What does Route 53 do?

Amazon Route 53 is a highly available and scalable DNS web service provided by Amazon Web Services (AWS). It acts as a digital phonebook for the internet, translating human-readable domain names into IP addresses that computers use to identify each other.

When you type in a web address, how does your computer know where to find that site? Route 53 takes the website name and resolves it to the corresponding IP address. This action helps you reach your desired destination on the internet.

Route 53 provides a reliable and cost-effective way to route end users to internet applications. You can benefit from Route 53 whether you're running a small personal blog or managing a large-scale enterprise application. Route 53 offers tools and flexibility to help your users access your online resources quickly and reliably.

You can use Route 53 to register domain names. You can work with both public and private hosted zones to route different kinds of traffic. Route 53 also provides a quick way to check the health of your resources. You can set up Amazon CloudWatch alarms and receive notifications if the status of the health check changes.

https://aws.amazon.com/route53/

What problems does Route 53 solve?

A primary problem that Route 53 addresses is the challenge of reliably and efficiently connecting users to applications hosted on AWS or any cloud. As organizations move their applications and infrastructure to the cloud, they need a way to map human-readable domain names (such as example.com) to their resources' IP addresses. These IP addresses can frequently change. Route 53 provides a highly available and scalable DNS service that translates domain names into IP addresses, to provide seamless connectivity for end users.

Additionally, Route 53 helps organizations overcome the complexities of managing DNS records across multiple Regions and resources. It simplifies the process of routing traffic to various AWS resources, based on predefined rules and conditions. For example, these resources might include Elastic Load Balancing (ELB), Amazon Simple Storage Service (Amazon S3) buckets, or other cloud services.

Route 53 helps you address the following challenges:

Domain name resolution
Route 53 seamlessly translates domain names to IP addresses.

Global traffic management
Route 53 routes traffic intelligently across multiple AWS Regions.

Health checking and failover
Route 53 monitors resource health and automatically redirects traffic.

DNS configuration management
Route 53 centrally manages and updates DNS records efficiently.

Scalability and availability
Route 53 is highly available and scales to handle high DNS query volumes.

Integration with AWS services
Route 53 integrates natively with other AWS services and resources.

What are the benefits of Route 53?

Review the following to learn more about the benefits of Route 53.

High availability and reliability

You can rely on the Route 53 global network of DNS servers to keep your domain names accessible, reduce downtime, and improve user experience. With its distributed infrastructure, Route 53 handles large volumes of queries and withstands potential failures.

Seamless AWS service integration

Route 53 integrates smoothly with other AWS services. It helps you manage DNS records for your AWS resources such as Amazon Elastic Compute Cloud (Amazon EC2) instances, S3 buckets, and Amazon CloudFront distributions. This streamlined connection simplifies your workflow and enhances overall efficiency.

<u>Advanced traffic management and routing policies</u>

You can optimize your application's performance, improve user experience, and implement complex routing scenarios tailored to your needs with the advanced features of Route 53. For example, some of these features include latency-based routing, geolocation routing, and weighted round-robin routing.

<u>Built-in health checks and monitoring</u>

Route 53 provides built-in health checks and monitoring for your resources. It automatically routes traffic away from unhealthy endpoints to ensure high availability for your applications. You can set up alerts to notify you of any issues and then facilitate proactive infrastructure management.

<u>Public, privtae, and hybrid zone resolution</u>

Route 53 gives you the option of public and private hosted zones. With public hosted zones, you can specify how you want to route traffic on the internet. With private hosted zones, you specify how to route traffic in your Amazon Virtual Private Cloud (Amazon VPC). You can also use Route 53 for hybrid networks.

Route 53 features https://aws.amazon.com/route53/features/

Route 53 offers tools and flexibility to help your users access your online resources quickly and reliably.

How much does Route 53 cost?

With Route 53, you don't need to pay any upfront fees or commit to the number of queries the service will answer for your domain. As with other AWS services, you pay as you go and only for what you use, which provides cost effectiveness and flexibility.

Route 53 charges you based on the resources that you use and the traffic that you route through the service. These charges include the following:

- You pay a monthly fee for each hosted zone, which varies by domain type (.com, .org, and others). This fee covers management of your DNS records.

- You pay an annual fee for each domain name that is registered by Route 53 or transferred into Route 53.

- You pay a fee per query for DNS queries to your hosted zones. The fee varies by query type and the originating query location.

- Optional features incur additional usage-based charges, including items such as

health checks and traffic flow policies.

Route 53 pricing https://aws.amazon.com/route53/pricing/

AWS Pricing Calculator https://calculator.aws/#/

Knowledge check

Q1. What is a key benefit of using Amazon Route 53?

- High availability and reliability for DNS services (Correct)
- Automatic scaling of Amazon EC2 instances
- Built-in content delivery network
- Serverless compute capabilities

Route 53 provides high availability and reliability through its global network of DNS servers, thus reducing downtime and improving user experience.

AWS Auto Scaling is a separate AWS service for EC2 instances. Though Route 53 can integrate with Amazon CloudFront, it is not itself a content delivery network. AWS Lambda provides serverless compute capabilities.

Q2. What is the primary function of Amazon Route 53?

- Load balancing of network traffic
- Encryption of data in transit
- Translation of domain names to IP addresses (Correct)
- Management of virtual private clouds

Amazon Route 53 primarily functions as a DNS web service. It translates human-readable domain names into IP addresses that computers use to identify each other on the internet.

Though Route 53 can be used in conjunction with load balancing, this function is not its primary purpose. Load balancing is primarily handled by services like Elastic Load Balancing. Route 53 does not handle encryption. Managing encryption is typically done by services such as AWS Certificate Manager (ACM) or through the use of SSL/TLS protocols. Virtual private clouds (VPCs) are managed by using the Amazon VPC service.

Q3. Which problem does Amazon Route 53 solve for organizations that are moving their applications to the cloud?

- Deploying code automatically
- Creating virtual private networks
- Managing server hardware

- Mapping domain names to changing IP addresses

Route 53 solves the challenge of mapping human-readable domain names to IP addresses of cloud resources, which can frequently change in dynamic cloud environments.

Automatic code deployment is typically handled by services like AWS CodeDeploy, not Route 53, which focuses on DNS and routing. Creating virtual private networks is managed by Amazon Virtual Private Cloud (Amazon VPC), not Route 53, which is primarily a DNS service. AWS manages the underlying hardware in the cloud. Route 53 is a DNS service and doesn't deal with physical hardware management.

Architecture and Use Cases

How is Route 53 used to architect a cloud solution?

Route 53 plays a vital role in architecting cloud solutions by seamlessly integrating with various AWS services. This integration enhances the reliability, performance, and scalability of applications deployed on AWS. Understanding how Route 53 interacts with other AWS components is essential for you to design robust and efficient cloud architectures.

To learn more about how Route 53 is used to architect a cloud solution, choose each of the seven numbered markers.

1 User request

A user opens a web browser, enters www.example.com in the address bar of their device for the first time, and presses Enter.

2 DNS resolver

The request for www.example.com is routed to a DNS resolver, which is typically managed by the user's internet service provider (ISP). The ISP might be a cable internet provider, a DSL broadband provider, or a corporate network.

3 DNS root name server

The DNS resolver for the ISP forwards the request for www.example.com to a DNS root name server. The DNS root name server directly answers requests for records in the root zone. It also answers other requests by returning a list of the name servers for the

appropriate top-level domain.

4 Reroute to top-level domain server

When it doesn't find the location of the website, the DNS resolver forwards the request again. This time, it forwards the request to one of the top-level domain name servers for .com domains.

5 Name server response

The name server for .com domains responds to the request with the names of the four Route 53 name servers associated with the example.com domain. The DNS resolver caches the four Route 53 name servers for an amount of time that you specify.

6 DNS resolver chooses Route 53

The DNS resolver chooses a Route 53 name server and forwards the request for www.example.com. The Route 53 name server looks in the example.com hosted zone for the www.example.com record. It gets the associated value, such as the IP address for a web server, 192.0.2.3, and returns the IP address to the DNS resolver.

7 IP address retrieved

The DNS resolver finally has the IP address that the user needs. The resolver returns that value to the web browser and continues with the user request.

Which AWS services integrate with Route 53?

Route 53 integrates with many AWS services. By using these connections, you can design architectures that are highly available, scalable, and high performing. You can seamlessly connect DNS management with compute, storage, content delivery, and serverless technologies to address complex application requirements and deliver optimal user experiences.

Review the following to learn more about AWS services that you can integrate with Route 53.

Amazon EC2

You can direct traffic to EC2 instances, to enable load balancing and high availability for web applications.

Elastic Load Balancing (ELB)

Route 53 works with ELB to distribute incoming traffic across multiple EC2 instances and improve application performance and fault tolerance.

Amazon S3

You can route requests to S3 buckets for static website hosting and to provide a cost-effective solution for serving content.

Amazon CloudFront

Integration with CloudFront lets you direct users to the nearest content delivery network (CDN) edge location, to reduce latency and improve user experience.

AWS Global Accelerator

You can work with Global Accelerator to optimize the path between users and applications and enhance performance for global audiences.

Amazon API Gateway

You can route traffic to API Gateway endpoints to facilitate the management and deployment of APIs.

AWS Lambda

What are the basic technical concepts of Route 53?

Route 53 connects user requests to infrastructure that runs in AWS, such as EC2 instances, load balancers, or S3 buckets. It translates human-readable domain names into IP addresses so that you can access applications and services that are hosted on AWS. Route 53 monitors the health of your resources and automatically routes traffic away from unhealthy resources to ensure high availability and reliability.

Review the following to learn more about the technical concepts of Route 53.

DNS

DNS is a naming system that translates human-readable domain names into machine-readable IP addresses, to help you access websites and applications on the internet.

Record set

A record set is a collection of records in a hosted zone that defines how you want to route traffic for a specific domain or subdomain. Record sets contain information about routing

policies, health checks, and other settings.

Routing policy

A routing policy is a set of rules that determines how Route 53 responds to queries for a record. Route 53 supports various routing policies, including simple, weighted, latency-based, geolocation, failover, and multi-value routing.

Health check

A health check is a monitoring system that continuously checks the health of your resources and routes traffic away from unhealthy resources. Route 53 health checks can monitor the health of web servers, email servers, and other resources.

Traffic flow

A traffic flow is a visual workflow. It models how you want to route traffic based on various conditions, such as resource health, geographic location, or resource type. Traffic flows can help you manage complex routing configurations.

Hosted zone

A hosted zone is a container in Route 53 that holds information about how you want to route traffic for a specific domain and its subdomains. You can create public hosted zones for internet-accessible domains or private hosted zones for domains accessible only within your virtual private cloud (VPC).

Route 53 Resolver

Route 53 Resolver is a component of Route 53 that manages DNS resolution for VPCs. It enables DNS queries to be resolved between VPCs and on-premises environments.

Private hosted zone

Private hosted zones contain records that specify how you want to route traffic in your Amazon Virtual Private Cloud (Amazon VPC).

- Routing for internal clients

- Resolve from inside the VPC

- Support hybrid networking

Public hosted zone

Public hosted zones contain records that specify how you want to route traffic on the internet (routing for external clients).

- Routing for external clients

- Resolve from the internet

- Use global routing policies

Route 53 acts as a digital phonebook for the internet.

What are typical use cases for Route 53?

Review the following to learn more about the typical use cases of Route 53.

Global DNS management

Manage domain names and DNS records for websites and applications distributed across multiple Regions worldwide.

Domain name registration

Register a new domain name or transfer an existing domain name with Route 53.

Load balancing

Distribute incoming traffic across multiple AWS resources or on-premises servers to improve availability and performance.

Failover routing

Implement automatic failover to back up resources in case of primary resource failure to ensure high availability.

Geolocation routing

Direct users to the nearest server or provide location-specific content based on the geographic origin of DNS queries.

Latency-based routing

Route traffic to the AWS Region with the lowest latency for improved user experience and application performance.

Multi-Region deployments

Manage complex, multi-Region application architectures by routing traffic efficiently across different AWS Regions.

Route 53 takes the website name that you've entered and quickly directs your request

to the correct server that is hosting that site.

What else should you keep in mind about Route 53?

SECURING AND MONITORING ROUTE 53 CONFIGURATIONS

Because Route 53 plays a vital role in directing traffic to your applications, you should implement robust security measures and monitoring practices. Configure appropriate access controls, enable logging and monitoring for Route 53 configurations, and set up alerts for any unauthorized changes or potential security incidents. Conduct regular audits and reviews of Route 53 configurations to ensure continued compliance with security best practices.

Now that you've reviewed securing and monitoring Route 53 configurations, move to the next tab to learn about integrating Route 53 with other AWS services.

INTEGRATING ROUTE 53 WITH OTHER AWS SERVICES

Route 53 integrates seamlessly with various AWS services to help you build comprehensive and scalable solutions. Use services such as AWS CloudTrail for auditing, Amazon CloudWatch for monitoring and logging, and AWS WAF for web application security. In this way, you can enhance the overall security, observability, and resilience of your applications that are powered by Route 53.

Now that you've reviewed integrating Route 53 with other AWS services, move to the next tab to learn about implementing failover and load balancing strategies.

IMPLEMENTING FAILOVER AND LOAD BALANCING STRATEGIES

The advanced routing policies of Route 53, such as failover and geolocation routing, help you implement robust failover and load balancing strategies for your applications. By configuring these policies, you can ensure high availability, minimize downtime, and optimize performance. You can route traffic to the most appropriate resources based on factors like health checks, geographic location, or latency.

Now that you've reviewed implementing failover and load balancing strategies, move to the next tab to learn about adopting operational best practices for Route 53.

ADOPTING OPERATIONAL BEST PRACTICES FOR ROUTE 53

To effectively manage and maintain Route 53 configurations, adopt operational best practices. Implement infrastructure as code (IaC) practices, version control, and automated deployment pipelines for managing Route 53 resources. To ensure optimal performance and security, regularly review and update Route 53 configurations to align

with evolving business requirements and industry best practices.

Now that you have reviewed adopting operational best practices for Route 53, move to the next section.

Getting started with Route 53
https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/getting-started.html

Knowledge check

Q1. What is a common use case for Amazon Route 53?

- Managing virtual private networks
- Encrypting data at rest
- Automating infrastructure deployment
- Global DNS management (Correct)

Route 53 is primarily used for managing domain names and DNS records for websites and applications distributed across multiple Regions worldwide.

VPN management is typically handled by AWS Virtual Private Network (AWS VPN) or similar networking services.

Data encryption is not a function of Route 53. This function is typically handled by services like AWS Key Management Service (AWS KMS) or through encryption features in storage services. Though Route 53 can be part of automated deployments, it's not primarily used for this purpose. Services such as AWS CloudFormation or AWS Cloud Development Kit (AWS CDK) are more suited for infrastructure automation.

Q2. What is a key feature of Amazon Route 53 that helps ensure high availability of your applications?

- Automatic scaling of resources
- Health checks and DNS failover (Correct)
- Built-in firewall protection
- Serverless computing capabilities

Route 53 provides health checking capabilities to monitor the health of your resources. It can automatically route traffic away from unhealthy resources, which ensures high availability of your applications.

Though important for high availability, automatic scaling is a feature of services such as Amazon EC2 Auto Scaling, not Route 53. Firewall protection is provided by services like AWS WAF or AWS Security Groups, not Route 53. Serverless computing is provided by

services such as AWS Lambda, not Route 53. Route 53 can work with serverless applications but does not provide serverless capabilities itself.

Q3. What is a key requirement when implementing Amazon Route 53 for secure and monitored configurations?

- Configuring appropriate access controls (Correct)
- Enabling automatic scaling for all resources
- Using only private hosted zones
- Implementing multi-factor authentication for all users

Configuring appropriate access controls is a crucial requirement when implementing Route 53 securely. This practice ensures that only authorized personnel can make changes to DNS configurations, thus protecting against unauthorized modifications.

Though scaling is important for many AWS services, it's not a specific requirement for implementing Route 53 securely and with proper monitoring. Private hosted zones have their uses, but using only private hosted zones is not a requirement for secure and monitored Route 53 configurations. Many implementations use public hosted zones securely. Though multi-factor authentication is a good security practice, it's not specifically a key requirement for implementing Route 53. It's more related to overall AWS account security.

Prerequisites for the demonstrations

Learn how to set up an environment for the Route 53 demonstrations.

Transcript: Amazon Route 53 Demo Prerequisites

In this demonstration, you will set up the prerequisites for the Amazon Route 53 demonstrations.

From the AWS Management Console, enter *EC2* in the search bar. Choose **EC2**.

First, navigate to the Amazon EC2 console.

Choose **Launch instance**.

On the Launch an instance page, enter *WebServer* as the instance name.

Choose **Amazon Linux from Application and OS Images (Amazon Machine Image)**. Leave the default Amazon Machine Image (AMI). In this demo, the default is the Amazon Linux 2023 AMI.

Keep the default selections for the other choices.

Under **Instance type**, choose **t2.micro**.

You will need a key pair to connect to the instance. Choose **Create new key pair**.

Choose a name for the key pair. Leave the key pair type and private key file format at the defaults.

Choose **Create key pair**.

A dialog box will open to save the key pair.

Save the key pair to your computer in a place where you will be able to easily retrieve it later.

For **Allow SSH traffic in Network** settings, choose the source of Secure Shell (SSH) connections to the EC2 instance.

Choose the checkbox next to **Allow HTTP traffic from the internet**.

Choose **My IP** if the displayed IP address is correct for SSH connections.

If you use 0.0.0.0/0 for SSH access, you make it possible for all IP addresses to access your public EC2 instances by using SSH. This approach is acceptable for a short time in a test environment, but it's unsafe for production environments. In production, authorize only a specific IP address or range of addresses to access your EC2 instances by using SSH.

Leave the default values for the remaining sections.

In the summary panel, review a summary of your EC2 instance configuration.

Choose **Launch instance**.

A success banner will show a successful launch of the instance.

Choose **View all instances**.

Choose the checkbox next to the **Instance**.

In the **Details** tab, note the following values, which you need when you connect by using SSH.

In the Instance summary, note the value for **Public IPv4 address**.

In Instance details, note the value for **Key pair name**.

You will need to SSH into the EC2 instance to install the Apache web server.

Elevate your permissions to root by running **sudo –i**.

Now, run the command **sudo dnf update –y** to update bug fixes and security updates on your EC2 instance.

Then, install the Apache server. Run the command **yum install httpd**.

Enter *y* for yes. Now the server has been installed.

Now you can begin interacting with the server.

Enter *systemctl start httpd*.

Then, *systemctl enable httpd*.

Now, you can look at the status. Enter *systemctl status httpd*.

This will show you the status, the time running, and other items about the server.

To test the server, navigate to a new browser tab. Enter the IP address and choose enter.

The page will show that the web server is up and running.

You are now ready to begin the Amazon Route 53 demonstrations.

Thank you for watching this demonstration.

Registering a domain name

In this demo, you will learn how to register a domain name in Route 53.

Transcript: Register a Domain Name in Amazon Route 53

In this demonstration, you will learn how to register a domain name in Route 53.

From the AWS Management Console, enter *Route 53* in the search bar.

Choose **Route 53**.

On the Route 53 dashboard, choose **Registered domains**.

Choose **Register domains**.

Next, enter the domain name that you would like to use in the **Search for domain** box.

Choose **Search**.

The search result will show whether the domain that you chose is available. You can also

review a list of suggested available domains, which follows the search results.

If the domain that you request is available, choose **Select**.

Now, in the Selected domains panel, you can review the domain name and domain registration fee per year.

Choose **Proceed to checkout**.

Now, begin the checkout process.

Review the pricing. **Auto renew** is the default.

Choose **Next**.

Now, enter the contact information for the domain. You will include name, email address, phone number, and address.

Then, you can add any additional contact information if there will be a different contact for admin, tech, or billing.

Choose **Next**.

Next, you will review the information that you entered. The contact information can be edited if needed.

Select the checkbox to agree to the **Terms and conditions**.

Then, choose **Submit**.

The request status will appear, and you will receive a separate email.

Thank you for watching this demonstration.

Creating a Public Hosted Zone

In this demo, you will learn how to create a public hosted zone in Route 53 for your domain. This step is the first step in managing DNS records for your domain.

Transcript: Creating a Public Hosted Zone in Amazon Route 53

In this demonstration you will learn how to create a ublic hosted zone in Amazon Route 53. You will add DNS records and run a test.

From the AWS Management Console, enter *Route 53* into the search bar.

Choose **Route 53**.

On the Amazon Route 53 console, choose **Get started**.

Next, you will choose your starting point. Choose **Create hosted zones**.

Then, choose **Get started**.

Now, you will enter a domain name. This name is the name of the domain that you will route traffic for.

Choose **Public hosted zone**.

Choose **Create hosted zone**.

Choose **Create record**.

In the Record name box, enter *www*. Choose the record type **A – Routes traffic to an IPv4 address** and some AWS resources.

In the Value field, enter the IP address of the EC2 instance that you created in the prerequisites demo.

Choose **Create records**.

A success banner will indicate that the record was created.

Now, you can test the record in the console. Choose **Test record**.

Choose the Record type that you want to test. For this demo, choose **A – Routes traffic to an IPv4 address** and some AWS resources.

Choose **Get response**.

The returned response appears. The DNS response code shows No Error.

Thank you for watching this demonstration.

Health Checking

Learn how to set up basic health checks that monitor an endpoint and route traffic away if unhealthy.

Transcript: Health Checking with Amazon Route 53

In this demonstration, you will learn about health checking with Amazon Route 53.

From the AWS Management Console enter *Route 53* into the search bar.

Choose **Route 53**.

From the navigation panel on the Route 53 dashboard, choose **Health checks**.

Choose **Create health check**.

Now, you will configure the health check. Start by entering a name for the health check.

Next, specify the type of endpoint. For this demo, you will choose **Domain name**.

Enter the domain name of the endpoint.

Choose **Create health check**.

The health check is now created. A success banner will appear.

Refresh the page to see an updated status. The status of the health check is Healthy.

Thank you for watching this demonstration.

How can I learn more about Route 53 ?

To learn more information about Route 53, choose from the following links.

Route 53 official guide https://aws.amazon.com/route53/

Amazon Route 53 Developer Guide
https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/Welcome.html

Route 53 blogs https://aws.amazon.com/blogs/aws/category/networking-content-delivery/amazon-route-53/

Ask the Route 53 team questions https://repost.aws/