**Route 53 and cloudfront**

What is Amazon Route 53?

Route 53 is a DNS service that connects the Internet traffic to appropriate servers hosting the requested Web application. Route 53 takes its name with reference to port 53. Unlike traditional DNS management services, Route 53, together with other AWS services, enables scalable, flexible, secure, and manageable traffic routing.

You can use Route 53 to perform three main functions: **domain registration, DNS routing, and health checking without coding requirements** using the AWS Management Console.

For an overview on how to start using Route 53 and creating/migrating a domain name, see the section on [how to set up Amazon Route 53](https://www.stormit.cloud/blog/amazon-route-53/#setup-route-53).

Route Internet traffic to the resources for your domain

When a user opens a web browser and enters your domain name (site.com) or subdomain name (videos.site.com) in the address bar, Route 53 helps connect the browser with your website or web application.

Route 53 provides an extension to DNS known as alias records. Similar to CNAME records, alias records let you route traffic to AWS resources, such as CloudFront distributions, Amazon Elastic Load Balancers and S3 buckets. For more information and comparison of alias and CNAME records, see [alias vs non-alias records](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resource-record-sets-choosing-alias-non-alias.html).

Route 53 **supports any of the most common record types** such as CNAMEs, AAAA, SOA, NS and A records, but also less common records. See [supported DNS record types](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/ResourceRecordTypes.html) for more info.

Route 53 has **seven types of Routing policies** that can help you with incoming traffic and point it to the right [origin server](https://www.stormit.cloud/blog/what-is-an-origin-server/) or any kind of resource.

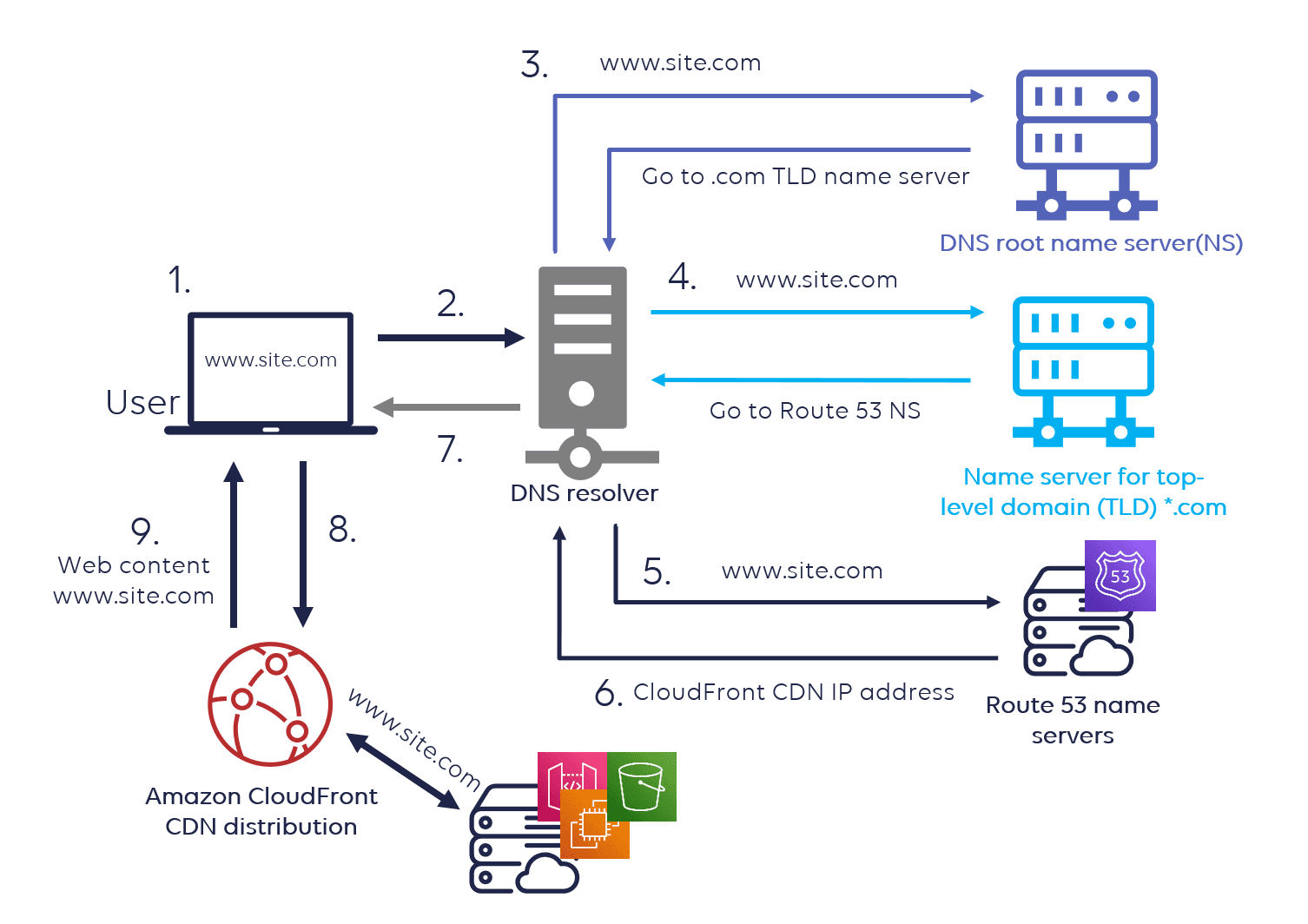
Route 53 can check the health of your resources

Route 53 health checks are a function that **allow you to monitor the health of selected types of AWS resources or any endpoints** that can respond to requests.

They can also **provide notifications of a change in the state** of the health check and can help Route 53 to recognize when a record is pointing to an unhealthy resource, allowing Route 53 to failover to an alternate record.

Learn more in our article: [Route 53: Health Checks and DNS Failover](https://www.stormit.cloud/blog/route-53-health-check/)

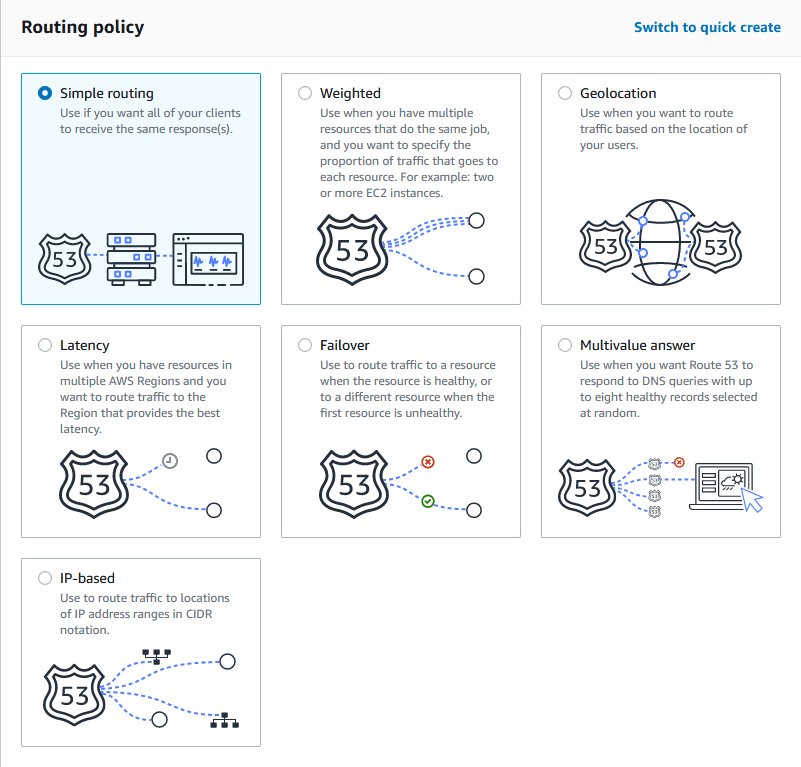
How does Route 53 work?

**1.**A user opens a web browser and sends a request for www.site.com.  
**2.**The request from www.site.com is routed to a DNS resolver, which is usually managed by the Internet Service Provider (ISP).  
**3.**The ISP DNS resolver forwards the request from www.site.com to a DNS root name server.  
**4.**The DNS resolver forwards the request from www.site.com again, this time to one of the top-level domain (TLD) name servers of .com domains. The .com domain name server responds 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 future use.  
**5.**The DNS resolver chooses a Route 53 name server and forwards the request from www.site.com to that Route 53 name server.  
**6.**The Route 53 name server looks for the record www.site.com in the hosted zone site.com, gets its value, such as the alias of [Amazon CloudFront](https://www.stormit.cloud/cdn-services/) distribution in the case of simple routing.  
**7.**The DNS resolver finally has the right route (CloudFront IP) the user needs and returns the value for the user's web browser.  
**8.**The web browser sends a request from www.site.com to the IP address of the CloudFront distribution.  
**9.**The example CloudFront distribution returns the web page from cache or origin server for www.site.com to the web browser.  


What are the different routing policies available in Route 53?

Route 53 offers powerful policies to allow for efficient DNS requests. Once you’ve got your domain up and running, you can choose a routing policy that best fits your needs. However, to get the most out of the service you’ll need to properly understand the **function of each policy type**.

When you create a record, you choose a routing policy, which determines how Amazon Route 53 responds to queries:



**1. Simple routing policy**: Use for a single resource that performs a given function for your domain, for example, an Amazon EC2 instance that serves content for the example.com website.

**2. Weighted**: This allows you to assign weights to resource record sets. For instance, you can specify 25 for one resource and 75 for another, meaning that 25% of requests will go to the first resource and 75% will be routed to the second.

**3. LBR (Latency based routing):** Use when you have resources in multiple AWS Regions and you want to route end users to the AWS region that provides the lowest latency.

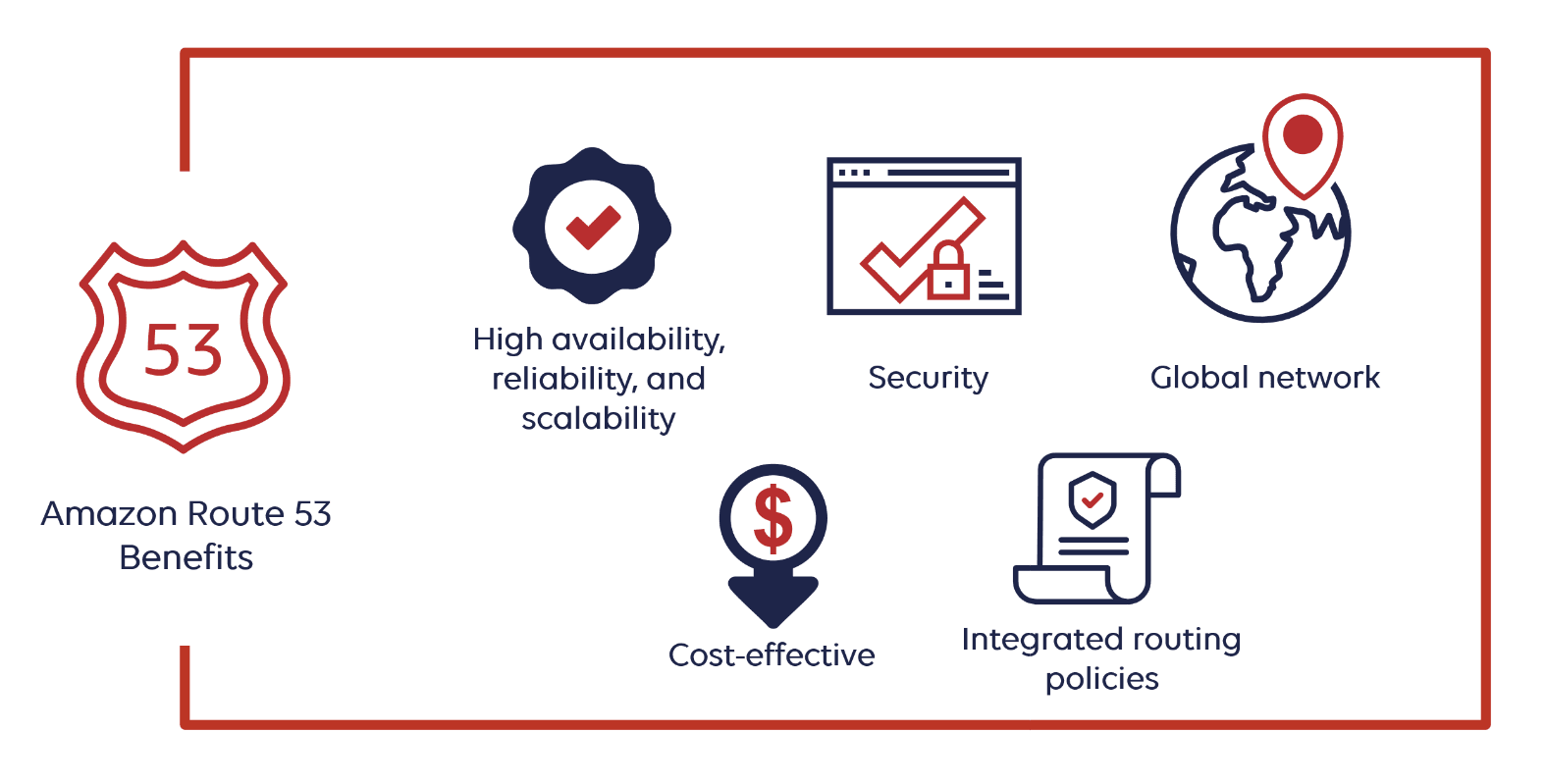
**4. Failover**: Use when you want to configure active-passive failover. More info in our blog post: Amazon Route 53: Health Checks and DNS Failover

**5. Geolocation**: This lets you balance the load on your resources by directing requests to specific endpoints based on the geographic location from which the request originates.

**6. Multivalue answer**: Use when you want Route 53 to respond to DNS queries with up to eight healthy records selected at random.

**7. IP-based**: With IP-based routing, you can create a series of Classless Inter-Domain Routing (CIDR) blocks that represent the client IP network range and associate these CIDR blocks with locations.

Benefits of Route 53



1. High availability, reliability, and scalability

Amazon Route 53 is built using **AWS’s highly available and reliable infrastructure** and is designed to automatically scale to handle very large query volumes.

The distributed nature of our DNS servers helps ensure a **consistent ability to route your end users to your application**. Route 53 is designed to provide the level of dependability required by important applications and is backed by the [Amazon Route 53 SLA (Service Level Agreement](https://aws.amazon.com/route53/sla/)).

**The close integration of AWS services** allows users to perform changes to their architecture and scale resources to accommodate increasing Internet traffic volume without significant configuration and management requirements.

2. Security

You can manage permissions for each user in your AWS account and **control who has access to which parts of the Route 53 service**. When you enable the Route 53 Resolver DNS firewall, you can configure it to check outbound DNS requests against a list of known malicious domains.

3. Global network

A global anycast network of Route 53 DNS servers distributed around the world helps take advantage of lightning-fast speeds. The **DNS database is replicated between regions**. This makes Route 53 a globally resilient service, meaning it can tolerate failure in one or more regions and continue to operate.

4. Cost-effective

You pay only for the resources you use, such as the **number of queries for each of your domains, hosted zones, and optional features such as routing policies and health checks**, all at a low cost and without minimum usage commitments or any up-front fees. For more information, see [Amazon Route 53 Pricing: How Does it Work?](https://www.stormit.cloud/blog/amazon-route-53-pricing/)

5. Integrated routing policies

The routing of traffic based on different criteria such as latency, endpoint health and geographic location is advantageous. The flexibility of route 53 allows the configuration of multiple traffic policies and determines the activity of policies at a particular point in time.

6. Compatibility with other AWS services

Route 53 can help in mapping domain names to Amazon CloudFront distributions, Elastic Load Balancers, EC2 instances, S3 buckets, and other AWS resources.

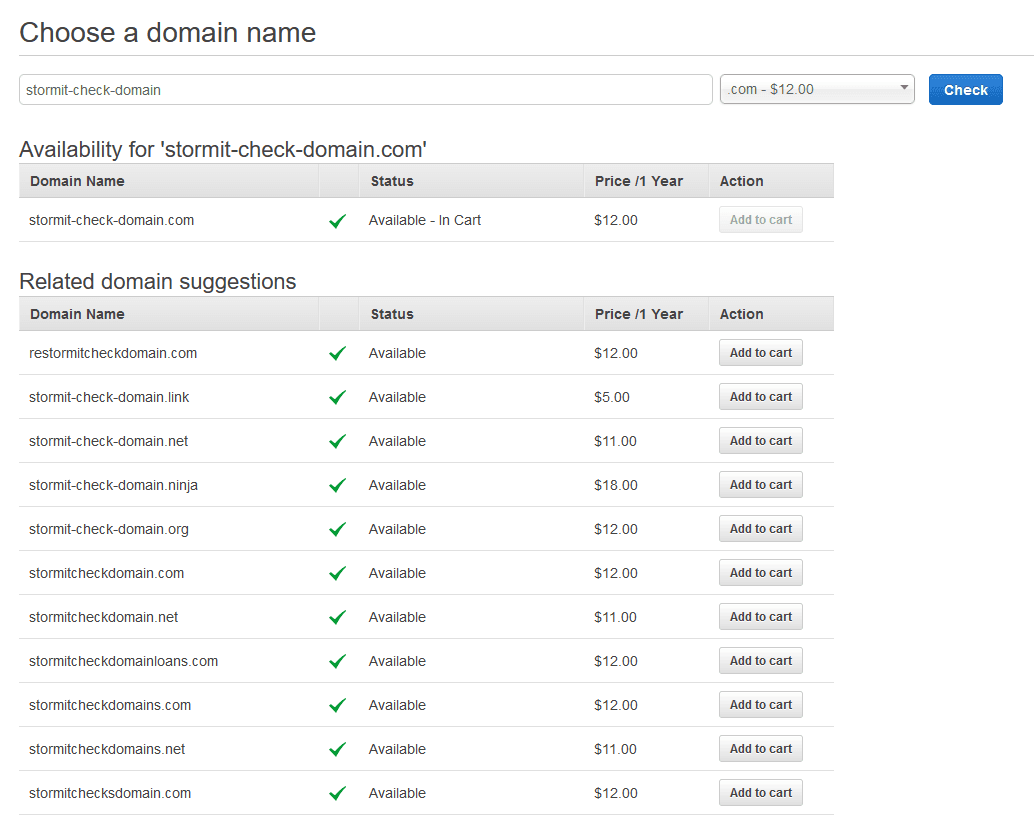
The use of AWS Identity and Access Management (IAM) with Route 53 helps with privileges for updating DNS data.

How to set up Route 53 DNS

First, you will need a domain name. You can register a domain in Route 53 or migrate from your current DNS.

1. Register domain names in Route 53

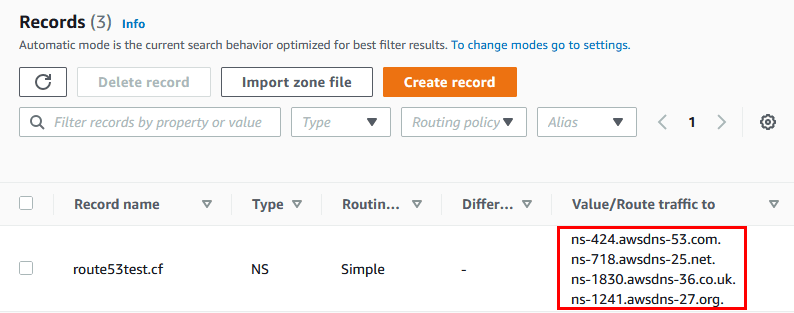
Your website needs a name, such as site.com. Route 53 lets you register a name for your website or web application, known as a **domain name**.



2. Route 53 migration options

Leave your domain name with your current registrar

If you would like to leverage Route 53 routing features but have no need to move your domain name, you simply need to **give your registrar the new name server** **addresses** you’ll get from the Route 53 records upon creating a hosted zone.



Once you have updated the NS records in your current DNS, **Route 53 will ensure the routing of all new domain requests through its name servers**, but this propagation can take some time, so be patient.

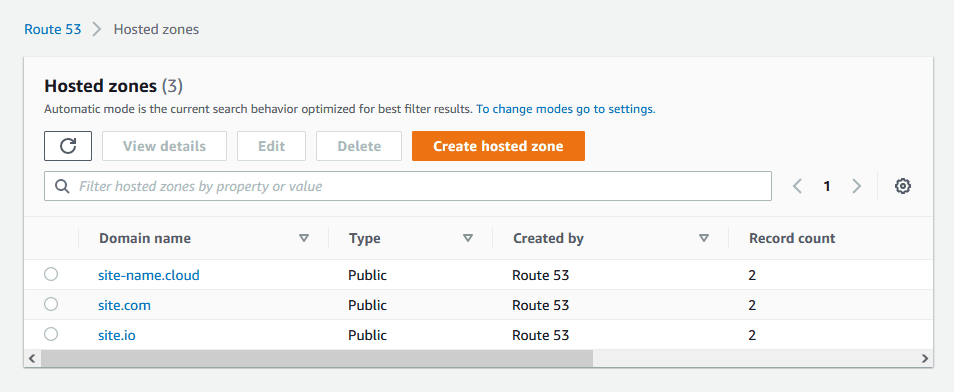
Migrate your domain name

When you decide to [transfer your domain to Route 53](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/domain-transfer-to-route-53.html), you will need to get the DNS record data from your DNS provider. You will then **import this data to a Route 53 hosted zone**, and replace the registrar’s name server records with AWS name servers that you get after creating Hosted Zones. Depending on your settings, changes usually take one day.

3. Route 53 hosted zone creation

As a second step, you will need to create a Route 53 hosted zone, such as site.com. There are two types of hosted zones:

* **Public hosted zones** are the most frequently used and specify how you want to route traffic on the Internet. Visit [working with public hosted zones](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/AboutHZWorkingWith.html) for more info.
* **Private hosted zones** specify how you want to route traffic in an Amazon VPC (Virtual Private Cloud). Visit [working with private hosted zones](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/hosted-zones-private.html) for more info.



4. Route 53 record creation

As the last step, you will need to [create records with routing policies](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/rrsets-working-with.html) applied to them. Every use case is different and it’s difficult to write a simple guide, so try visiting the links below for more info.

For more information about how to start using Route 53, see the following:

* [Set up an Amazon CloudFront distribution with SSL, Route 53 custom domain and S3](https://www.stormit.cloud/blog/setup-an-amazon-cloudfront-distribution-with-ssl-custom-domain-and-s3/), which describes how to host a static website in an S3 bucket, set up a custom domain in Route 53, and how to create an Amazon CloudFront distribution.
* [Creating a CloudFront distribution for Amazon EC2, ALB and Route 53 domain with SSL](https://www.stormit.cloud/blog/cloudfront-distribution-for-amazon-ec2-alb/), which explains how you can use Route53, CloudFront, Application Load Balancer (ALB) and EC2 instances behind it to speed access to your dynamic and static content.
* [Setting up Route 53](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/setting-up-route-53.html), which explains how to sign up, how to secure access to your AWS account, and how to set up programmatic access to Route 53.

Amazon Route 53 pricing and billing

Typically, Route 53 doesn't tend to be a large cost center for the vast majority of businesses operating in the AWS Cloud. But when deciding where you will buy or host your domain name, it can be difficult to understand the final costs of Route 53.

For full details, visit our blog post: [Amazon Route 53 Pricing: How Does it Work?](https://www.stormit.cloud/blog/amazon-route-53-pricing/)

Conclusion

Amazon Route 53 DNS helps to make connections between domain names and IP addresses, and it works very well with AWS services, but also with services from other providers or on-premises infrastructures.

**CLOUD FRONT:**

Q: What is CloudFront?

Answer: Amazon CloudFront is a [content delivery network (CDN)](https://cloudiofy.com/amazon-content-delivery-network/) offered by Amazon Web Services (AWS). It is designed to improve the performance, scalability, and availability of web content and applications by delivering them to users with low latency and high data transfer speeds. CloudFront helps to distribute content across a global network of edge locations strategically located around the world.

Q: What is Geo Restriction in CloudFront?

Answer: Geo restriction feature in Amazon CloudFront allows you to restrict access to your content based on the geographical location of the users. With geo restriction, you can choose to allow or deny access to your content from specific countries or regions.

Q: How to deliver content faster?

Answer: By Utilizing Content Delivery Network (CDN) like Amazon CloudFront or other similar services. CDNs cache your content in multiple edge locations distributed across the globe. When users request content, it is delivered from the nearest edge location, reducing latency and improving response times.

Q: What are the Uses AWS CloudFront?

Answer: AWS CloudFront is a content delivery network (CDN) that accelerates the delivery of web content to end-users globally. It can be used for improving website performance, streaming video content, distributing software and application updates, delivering live and on-demand media streaming, and enhancing security with DDoS protection. CloudFront caches content at edge locations for reducing latency and improving the user experience.

Q: Is Digital Rights Management in built as part of cloudfront?

Answer: No, Digital Rights Management (DRM) is not built directly into Amazon CloudFront. CloudFront itself is primarily a content delivery network (CDN) that focuses on efficient and fast delivery of content to end-users.

By combining the DRM capabilities with CloudFront, you can provide secure and protected content to your users while ensuring efficient delivery and low latency.

Q: When to use Amazon Cloudfront?

Answer: [Amazon CloudFront](https://aws.amazon.com/cloudfront/) is like a super-fast delivery service for web content. It stores copies of your website’s data in many different places around the world, so when someone visits your site, they get the data from the closest location to them.

If your users are spread out all over the world and you want your site to load quickly for everyone, CloudFront is a great choice. It makes sure your content gets to your users as fast as possible, no matter where they are.

Q: How to disable Cache for CloudFront?

Answer: For disable caching – Change the following caching settings –

* Minimum TTL: 0 seconds
* Maximum TTL: 0 seconds
* Default TTL: 0 seconds

Q: Can I Use CloudFront for dynamic content?

Answer: Yes, you can use Amazon CloudFront for dynamic content delivery. While CloudFront is commonly known for accelerating the delivery of static content. To support dynamic content delivery, you can configure caching rules based on query strings, cookies, or headers to determine which dynamic content should be cached and for how long.

Q: Can a Single web server be used to implement CDN at amazon?

Answer: Yes, it is possible to implement a basic content delivery network (CDN) using a single web server on Amazon Web Services (AWS).

However, it is important to note that this approach may not provide the same level of performance and scalability as a dedicated CDN service like Amazon CloudFront.