

Route 53

If you are studying for AWS Developer Associate Exam, this guide will help you with quick revision before the exam. It can be used as study notes for your preparation.

Dashboard

Other Certification Notes

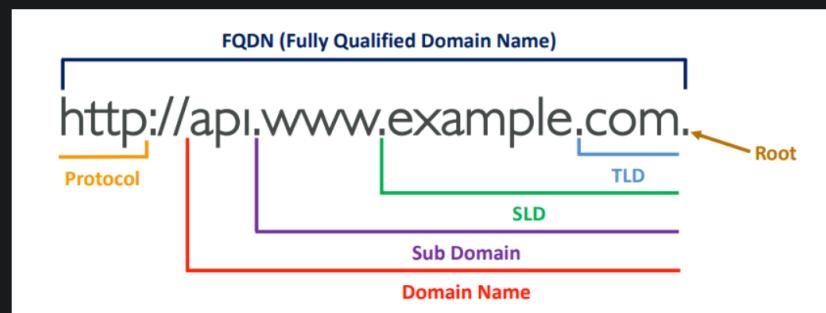
Route 53

- Route 53
 - What is DNS?
 - DNS Terminologies
 - How DNS Works
 - Amazon Route 53
 - Route 53 - Records
 - Route 53 - RecordTypes
 - Route 53 - Hosted Zones
 - Route 53 - RecordTTL (TimeTo Live)
 - CNAME vs Alias
 - Route 53 - Alias Records
 - Route 53 - Alias Records Targets
 - Route 53 - Routing Policies
 - Routing Policies - Simple
 - Routing Policies - Multi-Value
 - Routing Policies - Weighted
 - Routing Policies - Latency-based
 - Route 53 - Health Checks
 - Health Checks - Monitor an Endpoint
 - Route 53 - Calculated Health Checks
 - Health Checks - Private Hosted Zones
 - Routing Policies - Failover (Active-Passive)
 - Routing Policies - Geolocation
 - Routing Policies - Geoproximity
 - Route 53 - Traffic flow
 - Domain Registrar vs. DNS Service
 - GoDaddy as Registrar & Route 53 as DNS Service
 - 3rd Party Registrar with Amazon Route 53

What is DNS?

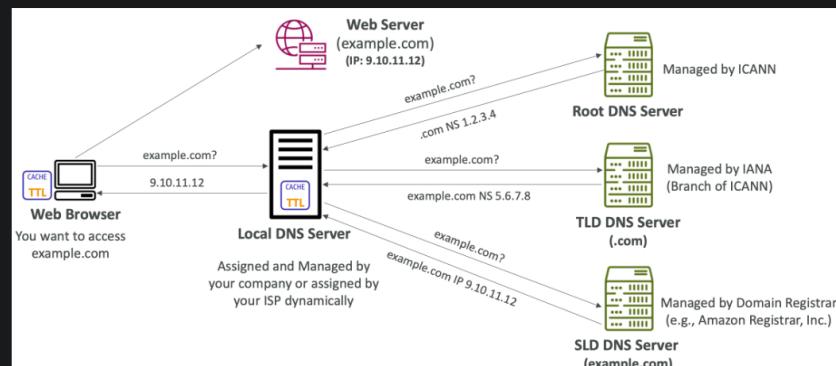
- Domain Name System which translates the human friendly hostname into the machine IP addresses
- www.google.com => 172.217.18.36
- DNS is the backbone of the Internet
- DNS uses hierarchical naming structure
 - .com
 - example.com
 - www.example.com
 - api.example.com

- Domain Registrar: Amazon Route 53, GoDaddy, ...
- DNS Records: A, AAAA, CNAME, NS, ...
- Zone File: contains DNS records
- Name Server: resolves DNS queries (Authoritative or Non-Authoritative) • Top Level Domain (TLD): .com, .us, .in, .gov, .org, ...
- Second Level Domain (SLD): amazon.com, google.com, ...



How DNS Works

- DNS is a system used to map domain names (such as www.example.com) to IP addresses (such as 9.10.11.12).
- When a user enters a domain name into their web browser, the browser sends a request to a DNS server to resolve the domain name to an IP address.
- The DNS server first checks its cache to see if it has recently resolved the domain name to an IP address.
- If the DNS server has the resolution in its cache, it returns that IP address to the browser.
- If the DNS server does not have the resolution in its cache, it sends a request to other DNS servers to resolve the domain name.
- This process continues until a DNS server that has the resolution is found.
- The IP address is returned to the browser, which then uses it to connect to the web server associated with that IP address.
- DNS is hierarchical in nature, with several types of DNS servers that work together to resolve domain names to IP addresses.
 - **Root DNS Server:** At the top of the hierarchy is the Root DNS server. It contains information about the top-level domain (TLD) servers, such as .com, .org, and .edu. When a DNS request is made, the root server is the first point of contact and it responds with the IP address of the appropriate TLD server.
 - **TLD DNS Server:** TLD servers are responsible for managing the information for the specific TLDs they are responsible for. For example, a .com TLD server will have information about all the domain names that end in .com. When a request is made to a TLD server, it responds with the IP address of the appropriate second-level domain (SLD) server.
 - **SLD DNS Server:** SLD servers are the final step in the DNS resolution process. They contain information about specific domain names and their associated IP addresses. When a request is made to an SLD server, it responds with the IP address associated with the domain name.
- Each step of the DNS resolution process is performed in order, starting with the root server and ending with the SLD server. This allows for efficient and accurate resolution of domain names to IP addresses, even as new domain names are added or existing ones change IP addresses.



Amazon Route 53

- A highly available, scalable, fully managed and Authoritative DNS
 - Authoritative = the customer (you) can update the DNS records
- Route 53 is also a Domain Registrar
- Ability to check the health of your resources
- The only AWS service which provides 100% availability SLA
- Why Route 53? 53 is a reference to the traditional DNS port

Route 53

Route 53 - Records

- How you want to route traffic for a domain
- Each record contains:
 - Domain/subdomain Name – e.g., example.com
 - Record Type – e.g., A or AAAA
 - Value – e.g., 12.34.56.78
 - Routing Policy – how Route 53 responds to queries
 - TTL – amount of time the record cached at DNS Resolvers
- Route 53 supports the following DNS record types:
 - (must know)A /AAAA / CNAME / NS
 - (advanced)CAA/DS/MX/NAPTR/PTR/SOA/TXT/SPF/SRV

Route 53 - Record Types

- **A** – maps a hostname to IPv4
- **AAAA** – maps a hostname to IPv6
- **CNAME** – maps a hostname to another hostname
 - The target is a domain name which must have an A or AAAA record
 - Can't create a CNAME record for the top node of a DNS namespace (Zone Apex)
 - Example: you can't create for example.com, but you can create for www.example.com
- **NS** – Name Servers for the Hosted Zone
- Control how traffic is routed for a domain

Route 53 - Hosted Zones

- A container for records that define how to route traffic to a domain and its subdomains
- **Public Hosted Zones** – contains records that specify how to route traffic on the Internet (public domain names) application1.mypublicdomain.com
- **Private Hosted Zones** – contain records that specify how you route traffic within one or more VPCs (private domain names) application1.company.internal
- You pay **\$0.50** per month per hosted zone

Feature	Public Hosted Zones	Private Hosted Zones
Purpose	To make your resources available to the internet	To make your resources available only to resources within your VPC
Resolves to	Public IP addresses	Private IP addresses within your VPC
Access	Available to anyone on the internet	Only available to resources within your VPC
Example Use Case	A website hosted on an EC2 instance	An application hosted on an EC2 instance in a private subnet

Hosted Zones

Route 53 - Records TTL (Time To Live)

- **High TTL – e.g., 24 hr**
 - Less traffic on Route 53
 - Possibly outdated records
- **Low TTL – e.g., 60 sec.**
 - More traffic on Route 53 (\$\$)
 - Records are outdated for less time
 - Easy to change records
- **Except for Alias records, TTL is mandatory for each DNS record**

CNAME vs Alias

- AWS Resources (Load Balancer, CloudFront...) expose an AWS hostname:
 - lb1-1234.us-east-2.elb.amazonaws.com and you want myapp.mydomain.com
- **CNAME:**
 - Points a hostname to any other hostname. (app.mydomain.com => blabla.anything.com)
 - **ONLY FOR NON ROOT DOMAIN (aka. something.mydomain.com)**
- **Alias:**
 - Points a hostname to an AWS Resource (app.mydomain.com => blabla.amazonaws.com)
 - **Works for ROOT DOMAIN and NON ROOT DOMAIN (aka mydomain.com)**
 - Free of charge
 - Native health check

Route 53 - Alias Records

- Maps a hostname to an AWS resource
- An extension to DNS functionality
- Automatically recognizes changes in the resource's IP addresses

- Automatically recognizes changes in the resource's IP addresses
- Unlike CNAME, it can be used for the top node of a DNS namespace (Zone Apex), e.g.: example.com Amazon Route 53
- Alias Record is always of type A/AAAA for AWS resources (IPv4 / IPv6)
- **You can't set the TTL**

Route 53 - Alias Records Targets

- Elastic Load Balancers
- CloudFront Distributions
- API Gateway
- Elastic Beanstalk environments
- S3 Websites
- VPC Interface Endpoints
- Global Accelerator accelerator
- Route 53 record in the same hosted zone
- **You cannot set an ALIAS record for an EC2 DNS name**

Route 53 - Routing Policies

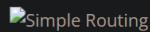
- Define how Route 53 responds to DNS queries
- Don't get confused by the word "Routing"
- It's not the same as Load balancer routing which routes the traffic
- DNS does not route any traffic, it only responds to the DNS queries
- Route 53 Supports the following Routing Policies
 - Simple
 - Multi-Value Answer
 - Weighted
 - Failover
 - Latency based
 - Geolocation
 - Geoproximity (using Route 53 Traffic Flow feature)

Routing Policies - Simple

- Typically, route traffic to a single resource
- Can specify multiple values in the same record
- **If multiple values are returned, a random one is chosen by the client**
- When Alias enabled, specify only one AWS resource
- Can't be associated with Health Checks

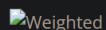
Routing Policies - Multi-Value

- Use when routing traffic to multiple resources
- Route 53 return multiple values/resources
- Can be associated with Health Checks (return only values for healthy resources)
- Up to 8 healthy records are returned for each Multi-Value query
- **Multi-Value is not a substitute for having an ELB**



Routing Policies - Weighted

- Control the % of the requests that go to each specific resource
- DNS records must have the same name and type
- Can be associated with Health Checks
- Use cases: load balancing between regions, testing new application versions.
- **Assign a weight of 0 to a record to stop sending traffic to a resource**
- **If all records have weight of 0, then all records will be returned equally**



Routing Policies - Latency-based

- Redirect to the resource that has the least latency close to us
- Super helpful when latency for users is a priority
- **Latency is based on traffic between users and AWS Regions**
- Can be associated with Health Checks (has a failover capability)

Route 53 - Health Checks

- HTTP Health Checks are only for **public resources**
- Health Check => Automated DNS Failover:
 - Health checks that monitor an endpoint (application, server, other AWS resource)
 - Health checks that monitor other health checks (Calculated Health Checks)
 - Health checks that monitor CloudWatch Alarms (full control !!) – e.g., throttles of DynamoDB, alarms on RDS, custom metrics (helpful for private resources)

- Health Checks are integrated with CW Metrics

Health Checks - Monitor an Endpoint

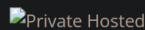
- About 15 global health checkers will check the endpoint health
 - Healthy/UnhealthyThreshold=3(default)
 - Interval – 30 sec (can set to 10 sec – higher cost)
 - Supported protocol:HTTP, HTTPS and TCP
 - If > 18% of health checkers report the endpoint is healthy, Route 53 considers it Healthy. Otherwise, it's Unhealthy
 - Ability to choose which locations you want Route 53 to use
- Health Checks pass only when the endpoint responds with the 2xx and 3xx status codes
- Health Checks can be setup to pass / fail based on the text in the first 5120 bytes of the response
- Configure your router/firewall to allow incoming requests from Route 53 Health Checkers

Route 53 - Calculated Health Checks

- Combine the results of multiple Health Checks into a single Health Check
- You can use **OR, AND, or NOT**
- Can monitor up to 256 Child Health Checks
- Specify how many of the health checks need to pass to make the parent pass
- Usage: perform maintenance to your website without causing all health checks to fail

Health Checks - Private Hosted Zones

- Route 53 health checkers are outside the VPC
- They can't access **private** endpoints (private VPC or on-premises resource)
- You can create a **CloudWatch Metric** and associate a **CloudWatch Alarm**, then create a Health Check that checks the alarm itself



Routing Policies - Failover (Active-Passive)

- A failover routing policy is a routing policy used to route traffic to a primary resource, such as an Amazon Elastic Compute Cloud (EC2) instance, while monitoring a secondary resource as a backup.
- If the primary resource becomes unavailable, traffic is automatically routed to the secondary resource.
- This type of routing policy is useful for ensuring high availability of your application by minimizing the impact of a failure of the primary resource.
- AWS Route 53 service supports failover routing policy, where you can configure a primary resource and a secondary resource and Route 53 will monitor the health of both the resources and route traffic to the healthy resource.
- This can be done by creating a health check for each resource, and then configuring the routing policy to route traffic to the primary resource as long as it is healthy, and to the secondary resource if the primary resource becomes unhealthy.



Routing Policies - Geolocation

- Different from Latency-based!
- **This routing is based on user location**
- Specify location by Continent, Country or by US State (if there's overlapping, most precise location selected)
- Should create a **"Default"** record (in case there's no match on location)
- Use cases: website localization, restrict content distribution, load balancing, ...
- Can be associated with Health Checks



Routing Policies - Geoproximity

- Route traffic to your resources based on the geographic location of users and resources
- Ability to **shift more traffic to resources based** on the defined bias
- To change the size of the geographic region, specify bias values:
 - To expand (1 to 99) – more traffic to the resource
 - To shrink (-1 to -99) – less traffic to the resource
- Resources can be:
 - AWS resources (specify AWS region)
 - Non-AWS resources (specify Latitude and Longitude)
- You must use Route 53 **Traffic Flow** to use this feature

Route 53 - Traffic flow

- Simplify the process of creating and maintaining records in large and complex configurations

Simplifying the process of creating and maintaining records in large and complex configurations

- Visual editor to manage complex routing decision trees
- Configurations can be saved as **Traffic Flow Policy**
- Can be applied to different Route 53 Hosted Zones (different domain names)
- Supports versioning

Domain Registrar vs. DNS Service

- You buy or register your domain name with a Domain Registrar typically by paying annual charges (e.g., GoDaddy, Amazon Registrar Inc., ...)
- The Domain Registrar usually provides you with a DNS service to manage your DNS records
- But you can use another DNS service to manage your DNS records
- Example: purchase the domain from GoDaddy and use Route 53 to manage your DNS records

GoDaddy as Registrar & Route 53 as DNS Service

- GoDaddy is a domain registrar, which means they allow you to register and manage domain names.
- Route 53 is a DNS (Domain Name System) service provided by Amazon Web Services (AWS). It allows you to manage and route internet traffic to your domain using DNS records, such as A, MX, and CNAME records.
- By using GoDaddy as your registrar and Route 53 as your DNS service, you would be able to register your domain with GoDaddy and then manage the DNS records for that domain using Route 53.



3rd Party Registrar with Amazon Route 53

- If you buy your domain on a 3rd party registrar, you can still use Route 53 as the DNS Service provider
 1. Create a Hosted Zone in Route 53
 2. Update NS Records on 3rd party website to use Route 53 **Name Servers**
- **Domain Registrar != DNS Service**
- But every Domain Registrar usually comes with some DNS features

