Amazon Route 53

General Route 53 Concepts

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) service.

Route 53 offers the following functions:

* Domain name registry.
* DNS resolution.
* Health checking of resources.

Route 53 can perform any combination of these functions.

Route 53 provides a worldwide distributed DNS service.

Route 53 is located alongside all edge locations.

Health checks verify Internet connected resources are reachable, available and functional.

Route 53 can be used to route Internet traffic for domains registered with another domain registrar (any domain).

When you register a domain with Route 53 it becomes the authoritative DNS server for that domain and creates a public hosted zone.

To make Route 53 the authoritative DNS for an existing domain without transferring the domain create a Route 53 public hosted zone and change the DNS Name Servers on the existing provider to the Route 53 Name Servers.

Changes to Name Servers may not take effect for up to 48 hours due to the DNS record Time To Live (TTL) values.

You can transfer domains to Route 53 only if the Top Level Domain (TLD) is supported.

You can transfer a domain from Route 53 to another registrar by contacting AWS support.

You can transfer a domain to another account in AWS however it does not migrate the hosted zone by default (optional).

It is possible to have the domain registered in one AWS account and the hosted zone in another AWS account.

Primarily uses UDP port 53 (can use TCP).

AWS offer a 100% uptime SLA for Route 53.

You can control management access to your Amazon Route 53 hosted zone by using IAM.

There is a default limit of 50 domain names but this can be increased by contacting support.

Private DNS is a Route 53 feature that lets you have authoritative DNS within your VPCs without exposing your DNS records (including the name of the resource and its IP address(es) to the Internet.

You can use the AWS Management Console or API to register new domain names with Route 53.

Hosted Zones

A hosted zone is a collection of records for a specified domain.

A hosted zone is analogous to a traditional DNS zone file; it represents a collection of records that can be managed together.

There are two types of zones:

* Public host zone – determines how traffic is routed on the Internet.
* Private hosted zone for VPC – determines how traffic is routed within VPC (resources are not accessible outside the VPC).

Amazon Route 53 automatically creates the Name Server (NS) and Start of Authority (SOA) records for the hosted zones.

Amazon Route 53 creates a set of 4 unique name servers (a delegation set) within each hosted zone.

You can create multiple hosted zones with the same name and different records.

NS servers are specified by Fully Qualified Domain Name (FQDN) but you can get the IP addresses from the command line (e.g. dig or nslookup).

For private hosted zones you can see a list of VPCs in each region and must select one.

For private hosted zones you must set the following VPC settings to “true”:

* enableDnsHostname.
* enableDnsSupport.

You also need to create a DHCP options set.

You can extend an on-premises DNS to VPC.

You cannot extend Route 53 to on-premises instances.

You cannot automatically register EC2 instances with private hosted zones (would need to be scripted).

Health checks check the instance health by connecting to it.

Health checks can be pointed at:

* Endpoints.
* Status of other health checks.
* Status of a CloudWatch alarm.

Endpoints can be IP addresses or domain names.

Records

Amazon Route 53 currently supports the following DNS record types:

* A (address record).
* AAAA (IPv6 address record).
* CNAME (canonical name record).
* CAA (certification authority authorization).
* MX (mail exchange record).
* NAPTR (name authority pointer record).
* NS (name server record).
* PTR (pointer record).
* SOA (start of authority record).
* SPF (sender policy framework).
* SRV (service locator).
* TXT (text record).
* Alias (an Amazon Route 53-specific virtual record).

The Alias record is a Route 53 specific record type.

Alias records are used to map resource record sets in your hosted zone to Amazon Elastic Load Balancing load balancers, Amazon CloudFront distributions, AWS Elastic Beanstalk environments, or Amazon S3 buckets that are configured as websites.

You can use Alias records to map custom domain names (such as api.example.com) both to API Gateway custom regional APIs and edge-optimized APIs and to Amazon VPC interface endpoints.

The Alias is pointed to the DNS name of the service.

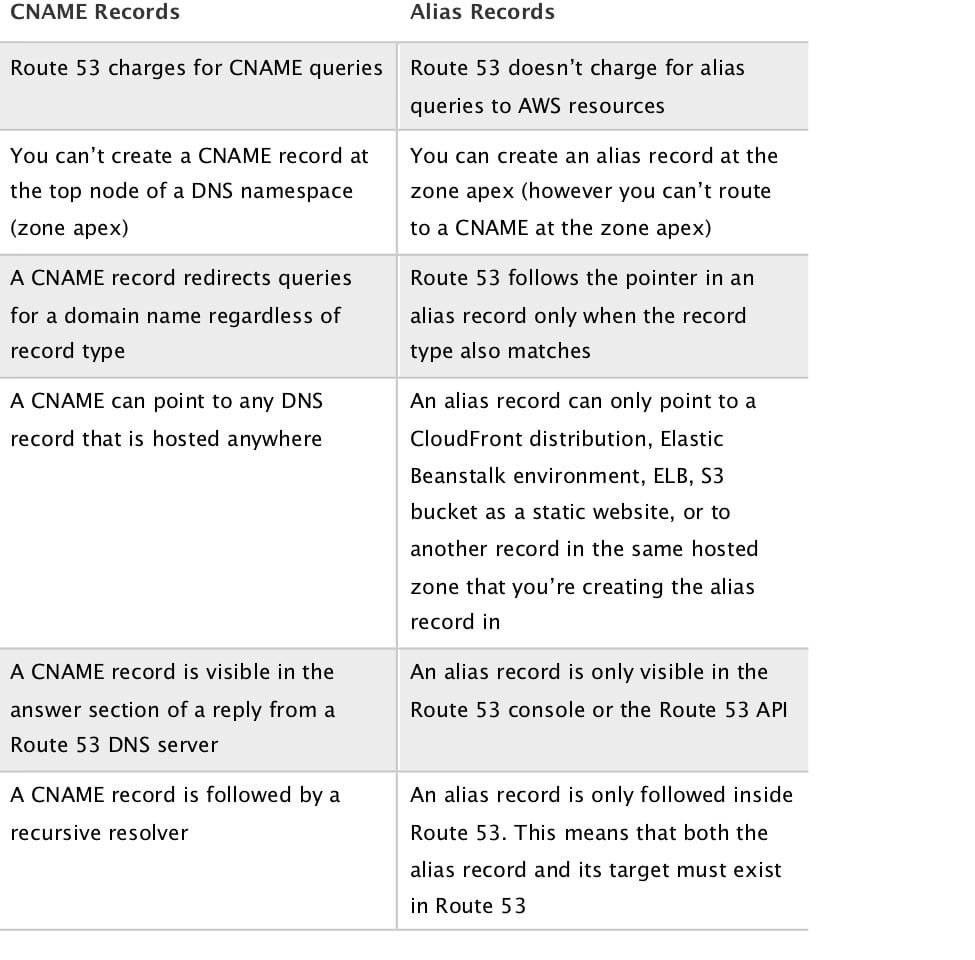
You cannot set the TTL for Alias records for ELB, S3, or Elastic Beanstalk environment (uses the service’s default).

Alias records work like a CNAME record in that you can map one DNS name (e.g. example.com) to another ‘target’ DNS name (e.g. elb1234.elb.amazonaws.com).

An Alias record can be used for resolving apex / naked domain names (e.g. example.com rather than sub.example.com).

A CNAME record can’t be used for resolving apex / naked domain names.

Generally use an Alias record where possible. The following table details the differences between Alias and CNAME records:

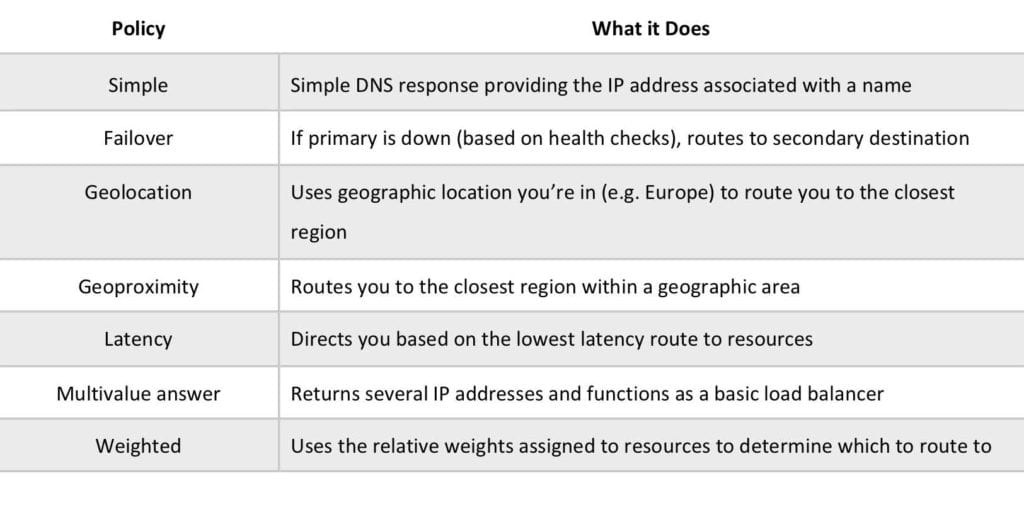


Route 53 supports wildcard entries for all record types, except NS records.

Routing Policies

Routing policies determine how Route 53 responds to queries.

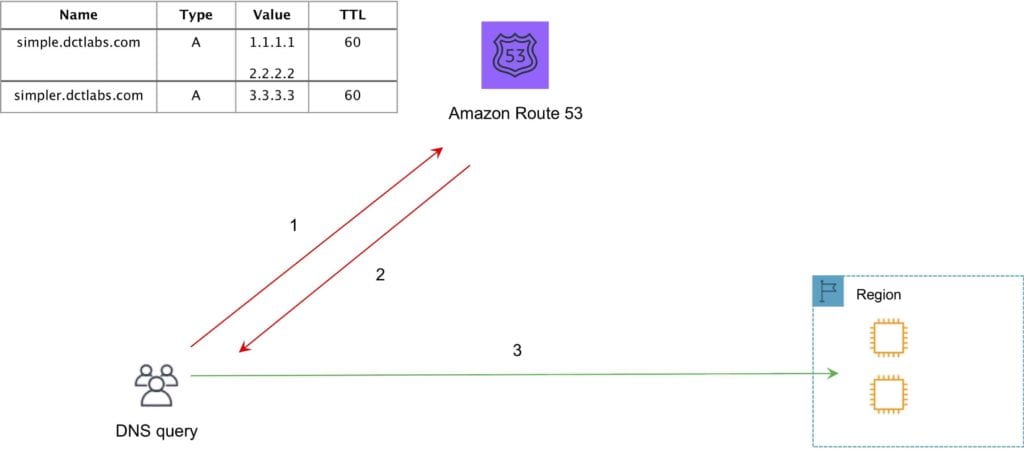
The following table highlights the key function of each type of routing policy:



Simple:

* An A record is associated with one or more IP addresses.
* Uses round robin.
* Does not support health checks.

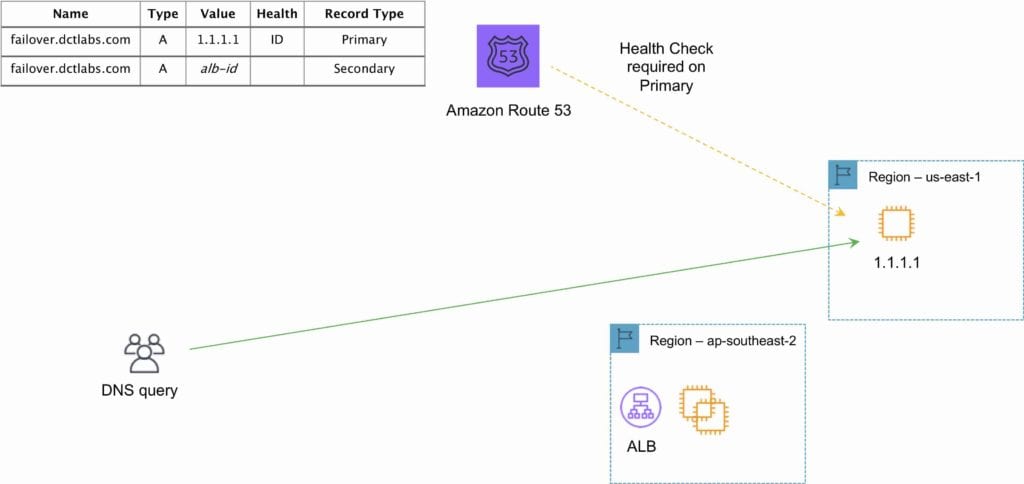
The following diagram depicts an Amazon Route 53 Simple routing policy configuration:



Failover:

* Failover to a secondary IP address.
* Associated with a health check.
* Used for active-passive.
* Routes only when the resource is healthy.
* Can be used with ELB.
* When used with Alias records set Evaluate Target Health to “Yes” and do not use health checks.

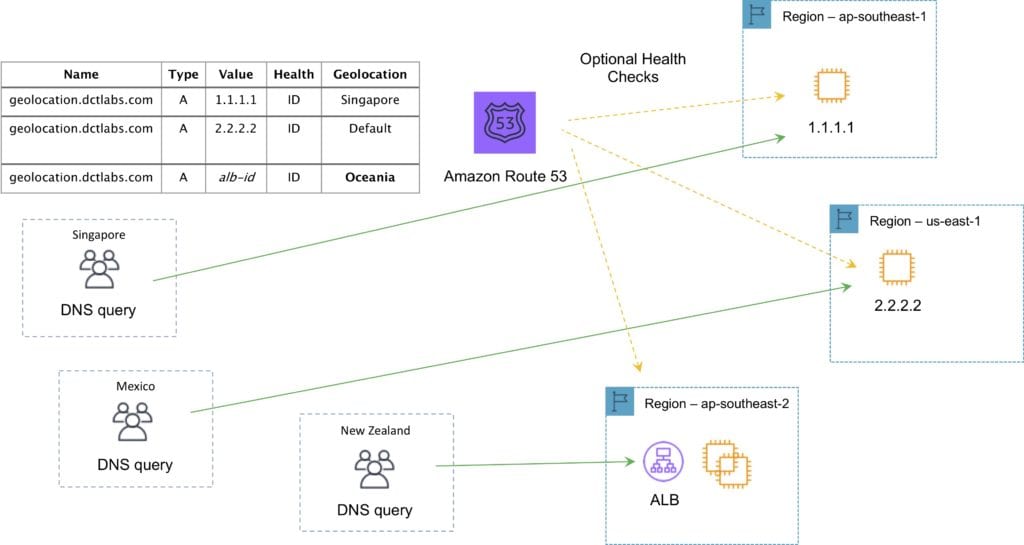
The following diagram depicts an Amazon Route 53 Failover routing policy configuration:



Geo-location:

* Caters to different users in different countries and different languages.
* Contains users within a particular geography and offers them a customized version of the workload based on their specific needs.
* Geolocation can be used for localizing content and presenting some or all of your website in the language of your users.
* Can also protect distribution rights.
* Can be used for spreading load evenly between regions.
* If you have multiple records for overlapping regions, Route 53 will route to the smallest geographic region.
* You can create a default record for IP addresses that do not map to a geographic location.

The following diagram depicts an Amazon Route 53 Geolocation routing policy configuration:



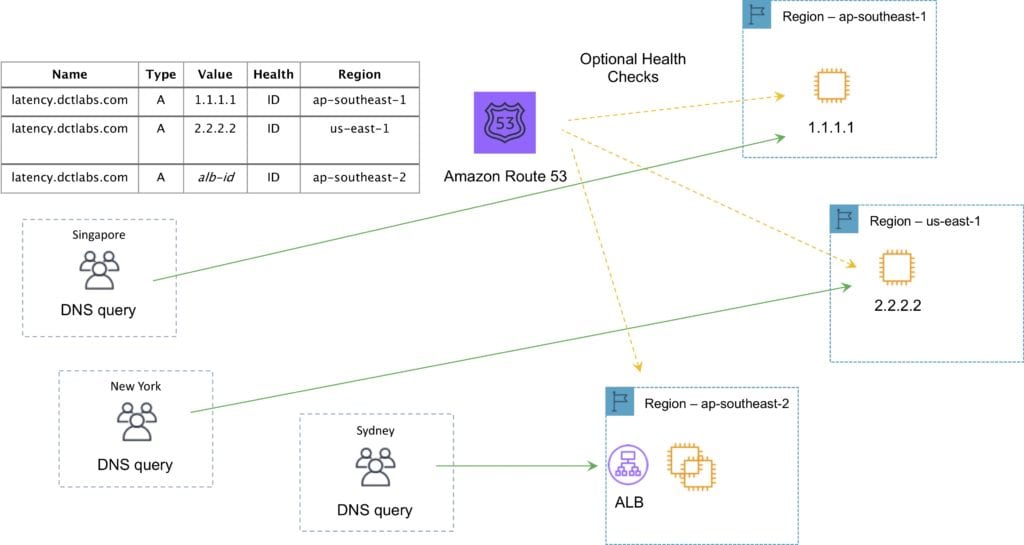
Geo-proximity routing policy (requires Route Flow):

* Use for routing traffic based on the location of resources and, optionally, shift traffic from resources in one location to resources in another.

Latency based routing:

* AWS maintains a database of latency from different parts of the world.
* Focussed on improving performance by routing to the region with the lowest latency.
* You create latency records for your resources in multiple EC2 locations.

The following diagram depicts an Amazon Route 53 Latency based routing policy configuration:



Multi-value answer routing policy:

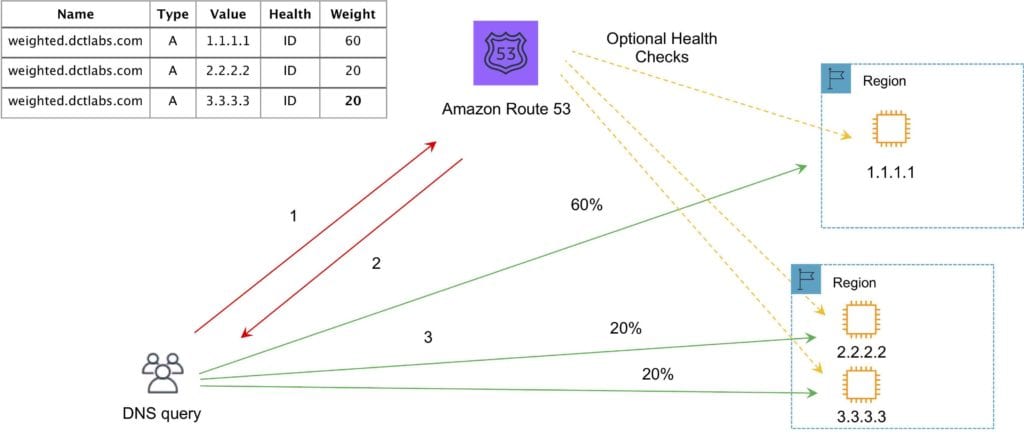
* Use for responding to DNS queries with up to eight healthy records selected at random.

The following diagram depicts an Amazon Route 53 Multivalue routing policy configuration:

Weighted:

* Similar to simple but you can specify a weight per IP address.
* You create records that have the same name and type and assign each record a relative weight.
* Numerical value that favours one IP over another.
* To stop sending traffic to a resource you can change the weight of the record to 0.

The following diagram depicts an Amazon Route 53 Weighted routing policy configuration:



Want to see this in action? In the AWS Hands-On Labs video tutorial below, you’ll learn about the Amazon Route 53 Weighted Routing Policy. We teach you how it works and show you it in action using EC2 instances spread around the world.

Traffic Flow

Route 53 Traffic Flow provides Global Traffic Management (GTM) services.

Traffic flow policies allow you to create routing configurations for resources using routing types such as failover and geolocation.

Create policies that route traffic based on specific constraints, including latency, endpoint health, load, geo-proximity and geography.

Scenarios include:

* Adding a simple backup page in Amazon S3 for a website.
* Building sophisticated routing policies that consider an end user’s geographic location, proximity to an AWS region, and the health of each of your endpoints.

Amazon Route 53 Traffic Flow also includes a versioning feature that allows you to maintain a history of changes to your routing policies, and easily roll back to a previous policy version using the console or API.

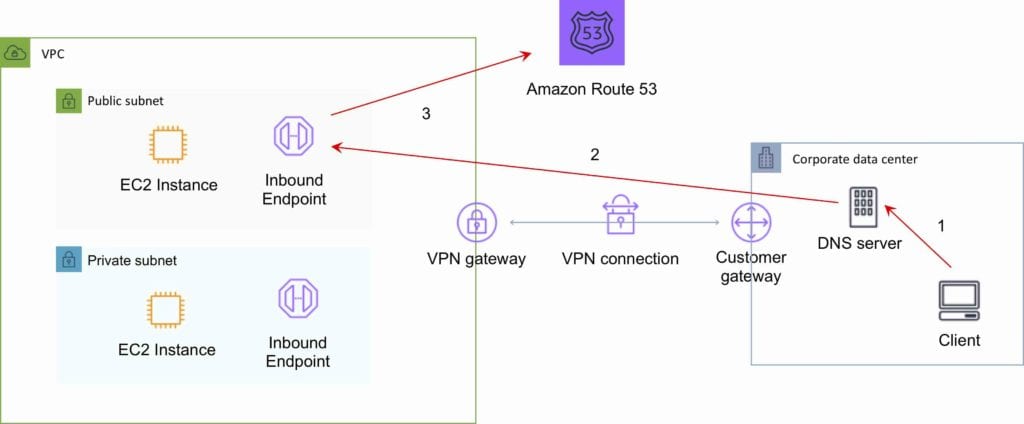
Route 53 Resolver

Route 53 Resolver is a set of features that enable bi-directional querying between on-premises and AWS over private connections.

Used for enabling DNS resolution for hybrid clouds.

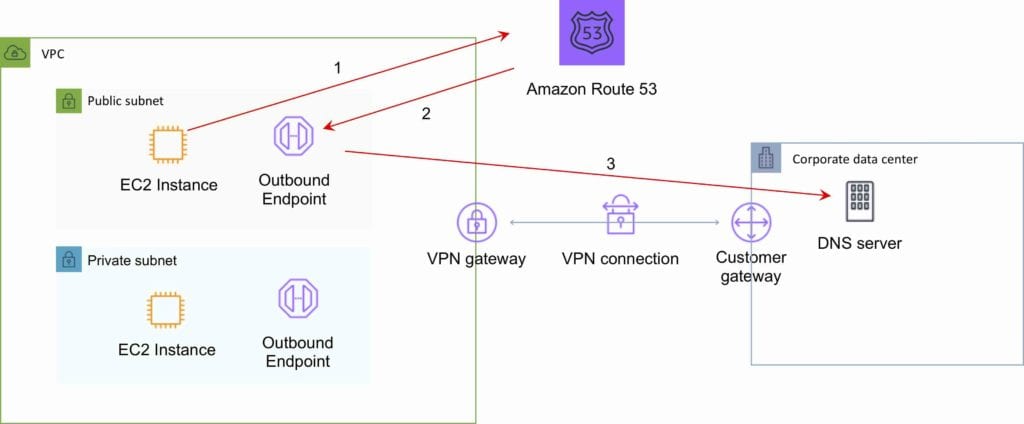
Route 53 Resolver Endpoints.

* Inbound query capability is provided by Route 53 Resolver Endpoints, allowing DNS queries that originate on-premises to resolve AWS hosted domains.
* Connectivity needs to be established between your on-premises DNS infrastructure and AWS through a Direct Connect (DX) or a Virtual Private Network (VPN).
* Endpoints are configured through IP address assignment in each subnet for which you would like to provide a resolver.



Conditional forwarding rules:

* Outbound DNS queries are enabled through the use of Conditional Forwarding Rules. .
* Domains hosted within your on-premises DNS infrastructure can be configured as forwarding rules in Route 53 Resolver.
* Rules will trigger when a query is made to one of those domains and will attempt to forward DNS requests to your DNS servers that were configured along with the rules.
* Like the inbound queries, this requires a private connection over DX or VPN.



Charges

You pay per hosted zone per month (no partial months).

A hosted zone deleted within 12 hours of creation is not charged (queries are charged).

Additional charges for:

* Queries.
* Traffic Flow.
* Health Checks.
* Route 53 Resolver ENIs + queries.
* Domain names.

Alias records are free of charge when the records are mapped to one of the following:

* Elastic Load Balancers.
* Amazon CloudFront distributions.
* AWS Elastic Beanstalk environments.
* Amazon S3 buckets that are configured as website endpoints.

Health checks are charged with different prices for AWS vs non-AWS endpoints.

You do not pay for the records that you add to your hosted zones.

Latency-based routing queries are more expensive.

Geo DNS and geo-proximity also have higher prices.