# DNS (Domain Name Server)

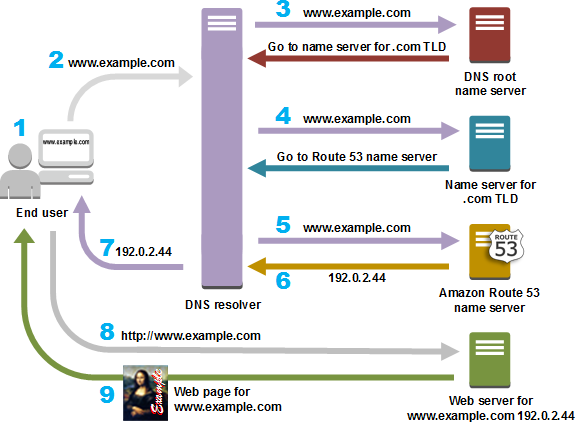
 DNS servers maintains records for [Refer Here](https://www.site24x7.com/learn/dns-record-types.html) for record types  name to ipv4 address mapping (A Record)

 name to ipv6 address mapping (AAAA Record)  alias name (C Record)

 Mail server Record (MX Record)

 DNS Servers are of two types

 public dns: These are hosted for public access. Generally all domain seller host public dns.  private dns: This is maintained by your organization for internal records



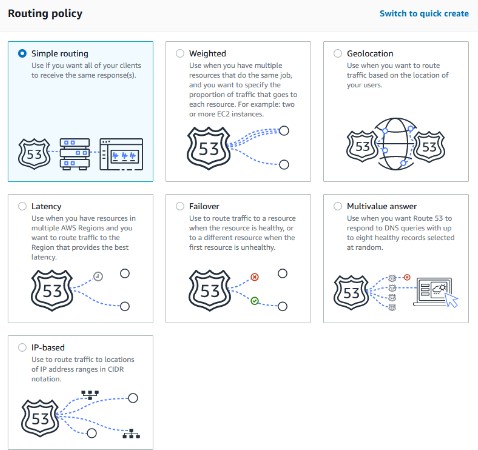
# DNS in AWS

 DNS as a service in AWS is called as Route 53

 using Route 53 we can host both public and private DNS  Route 53 also has a Domain purchase options

 Route 53 can host Name servers as part of hosting zones where we can maintain DNS Records

## Routing Polcies

 Overview

Amazon Route 53 is a scalable Domain Name System (DNS) web service that provides various routing policies to manage how DNS queries are handled. These routing policies allow users to direct traffic based on specific criteria, enhancing performance and reliability. Below are the seven main routing policies offered by Route 53:

1. **Simple Routing Policy**

This is the most straightforward option, used when a single resource (like a web server) handles requests for a domain. It allows for multiple values (e.g., IP addresses) to be returned, but does not support health checks. If a resource is unhealthy, Route 53 will still return its IP address to clients.

1. **Weighted Routing Policy**

This policy allows users to assign weights to different resources, controlling the proportion of traffic directed to each. For example, if one resource is assigned a weight of 25 and another a weight of 75, 25% of the

requests will go to the first resource, while 75% will go to the second. This is useful for testing new features or balancing traffic.

1. **Latency-Based Routing**

Latency-based routing directs users to the AWS region that offers the lowest latency, improving the speed of content delivery. This is particularly beneficial for applications with resources deployed in multiple regions.

1. **Failover Routing Policy**

This policy is designed for active-passive failover setups. It routes traffic to a primary resource and switches to a secondary resource if the primary fails. Health checks are integral to this policy, ensuring that traffic only

goes to healthy resources[1][3].

1. **Geolocation Routing Policy**

Geolocation routing directs traffic based on the geographic location of the user. This can be useful for

localizing content or complying with regulatory requirements by serving different content based on where the request originates.

1. **Multivalue Answer Routing Policy**

This policy allows Route 53 to return multiple healthy resource records in response to a DNS query, up to eight records. It also performs health checks on the resources, ensuring that only healthy IP addresses are returned. This is not a substitute for a load balancer but can enhance availability and load distribution

1. **IP-Based Routing Policy**

IP-based routing allows users to route traffic based on the IP address of the client. This can be useful for directing users to specific resources based on their IP address range, enhancing control over traffic

management.

**Conclusion**

Each routing policy in Amazon Route 53 serves specific use cases, allowing for flexible and efficient traffic

management. By selecting the appropriate policy, users can optimize the performance and reliability of their applications hosted on AWS.

# DNS in Azure

 DNS as a service in Azure is called as Azure DNS

 using Azure DNS we can host both public and private DNS  Azure DNS does not have Domain purchase options

 Azure DNS also can host Name servers as part of hosting zones where we can maintain DNS Records  Azure Traffic manager can make routing decision [Refer Here](https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview)

Azure Traffic Manager

Azure Traffic Manager offers several routing methods to distribute traffic across endpoints based on different criteria. Here are the main routing methods:

# Priority Routing

This method directs traffic to the endpoint with the highest priority. If the primary endpoint is unavailable, Traffic Manager automatically fails over to the secondary endpoint. This is useful for implementing active- passive failover scenarios[1][2][3].

# Weighted Routing

The Weighted routing method allows you to distribute traffic across endpoints based on predefined weights. You assign a weight (1-1000) to each endpoint. Traffic is distributed randomly among the available endpoints, with the probability proportional to the assigned weights[1][2][3].

# Performance Routing

Performance routing directs traffic to the endpoint with the lowest latency relative to the user. Traffic Manager maintains a database of network latencies between IP address ranges and Azure datacenters. It uses this data to determine the optimal endpoint for each user[1][2][3].

# Geographic Routing

Geographic routing enables you to route traffic based on the geographic location of the user. You define geographic regions and map endpoints to each region. Traffic Manager then directs users to the endpoint associated with their geographic location[1][3].

# Multivalue Routing

Multivalue routing returns multiple healthy endpoints in response to each DNS query. This allows clients to directly connect to any of the returned endpoints. This is useful for scenarios like content delivery networks where multiple endpoints can serve the same content[1][3].

# Subnet Routing

Subnet routing allows you to map groups of client IP address ranges to specific endpoints. When a client makes a request, Traffic Manager returns the endpoint mapped to the client's IP address[3].

The choice of routing method depends on the specific needs of your application, such as failover

requirements, load distribution, performance optimization, or geographic constraints. Traffic Manager

supports mixing and nesting of routing methods to create sophisticated traffic management configurations[1] [2].

Citations: [1] [https://www.cloudthat.com/resources/blog/choosing-the-right-azure-traffic-manager-routing-](http://www.cloudthat.com/resources/blog/choosing-the-right-azure-traffic-manager-routing-)

method-strategy-for-application-deployment [2] <http://gowie.eu/index.php/traffic-manager/routing-methods>

[3] https://k21academy.com/microsoft-azure/az-303/azure-traffic-manager/ [4]

https://intellipaat.com/blog/azure-traffic-manager/ [5] https://learn.microsoft.com/th-th/azure/traffic-

manager/traffic-manager-overview [6] https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-

manager-how-it-works [7] https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-

configure-priority-routing-method [8] https://disaster- recovery.workshop.aws/en/services/networking/route53/routing-policies.html

## Exercise: Demonstrate path based routing

 Demonstrate path based routing in  AWS

 Azure