

## CASE STUDY

### **1.what is the purpose of route 53 and significance of DNS in the context of Route 53?**

#### **Route 53:**

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. You can use Route 53 to perform three main functions in any combination: domain registration, DNS routing, and health checking.

The name is reference to the TCP/UDP port 53, where DNS server requests are addressed.

In addition to being able to route users to various AWS services, including ec2 instances, Route 53 also enables AWS customers to route users to non- AWS infrastructure and to monitor the health of their application and its end points.

Route 53's servers are distributed through out the world. Amazon route 53 supports full, end-to-end DNS resolution over IPv6.

Recursive DNS resolvers on IPv6 networks can use either IPv4 or IPv6 transport to send DNS quires to Amazon route 53.

Customers create "hosted zones" that act as a container for four name servers. The name servers are spread across four different TLDs. customer able to add, delete, and change any DNS records in their hosted zones.

Amazon also offers domain registration services to AWS customers Through Route 53.

Amazon provides an SLA of the service always Being available at all times.

One of the key features of route 53 is programmatic access to the service that follows customers to modify DNS records via web service calls. Combined with other features in AWS, this allows a Developers to programmatically bring up a machine and point to components have been created via other service calls such as those to create new s3 buckets or ec2 instances.

### **2)What is amazon cloud front, and what problem does it Slove?**

#### **Amazon cloud front:**

Amazon CloudFront is a web service that speeds up distribution of your static and dynamic web content, such as .html, .CSS, .JS, and image files, to your users. CloudFront delivers your content through a worldwide network of data centers called edge locations. When a user requests content that you're serving with CloudFront, the request is routed to the edge location that provides the lowest latency (time delay), so that content is delivered with the best possible performance.

If the content is already in the edge location with the lowest latency, CloudFront delivers it immediately.

If the content is not in that edge location, CloudFront retrieves it from an origin that you've defined such as an Amazon S3 bucket, a Media Package channel, or an HTTP server (for example, a web server) that you have identified as the source for the definitive version of your content.

### **3) Explain the concept of “server less” Computing in the context of AWS Lambda?**

#### **Concept of “server less” computing in the context of AWS Lambda:**

AWS Lambda is a serverless compute service through which you can run your code without provisioning any server. It only runs your code when needed and also scales automatically when the request count increases. AWS Lambda follows the pay per use principle—it means there is no charge when your code is not running.

Lambda allows you to run your code for any application or backend service with zero administration. Lambda can run code in response to the events. ex: update I DynamoDB table or change in s3 bucket. You can even run your code in response to HTTP requests using Amazon API Gateway.

Serverless computing is a method of providing backend services on a pay per use basis. Serverless vendor allows you to write and deploy code without worrying about the underlying infrastructure. Servers are still there, but you are not managing them, and the vendor will charge you based on usage.

When using AWS Lambda, you are only responsible for your code. AWS Lambda manages the memory, CPU, network, and other resources. It means you cannot log in to the compute instances or customize the operating system. If you want to manage your own compute resources, you can use services such as EC2, Elastic Beanstalk.

