AWS Route 53 - Study Notes for AWS Certified Solutions Architect Associate Exam

---

What is DNS (Domain Name System)?

DNS is a hierarchical and decentralized naming system that translates human-readable domain names (like www.example.com) into IP addresses (like 192.0.2.1), which computers use to identify each other on the network.

It is often referred to as the "phonebook" of the internet, resolving domain names to their corresponding IP addresses.

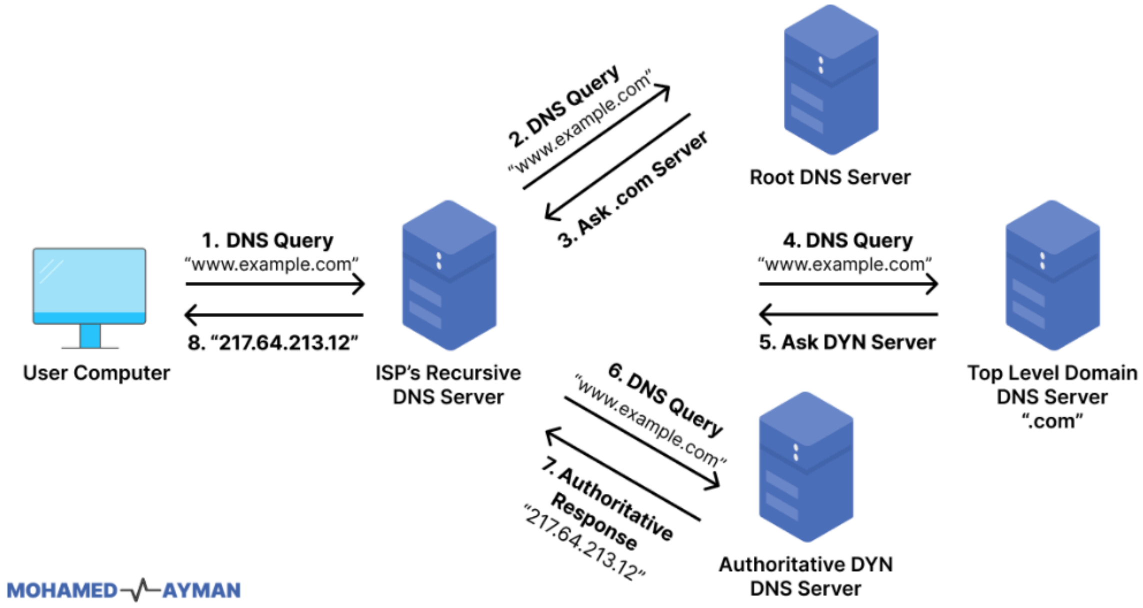
DNS Hierarchical Structure

Given the URL: `http://api.www.example.com.`

Components:

- **Protocol:** `http`
  - Defines the communication protocol used between client and server.
- **URL:** Uniform Resource Locator.
  - Example: `http://api.www.example.com` is the full address used to access a resource over the web.
- **FQDN (Fully Qualified Domain Name):** `api.www.example.com.`
  - A complete domain name including all levels in the hierarchy, ending with a dot (.) to represent the root.
- **Root:** Represented by a trailing dot (.) – the top of the DNS hierarchy.

- **TLD (Top Level Domain):** `com`
  - Managed by registries (e.g., Verisign for .com).
- **SLD (Second Level Domain):** `example`
  - The registered domain name (usually owned by an individual or organization).
- **Subdomain:** `www`, `api`
  - Used to organize or structure different services of a domain.



AWS Route 53

Amazon Route 53 is a scalable and highly available DNS web service designed to give developers and businesses an extremely reliable and cost-effective way to route end users to internet applications.

Main Functions of Route 53:

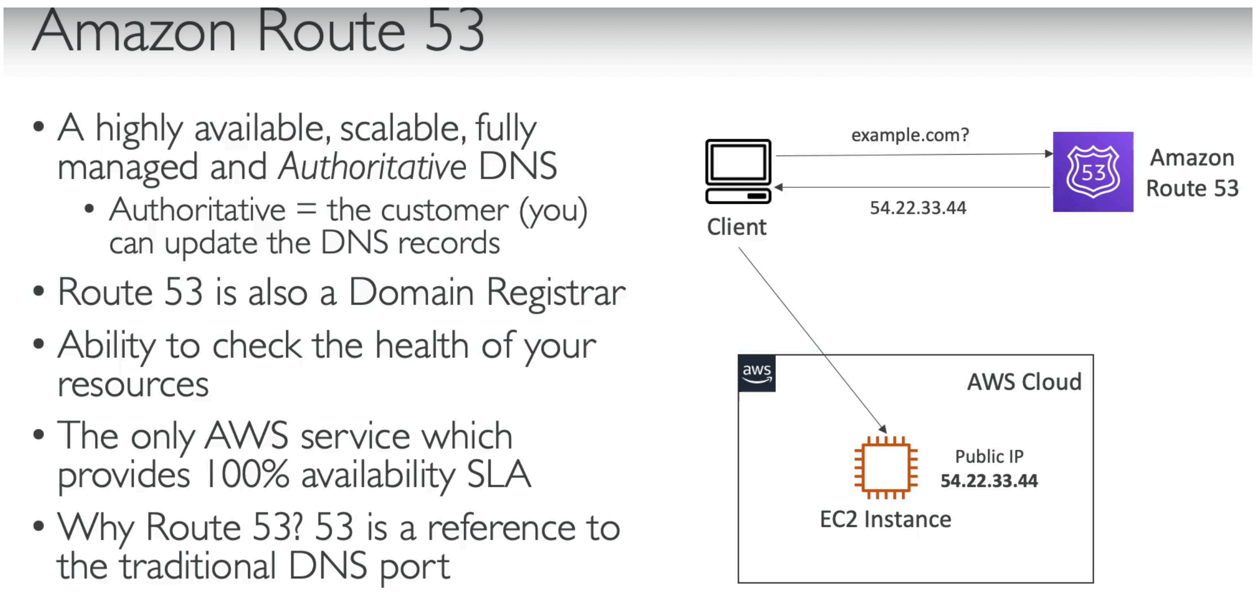
1. **Domain registration**
2. **DNS service** (resolve domain names)
3. **Health checking**

Route 53 Records

A DNS record in Route 53 defines how you want to route traffic for a domain or subdomain.

Each record includes:

- **Name:** The domain or subdomain (e.g., www.example.com)
- **Type:** The DNS record type (e.g., A, CNAME)
- **TTL (Time to Live):** Duration (in seconds) that the record is cached by DNS resolvers.
- **Value:** The data associated with the record (e.g., IP address or another domain name).
- **Routing Policy:** Determines how Route 53 responds to DNS queries (e.g., simple, weighted, latency-based, etc.)



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

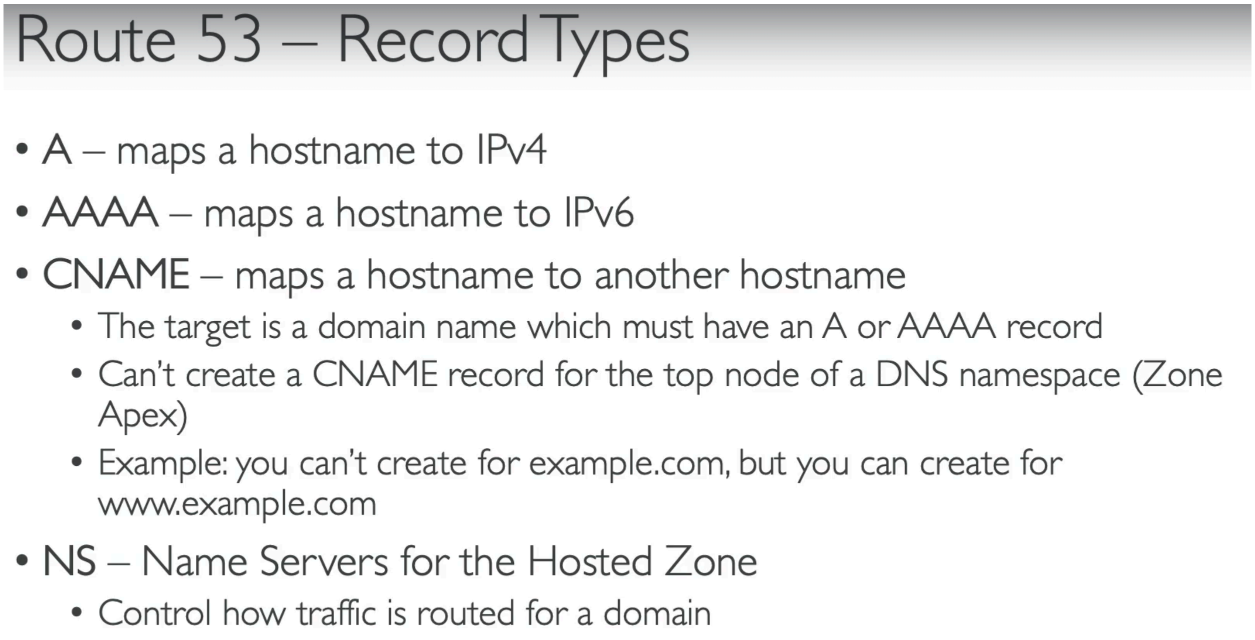
example.com?
54.22.33.44
Client
Amazon Route 53
53
aws
AWS Cloud
Public IP
54.22.33.44
EC2 Instance

Common Route 53 Record Types

1. **A Record (Address Record)**
   - Maps a domain name to an IPv4 address.
   - Example: `www.example.com -> 192.0.2.1`
2. **AAAA Record (IPv6 Address Record)**
   - Maps a domain name to an IPv6 address.
   - Example: `www.example.com -> 2001:db8::1`
3. **CNAME Record (Canonical Name)**
   - Maps a domain name to another domain name.
   - Used to alias one domain to another.
   - Example: `blog.example.com -> www.example.com`

4. **NS Record (Name Server)**

   - Specifies the authoritative name servers for the domain.
   - Example: `example.com NS -> ns-123.awsdns-45.org`



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 **Hosted Zone** is a container for records associated with a domain name.

There are two types of hosted zones:

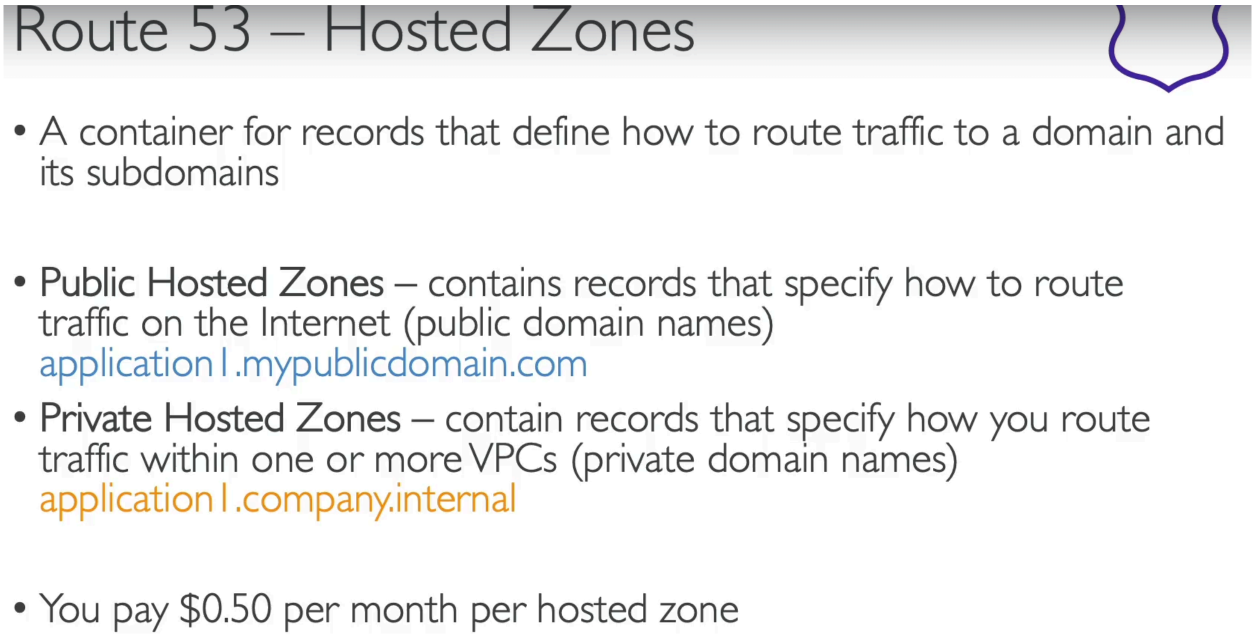
1. **Public Hosted Zone**

   - Used to manage public DNS records for a domain (accessible over the internet).

2. **Private Hosted Zone**

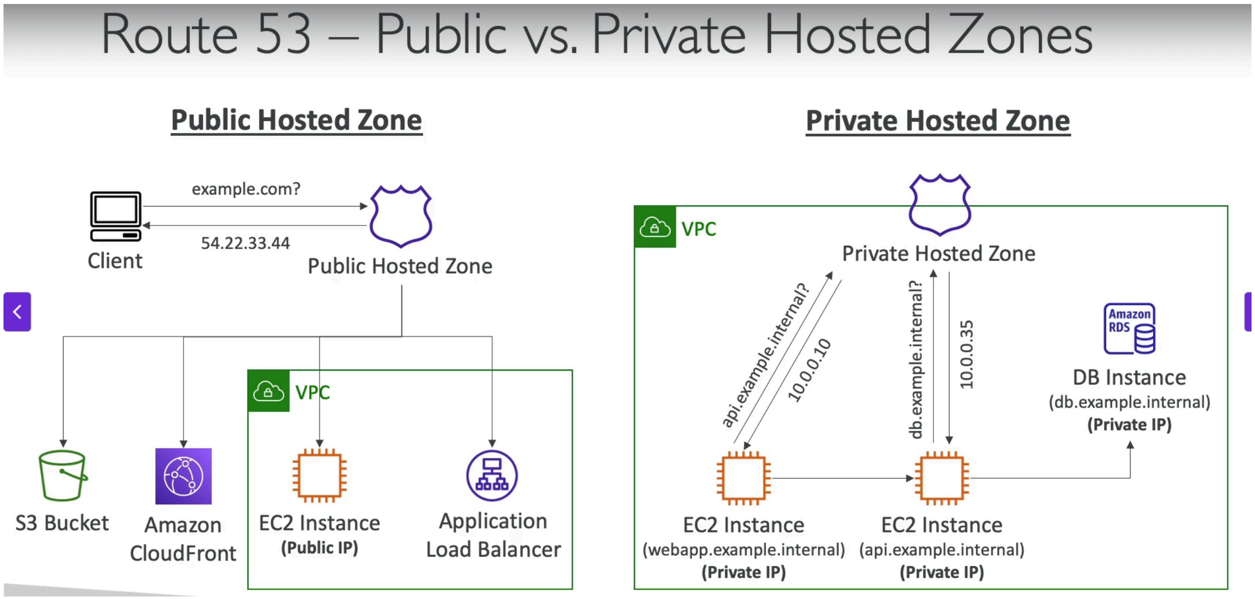
   - Used to manage DNS records for a VPC (Virtual Private Cloud).
   - Not accessible from the internet.

When you create a hosted zone, Route 53 automatically creates default NS and SOA (Start of Authority) records.



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



Route 53 – Public vs. Private Hosted Zones

Route 53 TTL and How It Works

TTL **(Time To Live)** is the amount of time, in seconds, that DNS resolvers (like your ISP or browser) are allowed to cache a **DNS record before querying Route 53 again** for updated information.

📘 How TTL Works – Step by Step:

1. When a user tries to access `example.com`, their device asks a DNS resolver (usually provided by an ISP) for the IP address.
2. The resolver checks if it has a cached answer for `example.com`.
3. If the record is **cached and TTL has not expired**, it returns the cached IP address.
4. If **TTL has expired**, the resolver queries Route 53 again to fetch the latest DNS record.
5. After retrieving the fresh record, it resets the TTL timer and caches it again.

🕒 TTL Example Timings:

- TTL = `300` seconds → Cache expires after 5 minutes.
- TTL = `3600` seconds → Cache expires after 1 hour.
- TTL = `86400` seconds → Cache expires after 24 hours.

🔼 High TTL

- **Definition**: TTL values of 1 hour (`3600s`) to 1 day (`86400s`) or more:
- **Pros**:
    - Reduces repeated DNS lookups → less load on Route 53.
    - Improves performance by returning cached results faster.
- **Cons**:
    - Changes to DNS (like IP address updates or failover switches) take longer to propagate.
- **Common TTL Values**: `3600`, `7200`, `86400`

- **Use Cases**:
    - Static websites
    - Stable IP addresses
    - Non-critical applications

🔽 Low TTL

- **Definition**: TTL values of a few seconds to a few minutes (e.g., `30s`, `60s`, `300s`)
- **Pros**:
    - Faster DNS updates and failover response.
      Ideal for dynamic or frequently changing services.
- **Cons**:
    - Higher DNS query volume → may increase cost and latency slightly.
- **Common TTL Values**: `30`, `60`, `300`
- **Use Cases**:
    - Load balancers
    - Blue/green deployments
    - Failover setup
    - Environments with frequent IP changes (like EC2 with dynamic IPs).

CNAME vs Alias Record in Route 53

Both CNAME and Alias records are used to map one domain name to another. However, there are critical differences in their behavior and use cases.

CNAME (Canonical Name) Record

- Maps a domain to another domain name.
- Cannot be used at the root (apex) level of a domain.
- Supported by standard DNS services.
- Adds a small delay since it requires an extra lookup.

Alias Record (Route 53-specific)

- Similar to CNAME but works at the apex level and integrates with AWS resources.
- Can point to AWS services like ELB, CloudFront, API Gateway, S3 (static website).
- Functions like an A record and returns an IP address.
- Does not incur a DNS query charge when pointing to AWS resources.

What is an APEX Domain?

An **APEX domain** (aka root domain or zone apex) is the base domain name without any subdomain prefix.

Examples:

- ✅ `example.com` → APEX domain
- ❌ `www.example.com` → Not APEX
- ❌ `api.example.com` → Not APEX

Standard DNS rules do not allow CNAME records at the apex domain.

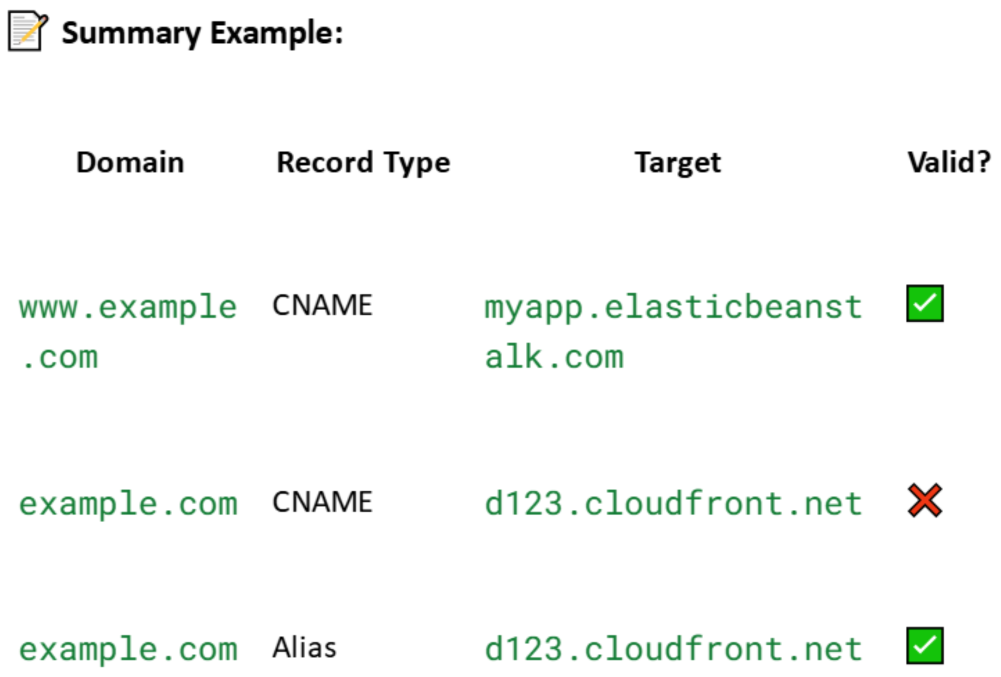
APEX Domain with CNAME and Alias

Why CNAME doesn’t work with APEX:

DNS protocol prohibits using **CNAME at the root domain**, because it can conflict with other essential DNS records (like NS or SOA).

How Alias Solves This:

Route 53 allows **Alias records at the APEX domain**, enabling users to point root domains to AWS services without violating DNS standards.



📝 Summary Example:

Domain	Record Type	Target	Valid?
`www.example.com`	CNAME	`myapp.elasticbeanstalk.com`	✅
`example.com`	CNAME	`d123.cloudfront.net`	❌
`example.com`	Alias	`d123.cloudfront.net`	✅

Routing Policies in Route 53

Routing policies define how Route 53 responds to DNS queries:

1. Simple Routing Policy

- Default routing method.
- Maps a domain to a single resource (IP or domain).
- No health checks.
- Best for simple, static websites.

2. Weighted Routing Policy

- Distributes traffic across multiple resources based on weights.
- Assign a weight to each record.
- Traffic(%) = Weight of record / Sum of all weights
- Use Cases: Load balancing, A/B testing, traffic shifting.
- Supports health checks.

3. Latency Routing Policy

- Routes traffic based on the lowest latency between the user and AWS region.
- Improves performance by reducing response time.
- Requires records in multiple AWS regions.

4. Failover Routing Policy

- Provides active-passive failover.
- Primary record is used unless a health check fails, then traffic is routed to the secondary.
- Use Case: High availability and DR (Disaster Recovery).

5. Geolocation Routing Policy

- Routes traffic based on the geographic location of the user.
- Helps serve region-specific content.

6. Geoproximity Routing Policy

- Routes traffic based on the location of AWS resources and optionally biases traffic distribution.
- Requires Route 53 traffic flow.

7. IP-Based Routing Policy

- Routes traffic based on the user's IP address or IP range.
- Useful for custom routing rules based on known client networks.

Using Route 53 with a Third-Party Domain Registrar

Even if you purchase your domain from a third-party registrar (like GoDaddy, Namecheap, etc.), you can **still use Amazon Route 53 as your DNS service provider**. Here's how:

Steps to Use Route 53 with a 3rd-Party Domain:

1. **Create a Hosted Zone in Route 53**:
   - Go to Route 53 console.
   - Create a Public Hosted Zone for your domain (e.g., `example.com`). Route 53 will automatically generate NS (Name Server) and SOA (Start of Authority) records.

2. **Update NS Records at the Registrar**:
   - Go to your domain registrar's website.
   - Find the option to update Name Servers (usually under DNS or domain settings).
   - Replace the default name servers with the **4 NS values provided by Route 53** (e.g., `ns-123.awsdns-45.com`).

3. **Propagation Time**:
   - DNS changes (like name server updates) may take **up to 48 hours** to propagate globally.
   - After that, Route 53 becomes the authoritative DNS provider for your domain.

Why Use Route 53?

- Seamless integration with AWS services (EC2, CloudFront, S3).
- Advanced routing policies (latency-based, geolocation, failover).
- Health checks and DNS failover support.