestParameters": null,
            "responseElements": null,
            "sourceIPAddress": "192.0.2.92",
            "userAgent": "Apache-HttpClient/4.3 (java 1.5)",
            "userIdentity": {
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "accountId": "111122223333",
                "arn": "arn:aws:iam::111122223333:user/smithj",
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "type": "IAMUser",
```

```
                "userName": "smithj"
            }
        },
        {
            "apiVersion": "2013-04-01",
            "awsRegion": "us-east-1",
            "eventID": "45ec906a-1325-4f61-b133-3ef1012b0cbc",
            "eventName": "CreateHealthCheck",
            "eventSource": "route53.amazonaws.com",
            "eventTime": "2018-01-16T00:41:57Z",
            "eventType": "AwsApiCall",
            "eventVersion": "1.02",
            "recipientAccountId": "444455556666",
            "requestID": "79915168-9d18-11e4-b752-f9c6311f3510",
            "requestParameters": {
                "callerReference": "2014-05-06 64832",
                "healthCheckConfig": {
                    "iPAddress": "192.0.2.249",
                    "port": 80,
                    "type": "TCP"
                }
            },
            "responseElements": {
                "healthCheck": {
                    "callerReference": "2014-05-06 64847",
                    "healthCheckConfig": {
                        "failureThreshold": 3,
                        "iPAddress": "192.0.2.249",
                        "port": 80,
                        "requestInterval": 30,
                        "type": "TCP"
                    },
                    "healthCheckVersion": 1,
                    "id": "b3c9cbc6-cd18-43bc-93f8-9e557example"
                },
                "location": "https://route53.amazonaws.com/2013-04-01/healthcheck/
b3c9cbc6-cd18-43bc-93f8-9e557example"
            },
            "sourceIPAddress": "192.0.2.92",
            "userAgent": "Apache-HttpClient/4.3 (java 1.5)",
            "userIdentity": {
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "accountId": "111122223333",
                "arn": "arn:aws:iam::111122223333:user/smithj",
```

```
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "type": "IAMUser",
                "userName": "smithj"
            }
        },
        {
            "additionalEventData": {
                "Note": "Do not use to reconstruct hosted zone"
            },
            "apiVersion": "2013-04-01",
            "awsRegion": "us-east-1",
            "eventID": "883b14d9-2f84-4005-8bc5-c7bf0cebc116",
            "eventName": "ChangeResourceRecordSets",
            "eventSource": "route53.amazonaws.com",
            "eventTime": "2018-01-16T00:41:43Z",
            "eventType": "AwsApiCall",
            "eventVersion": "1.02",
            "recipientAccountId": "444455556666",
            "requestID": "7081d4c6-9d18-11e4-b752-f9c6311f3510",
            "requestParameters": {
                "changeBatch": {
                    "changes": [
                        {
                            "action": "CREATE",
                            "resourceRecordSet": {
                                "name": "prod.example.com.",
                                "resourceRecords": [
                                    {
                                        "value": "192.0.1.1"
                                    },
                                    {
                                        "value": "192.0.1.2"
                                    },
                                    {
                                        "value": "192.0.1.3"
                                    },
                                    {
                                        "value": "192.0.1.4"
                                    }
                                ],
                                "tTL": 300,
                                "type": "A"
                            }
                        },
```

```
                    {
                        "action": "CREATE",
                        "resourceRecordSet": {
                            "name": "test.example.com.",
                            "resourceRecords": [
                                {
                                    "value": "192.0.1.1"
                                },
                                {
                                    "value": "192.0.1.2"
                                },
                                {
                                    "value": "192.0.1.3"
                                },
                                {
                                    "value": "192.0.1.4"
                                }
                            ],
                            "tTL": 300,
                            "type": "A"
                        }
                    }
                ],
                "comment": "Adding subdomains"
            },
            "hostedZoneId": "Z1PA6795UKMFR9"
        },
        "responseElements": {
            "changeInfo": {
                "comment": "Adding subdomains",
                "id": "/change/C156SRE0X2ZB10",
                "status": "PENDING",
                "submittedAt": "Jan 16, 2018 12:41:43 AM"
            }
        },
        "sourceIPAddress": "192.0.2.92",
        "userAgent": "Apache-HttpClient/4.3 (java 1.5)",
        "userIdentity": {
            "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
            "accountId": "111122223333",
            "arn": "arn:aws:iam::111122223333:user/smithj",
            "principalId": "A1B2C3D4E5F6G7EXAMPLE",
            "type": "IAMUser",
            "userName": "smithj"
```

```
            }
        },
        {
            "apiVersion": "2013-04-01",
            "awsRegion": "us-east-1",
            "eventID": "0cb87544-ebee-40a9-9812-e9dda1962cb2",
            "eventName": "DeleteHostedZone",
            "eventSource": "route53.amazonaws.com",
            "eventTime": "2018-01-16T00:41:37Z",
            "eventType": "AwsApiCall",
            "eventVersion": "1.02",
            "recipientAccountId": "444455556666",
            "requestID": "6d5d149f-9d18-11e4-b752-f9c6311f3510",
            "requestParameters": {
                "id": "Z1PA6795UKMFR9"
            },
            "responseElements": {
                "changeInfo": {
                    "id": "/change/C1SIJYUYIKVJWP",
                    "status": "PENDING",
                    "submittedAt": "Jan 16, 2018 12:41:36 AM"
                }
            },
            "sourceIPAddress": "192.0.2.92",
            "userAgent": "Apache-HttpClient/4.3 (java 1.5)",
            "userIdentity": {
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "accountId": "111122223333",
                "arn": "arn:aws:iam::111122223333:user/smithj",
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "type": "IAMUser",
                "userName": "smithj"
            }
        },
        {
            "eventVersion": "1.05",
            "userIdentity": {
                "type": "IAMUser",
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "arn": "arn:aws:iam::111122223333:user/smithj",
                "accountId": "111122223333",
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "userName": "smithj",
                "sessionContext": {
```

```
                    "attributes": {
                        "mfaAuthenticated": "false",
                        "creationDate": "2018-11-01T19:43:59Z"
                    }
                },
                "invokedBy": "test"
            },
            "eventTime": "2018-11-01T19:49:36Z",
            "eventSource": "route53domains.amazonaws.com",
            "eventName": "updateDomainContact",
            "awsRegion": "us-west-2",
            "sourceIPAddress": "192.0.2.92",
            "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.12; rv:52.0)
 Gecko/20100101 Firefox/52.0",
            "requestParameters": {
                "domainName": {
                    "name": "example.com"
                }
            },
            "responseElements": {
                "requestId": "034e222b-a3d5-4bec-8ff9-35877ff02187"
            },
            "additionalEventData": "Personally-identifying contact information is not
 logged in the request",
            "requestID": "015b7313-bf3d-11e7-af12-cf75409087f6",
            "eventID": "f34f3338-aaf4-446f-bf0e-f72323bac94d",
            "eventType": "AwsApiCall",
            "recipientAccountId": "444455556666"
        },
        {
            "eventVersion": "1.05",
            "userIdentity": {
                "type": "IAMUser",
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "arn": "arn:aws:iam::111122223333:user/smithj",
                "accountId": "111122223333",
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "sessionContext": {
                    "attributes": {
                        "mfaAuthenticated": "false",
                        "creationDate": "2018-11-01T14:33:09Z"
                    },
                    "sessionIssuer": {
                        "type": "Role",
```

```
                        "principalId": "AROAIUZEZLWWZOEXAMPLE",
                        "arn": "arn:aws:iam::123456789012:role/Admin",
                        "accountId": "123456789012",
                        "userName": "Admin"
                    }
                }
            },
            "eventTime": "2018-11-01T14:37:19Z",
            "eventSource": "route53resolver.amazonaws.com",
            "eventName": "CreateResolverEndpoint",
            "awsRegion": "us-west-2",
            "sourceIPAddress": "192.0.2.176",
            "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.12; rv:52.0)
 Gecko/20100101 Firefox/52.0",
            "requestParameters": {
                "creatorRequestId": "123456789012",
                "name": "OutboundEndpointDemo",
                "securityGroupIds": [
                    "sg-05618b249example"
                ],
                "direction": "OUTBOUND",
                "ipAddresses": [
                    {
                        "subnetId": "subnet-01cb0c4676example"
                    },
                    {
                        "subnetId": "subnet-0534819b32example"
                    }
                ],
                "tags": []
            },
            "responseElements": {
                "resolverEndpoint": {
                    "id": "rslvr-out-1f4031f1f5example",
                    "creatorRequestId": "123456789012",
                    "arn": "arn:aws:route53resolver:us-west-2:123456789012:resolver-
endpoint/rslvr-out-1f4031f1f5example",
                    "name": "OutboundEndpointDemo",
                    "securityGroupIds": [
                        "sg-05618b249example"
                    ],
                    "direction": "OUTBOUND",
                    "ipAddressCount": 2,
                    "hostVPCId": "vpc-0de29124example",
```

```
                    "status": "CREATING",
                    "statusMessage": "[Trace id: 1-5bd1d51e-f2f3032eb75649f71example]
 Creating the Resolver Endpoint",
                    "creationTime": "2018-11-01T14:37:19.045Z",
                    "modificationTime": "2018-11-01T14:37:19.045Z"
                }
            },
            "requestID": "3f066d98-773f-4628-9cba-4ba6eexample",
            "eventID": "cb05b4f9-9411-4507-813b-33cb0example",
            "eventType": "AwsApiCall",
            "recipientAccountId": "123456789012"
        }
    ]
}
```

Troubleshooting Amazon Route 53

This page covers the following troubleshooting topics for Amazon Route 53:

1. **Domain unavailability:**
   - Understand common reasons why your domain might be unavailable on the internet, such as not confirming the registrant email, DNS service transfer issues, incorrect name server settings, or deleted hosted zones.
2. **Domain suspension:**
   - Learn about the causes of domain suspension (ClientHold status) and how to get your domain unsuspended, including expired domains, unverified registrant email changes, and payment processing issues.
3. **Failed domain operation:**
   - Resolve domain operation failures including registration, transfer, renewal, and contact update issues caused by invalid contact information.
4. **Failed domain transfer:**
   - Discover common reasons for a failed domain transfer to Route 53, such as not authorizing the transfer, invalid authorization codes, or issues with internationalized domain names.
5. **DNS settings not taking effect:**
   - Troubleshoot situations where your DNS settings changes haven't taken effect yet, including DNS resolver caching, incorrect name server updates, and multiple hosted zones with the same name.
6. **"Server Not Found" error:**
   - Find solutions for "Server Not Found" errors in your browser, such as missing records, incorrect record values, or unavailable resources.
7. **Routing traffic to S3 buckets:**
   - Resolve issues when trying to route traffic to an Amazon S3 bucket configured for website hosting.
8. **Billing issues:**
   - Understand common billing scenarios, including being billed twice for the same hosted zone, multiple invoices for domains, and domain registration concerns when your AWS account is closed or permanently closed.

Topics

- My domain is unavailable on the internet
- My domain is suspended (status is ClientHold)
- My domain operation failed
- Transferring my domain to Amazon Route 53 failed
- I changed DNS settings, but they haven't taken effect
- My browser displays a "Server not found" error
- I can't route traffic to an Amazon S3 bucket that's configured for website hosting
- I was billed twice for the same hosted zone
- I was charged multiple invoices for my domain
- My AWS account is closed or permanently closed, and my domain is registered with Route 53

My domain is unavailable on the internet

Here are the most common reasons that your domain is not available on the internet.

Topics

- You registered a new domain, but you didn't choose the link in the confirmation email
- You transferred domain registration to Amazon Route 53, but you didn't transfer DNS service
- You transferred domain registration and specified the wrong name servers in the domain settings
- You transferred DNS service first, but you didn't wait long enough before transferring domain registration
- You deleted the hosted zone that Route 53 is using to route internet traffic for the domain
- Your domain has been suspended

You registered a new domain, but you didn't choose the link in the confirmation email

When you register a new domain, ICANN requires that we get confirmation that the email address for the registrant contact is valid. To get confirmation, we send an email that contains a link. You have between 3 and 15 days to choose the link, depending on the top-level domain. After that time, the link stops working.

If you don't choose the link in the email in the allotted amount of time, ICANN requires that we suspend the domain, which makes it unavailable on the internet. For information about how to resend the confirmation email to the registrant contact, see Resending authorization and confirmation emails.

You transferred domain registration to Amazon Route 53, but you didn't transfer DNS service

If your previous registrar offered free DNS service with domain registration, the registrar might have stopped providing DNS service when you transferred domain registration to Route 53. Perform the following procedure to determine whether this is the problem and, if so, to resolve it.

To restore DNS service if your previous registrar canceled it after you transferred domain registration to Route 53

1. Contact your previous registrar and confirm that they canceled DNS service for your domain. If so, here are the three quickest ways to restore DNS service for the domain, in order of desirability:

   - If the previous registrar provides paid DNS service, ask them to restore DNS service using the old DNS records and name servers for your domain.
   - If the previous registrar doesn't provide paid DNS service without domain registration, ask whether you can transfer domain registration back to them and have them restore DNS service using the old DNS records and name servers for your domain.
   - If you can transfer domain registration back to the previous registrar but they don't have your DNS records any longer, ask whether you can transfer domain registration back to them and get the same set of name servers that were formerly assigned to the domain. If this is possible, you'll have to recreate your old DNS records yourself. However, as soon as you do that, your domain will become available again.

   If your previous registrar can't help with any of these options, continue with step 2.

   > **Important**
   >
   > If you can't restore DNS service using the name servers that you specified when you transferred your domain to Route 53, it can take up to two days after you complete the remaining steps in this procedure for your domain to become available again on the

internet. DNS resolvers typically cache the names of the name servers for a domain for 24 to 48 hours, and it will take that long before all DNS resolvers get the names of the new name servers.

2. Choose a new DNS service, for example, Route 53.
3. Using the method provided by the new DNS service, create a hosted zone and records:

   a. Create a hosted zone that has the same name as your domain, such as example.com.

   b. Use the zone file that you got from the previous registrar to create records.

   If you chose Route 53 as your new DNS service, you can create records by importing the zone file. For more information, see Creating records by importing a zone file.
4. Get the name servers for the new hosted zone. If you chose Route 53 as the DNS service, see Getting the name servers for a public hosted zone.
5. Change the name servers for your domain to the name servers that you got in step 4. For more information, see Adding or changing name servers and glue records for a domain.

You transferred domain registration and specified the wrong name servers in the domain settings

When you transfer domain registration to Amazon Route 53, one of the settings that you specify for the domain is the set of name servers that will respond to DNS queries for the domain. These name servers come from the hosted zone that has the same name as the domain. The hosted zone contains information about how you want to route traffic for the domain, such as the IP address of a web server for www.example.com.

You might have accidentally specified the name servers for the wrong hosted zone, which is especially easy if you have more than one hosted zone that has the same name as the domain. To confirm that the domain is using the name servers for the correct hosted zone and, if necessary, update the name servers for the domain, perform the following procedures.

Important

If you specified the wrong name server records when you transferred the domain to Route 53, it can take up to two days after you correct the name servers for the domain

before DNS service is fully restored. This is because DNS resolvers across the internet typically request the name servers only once every two days and cache the answer.

To get the name servers for your hosted zone

1. If you're using another DNS service for the domain, use the method provided by the DNS service to get the name servers for the hosted zone. Then skip to the next procedure.

   If you're using Route 53 as the DNS service for the domain, sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted Zones**.
3. On the **Hosted Zones** page, choose the radio button (not the name) for the hosted zone.

   Important

   If you have more than one hosted zone with the same name, make sure you're getting the name servers for the correct hosted zone.
4. In the right pane, make note of the four servers listed for **Name Servers**.

To confirm that the domain is using the correct name servers

1. If you're using another DNS service for the domain, sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/

   If you're using Route 53, skip to the next step.
2. In the navigation pane, choose **Registered Domains**.
3. Choose the name of the domain for which you want to edit settings.
4. Choose **Add or Edit Name Servers**.
5. Compare the list of name servers that you got in the previous procedure with the name servers that are listed in the **Edit Name Servers for** *domain name* dialog box.
6. If the name servers listed here don't match the name servers that you got in the previous procedure, change the name servers here, and then choose **Update**.

You transferred DNS service first, but you didn't wait long enough before transferring domain registration

When you transferred DNS service to Amazon Route 53 or another DNS service, you updated the configuration for your domain with the domain registrar to use the name servers for the new DNS service.

DNS resolvers, which respond to requests for your domain, commonly cache the names of name servers for 24 to 48 hours. If you change the DNS service for a domain and replace the name servers from one DNS service with the name servers for another DNS service, it can take up to 48 hours before DNS resolvers start using the new name servers and, therefore, the new DNS service.

Here's how transferring your DNS service and then transferring your domain too soon after can cause your domain to become unavailable on the internet:

1. You transferred DNS service for your domain.
2. You transferred your domain to Route 53 before DNS resolvers started to use the name servers for your new DNS service.
3. Your previous registrar canceled DNS service for your domain as soon as the domain was transferred to Route 53.
4. DNS resolvers are still routing queries to your old DNS service, but there are no longer any records that tell how to route your traffic.

When caching expires for the name servers for the old DNS service, DNS will start to use your new DNS service. Unfortunately, there is no way to accelerate this process.

You deleted the hosted zone that Route 53 is using to route internet traffic for the domain

If Route 53 is the DNS service for your domain and if you delete the hosted zone that is used to route internet traffic for the domain, the domain will become unavailable on the internet. This is true regardless of whether the domain is registered with Route 53.

> **Important**
>
> Restoring internet service for the domain can take up to 48 hours.

To restore internet service if you delete a hosted zone that Route 53 is using to route internet traffic for a domain

1. Create another hosted zone that has the same name as the domain. For more information, see Creating a public hosted zone.
2. Recreate the records that were in the hosted zone that you deleted. For more information, see Working with records.
3. Get the names of the name servers that Route 53 assigned to the new hosted zone. For more information, see Getting the name servers for a public hosted zone.
4. Update the domain registration to use the name servers that you got in step 3:
   - If the domain is registered with Route 53, see Adding or changing name servers and glue records for a domain.
   - If the domain is registered with another domain registrar, use the method provided by the registrar to update the domain registration to use the new name servers.
5. Wait for the TTL for the name servers to pass for recursive resolvers that have cached the names of the name servers for the deleted hosted zone. After the TTL has passed, when a browser or application submits a DNS query for the domain or one of its subdomains, a recursive resolver forwards the query to the Route 53 name servers for the new hosted zone. For more information, see How Amazon Route 53 routes traffic for your domain.

   The TTL for name servers can be as long as 48 hours, depending on the TLD of the domain.

Your domain has been suspended

Your domain might be unavailable on the internet because we had to suspend it. For more information, see My domain is suspended (status is ClientHold).

My domain is suspended (status is ClientHold)

If Amazon Route 53 suspends your domain, the domain becomes unavailable on the internet. You can use either of the following methods to determine whether a domain has been suspended:

- On the **Registered domains** page of the Route 53 console, find the domain name in the **Alerts** table at the bottom of the page. If the value of the **Status** column is **clientHold**, the domain has been suspended.

- Send a WHOIS query for the domain. If the value of **Domain Status** is **clientHold**, the domain has been suspended. The WHOIS command is available in many operating systems, and it's also available as a web application on many websites.

In addition, when we suspend a domain, we generally send an email to the email address for the registrant contact for the domain. However, if the domain was suspended based on a court order, the court might not let us notify the registrant contact.

To make a domain available on the internet again, you must get it unsuspended. Here are the reasons that a domain can be suspended and how you get it unsuspended.

> **Note**
>
> If you need help getting your domain unsuspended, you can contact AWS Support free of charge. For more information, see Contacting AWS Support about domain registration issues.

Topics

- You registered a new domain, but you didn't choose the link in the confirmation email
- You disabled automatic renewal for the domain, and the domain expired
- You changed the email address for the registrant contact, but you didn't verify that the new email address is valid
- We suspended the domain for a violation of the AWS Acceptable Use Policy
- We suspended the domain because of a court order

You registered a new domain, but you didn't choose the link in the confirmation email

When you register a domain with AWS for the first time, ICANN requires that we get confirmation that the email address for the registrant contact is valid. To get confirmation, we send an email that contains a link. You have between 3 and 15 days to choose the link, depending on the top-level domain. After that time, the link stops working.

> **Note**
>
> If you have already registered one or more domains with Amazon Route 53 and used the same email address for the registrant contact, we don't send a confirmation email.

If you don't choose the link in the email in the allotted amount of time, ICANN requires that we suspend the domain, which makes it unavailable on the internet. For information about how to resend the confirmation email to the registrant contact, see Resending authorization and confirmation emails. When you confirm that the email address is valid, we automatically unsuspend the domain.

You disabled automatic renewal for the domain, and the domain expired

When automatic renewal is enabled for a domain (the default value for a new or transferred domain), we automatically renew registration for the domain shortly before the expiration date. If you disable automatic renewal, we send reminder emails that the domain registration is about to expire to the email address for the registrant contact. We start to send these emails 45 days before the domain expires.

If you disable automatic renewal for the domain and you don't manually extend the registration period for the domain, we generally suspend the domain on the expiration date. Note that the registries for some domains delete the domain even before the expiration date.

For information about how to renew an expired domain, see Renewing registration for a domain.

You changed the email address for the registrant contact, but you didn't verify that the new email address is valid

If you change the email address for the registrant contact to an address that you haven't previously verified, ICANN requires that we get confirmation that the email address for the registrant contact is valid. To get confirmation, we send an email that contains a link. You have between 3 and 15 days to choose the link, depending on the top-level domain. After that time, the link stops working.

If you don't choose the link in the email in the amount of time allowed by the TLD registry, ICANN requires that we suspend the domain, which makes it unavailable on the internet. For information

about how to resend the confirmation email to the registrant contact, see Resending authorization and confirmation emails. When you confirm that the email address is valid, we automatically unsuspend the domain.

We suspended the domain for a violation of the AWS Acceptable Use Policy

If we suspend a domain for a violation of the AWS Acceptable Use Policy, we send an email notification to the registrant contact for the domain. (We don't send a notification email if the AWS account is already suspended for fraud.)

To contest a suspension, send an email to trustandsafety@support.aws.com.

We suspended the domain because of a court order

If a domain is suspended as a result of a court order, we can't unsuspend the domain until the court order has been lifted. To contest the validity of a court order, send an email to trustandsafety@support.aws.com and attach the applicable documents.

My domain operation failed

Here are some troubleshooting tips if your domain operation fails. Operations can be registration, transfer, renewal, or contact update.

Domain operation failed due to invalid contact information

If a domain operation fails with an error about invalid contact information, the contact details that you provided don't meet the registry's validation requirements.

To resolve this issue, verify that all contact details are correctly formatted:

- **Name** – Make sure the name contains only valid characters and is properly formatted.
- **Address** – Confirm that the address format matches the postal standards for the country.
- **Phone number** – Use the correct international format for the country.
- **Identification fields** – For domains that require identification numbers, make sure the format is correct for the specified country.

After you correct the contact information, try the domain operation again. For specific requirements for different top-level domains, see Domains that you can register with Amazon Route 53.

Transferring my domain to Amazon Route 53 failed

If your domain transfer has already failed or been rejected, use this section to diagnose and resolve the failure.

> **Note**
>
> For preventing issues during active transfers, see Preventing common Route 53 transfer issues.

Topics

- You didn't choose the link in the authorization email
- The authorization code that you got from the current registrar is not valid
- "Parameters in request are not valid" error when trying to transfer a .es domain to Amazon Route 53
- Is the internationalized domain name you're transferring to Amazon Route 53 listed in punycode?
- Common transfer error messages

You didn't choose the link in the authorization email

For some top-level domains, the registry requires the registrant contact to authorize the transfer. In those cases, we send an email that contains a link. You have between 5 and 15 days to choose the link, depending on the top-level domain. After that time, the link stops working.

If you don't choose the link in the email in the allotted amount of time, we cancel the transfer. For information about how to resend the authorization email to the registrant contact, see Resending authorization and confirmation emails.

The authorization code that you got from the current registrar is not valid

If you request the transfer of a domain to Amazon Route 53 and you don't receive the authorization email, check the status page in the Route 53 console. If the status page shows that the transfer authorization code that you got from your registrar is not valid, perform the following steps:

1. Contact the current registrar for the domain and request a new authorization code. Confirm the following:
   - How long the new authorization code will remain active. You must request a domain transfer before the code expires.
   - The new authorization code is different from the code that isn't valid. If not, ask the current registrar to refresh the authorization code.
2. Submit another request to transfer the domain. For more information, see Step 5: Request the transfer in the topic Transferring registration for a domain to Amazon Route 53.

"Parameters in request are not valid" error when trying to transfer a .es domain to Amazon Route 53

If Amazon Route 53 returns a "Parameters in request are not valid" error when you try to transfer a .es domain to Route 53, verify that all required identification fields for the registrant contact are correctly formatted. For .es domains, the registry requires specific identification details depending on whether the registrant is a person or a company. Also verify that the contact type of the administrative, technical, and billing contacts is **Person**; only the registrant contact can be a company. For more information about required fields, see Domains that you can register with Amazon Route 53.

Is the internationalized domain name you're transferring to Amazon Route 53 listed in punycode?

When you register a new domain name or create hosted zones and records, you can specify letters other than a-z (for example, the ç in French), characters in other alphabets (for example, Cyrillic or Arabic), and characters in Chinese, Japanese, or Korean. Amazon Route 53 stores these

internationalized domain names (IDNs) in Punycode, which represents Unicode characters as ASCII strings.

If you get an error while transferring an IDNs to Route 53, use punycode to represent it and try again. For more information, see Formatting internationalized domain names.

Common transfer error messages

When a domain transfer fails, you'll typically receive an error message either in the Route 53 console or by email. Here are the most common error messages and how to resolve them:

"Transfer was rejected by the registry"

This error occurs when the domain registry blocks the transfer for one of several reasons. The most common causes are:

- The domain doesn't meet the 60-day ICANN rule. ICANN requires that transfers be blocked within 60 days of initial registration or a registrant contact change.
- The domain has transfer lock enabled at your current registrar.
- The domain is in a status that prevents transfers, such as `clientTransferProhibited`, `serverTransferProhibited`, `pendingDelete`, `pendingTransfer`, or `redemptionPeriod`.

To resolve this issue, check the following with your current registrar:

- Verify that the domain is more than 60 days old from the date of registration. If you recently changed the registrant contact information, you might need to wait 60 days from that change. Some registrars allow you to opt out of this restriction, so ask your current registrar if this option is available.
- Unlock the domain by disabling any transfer locks or domain locks.
- Check that your domain status allows transfers. You can verify the domain status using a WHOIS lookup or by contacting your current registrar.

"Invalid authorization code" or "Auth code is incorrect"

This error means that the authorization code (also called an EPP code or transfer code) that you entered is not valid. This can happen if the code has expired, was entered incorrectly, or wasn't obtained from your current registrar.

To resolve this issue:

- Contact your current registrar to request a new authorization code. Make sure to ask how long the code will remain active so you can complete the transfer before it expires.
- When you receive the new code, verify that it's different from the previous code. If it's the same, ask your registrar to refresh or regenerate the authorization code.
- Double-check that you're entering the code exactly as provided, including any uppercase or lowercase letters, numbers, and special characters.

"Transfer authorization email not received" or "Authorization timeout"

This error occurs when you don't choose the authorization link in the transfer confirmation email within the required timeframe (typically 5 days). For some top-level domains, the registry requires this authorization step.

If you didn't receive the authorization email or missed the deadline, check the following:

- Check your spam or junk mail folder. Transfer authorization emails are sometimes filtered by email providers.
- Verify that your mailbox isn't full and doesn't have storage limitations that might prevent email delivery.
- Check if your email provider has policies that filter incoming mail from unknown senders.
- Confirm that you can access the registrant email account and that the email address is still active.

If necessary, update your contact information with your current registrar before requesting another transfer. For information about resending the authorization email, see Resending authorization and confirmation emails.

"Domain is locked" or "Transfer prohibited"

This error indicates that your domain has a lock enabled that prevents transfers. Most registrars enable domain locks by default as a security measure to prevent unauthorized transfers.

To resolve this issue, log into your account with your current registrar and disable the domain lock or transfer lock settings. The exact steps vary by registrar, but you'll typically find these settings in

your domain management or security settings. If you need help finding these settings, contact your current registrar for assistance.

"Contact information verification required"

This error occurs when you've recently changed your domain's contact information. ICANN rules require a 60-day waiting period after contact changes before a domain can be transferred to prevent fraudulent transfers.

To resolve this issue, you have two options:

- Wait 60 days from the date of the contact change before attempting the transfer again.
- Contact your current registrar to verify your current contact information. Some registrars might allow you to opt out of the 60-day restriction if you can verify your identity.

> **Note**
>
> If you receive an error message that isn't listed here, contact AWS Support and include the exact error text. Our support team can help you diagnose and resolve transfer issues specific to your situation.

I changed DNS settings, but they haven't taken effect

If you changed DNS settings, here are some common reasons that the changes haven't taken effect yet.

Topics

- You transferred DNS service to Amazon Route 53 in the last 48 hours, so DNS is still using your previous DNS service
- You recently transferred DNS service to Amazon Route 53, but you didn't update the name servers with the domain registrar
- DNS resolvers still are using the old settings for the record
- You have more than one hosted zone with the same name, and you updated the one that isn't associated with the domain

You transferred DNS service to Amazon Route 53 in the last 48 hours, so DNS is still using your previous DNS service

When you transferred DNS service to Amazon Route 53, you used the method provided by the registrar for your domain to replace the name servers for the previous DNS service with the four name servers for Route 53.

> **Note**
>
> If you aren't sure you did this part, see You recently transferred DNS service to Amazon Route 53, but you didn't update the name servers with the domain registrar.

Domain registrars typically use a TTL (time to live) of 24 to 48 hours for name servers. This means that when a DNS resolver gets the name servers for your domain, it uses that information for up to 48 hours before it submits another request for the current name servers for the domain. If you transferred DNS service to Route 53 in the last 48 hours and then changed DNS settings, some DNS resolvers are still using your old DNS service to route traffic for the domain.

You recently transferred DNS service to Amazon Route 53, but you didn't update the name servers with the domain registrar

The registrar for your domain has a variety of information about the domain, including the name servers for the DNS service for the domain. Typically, the domain registrar is also your DNS service, so the name servers that are associated with your domain belong to the registrar. These name servers tell DNS where to get information about how you want traffic for your domain to be routed, for example, to the IP address of a web server for your domain.

When you transfer DNS service to Amazon Route 53, you need to use the method that is provided by your domain registrar to change the name servers that are associated with your domain. You're usually replacing the name servers that are provided by the registrar with the four Route 53 name servers that are associated with the hosted zone that you created for the domain.

If you created a new hosted zone and records for your domain and specified different settings than you used for the previous DNS service, and if DNS is still routing traffic to the old resources, it's possible that you didn't update the name servers with the domain registrar. To determine whether the registrar is using the name servers for your Route 53 hosted zone and, if necessary, to update the name servers for the domain, perform the following procedure:

To get the name servers for your hosted zone and update the name server setting with the domain registrar

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. In the navigation pane, choose **Hosted Zones**.
3. On the **Hosted Zones** page, choose the name of the hosted zone (not the radio button) for the hosted zone.

   > **Important**
   >
   > If you have more than one hosted zone with the same name, make sure you're getting the name servers for the correct hosted zone.

4. In the **Record name** list, make note of the four servers listed for **Name Servers**.
5. Using the method provided by the registrar for the domain, display the list of name servers for the domain.
6. If the name servers for the domain match the name servers that you got in step 4, then the domain configuration is correct.

   If the name servers for the domain don't match the name servers that you got in step 4, update the domain to use the Route 53 name servers.
7.

> **Important**
>
> When you change the name servers for the domain to the name servers from your Route 53 hosted zone, it can take up to two days for the change to take effect and for Route 53 to become your DNS service. This is because DNS resolvers across the internet typically request the name servers only once every two days and cache the answer.

DNS resolvers still are using the old settings for the record

If you changed the settings in a record but your traffic is still being routed to the old resource, such as a web server for your website, one possible cause is that DNS still has the previous settings cached. Each record has a TTL (time to live) value that specifies how long, in seconds, that you want

DNS resolvers to cache the information in the record, such as the IP address for a web server. Until the amount of time that is specified by the TTL passes, DNS resolvers will continue to return the old value in response to DNS queries. If you want to know what the TTL is for a record, perform the following procedure.

> **Note**
>
> For alias records, the TTL is determined by the AWS resource that the record routes traffic to. For more information, see Choosing between alias and non-alias records.

To view the TTL for a record

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
2. On the **Hosted Zones** page, choose the name of the hosted zone that includes the record.
3. In the list of records, find the record that you want the TTL value for, and check the value of the **TTL** column.

   > **Note**
   >
   > Changing the TTL now won't make your change take effect faster. DNS resolvers already have the value cached, and they won't get the new setting until the amount of time that was specified by the old setting passes.

You have more than one hosted zone with the same name, and you updated the one that isn't associated with the domain

You can create more than one hosted zone that has the same name, either using the same account or using multiple accounts. To specify the hosted zone that Route 53 uses to route internet traffic for your domain, you get the four Route 53 name servers for that hosted zone, and you update the domain registration to use those name servers.

If you add, change, or delete records in one hosted zone but your domain registration is using the name servers for another hosted zone, Route 53 responses to DNS queries won't reflect your changes. To determine whether your domain registration is using the name servers for the hosted zone that you updated records in, perform the following tasks:

1. Determine which name servers are associated with your domain registration. See Adding or changing name servers or glue records.
2. Compare the name servers that you got in step 1 with the name servers that Route 53 assigned to the hosted zone that you updated records in. See Getting the name servers for a public hosted zone.

If the name servers for the domain registration don't match the name servers for the hosted zone that you updated records in, you have two options:

Change records in the hosted zone that's currently associated with the domain (recommended)

Make note of the changes that you made in the hosted zone that is not currently associated with your domain registration. Then go to the hosted zone that is associated with the domain registration, and make the same changes. This is the preferred method because the changes take effect almost immediately. For more information, see Editing records.

Update your domain registration to use different name servers

Change your domain registration to use the name servers in the hosted zone that you updated.

> **Important**
>
> If you change the name servers that are associated with your domain registration, your domain will be unavailable on the internet for up to 2 days. This is because DNS resolvers typically cache the names of name servers for 2 days. For an overview of how DNS works, including information about resolver caching, see How Amazon Route 53 routes traffic for your domain.

By changing the name servers that are associated with your domain registration, you're essentially changing the DNS service for the domain. You have two options, depending on whether the domain is currently in use:

- **If the domain is in use**, see Making Route 53 the DNS service for a domain that's in use.
- **If the domain is currently inactive**, perform the following tasks:
  1. Get the name servers for the hosted zone that you want to use to route traffic to your domain. See Getting the name servers for a public hosted zone.
  2. In the hosted zone that you got name servers for in step 1, confirm that the NS record is using the same four name servers. If not, update the NS record. See Editing records.

3. Update the domain registration to use the name servers that you got in step 1. See Adding or changing name servers or glue records.

My browser displays a "Server not found" error

If your browser displays a "Server not found" error when you try to browse to a domain (example.com) or a subdomain (www.example.com), here are some common explanations.

Topics

- You didn't create a record for the domain or subdomain name
- You created a record but specified the wrong value
- The resource that you're routing traffic to is unavailable

You didn't create a record for the domain or subdomain name

If you don't create a record for the domain or subdomain, then DNS doesn't know where to route traffic when someone enters that name in a browser. For more information, see Working with records.

You created a record but specified the wrong value

When you create a record, it's easy to specify the wrong value, such as the IP address for a web server or the domain name that CloudFront assigned to your web distribution. If the record exists but you're still getting a "Server not found" error, we recommend that you confirm that the value is correct.

The resource that you're routing traffic to is unavailable

If a record specifies a resource such as a web server that's unavailable, a browser will return a "Server not found" error. We recommend that you check the status of the resource that you're routing traffic to.

I can't route traffic to an Amazon S3 bucket that's configured for website hosting

When you configure an Amazon S3 bucket for website hosting, you must give the bucket the same name as the record that you want to use to route traffic to the bucket. For example, if you want to route traffic for example.com to an S3 bucket that is configured for website hosting, the name of the bucket must be example.com.

If you want to route traffic to an S3 bucket that is configured for website hosting but the name of the bucket doesn't appear in the **Alias Target** list in the Amazon Route 53 console, or if you're trying to create an alias record programmatically and you're getting an InvalidInput error from the Route 53 API, one of the AWS SDKs, the AWS CLI, or AWS Tools for Windows PowerShell, check the following:

- The name of the bucket exactly matches the name of the record, such as example.com or www.example.com.
- The S3 bucket is correctly configured for website hosting. For more information, see Hosting a static website on Amazon S3 in the *Amazon Simple Storage Service User Guide*.

I was billed twice for the same hosted zone

We don't bill you if you delete a hosted zone within 12 hours after you create it. After 12 hours, we immediately charge the standard monthly fee for a hosted zone. The monthly charge for a hosted zone is not prorated for partial months. (The same charge applies for the hosted zone that we automatically create when you register a domain.)

If you create a hosted zone on the last day of the month (for example, January 31), the charge for January might appear on the February invoice, along with the charge for February. Note that Amazon Route 53 uses Coordinated Universal Time (UTC) as the time zone to determine when a hosted zone was created.

I was charged multiple invoices for my domain

When you sign up for a subscription, pay a registration fee, a transfer fee, or a renewal fee with an upfront cost, a unique invoice is generated. This invoice remains on the Billing console, even if the payment transaction is unsuccessful. The related billing line item is shown as **[x] Quantity** under the **Registrar-Global** sub-section of the **Bill details by service** tab on the Billing console.

To view the waived invoices, complete the following steps:

To view the waived invoices on the Billing console

1. Sign in to the AWS Management Console and open the AWS Billing and Cost Management console at https://console.aws.amazon.com/costmanagement/.
2. In the navigation pane, choose **Bills**.
3. Choose **Invoices** to see details for any waived invoices.

To view the successful payments and refunds on the Billing console, complete the following steps:

To confirm the payments or refunds that were successfully processed

1. In the navigation pane, choose **Payments**.
2. Choose the **Transactions** tab to view the **Transactions** table for all the completed transactions with AWS.

My AWS account is closed or permanently closed, and my domain is registered with Route 53

If you close your AWS account, or if your account is closed or permanently closed, your domains will go through a deletion process:

1. We will notify you that your account is closed and your domain will be suspended in the next 5 days on a daily basis.
2. After your domain is suspended, the following will take place:
   - If your registrar is Amazon Registrar, we will notify you that we will delete your domain in 30 days. For more information, see Finding your registrar and other information about your domain.
   - If your registrar is Gandi, we will notify you that we will release your domain to Gandi when your account becomes permanently closed.
3. After waiting 30 days, we will delete all the domains registered with Amazon Registrar in the account and send you an update.
4. When your account becomes permanently closed, we will release all the domains registered with Gandi in the account to Gandi.

If you reopen your account during the time period that your domains can be recovered, we will unsuspend your domains, or inform you that your domains were deleted but they might be able to be restored. For more information, see Domains that you can register with Amazon Route 53.

> **Note**
>
> After 90 days have passed from when you closed your account, you can no longer reopen it. For more information, see Closing an account in the *AWS Account Management guide*.

For more information, see Contacting AWS Support about domain registration issues.

IP address ranges of Amazon Route 53 servers

Amazon Web Services (AWS) publishes its current IP address ranges in JSON format. If your firewalls or security groups restrict incoming traffic based on source IP addresses, confirm that your configuration allows traffic from the applicable IP address ranges.

To view the current IP address ranges for Route 53, download ip-ranges.json, and search the file for the following values:

- `"service": "ROUTE53"`
- `"service": "ROUTE53_HEALTHCHECKS"`
- `"service": "ROUTE53_HEALTHCHECKS_PUBLISHING"`

For more information about IP addresses for AWS resources, see AWS IP address ranges in the *Amazon Web Services General Reference*.

IP address ranges of Route 53 name servers

`"service": "ROUTE53"` – These IP address ranges are used by Route 53 name servers. Add these ranges to the list of allowed IP address ranges if you're using Route 53 as the DNS service for one or more domains and you want to be able to use the `dig` or `nslookup` commands to query Route 53 name servers.

> **Note**
>
> Route 53 name server IP addresses are static.

IP address ranges of Route 53 health checks

`"service": "ROUTE53_HEALTHCHECKS"` – These IP address ranges are used by Route 53 health checkers. Add these ranges to the list of allowed IP address ranges if you're using Route 53 health checks to check the health of resources on your network.

> **Note**
>
> We rarely change the IP address ranges of health checkers. Monitor the ip-ranges.json file for any updates to these ranges.

For more information about IP addresses for health checks, see Configuring router and firewall rules for Amazon Route 53 health checks.

Referencing prefix lists

A *prefix list* is a set of one or more CIDR block entries that you can use to configure security groups. Your router and firewall for the rules for the Amazon EC2 instance must allow inbound traffic from the IP addresses that the Route 53 health checkers use. A reference to a prefix list helps you to simplify the management of the CIDR blocks in your rules. If you frequently use the same CIDRs across multiple rules, you can manage those CIDRs in a single prefix list, instead of repeatedly referencing the same CIDRs in each rule. If you need to remove a CIDR block, you can remove its entry from the prefix list instead of removing the CIDR from every affected rule. For more information about prefix lists in general, see Group CIDR blocks using managed prefix lists in the *Amazon VPC User Guide*.

AWS-managed prefix lists are sets of IP address ranges for AWS services. AWS-managed prefix lists are created and maintained by AWS and can be used by anyone with an AWS account. You cannot create, modify, share, or delete an AWS-managed prefix list.

For more information about AWS-managed prefix lists, see Work with AWS-managed prefix lists in the *Amazon VPC User Guide*.

Internal IP address ranges of Route 53 health checks

`"service": "ROUTE53_HEALTHCHECKS_PUBLISHING"` –. Route 53 uses these IP address ranges only internally. You don't need to add these ranges to the list of allowed ranges.

Tagging Amazon Route 53 resources

A tag is a label that you assign to an AWS resource. Each tag consists of a *key* and a *value*, both of which you define. For example, the key might be "domain" and the value might be "example.com". Tags serve many purposes. One common use is to categorize and track your Amazon Route 53 costs. When you apply tags to Route 53 hosted zones, domains, and health checks, AWS generates a cost allocation report as a CSV file. This report shows your usage and costs grouped by your tags. You can apply tags that represent business categories (such as cost centers, application names, or owners) to organize your costs across multiple services. For more information about using tags for cost allocation, see Using cost allocation tags in the AWS Billing User Guide.

For best results, use Tag Editor in the AWS Management Console. Tag Editor provides a central way to create and manage your tags. For more information, see Working with Tag Editor in Getting Started with the AWS Management Console. You can also use the Route 53 console to apply tags for some resources:

- **Health checks** – For more information, see Naming and tagging health checks.
- **Route 53 VPC Resolver inbound endpoints** – For more information, see Values that you specify when you create or edit inbound endpoints.
- **Resolver outbound endpoints** – For more information, see Values that you specify when you create or edit outbound endpoints.
- **Resolver rules** – For more information, see Values that you specify when you create or edit rules.
- **Hosted zones** – For more information, see Working with hosted zones.

> **Note**
>
> Charges for Resolver endpoints are allocated per VPC Resolver network interface. You can't tag VPC Resolver network interfaces, so tag-based cost allocation isn't supported for Resolver endpoints. For information about pricing for VPC Resolver, see Amazon Route 53 pricing.

You can also apply tags to resources by using the Route 53 API. For more information, see the actions related to tags in the topic Route 53 API actions by function in the *Amazon Route 53 API Reference*.

Tutorials

This section covers the following tutorials:

Using Route 53 as the DNS service for subdomains

Learn how to use Route 53 as the DNS service for a new or existing subdomain while still using another DNS service for the parent domain.

Transitioning to Latency-based routing

Discover how to gradually migrate from standard routing to latency-based routing in Route 53, directing users to the lowest-latency AWS endpoint available.

Combine weighted and latency records for a smooth, low-risk transition with full control and rollback capability.

Adding another Region to latency-based routing

Expand your latency-based routing setup by adding a new AWS Region and gradually shifting traffic to the new Region.

Routing traffic to multiple Amazon EC2 instances in a Region

Use a combination of latency and weighted records to route traffic to multiple Amazon EC2 instances within a specific AWS Region.

Managing over 100 weighted records

Learn how to direct traffic to more than 100 endpoints by creating a tree of weighted alias records and weighted records.

Weighting fault-tolerant multi-record answers

Understand how to weight DNS responses that contain multiple records, providing fault tolerance and load balancing across multiple endpoints.

These tutorials cover various use cases and scenarios, helping you effectively leverage Route 53's routing policies, weighted records, and latency-based routing to optimize your DNS management and traffic routing.

Topics

- Using Amazon Route 53 as the DNS service for subdomains without migrating the parent domain
- Transitioning to latency-based routing in Amazon Route 53
- Adding another Region to your latency-based routing in Amazon Route 53
- Using latency and weighted records in Amazon Route 53 to route traffic to multiple Amazon EC2 instances in a Region
- Managing over 100 weighted records in Amazon Route 53
- Weighting fault-tolerant multi-record answers in Amazon Route 53

Using Amazon Route 53 as the DNS service for subdomains without migrating the parent domain

Amazon Route 53 provides flexibility in managing DNS for subdomains, allowing you to leverage its features without the need to migrate the entire parent domain.

You can either create a new subdomain or migrate an existing one to Route 53, while keeping the parent domain hosted with another DNS service provider.

Creating a new subdomain with Route 53:

1. Create a hosted zone for the new subdomain.
2. Add the desired DNS records (e.g., A, CNAME, MX) for the subdomain to the hosted zone.
3. Obtain the Route 53 name servers assigned to the hosted zone.
4. Update the DNS configuration of the parent domain by adding NS (Name Server) records for the subdomain, pointing to the Route 53 name servers.

Migrating an existing subdomain to Route 53:

1. Create a hosted zone for the subdomain.
2. Obtain the current DNS configuration for the subdomain from your existing DNS service provider.
3. Add the corresponding DNS records to the hosted zone.
4. Obtain the Route 53 name servers assigned to the hosted zone.
5. Update the DNS configuration of the parent domain by adding NS records for the subdomain, pointing to the Route 53 name servers.

By following these steps, you can leverage Route 53's advanced features, such as health checks, routing policies, and traffic flow management, for your subdomains while maintaining the parent domain's DNS configuration with your existing provider.

Topics

- Creating a subdomain that uses Amazon Route 53 as the DNS service without migrating the parent domain
- Migrating DNS service for a subdomain to Amazon Route 53 without migrating the parent domain

Creating a subdomain that uses Amazon Route 53 as the DNS service without migrating the parent domain

You can create a subdomain that uses Amazon Route 53 as the DNS service without migrating the parent domain from another DNS service.

The process has these basic steps:

1. Determine whether this procedure applies to your use case.
2. Create a Route 53 hosted zone for the subdomain.
3. Add records for the new subdomain to your Route 53 hosted zone.
4. *API only:* Confirm that your changes have propagated to all Route 53 DNS servers.

> **Note**
>
> Currently, the only way to verify that changes have propagated is to use the GetChange API action. Changes generally propagate to all Route 53 name servers within 60 seconds.

5. Update the DNS service for the parent domain by adding name server records for the subdomain.

Deciding which procedures to use for creating a subdomain

These steps explain an uncommon task. If you already use Route 53 as the DNS service for your domain and you want to route traffic for a subdomain, see Routing traffic for subdomains.

Use this procedure *only* if you're using another DNS service for a domain, such as example.com, and you want to start using Route 53 as the DNS service for an new subdomain of that domain, such as www.example.com.

Creating a hosted zone for the new subdomain

To use Amazon Route 53 as the DNS service for a new subdomain without migrating the parent domain, first create a hosted zone for the subdomain. Route 53 stores your subdomain data in this zone.

For information about how to create a hosted zone using the Route 53 console, see Creating a public hosted zone.

Creating records

You can create records using the Amazon Route 53 console or the Route 53 API. The records that you create become the records that DNS uses after you delegate the subdomain to Route 53, as explained in Updating your DNS service with name server records for the subdomain.

> **Important**
>
> Do not create extra name server (NS) or start of authority (SOA) records in the Route 53 hosted zone. Do not delete the existing NS and SOA records.

To create records using the Route 53 console, see Working with records. To create records using the Route 53 API, use `ChangeResourceRecordSets`. For more information, see ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.

Checking the status of your changes (API only)

Creating a new hosted zone and changing records take time to propagate to the Route 53 DNS servers. If you used ChangeResourceRecordSets to create your records, you can use the `GetChange` action to determine whether your changes have propagated. (`ChangeResourceRecordSets` returns a value for `ChangeId`, which you can include in a subsequent `GetChange` request. `ChangeId` is not available if you created the records by using the console.) For more information, see GET GetChange in the *Amazon Route 53 API Reference*.

> **Note**
>
> Changes generally propagate to all Route 53 name servers within 60 seconds.

Updating your DNS service with name server records for the subdomain

After your changes to Amazon Route 53 records have spread (see Checking the status of your changes (API only)), update the DNS service for the parent domain by adding NS records for the subdomain. This delegates the subdomain to Route 53. For example, if example.com uses another DNS service and you created test.example.com in Route 53, update the DNS for example.com with NS records for test.example.com.

To update your DNS service with name server records for the subdomain

1. Using the method provided by your DNS service, back up the zone file for the parent domain.
2. In the Route 53 console, get the name servers for your Route 53 hosted zone:

   a. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

   b. In the navigation pane, click **Hosted zones**.

   c. On the **Hosted zones** page, choose the radio button (not the name) for the hosted zone, then choose **View details**.

   d. On the details page for the hosted zone, choose **Hosted zone details**.

   e. Make note of the four servers listed for **Name servers**.

   Alternatively, you can use the `GetHostedZone` action. For more information, see GetHostedZone in the *Amazon Route 53 API Reference*.
3. Add NS records for the subdomain to the zone file of the parent domain, using the method from the DNS service. In these NS records, specify the four Route 53 name servers linked to the hosted zone that you created in Step 1.

> **Important**
>
> Do not add a start of authority (SOA) record to the zone file for the parent domain. The subdomain uses Route 53, so the parent's DNS service does not control the subdomain.

If your DNS service added an SOA record for the subdomain, delete it. Do not delete the SOA record for the parent domain.

Migrating DNS service for a subdomain to Amazon Route 53 without migrating the parent domain

You can migrate a subdomain to use Amazon Route 53 as the DNS service without migrating the parent domain from another DNS service.

The process has these basic steps:

1. Determine whether this procedure applies to your use case.
2. Create a Route 53 hosted zone for the subdomain.
3. Get the current DNS configuration from the current DNS service provider for the parent domain.
4. Add records for the subdomain to your Route 53 hosted zone.
5. *API only:* Confirm that your changes have propagated to all Route 53 DNS servers.

> **Note**
>
> Currently, the only way to verify that changes have propagated is to use the GetChange API action. Changes generally propagate to all Route 53 name servers within 60 seconds.

6. Update the DNS configuration with the DNS service provider for the parent domain by adding name server records for the subdomain.

Deciding which procedures to use for creating a subdomain

The procedures in this topic explain how to perform an uncommon task. If you already use Route 53 as the DNS service for your domain and you want to route traffic for a subdomain, such as www.example.com, to your resources, see Routing traffic for subdomains.

Use this procedure *only* if you're using another DNS service for a domain, such as example.com, and you want to start using Route 53 as the DNS service for an existing subdomain of that domain, such as www.example.com.

Creating a hosted zone for the subdomain

To migrate a subdomain from another DNS service to Amazon Route 53 without migrating the parent domain, start by creating a hosted zone for the subdomain. Route 53 stores your subdomain information in this hosted zone.

For information about how to create a hosted zone using the Route 53 console, see Creating a public hosted zone.

Getting your current DNS configuration from your DNS service provider

Contact your current DNS service provider and request the current DNS configuration. Ask your current DNS service provider how to get a *zone file* or a *records list*. You can use this as a basis for configuring Route 53 as the DNS service for the subdomain.

What you ask for and the format depends on your current DNS service provider. Ideally, they give you a zone file, which has all the records in your current configuration. These records tell DNS how to route traffic for your domains and subdomains. For example, a record can route traffic for your domain name to a web server in your data center, an Amazon EC2 instance, or a CloudFront distribution. If you can get a zone file, you can edit it to remove records you don't want to migrate to Amazon Route 53. Then import the remaining records into your Route 53 hosted zone. This greatly simplifies the process.

Creating records

Using the records from your current DNS service provider as a starting point, create matching records in the Amazon Route 53 hosted zone that you created for the subdomain. These records will become the records that DNS uses after you delegate the subdomain to Route 53, as explained in Updating your DNS service with name server records for the subdomain.

> **Important**
>
> Do not create additional name server (NS) or start of authority (SOA) records in the Route 53 hosted zone, and do not delete the existing NS and SOA records.

To create records using the Route 53 console, see Working with records. To create records using the Route 53 API, use `ChangeResourceRecordSets`. For more information, see ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.

Checking the status of your changes (API only)

Creating a new hosted zone and changing records take time to propagate to the Route 53 DNS servers. If you used ChangeResourceRecordSets to create your records, you can use the `GetChange` action to determine whether your changes have propagated. (`ChangeResourceRecordSets` returns a value for `ChangeId`, which you can include in a subsequent `GetChange` request. `ChangeId` is not available if you created the records by using the console.) For more information, see GET GetChange in the *Amazon Route 53 API Reference*.

> **Note**
>
> Changes generally propagate to all Route 53 name servers within 60 seconds.

Updating your DNS service with name server records for the subdomain

After your changes to Amazon Route 53 records have propagated (see Checking the status of your changes (API only)), update the DNS service for the parent domain by adding NS records for the subdomain. This delegates the subdomain to Route 53. For example, suppose example.com is hosted with another DNS service and you're migrating test.example.com to Route 53. You must update the DNS service for example.com with the NS records that Route 53 assigned to the new hosted zone.

To update your DNS service with name server records for the subdomain

1. Using the method provided by your DNS service, back up the zone file for the parent domain.
2. If your previous DNS provider lets you change TTL settings for their name servers, change the settings to 900 seconds. This limits the time during which client requests try to resolve domain names using obsolete name servers. If the current TTL is 172800 seconds (two days), you still need to wait two days for resolvers and clients to stop caching DNS records. After the TTL settings expire, you can safely delete the records stored at the previous provider and make changes only to Route 53.
3. In the Route 53 console, get the name servers for your Route 53 hosted zone:
   a. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.
   b. In the navigation pane, click **Hosted zones**.

c. On the **Hosted zones** page, choose the radio button (not the name) for the hosted zone, then choose **View details**.

d. On the details page for the hosted zone, choose **Hosted zone details**.

e. Make note of the four servers listed for **Name servers**.

Alternatively, you can use the `GetHostedZone` action. For more information, see GetHostedZone in the *Amazon Route 53 API Reference*.

4. Using the method provided by the DNS service of the parent domain, add NS records for the subdomain to the zone file. Name the NS records with the subdomain name. For the values, specify the four Route 53 name servers linked to the hosted zone you created in Step 2. Different DNS services use different terms. You might need to contact support to learn how to do this.

> **Important**
>
> Do not add a start of authority (SOA) record to the zone file for the parent domain. Because the subdomain will use Route 53, the DNS service for the parent domain is not the authority for the subdomain.
> If your DNS service automatically added an SOA record for the subdomain, delete the record for the subdomain. However, do not delete the SOA record for the parent domain.

Changes to DNS resolvers can take 48 hours or more to propagate. During this period, DNS resolvers might still answer with the name servers for the DNS service of the parent domain. Also, client computers might still have the previous name servers for the subdomain in their cache.

5. After the registrar's TTL settings for the domain expire (see Step 2), delete these records from the zone file for the parent domain:

- The records that you added to Route 53 as described in Creating records.
- Your DNS service's NS records. When you finish deleting NS records, the only NS records in the zone file will be the ones you created in Step 4.

Transitioning to latency-based routing in Amazon Route 53

With latency-based routing, Amazon Route 53 can direct your users to the lowest-latency AWS endpoint available. For example, you might associate a DNS name like `www.example.com` with an ELB Classic, Application, or Network Load Balancer, or with Amazon EC2 instances or Elastic IP addresses that are hosted in the US East (Ohio) and Europe (Ireland) regions. The Route 53 DNS servers decide, based on network conditions of the past couple of weeks, which instances in which regions should serve particular users. A user in London will likely be directed to the Europe (Ireland) instance, a user in Chicago will likely be directed to the US East (Ohio) instance, and so on. Route 53 supports latency-based routing for A, AAAA, TXT, and CNAME records, as well as aliases to A and AAAA records.

> **Note**
>
> Data about the latency between users and your resources is based entirely on traffic between users and AWS data centers. If you aren't using resources in an AWS Region, the actual latency between your users and your resources can vary significantly from AWS latency data. This is true even if your resources are located in the same city as an AWS Region.

For a smooth, low-risk transition, you can combine weighted and latency records to gradually migrate from standard routing to latency-based routing with full control and rollback capability at each stage. Let's consider an example in which `www.example.com` is currently hosted on an Amazon EC2 instance in the US East (Ohio) region. The instance has the Elastic IP address `W.W.W.W`. Suppose you want to continue routing traffic to the US East (Ohio) region when applicable while also beginning to direct users to additional Amazon EC2 instances in the US West (N. California) region (Elastic IP `X.X.X.X`) and in the Europe (Ireland) region (Elastic IP `Y.Y.Y.Y`). The Route 53 hosted zone for `example.com` already has a record for `www.example.com` that has a **Type** of A and a **Value** (an IP address) of `W.W.W.W`.

When you're finished with the following example, you'll have two weighted alias records:

- You'll convert your existing record for `www.example.com` into a weighted alias record that continues to direct the majority of your traffic to your existing Amazon EC2 instance in the US East (Ohio) region.
- You'll create another weighted alias record that initially directs only a small portion of your traffic to your latency records, which route traffic to all three regions.

By updating the weights in these weighted alias records, you can gradually shift from routing traffic only to the US East (Ohio) region to routing traffic to all three regions in which you have Amazon EC2 instances.

To transition to latency-based routing

1. Make a copy of the record for `www.example.com`, but use a new domain name, for example, `copy-www.example.com`. Give the new record the same **Type** (A) and **Value** (`W.W.W.W`) as the record for `www.example.com`.
2. Update the existing A record for `www.example.com` to make it a weighted alias record:

   - For **Value/Route traffic to**, choose **Alias to another record in this hosted zone**, and specify `copy-www.example.com`.
   - For **Weight**, specify 100.

   When you're finished with the update, Route 53 will continue to use this record to route all traffic to the resource that has an IP address of `W.W.W.W`.
3. Create a latency record for each of your Amazon EC2 instances, for example:

   - US East (Ohio), Elastic IP address `W.W.W.W`
   - US West (N. California), Elastic IP address `X.X.X.X`
   - Europe (Ireland), Elastic IP address `Y.Y.Y.Y`

   Give all of the latency records the same domain name, for example, `www-lbr.example.com` and the same type, A.

   When you're finished creating the latency records, Route 53 will continue to route traffic using the record that you updated in Step 2.

   You can use `www-lbr.example.com` for validation testing, for example, to make sure that each endpoint can accept requests.
4. Let's now add the `www-lbr.example.com` latency record into the `www.example.com` weighted record and begin routing limited traffic to the corresponding Amazon EC2 instances. This means that the Amazon EC2 instance in the US East (Ohio) region will be getting traffic from both weighted records.

   Create another weighted alias record for `www.example.com`:

- For **Value/Route traffic to**, choose **Alias to another record in this hosted zone**, and specify `www-lbr.example.com.`
- For **Weight**, specify 1.

When you finish and your changes are synchronized to Route 53 servers, Route 53 will begin to route a tiny fraction of your traffic (1/101) to the Amazon EC2 instances for which you created latency records in Step 3.

5. As you develop confidence that your endpoints are adequately scaled for the incoming traffic, adjust the weights accordingly. For example, if you want 10% of your requests to be based on latency-based routing, change the weights to 90 and 10, respectively.

For more information about creating latency records, see Creating records by using the Amazon Route 53 console.

Adding another Region to your latency-based routing in Amazon Route 53

If you're using latency based routing and you want to add an instance in a new region, you can gradually shift traffic to the new region in the same way that you gradually shifted traffic to latency-based routing in Transitioning to latency-based routing in Amazon Route 53.

For example, suppose you're using latency-based routing to route traffic for `www.example.com`, and you want to add an Amazon EC2 instance in Asia Pacific (Tokyo) to your instances in US East (Ohio), US West (N. California), and Europe (Ireland). The following example procedure explains one way that you could add an instance in another Region.

For this example, the Amazon Route 53 hosted zone for `example.com` already has a weighted alias record for `www.example.com` that is routing traffic to the latency-based records for `www-lbr.example.com`:

- US East (Ohio), Elastic IP address `W.W.W.W`
- US West (N. California), Elastic IP address `X.X.X.X`
- Europe (Ireland), Elastic IP address `Y.Y.Y.Y`

The weighted alias record has a weight of 100. After you transitioned to latency-based routing, assume that you deleted the other weighted record that you used for the transition.

To add another Region to your latency-based routing in Route 53

1. Create four new latency-based records that include the three original regions as well as the new region to which you want to start routing traffic.

   - US East (Ohio), Elastic IP address `W.W.W.W`
   - US West (N. California), Elastic IP address `X.X.X.X`
   - Europe (Ireland), Elastic IP address `Y.Y.Y.Y`
   - Asia Pacific (Tokyo), Elastic IP address `Z.Z.Z.Z`

   Give all of the latency records the same new domain name, for example, `www-lbr-2012-04-30.example.com`, and the same type, A.

   When you're finished creating the latency records, Route 53 will continue to route traffic using the original weighted alias record (`www.example.com`) and latency records (`www-lbr.example.com`).

   You can use the `www-lbr-2012-04-30.example.com` records for validation testing, for example, to make sure that each endpoint can accept requests.
2. Create a weighted alias record for the new latency records:

   - For the domain name, specify the name for the existing weighted alias record, `www.example.com`.
   - For **Value/Route traffic to**, choose **Alias to another record in this hosted zone**, and specify `www-lbr-2012-04-30.example.com`.
   - For **Weight**, specify 1.

   When you finish, Route 53 will begin to route a tiny fraction of your traffic (1/101) to the Amazon EC2 instances for which you created the `www-lbr-2012-04-30.example.com` latency records in Step 1. The remainder of the traffic will continue to be routed to the `www-lbr.example.com` latency records, which do not include the Amazon EC2 instance in the Asia Pacific (Tokyo) region.

3. As you develop confidence that your endpoints are adequately scaled for the incoming traffic, adjust the weights accordingly. For example, if you want 10% of your requests to be routed to the latency records that include the Tokyo region, change the weight for `www-lbr.example.com` from 100 to 90 and the weight for `www-lbr-2012-04-30.example.com` from 1 to 10.

For more information about creating records, see Creating records by using the Amazon Route 53 console.

Using latency and weighted records in Amazon Route 53 to route traffic to multiple Amazon EC2 instances in a Region

If your application is running on Amazon EC2 instances in two or more Amazon EC2 regions, and if you have more than one Amazon EC2 instance in one or more regions, you can use latency-based routing to route traffic to the correct region and then use weighted records to route traffic to instances within the Region based on weights that you specify.

For example, suppose you have three Amazon EC2 instances with Elastic IP addresses in the US East (Ohio) region and you want to distribute requests across all three IPs evenly for users for whom US East (Ohio) is the appropriate region. Just one Amazon EC2 instance is sufficient in the other regions, although you can apply the same technique to many regions at once.

To use latency and weighted records in Amazon Route 53 to route traffic to multiple Amazon EC2 instances in a Region

1. Create a group of weighted records for the Amazon EC2 instances in the Region. Note the following:

   - Give each weighted record the same value for **Record name** (for example, `us-east.example.com`) and **Record type**.
   - For **Value/Route traffic to**, choose **IP address or another value depending on the record type**, and specify the value of one of the Elastic IP addresses.
   - If you want to weight the Amazon EC2 instances equally, specify the same value for **Weight**.
   - Specify a unique value for **Set ID** for each record.

   For more information about weighted record values, see Weighted routing

2. If you have multiple Amazon EC2 instances in other regions, repeat Step 1 for the other regions. Specify a different value for **Name** in each Region.
3. For each Region in which you have multiple Amazon EC2 instances (for example, US East (Ohio)), create a latency alias record. For **Value/Route traffic to**, choose **Alias to another record in this hosted zone**, and specify the value of the **Record name** field (for example, `us-east.example.com`) that you assigned to the weighted records in that Region.
4. For each Region in which you have one Amazon EC2 instance, create a latency record. For **Record name**, specify the same value that you specified for the latency alias records that you created in Step 3. For **Value/Route traffic to**, choose **IP address or another value depending on the record type**, and specify the Elastic IP address of the Amazon EC2 instance in that Region.

   For more information about adding alias records to Amazon EC2 instances, see Routing traffic to an Amazon EC2 instance

For more information about creating records, see Creating records by using the Amazon Route 53 console.

Managing over 100 weighted records in Amazon Route 53

Amazon Route 53 lets you configure weighted records. For a given name and type (for example, `www.example.com`, type A), you can configure up to 100 alternative responses, each with its own weight. When responding to queries for `www.example.com`, Route 53 DNS servers select a weighted random response to return to DNS resolvers. The value of a weighted record that has a weight of 2 is returned, on average, twice as often as the value of a weighted record that has a weight of 1.

If you need to direct traffic to more than 100 endpoints, one way to achieve this is to use a tree of weighted alias records and weighted records. For example, the first "level" of the tree might be up to 100 weighted alias records, each of which can, in turn, point to up to 100 weighted records. Route 53 permits up to three levels of recursion, allowing you to manage up to 1,000,000 unique weighted endpoints.

A simple two-level tree might look like this:

Weighted alias records

- `www.example.com` aliases to `www-a.example.com` with a weight of 1

- www.example.com aliases to www-b.example.com with a weight of 1

Weighted records

- www-a.example.com, type A, value 192.0.2.1, weight 1
- www-a.example.com, type A, value 192.0.2.2, weight 1

- www-b.example.com, type A, value 192.0.2.3, weight 1
- www-b.example.com, type A, value 192.0.2.4, weight 1

For more information about creating records, see Working with records.

Weighting fault-tolerant multi-record answers in Amazon Route 53

> **Note**
>
> Records that use the multivalue answer routing policy behave in much the same way as the configuration that is documented in this tutorial. The main difference is that the tutorial configuration lets you specify weights, which can be useful when your endpoints have different capacities. For more information, see Multivalue answer routing.

An Amazon Route 53 weighted record can only be associated with one record, meaning a combination of one name (for example, example.com) and one record type (for example, A). But it is often desirable to weight DNS responses that contain multiple records.

For example, you might have eight Amazon EC2 instances or Elastic IP endpoints for a service. If the clients of that service support connection retries (as all common browsers do), then providing multiple IP addresses in DNS responses provides those clients with alternative endpoints in the event of the failure of any particular endpoint. You can even protect against the failure of an availability zone if you configure responses to contain a mix of IPs hosted in two or more availability zones.

Multi-record answers are also useful when a large number of clients (for example, mobile web applications) share a small set of DNS caches. In this case, multi-record answers allow clients to

direct requests to several endpoints even if they receive a common DNS response from the shared cache.

These types of weighted multi-record answers can be achieved by using a combination of records and weighted alias records. You can group eight endpoints into two distinct record sets containing four IP addresses each:

`endpoint-a.example.com`, type A, with the following values:

- 192.0.2.1
- 192.0.2.2
- 192.0.2.128
- 192.0.2.129

`endpoint-b.example.com`, type A, with the following values:

- 192.0.2.3
- 192.0.2.4
- 192.0.2.130
- 192.0.2.131

You can then create a weighted alias record that points to each group:

- `www.example.com` aliases to `endpoint-a.example.com`, type A, weight 1
- `www.example.com` aliases to `endpoint-b.example.com`, type A, weight 1

For more information about creating records, see Working with records.

Best practices for Amazon Route 53

This section provides best practices for various components of Amazon Route 53, including:

1. **DNS best practices:**
   - Understand the trade-offs between time to live (TTL) values and responsiveness versus reliability.
   - Use alias records instead of CNAME records when possible for improved performance and cost savings.
   - Configure default routing policies to make sure all clients receive a response.
   - Leverage latency-based routing for minimizing application latency and geolocation/geoproximity routing for stability and predictability.
   - Verify change propagation using the `GetChange` API for automated workflows.
   - Delegate subdomains from the parent zone for consistent routing.
   - Avoid large single responses by using multivalue answer routing.
2. **Resolver best practices:**
   - Prevent routing loops by avoiding associating the same VPC with both a Resolver rule and its inbound endpoint.
   - Implement security group rules to reduce connection tracking overhead and maximize query throughput.
   - Configure inbound endpoints with IP addresses in multiple Availability Zones for redundancy.
   - Be aware of potential DNS zone walking attacks and contact AWS Support if your endpoints experience throttling.
3. **Health checks best practices:**
   - Follow recommendations for optimizing Route 53 health checks to ensure reliable monitoring of your resources
4. **Domain registration best practices:**
   - Avoid registering a domain in Route 53 on the same AWS account that uses the domain for root user email, to prevent a circular dependency that can permanently delete your domain.

By adhering to these best practices, you can optimize the performance, reliability, and security of your DNS infrastructure, ensuring efficient and effective routing of traffic to your applications and services

Topics

- Best practices for Amazon Route 53 DNS
- Best practices for VPC Resolver
- Best practices for Amazon Route 53 health checks
- Best practices for Amazon Route 53 domain registration

Best practices for Amazon Route 53 DNS

Follow these best practices to get the best results when using the Amazon Route 53 DNS service.

Use data plane functions for DNS failover and app recovery

The data planes for Route 53, including health checks, and Amazon Application Recovery Controller (ARC) routing control are globally distributed, and are designed for 100% availability and functionality, even during severe events. They integrate with each other and don't depend on control plane functionality. While the control planes for these services, including their consoles, are generally very reliable, they're designed in a more centralized way and prioritize durability and consistency rather than high availability. For scenarios such as failover during disaster recovery, we recommend that you use features like Route 53 health checks and ARC routing control that rely on data plane functionality to update DNS. For more information, see Control and data plane concepts and Blog: Creating Disaster Recovery Mechanisms Using Amazon Route 53.

Choosing TTL values for DNS records

The DNS TTL is the numeric value (in seconds) that DNS resolvers use to decide how long a record can be cached for without making another query to Route 53. All DNS records must have a TTL specified for them. The recommended range for TTL values is 60 to 172,800 seconds.

The choice of a TTL is a trade-off between latency and reliability, and responsiveness to change. With shorter TTLs on a record, DNS resolvers notice updates to the record quicker as they must query more frequently. This increases the query volume (and cost). As you lengthen the TTL, DNS resolvers answer queries from cache more often, which is typically faster, cheaper, and in some situations, more reliable, because it avoids queries across the internet. There is no correct value, but it is worthwhile to think about whether responsiveness or reliability is more important to you.

Things to consider when you set TTL values include:

- Set DNS record TTLs for the length of time that you can afford to wait for a change to take effect. This is especially true on delegations (NS record sets), or other records that rarely change, for example MX records. For these records, longer TTLs are recommended. A value between an hour (3600s) and a day (86,400s) is a common choice.
- For records that need to be altered as part of a rapid failover mechanism (especially records that are health checked), lower TTLs are appropriate. Setting a TTL of 60 or 120 seconds is a common choice for this scenario.
- When you want to make changes to critical DNS entries, we recommend that you temporarily shorten the TTLs. Then you can make the changes, observe, and rollback quickly if you need to. After the changes are finalized and working as expected, you can increase the TTL.

For more information see TTL (seconds).

CNAME records

DNS CNAME records are a way to point one domain name to another. If a DNS resolver resolves domain-1.example.com and finds a CNAME pointing at domain-2.example.com, the DNS resolver must proceed to resolve domain-2.example.com before it can respond. These records are useful in many situations, for example, to ensure consistency when a website has more than one domain name.

However, DNS resolvers must make more queries to answer CNAMEs, which increases latency and costs. Where possible, a faster and cheaper alternative is to use a Route 53 alias record. Alias records allow Route 53 to respond with a direct answer for AWS resources (for example, a load balancer) and for other domains within the same hosted zone.

For more information, see Routing internet traffic to your AWS resources.

Advanced DNS routing

- When using az-affinity, geolocation, geoproximity, or latency-based routing, always set a default (for az-affinity routing, a Global record), unless you want some clients to receive *no answer* responses.
- To minimize application latency, use latency-based routing. This type of routing data can change frequently.
- To provide routing stability and predictability, use either geolocation or geoproximity routing.

For more information, see AZ-affinity routing, Geolocation routing, Geoproximity routing, and Latency-based routing.

DNS change propagation

When you create or update a record or hosted zone by using the Route 53 console or API, it takes some time for the change to be reflected across the internet. This is called *change propagation*. While propagation typically takes less than one minute globally, there are occasionally delays, for example, due to problems syncing to one location, or in rare cases, problems within the central control plane. If you are building automated provisioning work flows, and it is important to wait for change propagation to complete before you move forward with the next work flow step, use the GetChange API to verify that your DNS changes have gone into effect (`Status =INSYNC`) .

DNS delegation

When you delegate multiple levels of subdomains in DNS, it is important to always delegate from the parent zone. For example, if you are delegating www.dept.example.com, you should do so from the dept.example.com zone, not from the example.com zone. Delegations from a *grandparent* to a *child* zone might not work at all or work only inconsistently. For more information, see Routing traffic for subdomains.

Size of DNS response

Avoid creating large single responses. If responses are larger than 512 bytes, many DNS resolvers must retry over TCP instead of UDP, which can reduce reliability and lead to slower responses. We recommend using multivalue answer routing, which chooses eight healthy random IP addresses to keep responses within the 512 byte boundary.

For more information, see Multivalue answer routing and DNS Reply Size Test Server.

Best practices for VPC Resolver

This section provides best practices for optimizing Route 53 VPC Resolver, covering the following topics:

1. **Avoiding Loop Configurations with Resolver Endpoints:**
   - Prevent routing loops by ensuring that the same VPC is not associated with both a Resolver rule and its inbound endpoint.
   - Utilize AWS RAM to share VPCs across accounts while maintaining proper routing configurations.

   For more information, see Avoid loop configurations with Resolver endpoints

2. **Scaling Resolver endpoints:**
   - Implement security group rules that permit traffic based on connection state to reduce connection tracking overhead
   - Follow recommended security group rules for inbound and outbound Resolver endpoints to maximize query throughput.
   - Monitor unique IP address and port combinations generating DNS traffic to avoid capacity limitations.

   For more information, see Resolver endpoint scaling
3. **High availability for Resolver endpoints:**
   - Create inbound endpoints with IP addresses in at least two Availability Zones for redundancy.
   - Provision additional network interfaces to ensure availability during maintenance or traffic surges

   For more information, see High availability for Resolver endpoints
4. **Preventing DNS zone walking attacks:**
   - Be aware of potential DNS zone walking attacks, where attackers attempt to retrieve all content from DNSSEC-signed DNS zones.
   - If your endpoints experience throttling due to suspected zone walking, contact AWS Support for assistance.

   For more information, see DNS zone walking
5. **Subnet compatibility for Resolver endpoints:**
   - We recommend using non-Outposts subnets for Resolver endpoints.
   - Outposts subnets with Local Network Interface (LNI) enabled are not compatible with VPC Resolver endpoints.
   - If you enable LNI on a subnet that contains VPC Resolver endpoint ENIs, those ENIs stop functioning.

   For more information, see Subnet compatibility for Resolver endpoints

By following these best practices, you can optimize the performance, scalability, and security of your VPC Resolver deployments, ensuring reliable and efficient DNS resolution for your applications and resources.

Avoid loop configurations with Resolver endpoints

Don't associate the same VPC to a Resolver rule and its inbound endpoint (whether it's a direct target of the endpoint, or through an on-premises DNS server). When the outbound endpoint in a Resolver rule points to an inbound endpoint that shares a VPC with the rule, it can cause a loop where the query is continually passed between the inbound and outbound endpoints.

The forwarding rule can still be associated with other VPCs that are shared with other accounts by using AWS Resource Access Manager (AWS RAM). Private hosted zones associated with the hub, or a central VPC, still resolve from queries to inbound endpoints because a forwarding resolver rule does not change this resolution.

Resolver endpoint scaling

Resolver endpoint security groups use connection tracking to gather information about traffic to and from the endpoints. Each endpoint interface has a maximum number of connections that can be tracked, and a high volume of DNS queries can exceed the connections and cause throttling and query loss. Connection tracking is AWS's default behavior for monitoring the state of traffic flowing through security groups (SGs). Using connection tracking in SGs will reduce the throughput of traffic, however, you can implement untracked connections to reduce overhead and improve performance. For more information see Untracked connections.

If the connection tracking is enforced either by using restrictive security group rules or queries are routed through Network Load Balancer (see Automatically tracked connections), the overall maximum queries per second per IP address for an endpoint can be as low as 1500.

The following tables list the recommended security group rules for inbound Resolver endpoints.

Inbound endpoint ingress rules

Protocol type	Port number	Source IP
TCP	53	0.0.0.0/0
UDP	53	0.0.0.0/0

Inbound endpoint egress rules

Protocol type	Port number	Destination IP
TCP	All	0.0.0.0/0
UDP	All	0.0.0.0/0

The following tables list the recommended security group rules for outbound Resolver endpoints.

Outbound endpoint ingress rules

Protocol type	Port number	Source IP
TCP	All	0.0.0.0/0
UDP	All	0.0.0.0/0

Outbound endpoint egress rules

Protocol type	Port number	Destination IP
TCP	53	0.0.0.0/0
UDP	53	0.0.0.0/0

Note

Security group port requirements:

- **Inbound endpoints** require ingress rules allowing TCP and UDP on port 53 to receive DNS queries from your network. Egress rules can allow all ports since the endpoint might need to respond to queries from various source ports.
- **Outbound endpoints** require egress rules allowing TCP and UDP access to the ports you're using for DNS queries on your network. Port 53 is shown in the example above because it's the most common DNS port, but your network might use different ports. Ingress rules can allow all ports to accommodate responses from your DNS servers.

Inbound Resolver endpoints

For clients using an inbound resolver endpoint, the capacity of the elastic network interface will be impacted if you have over 40,000 unique IP address and port combinations generating the DNS traffic.

High availability for Resolver endpoints

When you create your VPC Resolver inbound endpoints, Route 53 requires that you create at least two IP addresses that the DNS resolvers on your network will forward queries to. You should also specify IP addresses in at least two Availability Zones for redundancy.

If you require more than one elastic network interface endpoint to be available at all times, we recommend that you create at least one more network interface than you need, to make sure you have additional capacity available for handling possible traffic surges. The additional network interface also ensures availability during service operations like maintenance or upgrades.

For more information, see this detailed blog article: How to achieve DNS high availability with Resolver endpoints and Values that you specify when you create or edit inbound endpoints.

DNS zone walking

A DNS zone walking attack attempts to get all content from DNSSEC-signed DNS zones. If VPC Resolver team detects a traffic pattern that matches the ones generated when DNS zones are walked on your endpoint, the service team throttles the traffic on your endpoint. As a consequence you might observe a high percentage of your DNS queries timing out.

If you observe reduced capacity on your endpoints and believe that the endpoint have been throttled erroneously, go to https://console.aws.amazon.com/support/home#/ to create a support case.

Subnet compatibility for Resolver endpoints

We recommend using VPC Resolver on AWS Outposts to create endpoints on AWS Outposts Racks.

> **Important**
>
> Outposts subnets with Local Network Interface (LNI) enabled are not compatible with VPC Resolver endpoints. If you enable LNI on a subnet that contains VPC Resolver endpoint elastic network interfaces (ENIs), those ENIs stop functioning.

To avoid this issue:

- Before creating a Resolver endpoint, verify that the target subnets don't have LNI enabled.
- Don't enable LNI on subnets that already contain Resolver endpoint ENIs.
- If you need both LNI and Resolver endpoints, use separate subnets for each.

For more information about LNI, see Local network interfaces in the *AWS Outposts User Guide*.

Best practices for Amazon Route 53 health checks

Effective health check configuration is essential for maintaining a highly available and resilient infrastructure. Here are some best practices to consider when setting up and managing Amazon Route 53 health checks:

1. **Use elastic IP addresses for health check endpoints:**
   - Utilize elastic IP addresses for your health check endpoints to ensure consistent monitoring.
   - If you are no longer using an Amazon EC2 instance, remember to delete any associated health checks to avoid potential security risks or data compromise.

   Fore more information, see Values that you specify when you create or update health checks.
2. **Configure appropriate health check intervals:**
   - Set health check intervals based on your application's requirements and the criticality of the monitored resources.
   - Shorter intervals provide faster failure detection but might increase Route 53 costs and load on your resources.
   - Longer intervals reduce costs and resource load but might delay failure detection.

   Fore more information, see Advanced configuration ("Monitor an Endpoint" only).
3. **Implement alarm notifications:**
   - Configure Amazon CloudWatchalarms to receive notifications when health checks fail or recover.
   - Set appropriate alarm thresholds based on your application's requirements and the expected behavior of your resources.
   - Integrate notifications with your monitoring and incident response processes.

   Fore more information, see Monitoring health checks using CloudWatch.

4. **Utilize health check Regions strategically:**
   - Choose health check Regions based on the geographic distribution of your users and resources.
   - Consider using multiple health check regions for critical resources to improve reliability and reduce the impact of regional outages.
5. **Monitor health check logs and metrics:**
   - Regularly review Route 53 health check logs and CloudWatch metrics to identify potential issues or performance bottlenecks
   - Analyze health check failure reasons and take appropriate actions to resolve underlying problems.
6. **Implement Failover and Failback Strategies:**
   - Leverage Route 53's failover routing policies to automatically route traffic to healthy resources in the event of failures.
   - Plan and test failover and failback processes to ensure seamless transition during outages and recovery.

   Fore more information, see Configuring DNS failover.
7. **Regularly Review and Update Health Checks:**
   - Update health check endpoints, intervals, and alarm thresholds as needed to maintain optimal monitoring and performance.

By following these best practices, you can effectively leverage Route 53 health checks to monitor the health and availability of y our resources, ensuring a reliable and high-performing infrastructure for your applications and services.

Best practices for Amazon Route 53 domain registration

Follow these best practices to avoid account and domain recovery issues when you register domains with Amazon Route 53.

Avoid using a Route 53 domain for root user email on the same account

> **Warning**
>
> If you register a domain in Route 53 and use that domain for your AWS account root user email address on the same account, you create a circular dependency that can permanently delete your domain.

When an AWS account is closed or suspended, Route 53 suspends the domain registration within 5 days. Domain suspension causes the domain's name servers to stop resolving, which breaks email delivery for the domain. Because the root user email address depends on the domain, you can no longer receive password reset or account recovery emails. Without access to the root user email, you cannot sign in, reopen the account, or reinstate the domain. If the domain remains suspended, it is permanently deleted after 30 days with no remediation path.

This sequence creates an unrecoverable circular dependency:

1. Account closure or suspension triggers domain suspension within 5 days.
2. Domain suspension stops DNS resolution, breaking email delivery.
3. Without email delivery, you cannot complete root user password recovery.
4. Without root user access, you cannot reopen the account or reinstate the domain.
5. After 30 days of suspension, the domain is permanently deleted.

> **Important**
>
> To prevent this scenario, use one of the following recommended architectures:
>
> - **Register the domain on a separate AWS account** – Use a dedicated AWS account (such as an AWS Organizations management account) to register domains that serve as root user email domains for other accounts. This ensures that the domain registration remains active even if the dependent account is closed.
> - **Use an email address hosted outside AWS** – Choose a root user email address on a domain that is not registered or hosted within AWS. This eliminates the dependency between your account status and email delivery.

Making API Requests

Topics

- Making API Requests for Hosted Zones, Resource Record Sets, and Health Checks
- Making API Requests for Domain Registration and for Route 53 Resolver
- Signing Amazon Route 53 API Requests
- Amazon Route 53 API actions by function
- Throttling for Amazon Route 53 API requests

This section describes how to make requests to the two Route 53 APIs:

- A REST API for hosted zones, resource record sets, health checks, and cost allocation tags
- RPC APIs for domain registration and for Route 53 Resolver

For each API, we describe the components of requests and the content of responses. We also describe how to authenticate requests.

Making API Requests for Hosted Zones, Resource Record Sets, and Health Checks

Topics

- REST Requests
- REST Responses

This section describes how to make REST requests to the Amazon Route 53 API for creating and managing your hosted zones, resource record sets, health checks, and cost allocation tags. We'll acquaint you with the components of requests, the content of responses, and how to authenticate requests.

> **Note**
>
> To register and manage domains, use the Route 53 API for domain registration. For more information, see Making API Requests for Domain Registration and for Route 53 Resolver.

REST Requests

Amazon Route 53 REST requests are HTTPS requests, as defined by RFC 2616. This section describes the structure of an Route 53 REST request.

A typical REST action consists of sending a single HTTPS request to Route 53, and waiting for the response. Like any HTTP request, a REST request to Route 53 contains a request method, a URI, request headers, and sometimes a query string or request body. The response contains an HTTP status code, response headers, and sometimes a response body.

Request URI

The request URI always starts with a forward slash and then the version of the Route 53 API you use (for example, 2013-04-01). The remainder of the URI indicates the particular resource you want to act on. For example, following is the URI you use when creating a new hosted zone. (For more information, see CreateHostedZone.)

```
/2013-04-01/hostedzone
```

About the Request Time Stamp

You must provide the time stamp in either the HTTP `Date` header or the AWS `x-amz-date` header (some HTTP client libraries don't let you set the `Date` header). When an `x-amz-date` header is present, the system ignores any `Date` header when authenticating the request.

The time stamp must be within 5 minutes of the AWS system time when the request is received. If it isn't, the request fails with the `RequestExpired` error code. This is to prevent replays of your requests by an adversary.

The date must be specified in ISO 8601 format, for example, `2016-03-03T19:20:25.177Z`. For more information about ISO 8601 format, see the Wikipedia article ISO 8601.

Request Body

Many of the Route 53 API actions require you to include XML in the body of the request. The XML conforms to the Route 53 schema.

Example Request

The following example request uses a simple XML statement to create a hosted zone named example.com with the reference identifier, *myUniqueIdentifier*.

The XML elements in your request must appear in the order listed.

```
POST /2013-04-01/hostedzone HTTP/1.1
host:route53.amazonaws.com
x-amz-date:date and time of the request
authorization:AWS4-HMAC-SHA256
              Credential=AKIAIOSFODNN7EXAMPLE/date of the request in yyyymmdd format/
us-east-1/route53domains/aws4_request,
              SignedHeaders=content-length;content-type;host;user-agent;x-amz-date;x-
amz-target,
              Signature=computed signature
[Other required headers]

<?xml version="1.0" encoding="UTF-8"?>
<CreateHostedZoneRequest xmlns="https://route53.amazonaws.com/
doc/2013-04-01/">
   <Name>example.com.</Name>
   <CallerReference>myUniqueIdentifier</CallerReference>
   <HostedZoneConfig>
      <Comment>This is my hosted zone.</Comment>
   </HostedZoneConfig>
</CreateHostedZoneRequest>
```

REST Responses

Amazon Route 53 responses are just standard HTTP responses. Some of the Route 53 actions return special information specific to Route 53 in the form of an HTTP header or XML in the body of the response. The specific details are covered in the API reference topic for the particular action.

Each response contains a request ID that you can use if you need to troubleshoot a request with Route 53. The ID is contained in an HTTP header called `x-amz-request-id`. An example of a request ID is `647cd254-e0d1-44a9-af61-1d6d86ea6b77`.

The following example shows a response to a request to create a hosted zone. The `CreatedHostedZoneResponse` element contains information about the hosted zone including an Route 53 identifier, the domain that the hosted zone is associated with, and a reference description and comment. The change request itself is associated with a submittal time, an identifier and a status, shown as `PENDING`. Most importantly, the `CreatedHostedZoneResponse` includes the Route 53 name servers assigned to the hosted zone; this information is contained in the `DelegationSet` element.

Example Response

```
HTTP/1.1 201 Created
x-amz-request-id: request_id
<?xml version="1.0" encoding="UTF-8"?>
<CreateHostedZoneResponse xmlns="https://route53.amazonaws.com/doc/
2013-04-01/">
   <HostedZone>
      <Id>/hostedzone/Z1PA6795UKMFR9</Id>
      <Name>example.com.</Name>
      <CallerReference>myUniqueIdentifier</CallerReference>
      <Config>
         <Comment>This is my first hosted zone.</Comment>
      </Config>
   </HostedZone>
   <ChangeInfo>
      <Id>/change/C1PA6795UKMFR9</Id>
      <Status>PENDING</Status>
      <SubmittedAt>2010-09-10T01:36:41.958Z</SubmittedAt>
   </ChangeInfo>
   <DelegationSet>
      <NameServers>
         <NameServer>ns-2048.awsdns-64.com</NameServer>
         <NameServer>ns-2049.awsdns-65.net</NameServer>
         <NameServer>ns-2050.awsdns-66.org</NameServer>
         <NameServer>ns-2051.awsdns-67.co.uk</NameServer>
      </NameServers>
   </DelegationSet>
</CreateHostedZoneResponse>
```

Error Responses

If a REST request results in an error, the HTTP response has:

- An XML error document as the response body
- Content-Type header: text/xml
- An appropriate 3xx, 4xx, or 5xx HTTP status code

Following is an example of the XML error document in a REST error response.

```
<ErrorResponse xmlns="https://route53.amazonaws.com/doc/2013-04-01/">
   <Error>
      <Type>Sender</Type>
      <Code>InvalidInput</Code>
      <Message>The input is not valid.</Message>
   </Error>
   <RequestId>410c2a4b-e435-49c9-8382-3770d80d7d4c</RequestId>
</ErrorResponse>
```

Making API Requests for Domain Registration and for Route 53 Resolver

Topics

- RPC Requests
- RPC Responses

This section describes how to make RPC requests to two Amazon Route 53 APIs. You use one to register and manage domains, and you use the other to manage Route 53 Resolver. We'll acquaint you with the components of requests, the content of responses, and how to authenticate requests.

> **Note**
>
> To create and manage hosted zones, resource record sets, health checks, and cost allocation tags, use the applicable Route 53 API. For more information, see Making API Requests for Hosted Zones, Resource Record Sets, and Health Checks.

RPC Requests

Amazon Route 53 RPC requests are HTTPS requests, as defined by RFC 2616. This section describes the structure of an Route 53 RPC request.

For an RPC action, you send an HTTPS request to Route 53 and wait for the response. An RPC request to Route 53 contains request headers and sometimes a query string or request body. The response contains an HTTP status code, response headers, and sometimes a response body.

About the Request Time Stamp

You must provide the time stamp in either the HTTP `Date` header or the AWS `x-amz-date` header (some HTTP client libraries don't let you set the `Date` header). When an `x-amz-date` header is present, the system ignores any `Date` header when authenticating the request.

The time stamp must be within 5 minutes of the AWS system time when the request is received. If it isn't, the request fails with the `RequestExpired` error code. This helps prevent someone else from resubmitting a request.

The date must be specified in ISO 8601 format, for example, `2016-03-03T19:20:25.177Z`. For more information, see the Wikipedia article ISO 8601.

Request Body

Many Route 53 API actions require you to include JSON in the body of the request. The JSON conforms to the Route 53 schema for either domain registration or Route 53 Resolver.

Example Request

The following example request uses a simple JSON statement to determine whether the domain name `example.com` is available.

The JSON elements in your request must appear in the order listed.

```
POST / HTTP/1.1
host:route53domains.us-east-1.amazonaws.com
x-amz-date:date and time of the request
authorization:AWS4-HMAC-SHA256
              Credential=AKIAIOSFODNN7EXAMPLE/date of the request in yyyymmdd format/
us-east-1/route53domains/aws4_request,
              SignedHeaders=content-length;content-type;host;user-agent;x-amz-date;x-
amz-target,
              Signature=computed signature
x-amz-target:Route53Domains_v20140515.CheckDomainAvailability
user-agent:information about the source of the request
content-type:application/x-amz-json- 1.1
content-length:length
connections:Keep-Alive
{
   "DomainName":"example.com"
```

```
}
```

RPC Responses

Amazon Route 53 responses are just standard HTTP responses. Some of the Route 53 actions return special information specific to Route 53 in the form of an HTTP header or JSON in the body of the response. The specific details are covered in the documentation for the action.

Each response contains a request ID that you can use if you need to troubleshoot a request with Route 53. The ID is contained in an HTTP header called `x-amz-request-id`. An example of a request ID is `647cd254-e0d1-44a9-af61-1d6d86ea6b77`.

The following example shows a response to a request to check the availability of a specified domain name. The `Availability` element indicates whether the domain that you specified in the request is available.

Example Response

```
HTTP/1.1 200
Content-Length:number of characters in the JSON string
{
   "Availability":"AVAILABLE"
}
```

Error Responses

If an RPC request results in an error, the HTTP response has:

- An error document in JSON format as the response body
- Content-Type header: text/xml
- An appropriate 3xx, 4xx, or 5xx HTTP status code

Following is an example of the JSON error document in an RPC error response.

```
{
 "__type": "com.amazon.coral.service#UnrecognizedClientException",
 "message": "The security token included in the request is invalid."
}
```

Signing Amazon Route 53 API Requests

Requests must be signed using an access key ID and a secret access key. We strongly recommend that you do not use your AWS account credentials for everyday work with Route 53. You can use the credentials for a user or you can use AWS STS to generate temporary security credentials.

To sign your API requests, we recommend that you use AWS Signature Version 4. For more information, see Access Management in the *IAM User Guide*.

In addition, you might also be interested in the following topics:

- AWS Security Credentials – Provides general information about the types of credentials used for accessing AWS.
- IAM Best Practices – Presents a list of suggestions for using IAM service to help secure your AWS resources.
- Temporary Security Credentials – Describes how to create and use temporary security credentials.

Amazon Route 53 API actions by function

This topic lists all Route 53 and Route 53 Resolver API actions in groups by the function they perform.

Topics

- Types of function
- Actions by function

Types of function

DNS

- Public and private hosted zones
- Public hosted zones
- Private hosted zones
- Resource record sets
- Public DNS query logs

- Reusable delegation sets

DNS—Traffic flow

- Traffic policies
- Traffic policy instances

Domain registration

- Register, renew, and transfer domains
- Transfer domains between AWS accounts
- Get domain information
- Change domain settings

DNS—DNSSEC

- DNSSEC signing
- DNSSEC validation

DNS—IP-based routing

Health checking

- Health checks
- Health checker IP ranges

Limits (quotas) for accounts, hosted zones, and reusable delegation sets

Route 53 profiles

- Route 53 profiles
- Profile VPC associations
- Profile resource associations

Route 53 Resolver

- Route 53 Resolver endpoints
- Private DNS query logs
- Route 53 Resolver rules
- Route 53 Resolver DNS Firewall
- Amazon Route 53 Resolver on Outposts
- Resolver configuration

Tags

- Tags for hosted zones and health checks

- Tags for domains
- Tags for Route 53 Resolver

Actions by function

DNS | DNS—Traffic flow | Domain registration | DNS—DNSSEC | DNS—IP-based routing | Health checking | Limits (quotas) for accounts, hosted zones, and reusable delegation sets | Route 53 profiles | Route 53 Resolver | Tags

DNS

Public and private hosted zones

- CreateHostedZone
- DeleteHostedZone
- GetHostedZone
- GetHostedZoneChangeBatch
- GetHostedZoneCount
- ListHostedZoneChangeBatches
- ListHostedZones
- ListHostedZonesByName
- UpdateHostedZoneComment
- UpdateHostedZoneFeatures
- UpdateHostedZoneTier

Public hosted zones

- TestDNSAnswer

Private hosted zones

- AssociateVPCWithHostedZone
- DisassociateVPCFromHostedZone
- CreateVPCAssociationAuthorization
- DeleteVPCAssociationAuthorization
- ListHostedZonesByVPC
- ListVPCAssociationAuthorizations

Resource record sets

- ChangeResourceRecordSets
- GetChange
- ListResourceRecordSets
- CreateResourceRecordSetProtections
- DeleteResourceRecordSetProtections
- ListResourceRecordSetProtections
- ListResourceRecordSetChanges
- GetGeoLocation
- ListGeoLocations

Public DNS query logs

- CreateQueryLoggingConfig
- DeleteQueryLoggingConfig
- GetQueryLoggingConfig
- ListQueryLoggingConfigs

Reusable delegation sets

- CreateReusableDelegationSet
- DeleteReusableDelegationSet
- GetReusableDelegationSet
- ListReusableDelegationSets

DNS—Traffic flow

Traffic policies

- CreateTrafficPolicy
- CreateTrafficPolicyVersion
- DeleteTrafficPolicy
- GetTrafficPolicy
- ListTrafficPolicies
- ListTrafficPolicyVersions
- UpdateTrafficPolicyComment

Traffic policy instances

- CreateTrafficPolicyInstance
- DeleteTrafficPolicyInstance
- GetTrafficPolicyInstance
- GetTrafficPolicyInstanceCount
- ListTrafficPolicyInstances
- ListTrafficPolicyInstancesByHostedZone
- ListTrafficPolicyInstancesByPolicy
- UpdateTrafficPolicyInstance

Domain registration

Register, renew, and transfer domains

- RegisterDomain
- RenewDomain
- ResendContactReachabilityEmail
- RestoreDomain
- RetrieveDomainAuthCode
- TransferDomain
- DeleteDomain
- PushDomain
- ResendOperationAuthorization

Transfer domains between AWS accounts

- AcceptDomainTransferFromAnotherAwsAccount
- CancelDomainTransferToAnotherAwsAccount
- RejectDomainTransferFromAnotherAwsAccount
- TransferDomainToAnotherAwsAccount

Get domain information

- CheckDomainAvailability
- CheckDomainTransferability
- GetContactReachabilityStatus

- GetDomainDetail
- GetDomainPrices
- GetDomainSuggestions
- GetOperationDetail
- ListDomains
- ListOperations
- ListPrices
- ViewBilling

Change domain settings

- DisableDomainAutoRenew
- DisableDomainTransferLock
- EnableDomainAutoRenew
- EnableDomainTransferLock
- UpdateDomainContact
- UpdateDomainContactPrivacy
- UpdateDomainNameservers

DNS—DNSSEC

DNSSEC signing

- ActivateKeySigningKey
- CreateKeySigningKey
- DeactivateKeySigningKey
- GetDNSSEC
- AssociateDelegationSignerToDomain
- DisassociateDelegationSignerFromDomain

DNSSEC validation

- GetResolverDnssecConfig
- ListResolverDnssecConfigs
- UpdateResolverDnssecConfig

DNS—IP-based routing

- ChangeCidrCollection

- CreateCidrCollection
- DeleteCidrCollection
- ListCidrBlocks
- ListCidrCollections
- ListCidrLocations

Health checking

Health checks

- CreateHealthCheck
- DeleteHealthCheck
- GetHealthCheck
- GetHealthCheckCount
- GetHealthCheckLastFailureReason
- GetHealthCheckStatus
- ListHealthChecks
- UpdateHealthCheck

Health checker IP ranges

- GetCheckerIpRanges

Limits (quotas) for accounts, hosted zones, and reusable delegation sets

- GetAccountLimit
- GetHostedZoneLimit
- GetReusableDelegationSetLimit

Route 53 profiles

Route 53 profiles

- CreateProfile
- DeleteProfile
- GetProfile
- ListProfiles

Profile VPC associations

- AssociateProfile

- DisassociateProfile
- GetProfileAssociation
- ListProfileAssociations
- UpdateProfileAssociation

Profile resource associations

- AssociateResourceToProfile
- DisassociateResourceFromProfile
- GetProfileResourceAssociation
- ListProfileResourceAssociations
- UpdateProfileResourceAssociation

Route 53 Resolver

Route 53 Resolver endpoints

- AssociateResolverEndpointIpAddress
- CreateResolverEndpoint
- DeleteResolverEndpoint
- DisassociateResolverEndpointIpAddress
- GetResolverEndpoint
- ListResolverEndpointIpAddresses
- ListResolverEndpoints
- UpdateResolverEndpoint

DNS query logs

- AssociateResolverQueryLogConfig
- CreateResolverQueryLogConfig
- DeleteResolverQueryLogConfig
- DisassociateResolverQueryLogConfig
- GetResolverQueryLogConfig
- GetResolverQueryLogConfigAssociation
- GetResolverQueryLogConfigPolicy
- ListResolverQueryLogConfigAssociations

- ListResolverQueryLogConfigs
- PutResolverQueryLogConfigPolicy

Route 53 Resolver rules

- AssociateResolverRule
- CreateResolverRule
- DeleteResolverRule
- DisassociateResolverRule
- GetResolverRule
- GetResolverRuleAssociation
- GetResolverRulePolicy
- ListResolverRuleAssociations
- ListResolverRules
- PutResolverRulePolicy
- UpdateResolverRule

Route 53 Resolver DNS Firewall

- AssociateFirewallRuleGroup
- CreateFirewallDomainList
- CreateFirewallRule
- CreateFirewallRuleGroup
- DeleteFirewallDomainList
- DeleteFirewallRule
- DeleteFirewallRuleGroup
- DisassociateFirewallRuleGroup
- GetFirewallConfig
- GetFirewallDomainList
- GetFirewallRuleGroup
- GetFirewallRuleGroupAssociation
- GetFirewallRuleGroupPolicy
- ImportFirewallDomains

- ListFirewallConfigs
- ListFirewallDomainLists
- ListFirewallDomains
- ListFirewallRuleGroupAssociations
- ListFirewallRuleGroups
- ListFirewallRules
- PutFirewallRuleGroupPolicy
- UpdateFirewallConfig
- UpdateFirewallDomains
- UpdateFirewallRule
- UpdateFirewallRuleGroupAssociation

Route 53 Resolver on Outposts

- CreateOutpostResolver
- DeleteOutpostResolver
- GetOutpostResolver
- ListOutpostResolvers
- UpdateOutpostResolver

VPC Resolver configuration

- GetResolverConfig
- ListResolverConfigs
- UpdateResolverConfig

Tags

Tags for hosted zones and health checks

- ChangeTagsForResource
- ListTagsForResource
- ListTagsForResources

Tags for domains

- DeleteTagsForDomain
- ListTagsForDomain
- UpdateTagsForDomain

Tags for Route 53 Resolver

- ListTagsForResource
- TagResource
- UntagResource

Throttling for Amazon Route 53 API requests

> **Important**
>
> Amazon Route 53 updated its API throttling behavior. The update includes increasing the requests per second limit and introducing change-based throttling. This page describes the updated limits in detail.

Amazon Route 53 throttles API requests on a per-account basis to maintain service stability and ensure fair usage for all customers. Route 53 applies two independent limits:

- **Request rate:** the number of API requests per second.
- **Change throughput:** the number of individual DNS record changes per second, aggregated across the API actions that modify DNS data.

A request can be throttled by either limit. When a request is throttled, Amazon Route 53 returns an HTTP 400 error (`Bad request`). The response header also includes a `Code` element with a value of `Throttling` and a `Message` element with a value of `Rate exceeded`.

How throttling is applied

Amazon Route 53 uses a token bucket algorithm. Each limit has a bucket that holds a maximum number of tokens. Each request (for the request rate limit) or each change (for the change throughput limit) removes tokens from the applicable bucket. The bucket refills at a fixed rate every second, up to its maximum capacity. If refill tokens arrive when the bucket is already full, Route 53 discards them.

Two values describe each bucket:

- **Bucket maximum capacity** is your burst: the number of requests or changes Route 53 can absorb at once when the bucket is full.

- **Bucket refill rate** is your sustained rate: the number of requests or changes per second you can maintain indefinitely.

You can use refill tokens as they are added; you do not need to wait for the bucket to refill completely.

Request rate token bucket sizes and refill rates

Route 53 applies request rate throttling at two levels:

- **Account level:** all Amazon Route 53 API requests from your account draw from one bucket.
- **Account and operation level:** each API action also has its own bucket.

A request consumes a token from both buckets and is throttled if either bucket is empty. The following actions have different default request rate limits.

Request rate limits per API action

API action	Bucket maximum capacity	Bucket refill rate
All Amazon Route 53 API actions combined (account level)	50	10
Any single API action not listed below (per-action default)	50	10
`AssociateVPCWithHostedZone`	20	5
`ChangeCidrCollection`	40	5
`CreateCidrCollection`	40	5
`CreateHealthCheck`	50	0.5
`CreateHostedZone`	40	2
`CreateReusableDelegationSet`	40	2
`CreateTrafficPolicyInstance`	1	1

API action	Bucket maximum capacity	Bucket refill rate
DeleteCidrCollection	40	5
DeleteHealthCheck	15	3
DeleteHostedZone	40	5
DeleteReusableDelegationSet	40	5
DeleteTrafficPolicyInstance	1	1
DisassociateVPCFromHostedZone	10	5
GetHealthCheckLastFailureReason	4	1
GetHealthCheckStatus	4	1
UpdateHealthCheck	50	5
UpdateTrafficPolicyInstance	1	1

For the complete list of Amazon Route 53 API actions, see Actions in the *Amazon Route 53 API Reference*.

`CreateHealthCheck` requests

You can submit one `CreateHealthCheck` request every 2 seconds per AWS account. This corresponds to the refill rate of 0.5 requests per second shown in the preceding table.

Change throughput limiting

In addition to the request rate limit, the API actions that modify DNS data are subject to a change throughput limit. This limit uses a separate token bucket that depletes based on the number of DNS record changes a request makes, not the number of requests. Change throughput is limited per AWS account, not per API action. All of the following actions draw from one bucket, which has a maximum capacity of 1,500 changes (burst) and refills at 100 changes per second (sustained).

Changes are counted as follows:

Tokens consumed per operation

Operation	Tokens consumed
`ChangeResourceRecordSets`	1 per `CREATE`, 1 per `DELETE`, 2 per `UPSERT`
`AssociateVPCWithHostedZone`	2
`DisassociateVPCFromHostedZone`	2
`CreateHostedZone`	2
`DeleteHostedZone`	2

All other Amazon Route 53 API actions consume no change throughput tokens and are limited only by the request rate.

Example: You can submit a single request containing 1,000 changes (a burst), which consumes 1,000 tokens. After the burst, the bucket refills at 100 tokens per second. If you continue submitting 100 changes per second, you can sustain that rate indefinitely. If you attempt to sustain 500 changes per second, the bucket depletes and subsequent requests are throttled until it refills.

Monitor API throttling

You can monitor your Amazon Route 53 API usage by watching for HTTP 400 responses in your application logs, or by tracking API usage through CloudWatch metrics. When you receive throttling errors, your requests are exceeding one of the limits described in the preceding sections.

Retries and exponential backoff

When you poll or retry an API request, we recommend using an exponential backoff algorithm to calculate the sleep interval between requests. Exponential backoff uses progressively longer waits between retries for consecutive error responses. Implement a maximum delay interval and a maximum number of retries, and consider adding jitter (randomized delay) to prevent successive collisions. For more information, see Timeouts, retries, and backoff with jitter in the AWS Builders' Library.

Each AWS SDK implements automatic retry logic, including an adaptive retry mode that adjusts the client-side request rate in response to throttling. For workloads that regularly approach these

limits, consider enabling adaptive retries. For more information, see Retry behavior in the *AWS SDKs and Tools Reference Guide*.

Requesting a limit increase

You can request an increase to either the API request rate or the change throughput limit through AWS Support. To request an increase:

- Open the AWS Support Center.
- Create a case and choose **Service limit increase**.
- For **Limit type**, choose **Route 53**.
- Provide your current usage and the limit you need.

Best practices for API throttling

- **Balance request rate against batch size:** If you are throttled by the request rate limit, send more changes per request. If you are throttled by the change throughput limit, reduce your total change rate. Neither very small nor very large batches are optimal on their own.
- **Use batching for atomicity:** All changes in a single `ChangeResourceRecordSets` request are applied atomically, so they succeed or fail together.
- **Spread changes over time:** Distribute changes evenly across seconds instead of submitting large batches simultaneously.
- **Rely on burst for occasional spikes, not sustained throughput:** Burst capacity accommodates legitimate traffic spikes; it is not a sustained operating ceiling.
- **Retry with exponential backoff:** When you receive HTTP 400 responses, retry after a delay that increases with each attempt (see the section called “Retries and exponential backoff”).
- **Request limit increases proactively:** If you anticipate workload growth, request an increase before you hit the limit (see the section called “Requesting a limit increase”).

For broader Amazon Route 53 guidance, see *Best practices*.

Quotas

Amazon Route 53 API requests and entities are subject to the following quotas (formerly called "limits").

Topics

- Using Service Quotas to view and manage quotas
- Quotas on entities
- Maximums on API requests

Using Service Quotas to view and manage quotas

You can use the Service Quotas service to view quotas and to request quota increases for many AWS services. For more information, see the Service Quotas User Guide. (You can use Service Quotas to view and manage domains, Route 53, and Route 53 VPC Resolver quotas.)

> **Note**
>
> To view quotas and request higher quotas for Route 53, you must change the Region to US East (N. Virginia). To view quotas and request higher quotas for VPC Resolver, change to the applicable Region.

Quotas on entities

Amazon Route 53 entities are subject to the following quotas.

For information on getting current quotas (formerly called "limits"), see the following Route 53 actions:

- GetAccountLimit – Gets quotas on health checks, health check sources, hosted zones, reusable delegation sets, traffic flow policies, and traffic flow policy records
- GetHostedZoneLimit – Gets quotas on records in a hosted zone and on Amazon VPCs that you can associate with a private hosted zone
- GetReusableDelegationSetLimit – Gets the quota on the number of hosted zones that you can associate with a reusable delegation set

Topics

- Quotas on domains
- Quotas on hosted zones
- Quotas on records
- Quotas on Route 53 VPC Resolver
- Quotas on health checks
- Quotas on query log configurations
- Quotas on traffic flow policies and policy records
- Quotas on reusable delegation sets
- Quotas on Route 53 Profiles

Quotas on domains

Domain quotas

Entity	Quota
Domains	20* per AWS account Request a higher quota.

*The limit is 20 for new customers as of March 2021.

If you have an existing account and your default limit is 50 now, it will remain at 50.

Quotas on hosted zones

Hosted zone quotas

Entity	Quota
Hosted zones	Initial quota of 500 per AWS account, but you can request a higher quota as needed. Request a higher quota.

Entity	Quota
Hosted zones that can use the same reusable delegation set	100 Request a higher quota.
Amazon VPCs that you can associate with a private hosted zone per hosted zone	300 If you want more than 300 associations, we recommend you use Route 53 Profiles. For more information, see What are Amazon Route 53 Profiles?.
Private hosted zones that you can associate a VPC with	No quota *
Authorizations that you can create so you can associate VPCs that were created by one account with a hosted zone that was created by another account	1000
The number of key signing keys (KSK) that you can create per hosted zone	2

* You can associate a VPC with any or all of the private hosted zones that you control through your AWS accounts. For example, suppose you have three AWS accounts and all three have the default quota of 500 hosted zones. If you create 500 private hosted zones for all three accounts, you can link a VPC with all 1,500 private hosted zones.

Quotas on records

Record quotas

Entity	Quota

Entity	Quota
Records	10,000 per hosted zone Request a higher quota. For a quota greater than 10,000 records in a hosted zone, an additional charge applies.For more information, see Amazon Route 53 Pricing.
Records in a record set	400 per record set
Geolocation, latency, multivalue answer, weighted, and IP-based records	100 records that have the same name and type
Geoproximity records	30 records that have the same name and type
Record set protections	10,000 per hosted zone To request a higher quota, contact AWS Support. Protections can exist on names that have no records, so this quota is separate from the quota on records.
CIDR collections	5 per AWS account. Request a higher quota.
CIDR blocks	1000 per CIDR collection. Request a higher quota.

Quotas on Route 53 VPC Resolver

This section includes all the Route 53 VPC Resolver quotas

Quotas on Route 53 VPC Resolver

Use the following procedure to increase quotas for Route 53 VPC Resolver.

To increase Resolver quotas

1. Open the Service Quotas console at https://console.aws.amazon.com/servicequotas/home/services/route53resolver/quotas.
2. Go to the Region where you want to increase the limit.
3. Select the Route 53 VPC Resolver **Quota name** you want to increase.
4. Select **Request quota increase**, enter the quota value, and then select **Request**.

Quotas on Route 53 VPC Resolver endpoints

Resolver endpoint quotas

Entity	Quota
Endpoints per AWS Region	4 per AWS account Request a higher quota.
IP addresses per endpoint	6 Request a higher quota.
IP addresses per rule	6
Rules per AWS Region	1000 per AWS account Request a higher quota.
Associations between rules and VPCs per AWS Region	2000 per AWS account Request a higher quota.

Entity	Quota
UDP Queries per second per IP address in an endpoint	10,000*

* Each IP address in an endpoint can process up to 10,000 UDP DNS queries per second (QPS). The actual DNS QPS varies by the type of query, size of the response, health of the target name servers, query response times, round trip latency, and the protocol in use. For example, queries to a target name server that is slow to respond can reduce the capacity of the network interface. Also, to ensure high availability, Route 53 Resolver sends redundant outbound queries for each DNS request it receives. As a result, the QPS for each outbound network interface will not match the QPS sent to Route 53 VPC Resolver. Use CloudWatch metrics to measure how many queries are sent to each network interface. For more information, see Metrics for Route 53 VPC Resolver IP addresses. If your maximum query rate exceeds 50% of the capacity for any network interface in the endpoint, you can add more network interfaces to increase the endpoint capacity.

Connections made through services like Network Load Balancer and AWS Lambda (for a full list see Automatically tracked connections) are tracked by default, even if the security group setup does not require tracking.

If connection tracking is enforced by restrictive security group rules or queries are routed through Network Load Balancer, the maximum queries per second per IP address for an inbound endpoint can be as low as 1500.

Quotas on Route 53 VPC Resolver query logs

Resolver query log quotas

Entity	Quota
Query log configurations per AWS Region	20 Request a higher quota.
Query log configuration VPC associations per AWS Region	100 * Request a higher quota.

Entity	Quota
Query log configuration VPC associations per account per AWS Region (including query long confgurations shared using RAM) for the account that the configuration was shared to.	100 Request a higher quota.

* This is a Regional limit that applies across all VPC Resolver query log configurations in a single Region. Creating more query log configurations in the same Region does not provide more VPC association capacity.

Quotas on Resolver DNS Firewall

DNS Firewall quotas

Entity	Quota
Number of rule groups associated to a VPC for a single account per AWS Region	5
Number of DNS Firewall domains in a single Amazon S3 file for a single account per AWS Region	250,000 Request a higher quota.
Number of DNS Firewall rule groups for a single account per AWS Region	1,000 Request a higher quota.
Number of rules within a rule group for a single account per AWS Region	100 Request a higher quota.
Number of domain lists for a single account per AWS Region	1000 Request a higher quota.

Entity	Quota
The maximum number of domains that you can specify across all of the domain lists for a single account per AWS Region	100,000 Request a higher quota.

Quotas for Resolver on first-generation AWS Outposts

Resolver on Outposts quotas

Entity	Quota
Resolver on Outpost instance limit	6 (with a minimum of 4 required)

Resolver on Outpost instance types and the number of DNS queries per second each instance type can accommodate:

Queries per second by instance type

Instance type	Queries per second
c5.large	Up to 7,000
c5.xlarge	Up to 12,000
c5.2xlarge	Up to 24,000
c5.4xlarge	Up to 56,000
c5d.large	Up to 7,000
c5d.xlarge	Up to 12,000

Instance type	Queries per second
c5d.2xlarge	Up to 24,000
c5d.4xlarge	Up to 56,000
m5.large	Up to 7,000
m5.xlarge	Up to 12,000
m5.2xlarge	Up to 24,000
m5.4xlarge	Up to 56,000
m5d.large	Up to 7,000
m5d.xlarge	Up to 12,000
m5d.2xlarge	Up to 24,000
m5d.4xlarge	Up to 56,000
r5.large	Up to 7,000
r5.xlarge	Up to 12,000
r5.2xlarge	Up to 24,000
r5.4xlarge	Up to 56,000
r5d.large	Up to 7,000

Instance type	Queries per second
r5d.xlarge	Up to 12,000
r5d.2xlarge	Up to 24,000
r5d.4xlarge	Up to 56,000

Resolver on Outpost endpoint instance types and the number of DNS queries per second each instance type can accommodate:

Queries per second by endpoint instance type

Instance type	Queries per second
c5.large	Up to 5,000
c5.xlarge	Up to 10,000
c5.2xlarge	Up to 18,000
c5.4xlarge	Up to 30,000
c5d.large	Up to 5,000
c5d.xlarge	Up to 10,000
c5d.2xlarge	Up to 18,000
c5d.4xlarge	Up to 30,000
m5.large	Up to 5,000

Instance type	Queries per second
m5.xlarge	Up to 10,000
m5.2xlarge	Up to 18,000
m5.4xlarge	Up to 30,000
m5d.large	Up to 5,000
m5d.xlarge	Up to 10,000
m5d.2xlarge	Up to 18,000
m5d.4xlarge	Up to 30,000
r5.large	Up to 5,000
r5.xlarge	Up to 10,000
r5.2xlarge	Up to 18,000
r5.4xlarge	Up to 30,000
r5d.large	Up to 5,000
r5d.xlarge	Up to 10,000
r5d.2xlarge	Up to 18,000
r5d.4xlarge	Up to 30,000

Quotas for Resolver on second-generation AWS Outposts

The following table lists the quota for Resolver on second-generation AWS Outposts.

Resolver on second-generation AWS Outposts quotas

Quota name	Quota value
DNS queries per second	Up to 75,000

Quotas on health checks

Health check quotas

Entity	Quota
Health checks	200 active health checks per AWS account Request a higher quota.
Health check sources	5 health check sources per AWS account Request a higher quota.
Health checks per health check source	200
Subnets per health check source	Exactly 3, in 3 Availability Zones. In US West (N. Californi a), exactly 2, in 2 Availability Zones.
Security groups per subnet	1 to 10
Child health checks that a calculated health check can monitor	255
	16,384 bytes (16K)

Entity	Quota
Maximum total length of headers in the response to a health check request	

Quotas on query log configurations

Query log configuration quotas

Entity	Quota
Query log configurations	1 per hosted zone

Quotas on traffic flow policies and policy records

Traffic flow quotas

Entity	Quota
Traffic policies For more information about Route 53 traffic flow, see Using Traffic Flow to route DNS traffic.	50 per AWS account Request a higher quota.
Traffic policy versions	1,000 per traffic policy
Traffic policy records (referred to as "policy instances" in the Route 53 API, AWS SDKs, AWS Command Line Interface, and AWS Tools for Windows PowerShell)	5 per AWS account Request a higher quota.

Quotas on reusable delegation sets

Reusable delegation set quotas

Entity	Quota
Reusable delegation sets	100 per AWS account Request a higher quota.

Quotas on Route 53 Profiles

Route 53 Profile quotas

Entity	Quota
Number of Route 53 Profiles per AWS account in a Region	5 Request a higher quota.
Number of VPCs that can be associated to a Profile	1000 Request a higher quota.
Number of Route 53 Profiles that can be associated to a VPC	2
Number of DNS Firewall rule groups per Profile	5
Number of Resolver rules per Profile	1000 Request a higher quota.
Number of private hosted zones per a Profile	5000

Entity	Quota
	Request a higher quota.
Number of VPC Resolver query logging configurations per Profile	2

Maximums on API requests

Amazon Route 53 API requests are subject to the following maximums.

Topics

- Number of elements and characters in ChangeResourceRecordSets requests
- Number of elements in record set protection requests
- Frequency of Amazon Route 53 API requests
- Frequency of Route 53 VPC Resolver API requests

Number of elements and characters in `ChangeResourceRecordSets` requests

`ResourceRecord` elements

A request cannot contain more than 1,000 `ResourceRecord` elements (including alias records). When the value of the `Action` element is UPSERT, each `ResourceRecord` element is counted twice.

Maximum number of characters

The sum of the number of characters (including spaces) in all `Value` elements in a request cannot exceed 32,000 characters. When the value of the `Action` element is UPSERT, each character in a `Value` element is counted twice.

Number of elements in record set protection requests

The following limits apply to the number of protections in record set protection requests and responses:

`CreateResourceRecordSetProtections` and `DeleteResourceRecordSetProtections` requests

A request must contain at least 1 and at most 100 protections.

`ListResourceRecordSetProtections` responses

A response contains at most 300 protections per page.

Frequency of Amazon Route 53 API requests

All Amazon Route 53 API requests

Amazon Route 53 throttles API requests on a per-account basis to maintain service stability and ensure fair usage for all customers. Route 53 applies two independent limits:

- **Request rate:** the number of API requests per second.
- **Change throughput:** the number of individual DNS record changes per second, aggregated across the API actions that modify DNS data.

A request can be throttled by either limit. When a request is throttled, Amazon Route 53 returns an HTTP 400 error (`Bad request`). The response header also includes a `Code` element with a value of `Throttling` and a `Message` element with a value of `Rate exceeded`.

For the limits that apply to each API action, bucket sizes and refill rates, and requesting an increase, see the section called “API throttling”.

> **Note**
>
> If your application exceeds this limit, we recommend that you implement exponential backoff for retries. For more information, see Error Retries and Exponential Backoff in AWS in the *Amazon Web Services General Reference*.

`ChangeResourceRecordSets` requests

If Route 53 can't process a request before the next request arrives, it will reject subsequent requests for the same hosted zone and return an HTTP 400 error (`Bad request`). The response header also includes a `Code` element with a value of `PriorRequestNotComplete` and a `Message` element with a value of `The request was rejected because Route 53 was still processing a prior request.`

Frequency of Route 53 VPC Resolver API requests

All requests

Five requests per second per AWS account per Region. If you submit more than five requests per second in a Region, VPC Resolver returns an HTTP 400 error (`Bad request`). The response header also includes a `Code` element with a value of `Throttling` and a `Message` element with a value of `Rate exceeded`.

> **Note**
>
> If your application exceeds this limit, we recommend that you implement exponential backoff for retries. For more information, see Error Retries and Exponential Backoff in AWS in the *Amazon Web Services General Reference*.

Related information

The following related resources can help you as you work with this service.

Topics

- AWS resources
- Third-party tools and libraries
- Graphical user interfaces

AWS resources

Several helpful guides, forums, and other resources are available from Amazon Web Services.

- Amazon Route 53 API Reference – A reference guide that includes the schema location; complete descriptions of the API actions, parameters, and data types; and a list of errors that the service returns.
- **AWS::Route53::RecordSet Type in the *AWS CloudFormation User Guide*** – A property for using Amazon Route 53 with CloudFormation to create customized DNS names for your CloudFormation stacks.
- **Discussion Forums** – A community-based forum for developers to discuss technical questions related to Route 53.
- AWS Support Center – This site brings together information about your recent support cases and results from AWS Trusted Advisor and health checks, as well as providing links to discussion forums, technical FAQs, the service health dashboard, and information about AWS support plans.
- AWS Premium Support Information – The primary web page for information about AWS Premium Support, a one-on-one, fast-response support channel to help you build and run applications on AWS Infrastructure Services.
- Contact Us – Links for inquiring about your billing or account. For technical questions, use the discussion forums or support links above.
- **Route 53 product information** – The primary web page for information about Route 53, including features, pricing, and more.

- Classes & Workshops – Links to role-based and specialty courses, in addition to self-paced labs to help sharpen your AWS skills and gain practical experience.

- AWS Developer Center – Explore tutorials, download tools, and learn about AWS developer events.
- AWS Developer Tools – Links to developer tools, SDKs, IDE toolkits, and command line tools for developing and managing AWS applications.
- Getting Started Resource Center – Learn how to set up your AWS account, join the AWS community, and launch your first application.
- Hands-On Tutorials – Follow step-by-step tutorials to launch your first application on AWS.
- AWS Whitepapers – Links to a comprehensive list of technical AWS whitepapers, covering topics such as architecture, security, and economics and authored by AWS Solutions Architects or other technical experts.
- AWS Support Center – The hub for creating and managing your AWS Support cases. Also includes links to other helpful resources, such as forums, technical FAQs, service health status, and AWS Trusted Advisor.
- Support – The primary webpage for information about Support, a one-on-one, fast-response support channel to help you build and run applications in the cloud.
- Contact Us – A central contact point for inquiries concerning AWS billing, account, events, abuse, and other issues.
- AWS Site Terms – Detailed information about our copyright and trademark; your account, license, and site access; and other topics.

Third-party tools and libraries

In addition to AWS resources, you can find a variety of third-party tools and libraries that work with Amazon Route 53.

- **AmazonRoute53AppsScript (via webos-goodies)**

  Google spreadsheet management of Amazon Route 53.

- **AWS Component for .NET (via SprightlySoft)**

  SprightlySoft .NET Component for Amazon Web Services with support for REST operations and Route 53.

- **Boto API download (via github)**

  Boto Python interface to Amazon Web Services.

- **cli53 (via github)**

  Command line interface for Route 53.

- **Dasein Cloud API**

  Java-based API.

- **R53.py (via github)**

  Maintains a canonical version of your DNS configurations under source control, and calculates the minimum set of changes that are required to change a configuration.

- **route53d**

  DNS front-end to Route 53 API (enables incremental zone transfer (IXFR)).

- **Route53Manager (via github)**

  Web-based interface.

- **Ruby Fog (via github)**

  The Ruby cloud services library.

- **WebService::Amazon::Route53 (via CPAN)**

  Perl interface to Amazon Route 53 API.

Graphical user interfaces

The following third-party tools provide graphical user interfaces (GUIs) for working with Amazon Route 53:

- R53 Fox
- Ylastic

FAQs on supported protocols

The Amazon Route 53 Domains service is adding SDK support for the Smithy RPC v2 CBOR protocol in addition to the existing AWS JSON protocol. The Smithy RPC v2 CBOR protocol is more performant than AWS JSON for most workloads. AWS SDKs will prioritize the protocol that is the most performant for each language.

What are the AWS JSON and Smithy RPC v2 CBOR protocols, and how do they differ?

A wire protocol acts as a mediator between a client and the service. The AWS SDK handles the serialization (converting an object to the wire format) and deserialization (converting the wire format back to an object) between the client and server transparently for both protocols.

AWS JSON - JSON is one of the most widely used and accepted data formats for communication between systems. This service uses AWS JSON as a medium to communicate between an AWS SDK client and the server. An HTTP request for a service API operation accepts input in the form of JSON, the operation is executed, and the response is returned to the SDK client in the form of JSON.

Smithy RPC v2 CBOR - Smithy RPC v2 CBOR (Concise Binary Object Representation) is a protocol developed by Amazon for more efficient data transmission. It is a data format designed to keep message sizes compact and to allow the format to evolve without requiring version negotiation between systems. The key difference from AWS JSON is that AWS JSON uses human-readable text (like `{"name": "John"}`), while Smithy RPC v2 CBOR uses a binary format (sequences of 1s and 0s). This binary approach creates smaller data packages that travel faster over the network and reduces serialization time since computers process binary data directly.

How do I get started with the Smithy RPC v2 CBOR protocol?

To get started, use the latest AWS SDK/CLI version to use the Smithy RPC v2 CBOR protocol for this service. The SDK will automatically use the Smithy RPC v2 CBOR protocol when available. Upgrade your AWS SDK to the specified version or any subsequent version.

Starting 2026-11-10, the following AWS SDKs will use the Smithy RPC v2 CBOR protocol, while the other SDKs will continue using the AWS JSON protocol:

- AWS SDK for C++
- AWS SDK for Go V2
- AWS SDK for Java 2.x
- AWS SDK for .NET v4
- AWS Tools for PowerShell v5
- AWS SDK for Swift
- AWS SDK for Kotlin

What are the risks of enabling the Smithy RPC v2 CBOR protocol?

The added support of Smithy RPC v2 CBOR in the SDKs is expected to be fully transparent and effortless for customers. However, if you are using a custom implementation of the AWS SDK, or a combination of custom clients and the AWS SDK, that depends on AWS JSON-specific serialization behavior, it may be incompatible with the Smithy RPC v2 CBOR protocol.

What if my solution does not support Smithy RPC v2 CBOR?

To use the AWS JSON protocol, downgrade your SDK to a version released before 2026-11-10.

What AWS Regions support the Smithy RPC v2 CBOR protocol?

The Smithy RPC v2 CBOR protocol is supported in all AWS Regions where this service is available.

What latency improvements can I expect when upgrading to SDK versions that support Smithy RPC v2 CBOR?

Smithy RPC v2 CBOR provides up to 13% lower latency and up to 15% reduction in payload size compared to AWS JSON. The binary encoding eliminates the overhead of JSON text parsing and string escaping, which reduces both network transfer time and serialization cost.

Will the AWS JSON protocol be deprecated?

AWS JSON protocol will continue to be supported through older versions of SDKs. Also, some language-specific SDKs will continue using AWS JSON as it is more performant than Smithy RPC

v2 CBOR due to the available JSON implementation for the language being faster than its CBOR implementation.

Where can I find more information about Smithy RPC v2 CBOR?

You can find more information about the AWS JSON protocol at AWS JSON 1.0 protocol and AWS JSON 1.1 protocol in the Smithy documentation, and about Smithy RPC v2 CBOR at Smithy RPC v2 CBOR protocol in the Smithy documentation.

Document history

The following entries describe important changes in each release of the Route 53 documentation. For notification about updates to this documentation, you can subscribe to an RSS feed.

Topics

- 2026 releases
- 2025 releases
- 2024 releases
- 2023 releases
- 2022 releases
- 2021 releases
- 2020 releases
- 2018 releases
- 2017 releases
- 2016 releases
- 2015 releases
- 2014 releases
- 2013 releases
- 2012 release
- 2011 releases
- 2010 release

2026 releases

December 1, 2026

You can now connect a global resolver to an AWS Cloud WAN (Cloud WAN) core network with a core network endpoint, so that the core network's segments can be used as access sources for DNS resolution. For more information about core network endpoints, see Connecting a global resolver to a core network.

November 20, 2026

Route 53 now checks changes to the records in a hosted zone with six Change Guardrails, which reject a change that would break DNS resolution for part of your domain or delete a record or delegation that you still use. For more information, see Change Guardrails for hosted zones.

November 20, 2026

You can now review who changed the records in an advanced hosted zone, what they changed, and when. Use the new `ListHostedZoneChangeBatches`, `GetHostedZoneChangeBatch`, and `ListResourceRecordSetChanges` APIs or the Route 53 console. Route 53 retains each change for 365 days. For more information, see Viewing the change history of a hosted zone.

November 20, 2026

You can now choose a tier for each public or private hosted zone. The `ADVANCED` tier adds Change History, the Change Guardrails that check record deletions, and public DNS fallback for private hosted zones. The `STANDARD` tier remains the default, at no additional cost. For more information, see Hosted zone tiers.

November 20, 2026

You can now enable public DNS fallback on a private hosted zone that uses the `ADVANCED` tier. When a query from your VPC asks for a name that doesn't exist in the zone, VPC Resolver retries the query against public DNS instead of returning NXDOMAIN. For more information, see Public DNS fallback for private hosted zones.

November 20, 2026

You can now use AZ-affinity routing to route DNS queries from your VPCs to resources in the same Availability Zone, Zone Group, or AWS Region that the query comes from. If a resource is unhealthy, Route 53 falls back to a less specific location, down to a Global record. For more information, see AZ-affinity routing.

November 17, 2026

You can now subscribe to Infoblox DNS threat protection through AWS Marketplace and use Infoblox threat feeds in DNS Firewall rules. For more information, see Using partner managed DNS threat protection with Infoblox in DNS Firewall.

November 16, 2026

You can now use Automated Allowlists in Resolver DNS Firewall to build an allowlist of domains from the DNS query traffic of your VPCs, so that you can block all other domains. For more information, see Automated Allowlists.

November 12, 2026

You can now protect records in a hosted zone from changes. A record set protection blocks every `ChangeResourceRecordSets` change to the records that have a specific name and type until you delete the protection. Use the new `CreateResourceRecordSetProtections`, `DeleteResourceRecordSetProtections`, and `ListResourceRecordSetProtections` APIs or the Route 53 console. For more information, see Protecting records from changes.

November 9, 2026

You can associate more than one Amazon Route 53 Profile to a VPC. Each association has a priority value from 100 to 9900, and VPC Resolver evaluates the associated Profiles in priority order, lowest value first. You set the priority when you associate a Profile to a VPC, and you can change it later. For more information, see What are Amazon Route 53 Profiles?.

November 9, 2026

`AmazonRoute53ProfilesFullAccess` adds permission for `UpdateProfileAssociation`, so principals can change the name or priority of a Profile association with a VPC. For more information, see AWS managed policy: AmazonRoute53ProfilesFullAccess.

October 21, 2026

You can now use AWS Network Firewall resolved IP validation with Resolver DNS Firewall to check that a connection's destination IP address was returned by the VPC Resolver. For more information, see Resolved IP validation with Network Firewall.

October 8, 2026

`AmazonRoute53FullAccess` adds permission for `iam:CreateServiceLinkedRole`, limited to the `route53healthchecks.amazonaws.com` service. This permission lets principals create the service-linked role for health checking when the role doesn't exist in the account yet. For more information, see AWS managed policy: AmazonRoute53FullAccess.

October 1, 2026

You can now monitor Route 53 Global Resolver DNS query logs with Amazon CloudWatch. Create metric filters and alarms for blocked and failed DNS queries, analyze high-impact

contributors with CloudWatch Contributor Insights, and interactively query your logs with CloudWatch Logs Insights. For more information, see Monitoring Route 53 Global Resolver with Amazon CloudWatch.

April 14, 2026

Route 53 VPC Resolver IPv6 and dual-stack outbound endpoints now support IPv6 connectivity to public DNS resolvers through an internet gateway. Enable IPv6 internet access when you create or edit an outbound endpoint to forward DNS queries to public IPv6 targets. For more information, see Values that you specify when you create or edit outbound endpoints.

April 14, 2026

Route 53 VPC Resolver inbound endpoints now support DNS64. When you enable DNS64 on an inbound endpoint, VPC Resolver synthesizes AAAA records for IPv4-only services, enabling IPv6-only clients on your on-premises network to communicate with IPv4-only AWS resources through NAT64. For more information, see Forwarding inbound DNS queries to your VPCs.

2025 releases

December 15, 2025

Added detailed CloudWatch metrics for Route 53 Resolver endpoints. The new metrics provide enhanced monitoring capabilities at the Resolver Network Interface and Target Name Server levels, including P90 response latency of DNS queries, tracking of SERVFAIL, NXDOMAIN, REFUSED, and FORMERR response statuses, and availability monitoring of target name servers for outbound Resolver endpoints. For more information, see CloudWatch Detailed Metrics for Route 53 VPC Resolver.

June 26, 2025

Added `ResolverEndpointCapacityStatus` to the `AWS/Route53Resolver` namespace. For more information, see Metrics for Route 53 VPC Resolver endpoints.

June 25, 2025

Amazon Route 53 now supports alias records for VPC Lattice service custom domain endpoints. For more information, see Routing traffic to Amazon VPC Lattice service domain endpoint.

June 24, 2025

You can now use inbound and outbound delegation with VPC Resolver. For more information, see How DNS resolvers on your network forward DNS queries to Resolver endpoints.

June 01, 2025

Added permissions for `cloudwatch:GetMetricData` ,`tag:GetResources`, `es:ListDomainNames`, `es:DescribeDomains`, `cloudfront:GetDistributionTenantByDomain`, `cloudfront:GetConnectionGroup` and `lightsail:GetContainerServices`. These permissions enable you to fetch up to 500 CloudWatch health check metrics, up to 100 names of health checks, get the domain configuration for the specified Amazon OpenSearch Service domains, and list the names of all Amazon OpenSearch Service domains owned by the current user in the active Region, fetch the CloudFront multi-tenant distributions and get the Lightsail container services. For more information, see AWS managed policy: AmazonRoute53FullAccess.

April 28, 2025

You can now associate interface VPC endpoints to Route 53 Profiles. For more information, see Associate interface VPC endpoints to a Route 53 Profile.

April 28, 2025

You can now add an alias record to a CloudFront distribution tenant. For more information, see Routing traffic to an Amazon CloudFront distribution by using your domain name.

February 27, 2025

Updated the Route 53 guide with the new console experience for Traffic Flow. For more information, see Creating and managing traffic policies and Creating and managing policy records.

January 14, 2025

Amazon Route 53 now supports alias records for OpenSearch Service custom domain endpoints. For more information, see Routing traffic to Amazon OpenSearch Service domain endpoint.

January 13, 2025

Added Resolver DNS Firewall findings to the Security Hub CSPM. For more information, see Sending findings from Resolver DNS Firewall to Security Hub CSPM.

2024 releases

November 15, 2024

Added Resolver DNS Firewall Advanced, a new set of features on Resolver DNS Firewall that allows you to identify and block DNS traffic associated with advanced DNS threats, such as DNS tunneling, Domain Generation Algorithm (DGA), and Dictionary DGA based threats. For more information, see DNS Firewall Advanced Rules.

October 29, 2024

Added support for HTTPS, SSHFP, SVCB, and TLSA DNS record types. For more information, see Supported DNS record types.

October 3, 2024

Added support for Service Name Indication (SNI) for DoH outbound Resolver endpoints. For more information, see Values that you specify when you create or edit rules.

September 3, 2024

You can now use `route53:VPCs` policy condition to grant fine-grained access to managing hosted zone associations to VPCs. For more information, see Using IAM policy conditions for fine-grained access control.

August 27, 2024

`AmazonRoute53ProfilesFullAccess` added permissions for `GetProfilePolicy` and `PutProfilePolicy`. These are permission-only IAM actions. If an IAM principal doesn't have these permissions granted, an error will occur when attempting to share the Profile using the AWS RAM service. For more information, see AWS managed policy: AmazonRoute53ProfilesFullAccess.

August 27, 2024

`AmazonRoute53ProfilesReadOnlyAccess` added permission for `GetProfilePolicy`. This is a permission-only IAM action. If an IAM principal doesn't have this permission granted, an error will occur when attempting to access the Profile's policy using the AWS RAM service. For more information, see AWS managed policy: AmazonRoute53ProfilesReadOnlyAccess.

August 5, 2024

Added a statement id (Sid) to uniquely identity the managed policy `AmazonRoute53ResolverFullAccess`. For more information, see AWS managed policy: AmazonRoute53ResolverFullAccess.

August 5, 2024

Added a statement id (Sid) to uniquely identity the managed policy `AmazonRoute53ResolverReadOnlyAccess`. For more information, see AWS managed policy: AmazonRoute53ResolverReadOnlyAccess.

July 18, 2024

Updated the entire Route 53 guide with the new console experience for health checks. For more information, see Creating, updating, and deleting health checks.

April 30, 2024

You can now decide if a DNS Firewall rule will either inspect (default), or trust the DNS redirection chain. For more information, see Resolver DNS Firewall components and settings and Rule settings in DNS Firewall.

April 22, 2024

You can now use Route 53 Profiles to share DNS-specific configurations with many VPCs and with AWS acounts. For more information, see What are Amazon Route 53 Profiles?.

April 22, 2024

Added the managed policies `AmazonRoute53ProfilesReadOnlyAccess` and `AmazonRoute53ProfilesFullAccess` to grant read-only and full access to Amazon Route 53 Profiles. For more information, see AWS managed policies for Amazon Route 53.

February 5, 2024

You can now use Amazon EventBridge for real time alerts with DNS Firewall. For more information, see Managing Resolver DNS Firewall events using Amazon EventBridge.

January 9, 2024

You can now use the DNS query type as an optional value for DNS Firewall rule to differentiate the rule's response for a specific DNS query type. For more information, see Resolver DNS Firewall components and settings and Rule settings in DNS Firewall.

January 9, 2024

You can now use the Quick create record or the Create record wizard to create geoproximity routing records. For more information, see Geoproximity routing, Values specific for geoproximity records, and Values specific for geoproximity alias records.

2023 releases

December 20, 2023

You can now use DNS over HTTPS with Resolver endpoints. For more information, see Choosing protocols for the endpoints.

July 20, 2023

Amazon Route 53 on Outposts is now available on AWS Outposts racks. It includes a VPC Resolver that caches all DNS queries that originate from the AWS Outposts. You can also set up hybrid connectivity between an Outpost and an on-premises DNS resolver when you deploy inbound and outbound endpoints. For more information, see What is Amazon Route 53 on Outposts?.

July 19, 2023

You can now use Local Zones with geoproximity routing (traffic flow only) after you enable them. For more information, see Geoproximity routing and Traffic Policy Document Format.

March 22, 2023

Updated the entire Route 53 guide with the new console experience for domains. You can also use the new console experience to transfer a domain from one AWS account to another AWS account. For more information, see Registering a new domain and Transferring domains.

March 10, 2023

You can now connect to your resources by using IPv4, IPv6, or dual-stack endpoints with Route 53 VPC Resolver. For more information, see Values that you specify when you create or edit inbound endpoints and Values that you specify when you create or edit outbound endpoints.

2022 releases

September 21, 2022

You can now use policy conditions for giving users fine-grained access to updating resource record sets in Amazon Route 53. For more information, see Using IAM policy conditions for fine-grained access control.

August 30, 2022

Amazon Route 53 now supports alias records for AWS App Runner services that are created after August 1, 2022. For more information, see Routing traffic to an AWS App Runner service.

June 1, 2022

IP-based routing option is now available in Amazon Route 53. For more information, see IP-based routing.

March 16, 2022

Geolocation and latency-based routing options are now supported for private hosted zones in Amazon Route 53. For more information, see Considerations when working with a private hosted zone.

January 25, 2022

The process for changing ownership for .com.au and .net.au TLDs has been simplified to include responding to two emails (by both old and new registrants) and doesn't include filling in forms. For more information, see .com.au (Australia) and .net.au (Australia).

2021 releases

October 26, 2021

Added support for disabling default reverse DNS rules with Amazon Route 53. You can now disable the creation of these rules and instead forward queries for reverse DNS namespaces to external servers if desired. For more information, see Forwarding rules for reverse DNS queries in VPC Resolver.

September 1, 2021

Added a new getting started topic that walks you through creating Amazon CloudFront distributions for a static website. For more information, see Use an Amazon CloudFront distribution to serve a static website.

July 14, 2021

Started tracking AWS managed policies for Amazon Route 53. For more information, see AWS managed policies for Amazon Route 53.

March 31, 2021

Added Resolver DNS Firewall. With DNS Firewall you can provide protection for outbound DNS requests from your VPCs. For more information, see Using DNS Firewall to filter outbound DNS traffic.

2020 releases

December 17, 2020

Added support for DNSSEC signing for Route 53 VPC Resolver. For more information, see Configuring DNSSEC signing in Amazon Route 53.

Added support for DNSSEC validation for Route 53 VPC Resolver. For more information, see Enabling DNSSEC validation in Amazon Route 53.

September 23, 2020

Updated the entire Route 53 guide with the new console experience. For more information, see What is Amazon Route 53?.

September 1, 2020

Added support for Resolver query logs. For more information, see Resolver query logging.

2018 releases

December 20, 2018

You can create Route 53 alias records that route traffic to API Gateway APIs or to Amazon VPC interface endpoints. For more information, see Value/route traffic to.

November 28, 2018

Route 53 Auto Naming (also known as Service Discovery) is now a separate service, AWS Cloud Map. For more information, see the AWS Cloud Map Developer Guide.

November 19, 2018

You can use Route 53 Resolver to configure DNS resolution between your VPC and your network over a Direct Connect or VPN connection. (Resolver is the new name for the recursive DNS service that is provided to all customers by default in Amazon Virtual Private Cloud (Amazon VPC).) This lets you forward DNS queries from resolvers on your network to Route 53 Resolver. Resolver also lets you forward queries for selected domain names (example.com) and subdomain names (api.example.com) from a VPC to resolvers on your network. For more information, see What is Route 53 VPC Resolver?.

November 7, 2018

When you're using Route 53 traffic flow and geoproximity routing, you can use an interactive map to visualize how your end users will be routed to your endpoints around the world. For more information, see Viewing a map that shows the effect of geoproximity settings.

October 18, 2018

You can use the Route 53 console and API to temporarily disable a Route 53 health check. This gives you an easy way to pause monitoring of an endpoint, such a web server, so that you can perform maintenance on it without triggering alarms or generating unnecessary logs or status messages. For more information, see "Disabled" in Values that you specify when you create or update health checks. The feature is available for all three types of Route 53 health checks: health checks that monitor an endpoint, health checks that monitor other health checks, and health checks that monitor a CloudWatch alarm.

March 13, 2018

If you're using auto naming, you can now use a third-party health checker to evaluate the health of your resources. This is useful when a resource isn't available over the internet, for example, because the instance is in an Amazon VPC. For more information, see HealthCheckCustomConfig in the *Amazon Route 53 API Reference*.

March 9, 2018

IAM now includes managed policies for auto naming. For more information, see AWS managed policies for Amazon Route 53.

February 6, 2018

You can now configure auto naming to create alias records that route traffic to ELB load balancers or to create CNAME records. For more information, see Attributes in the documentation for the RegisterInstance API in the *Amazon Route 53 API Reference*.

2017 releases

December 5, 2017

You can now use the Route 53 autonaming API to provision instances for microservices. Autonaming lets you automatically create DNS records and, optionally, health checks based on a template that you define. For more information, see What is AWS Cloud Map? in the *AWS Cloud Map Developer Guide*.

November 16, 2017

You can now programmatically get both the current quotas on Route 53 resources such as hosted zones and health checks, and the number of each resource that you're currently using. For more information, see GetAccountLimit, GetHostedZoneLimit, and GetReusableDelegationSetLimit in the *Amazon Route 53 API Reference*.

October 3, 2017

Route 53 is now a HIPAA eligible service. For more information, see Compliance validation for Amazon Route 53.

September 29, 2017

You can now programmatically check whether a domain can be transferred to Route 53. For more information, see CheckDomainTransferability in the *Amazon Route 53 API Reference*.

September 11, 2017

You can now create Route 53 alias records that route internet traffic to Elastic Load Balancing Network Load Balancers. For more information about alias records, see Choosing between alias and non-alias records.

September 7, 2017

If you're using Route 53 as your public, authoritative DNS service, you can now log DNS queries that Route 53 receives. For more information, see Public DNS query logging.

September 1, 2017

If you're using Route 53 traffic flow, you can now use geoproximity routing, which lets you route traffic based on the physical distance between your users and your resources. You can also route more or less traffic to each resource by specifying a positive or negative bias. For more information, see Geoproximity routing.

August 21, 2017

You can now use Route 53 to create Certification Authority Authorization (CAA) records, which let you specify the certificate authorities that can issue certificates for your domains and subdomains. For more information, see CAA record type.

August 18, 2017

You can now transfer large numbers of domains to Route 53 using the Route 53 console. For more information, see Transferring registration for a domain to Amazon Route 53.

August 4, 2017

When you register a domain, the registries for some top-level domains (TLDs) require you to verify that you specified a valid email address for the registrant contact. You can now send the verification email and get confirmation that you successfully verified the email address during the domain registration process. For more information, see Registering a new domain.

June 21, 2017

If you want to route traffic approximately randomly to multiple resources, such as web servers, you can now create one multivalue answer record for each resource and, optionally, associate a Route 53 health check with each record. Route 53 responds to DNS queries with up to eight healthy records in response to each DNS query, and gives different answers to different DNS resolvers. For more information, see Multivalue answer routing.

April 10, 2017

When you use the Route 53 console to transfer a domain registration to Route 53, you can now choose one of the following options for associating the name servers for the DNS service for the domain with the transferred domain registration:

- Use the name servers for a Route 53 hosted zone that you choose
- Use the name servers for the current DNS service for the domain
- Use name servers that you specify

Route 53 automatically associates these name servers with the transferred domain registration.

2016 releases

November 21, 2016

You can now create health checks that use IPv6 addresses to check the health of endpoints. For more information, see Creating and updating health checks.

November 15, 2016

You can now use a Route 53 API action to associate an Amazon VPC that you created with one account with a private hosted zone that you created with another account. For more information, see Associating an Amazon VPC and a private hosted zone that you created with different AWS accounts.

August 30, 2016

With this release, Route 53 adds the following new features:

- **Name Authority Pointer (NAPTR) records** – You can now create NAPTR records, which are used by Dynamic Delegation Discovery System (DDDS) applications to convert one value to another or to replace one value with another. For example, one common use is to convert phone numbers into SIP URIs. For more information, see NAPTR record type.
- **DNS query test tool** – You can now simulate DNS queries for a record and see the value that Route 53 returns. For geolocation and latency records, you can also simulate requests from a particular DNS resolver and/or client IP address to find out what response Route 53 would return to a client with that resolver and/or IP address. For more information, see Checking DNS responses from Route 53.

August 11, 2016

With this release, you can create alias records that route traffic to ELB Application Load Balancers. The process is the same as for Classic Load Balancers. For more information, see Value/route traffic to.

August 9, 2016

With this release, Route 53 adds support for DNSSEC for domain registration. DNSSEC lets you protect your domain from DNS spoofing attacks, which are also known as man-in-the-middle attacks. For more information, see Configuring DNSSEC for a domain.

July 7, 2016

You can now manually extend the registration for a domain and register a domain with an initial registration period longer than the minimum registration period specified by the registry. For more information, see Extending the registration period for a domain.

July 6, 2016

If you're an AISPL customer with a contact address in India, you can now use Route 53 to register domains. For more information, see Managing an Account in India.

May 26, 2016

With this release, Route 53 adds the following new features:

- **Domain billing report** – You can now download a report that lists all domain registration charges, by domain, for a specified time period. The report includes all domain registration operations for which there is a fee, including registering domains, transferring domains to Route 53, renewing domain registration, and (for some TLDs), changing the owner of a domain. For more information, see the following documentation:
  - **Route 53 console** – See Downloading a domain billing report
  - **Route 53 API** – See ViewBilling in the *Amazon Route 53 API Reference*.
- **New TLDs** – You can now register domains that have the following TLDs: .college, .consulting, .host, .name, .online, .republican, .rocks, .sucks, .trade, .website, and .uk. For more information, see Domains that you can register with Amazon Route 53.
- **New APIs for domain registration** – For operations that require confirmation that the email address for the registrant contact is valid, such as registering a new domain, you can now programmatically determine whether the registrant contact has clicked the link in the confirmation email and, if not, whether the link is still valid. You can also programmatically request that we send another confirmation email. For more information, see the following documentation in the *Amazon Route 53 API Reference*:
  - GetContactReachabilityStatus
  - ResendContactReachabilityEmail

April 5, 2016

With this release, Route 53 adds the following new features:

- **Health checks based on CloudWatch metrics** – You can now create health checks that are based on the alarm state of any CloudWatch metric. This is useful for checking the health

of endpoints that can't be reached by a standard Route 53 health check, such as instances within an Amazon Virtual Private Cloud (VPC) that have only private IP addresses. For more information, see the following documentation:

- **Route 53 console** – See Monitoring a CloudWatch alarm in the "Values that You Specify When You Create or Update Health Checks" topic.
- **Route 53 API** – See CreateHealthCheck and UpdateHealthCheck in the *Amazon Route 53 API Reference*.

- **Configurable health check locations** – You can now choose the Route 53 health checking regions that check the health of your resources, which reduces the load on the endpoint from health checks. This is useful if your customers are concentrated in one or a few geographic regions. For more information, see the following documentation:
  - **Route 53 console** – See Advanced configuration ("Monitor an Endpoint" only) in the "Values that You Specify When You Create or Update Health Checks" topic.
  - **Route 53 API** – See the `Regions` element for CreateHealthCheck and UpdateHealthCheck in the *Amazon Route 53 API Reference*.
- **Failover in private hosted zones** – You can now create failover and failover alias records in a private hosted zone. When you combine this feature with metric-based health checks, you can configure DNS failover even for endpoints that have only private IP addresses and can't be reached by using standard Route 53 health checks. For more information, see the following documentation:
  - **Route 53 console** – See Configuring failover in a private hosted zone.
  - **Route 53 API** – See ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.
- **Alias records in private hosted zones** – In the past, you could create alias records that route DNS queries only to other Route 53 records in the same hosted zone. With this release, you can also create alias records that route DNS queries to Elastic Beanstalk environments that have regionalized subdomains, Elastic Load Balancing load balancers, and Amazon S3 buckets. (You still can't create alias records that route DNS queries to a CloudFront distribution.) For more information, see the following documentation:
  - **Route 53 console** – See Choosing between alias and non-alias records.
  - **Route 53 API** – See ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.

February 23, 2016

When you create or update HTTPS health checks, you can now configure Route 53 to send the host name to the endpoint during TLS negotiation. This allows the endpoint to respond to the HTTPS request with the applicable SSL/TLS certificate. For more information, see the

description for the SNI in Advanced configuration ("Monitor an Endpoint" only) field in the "Values that You Specify When You Create or Update Health Checks" topic. For information about how to enable SNI when you use the API to create or update a health check, see CreateHealthCheck and UpdateHealthCheck in the *Amazon Route 53 API Reference*.

January 27, 2016

You can now register domains for over 100 additional top-level domains (TLDs) such as .accountants, .band, and .city. For a complete list of supported TLDs, see Domains that you can register with Amazon Route 53.

January 19, 2016

You can now create alias records that route traffic to Elastic Beanstalk environments. For information about creating records by using the Route 53 console, see Creating records by using the Amazon Route 53 console. For information about using the API to create records, see ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.

2015 releases

December 3, 2015

The Route 53 console now includes a visual editor that lets you quickly create complex routing configurations that use a combination of Route 53 weighted, latency, failover, and geolocation routing policies. You can then associate the configuration with one or more domain names (such as example.com) or subdomain names (such as www.example.com), in the same hosted zone or in multiple hosted zones. In addition, you can roll back the updates if the new configuration isn't performing as you expected it to. The same functionality is available by using the Route 53 API, AWS SDKs, the AWS CLI, and AWS Tools for Windows PowerShell. For information about using the visual editor, see Using Traffic Flow to route DNS traffic. For information about using the API to create traffic flow configurations, see the Amazon Route 53 API Reference.

October 19, 2015

With this release, Route 53 adds the following new features:

- **Domain registration for .com and .net domains by Amazon Registrar, Inc.** – Amazon is now an ICANN-accredited registrar for the .com and .net top-level domains (TLDs) through Amazon Registrar, Inc. When you use Route 53 to register a .com or .net domain, Amazon Registrar will be the registrar of record and will be listed as the "Sponsoring Registrar" in

your Whois query results. For information about using Route 53 to register domains, see Registering and managing domains using Amazon Route 53.

- **Privacy protection for .com and .net domains** – When you register a .com or .net domain with Route 53, all of your personal information, including first and last name, is now hidden. First and last name are not hidden for other domains that you register with Route 53. For more information about privacy protection, see Enabling or disabling privacy protection for contact information for a domain.

September 15, 2015

With this release, Route 53 adds the following new features:

- **Calculated health checks** – You can now create health checks whose status is determined by the health status of other health checks. For more information, see Creating and updating health checks. In addition, see CreateHealthCheck in the *Amazon Route 53 API Reference*.
- **Latency measurements for health checks** – You can now configure Route 53 to measure the latency between health checkers and your endpoint. Latency data appears in Amazon CloudWatch graphs in the Route 53 console. To enable latency measurements for new health checks, see the **Latency measurements** setting under Advanced configuration ("Monitor an Endpoint" only) in the topic Values that you specify when you create or update health checks. (You can't enable latency measurements for existing health checks.) In addition, see **MeasureLatency** in the topic CreateHealthCheck in the *Amazon Route 53 API Reference*.
- **Updates to the health checks dashboard in the Route 53 console** – The dashboard for monitoring health checks has been improved in a variety of ways, including CloudWatch graphs for monitoring latency between Route 53 health checkers and your endpoints. For more information, see Monitoring health check status and getting notifications.

March 3, 2015

The *Amazon Route 53 Developer Guide* now explains how to configure white-label name servers for Route 53 hosted zones. For more information, see Configuring white-label name servers.

February 26, 2015

You can now use the Route 53 API to list the hosted zones that are associated with an AWS account in alphabetical order by name. You can also get a count of the hosted zones that are associated with an account. For more information, see ListHostedZonesByName and GetHostedZoneCount in the *Amazon Route 53 API Reference*.

February 11, 2015

With this release, Route 53 adds the following new features:

- **Health Check Status** – The health checks page in the Route 53 console now includes a **Status** column that lets you view the overall status of all of your health checks. For more information, see Viewing health check status and the reason for health check failures.
- **Integration with AWS CloudTrail** – Route 53 now works with CloudTrail to capture information about every request that your AWS account sends to the Route 53 API. Integrating Route 53 and CloudTrail lets you determine which requests were made to the Route 53 API, the source IP address from which each request was made, who made the request, when it was made, and more. For more information, see Logging Amazon Route 53 API calls with AWS CloudTrail.
- **Quick Alarms for Health Checks** – When you create a health check by using the Route 53 console, you can now simultaneously create an Amazon CloudWatch alarm for the health check and specify who to notify when Route 53 considers the endpoint unhealthy for one minute. For more information, see Creating and updating health checks.
- **Tagging for Hosted Zones and Domains** – You can now assign tags, which are commonly used for cost allocation, to Route 53 hosted zones and domains. For more information, see Tagging Amazon Route 53 resources.

February 5, 2015

You can now use the Route 53 console to update contact information for a domain. For more information, see Values that you specify when you register or transfer a domain.

January 22, 2015

You can now specify internationalized domain names when you're registering a new domain name with Route 53. (Route 53 already supported internationalized domain names for hosted zones and records.) For more information, see DNS domain name format.

2014 releases

November 25, 2014

With this release, you can now edit the comment that you specified for a hosted zone when you created it. In the console, you just choose the pencil icon next to the **Comment** field and enter a new value. For more information about changing the comment by using the Route 53 API, see UpdateHostedZoneComment in the *Amazon Route 53 API Reference*.

November 5, 2014

With this release, Route 53 adds the following new features:

- **Private DNS for VPCs created using the Amazon Virtual Private Cloud service** – You can now use Route 53 to manage your internal domain names for VPCs without exposing DNS data to the public internet. For more information, see Working with private hosted zones.
- **Health check failure reasons** – You can now see the current status of a selected health check, as well as details on why the health check last failed, as reported by each of the Route 53 health checkers. The status includes the HTTP status code, and failure reasons include information about numerous types of failures, such as string matching failures and response timeouts. For more information, see Viewing health check status and the reason for health check failures.
- **Reusable delegation sets** – You can now apply the same set of four authoritative name servers, known collectively as a delegation set, to multiple hosted zones that correspond with different domain names. This greatly simplifies the process of migrating DNS service to Route 53 and managing large numbers of hosted zones. Using reusable delegation sets currently requires that you use the Route 53 API or an AWS SDK. For more information, see the Amazon Route 53 API Reference.
- **Improved geolocation routing** – We further improved the accuracy of geolocation routing by adding support for the edns-client-subnet extension of EDNS0. For more information, see Geolocation routing.
- **Support for Signature v4** – You can now sign all Route 53 API requests using Signature version 4. For more information, see Signing Route 53 API Requests in the *Amazon Route 53 API Reference*.

July 31, 2014

With this release, you can now do the following:

- Register domain names using Route 53. For more information, see Registering and managing domains using Amazon Route 53.
- Configure Route 53 to respond to DNS queries based on the geographic location that the queries originate from. For more information, see Geolocation routing.

July 2, 2014

With this release, you can now do the following:

- Edit most values in health checks. For more information, see Creating, updating, and deleting health checks.
- Use the Route 53 API to get a list of the IP ranges that Route 53 health checkers use to check the health of your resources. You can use these IP addresses to configure your router

and firewall rules to allow health checkers to check the health of your resources. For more information, see GetCheckerIpRanges in the *Amazon Route 53 API Reference*.

- Assign cost allocation tags to health checks, which also lets you assign a name to health checks. For more information, see Naming and tagging health checks.
- Use the Route 53 API to get the number of health checks that are associated with your AWS account. For more information, see GetHealthCheckCount in the *Amazon Route 53 API Reference*.

April 30, 2014

With this release, you can now create health checks and use a domain name instead of an IP address to specify the endpoint. This is helpful when an endpoint's IP address either is not fixed or is served by multiple IPs, such as Amazon EC2 or Amazon RDS instances. For more information, see Creating and updating health checks.

In addition, some information about using the Route 53 API that formerly appeared in the *Amazon Route 53 Developer Guide* has been moved. Now all API documentation appears in the *Amazon Route 53 API Reference*.

April 18, 2014

With this release, Route 53 passes a different value in the `Host` header when the health check **Port** value is **443** and the **Protocol** value is **HTTPS**. During a health check, Route 53 now passes to the endpoint a `Host` header that contains the value of the **Host Name** field. If you created the health check by using the `CreateHealthCheck` API action, this is the value of the `FullyQualifiedDomainName` element.

For more information, see Creating, updating, and deleting health checks.

April 9, 2014

With this release, you can now view what percentage of Route 53 health checkers are currently reporting that an endpoint is healthy.

In addition, behavior of the Health Check Status metric in Amazon CloudWatch now shows only zero (if your endpoint was unhealthy during a given time period) or one (if the endpoint was healthy for that time period). The metric no longer shows values between 0 and 1 reflecting the portion of Route 53 health checks that are reporting the endpoint as healthy.

For more information, see Monitoring health checks using CloudWatch.

February 18, 2014

With this release, Route 53 adds the following features:

- **Health check failover threshold:** You can now specify how many consecutive health checks an endpoint must fail before Route 53 considers the endpoint unhealthy, between 1 and 10 consecutive checks. An unhealthy endpoint must pass the same number of checks to be considered healthy. For more information, see How Amazon Route 53 determines whether a health check is healthy.
- **Health check request interval:** You can now specify how frequently Route 53 sends requests to an endpoint to determine whether the endpoint is healthy. Valid settings are 10 seconds and 30 seconds. For more information, see How Amazon Route 53 determines whether a health check is healthy.

January 30, 2014

With this release, Route 53 adds the following features:

- **HTTP and HTTPS string-match health checks:** Route 53 now supports health checks that determine the health of an endpoint based on the appearance of a specified string in the response body. For more information, see How Amazon Route 53 determines whether a health check is healthy.
- **HTTPS health checks:** Route 53 now supports health checks for secure, SSL-only websites. For more information, see How Amazon Route 53 determines whether a health check is healthy.
- **UPSERT for the `ChangeResourceRecordSets` API Action:** When creating or changing records using the `ChangeResourceRecordSets` API action, you can now use the UPSERT action either to create a new record if none exists with a given name and type, or to update an existing record. For more information, see ChangeResourceRecordSets in the *Amazon Route 53 API Reference*.

January 7, 2014

With this release, Route 53 adds support for health checks that determine the health of an endpoint based on whether a specified string appears in the response body. For more information, see How Amazon Route 53 determines whether a health check is healthy.

2013 releases

August 14, 2013

With this release, Route 53 adds support for creating records by importing a BIND-formatted zone file. For more information, see Creating records by importing a zone file.

In addition, CloudWatch metrics for Route 53 health checks have been integrated into the Route 53 console and streamlined. For more information, see Monitoring health checks using CloudWatch.

June 26, 2013

With this release, Route 53 adds support for integrating health checks with CloudWatch metrics so you can do the following:

- Verify that a health check is properly configured.
- Review the health of a health check endpoint over a specified period of time.
- Configure CloudWatch to send an Amazon Simple Notification Service (Amazon SNS) alert when all Route 53 health checkers consider your specified endpoint to be unhealthy.

For more information, see Monitoring health checks using CloudWatch.

June 11, 2013

With this release, Route 53 adds support for creating alias records that route DNS queries to alternate domain names for Amazon CloudFront distributions. You can use this feature both for alternate domain names at the zone apex (example.com) and alternate domain names for subdomains (www.example.com). For more information, see Routing traffic to an Amazon CloudFront distribution by using your domain name.

May 30, 2013

With this release, Route 53 adds support for evaluating the health of ELB load balancers and the associated Amazon EC2 instances. For more information, see Creating Amazon Route 53 health checks.

March 28, 2013

The documentation about health checks and failover was rewritten to enhance usability. For more information, see Creating Amazon Route 53 health checks.

February 11, 2013

With this release, Route 53 adds support for failover and health checks. For more information, see Creating Amazon Route 53 health checks.

2012 release

March 21, 2012

With this release, Route 53 lets you create latency records. For more information, see Latency-based routing.

2011 releases

December 21, 2011

With this release, the Route 53 console in the AWS Management Console lets you create an alias record by choosing an Elastic Load Balancer from a list instead of manually entering the hosted zone ID and the DNS name of the load balancer. New functionality is documented in the *Amazon Route 53 Developer Guide*.

November 16, 2011

With this release, you can use the Route 53 console in the AWS Management Console to create and delete hosted zones, and to create, change, and delete records. New functionality is documented throughout the *Amazon Route 53 Developer Guide*, as applicable.

October 18, 2011

The *Amazon Route 53 Getting Started Guide* was merged into the *Amazon Route 53 Developer Guide*, and the *Developer Guide* was reorganized to enhance usability.

May 24, 2011

This release of Amazon Route 53 introduces alias records, which allow you to create zone apex aliases; weighted records; a new API (2011-05-05); and a service-level agreement. In addition, after six months in beta, Route 53 is now generally available. For more information, see the Amazon Route 53 product page and Choosing between alias and non-alias records in the *Amazon Route 53 Developer Guide*.

2010 release

December 5, 2010

This is the first release of *Amazon Route 53 Developer Guide*.