**Three Tier App Architecture**

AWS S3

AWS

LAMBDA

With ALB targets

RDS

API

AWS CDN

APAP

WAF

**Presentation Layer Application Layer Data Layer**

**Monitoring enabled**

**Monitoring enabled**

**Presentation Tier:**

Amazon S3 :

Bucket storage access

**Usage**: Amazon S3 has a simple web services interface that you can use to store and retrieve any amount of data, at any time, from anywhere on the web. It gives any developer access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of web sites. The service aims to maximize benefits of scale and to pass those benefits on to developers.

As security for any component in a design is important I have added the security section for each component that is necessary to ensure

**Security**: Access control managed for Buckets and its objects with proper encryption.

Amazon Cloudfront:

**Usage** : Cloud Front can speed up the delivery of your static content (for example, images, style sheets, JavaScript, and so on) to viewers across the globe. By using CloudFront, you can take advantage of the AWS backbone network and CloudFront edge servers to give your viewers a fast, safe, and reliable experience when they visit your website.

A simple approach for storing and delivering static content is to use an Amazon S3 bucket. Using S3 together with CloudFront has a number of advantages, including the option to use Origin Access Identity (OAI) to easily restrict access to your S3 content.

**Security**: Geo restriction to cross border security , Singed URL, HTTPS and DDoS protection.

**App Tier :**

Amazon API Gateway:

Versions: Test,Dev,Prod,DR

Usage: Amazon API Gateway is an AWS service for creating, publishing, maintaining, monitoring, and securing REST and WebSocket APIs at any scale. API developers can create APIs that access AWS or other web services as well as data stored in the [AWS Cloud](https://aws.amazon.com/what-is-cloud-computing/). As an API Gateway API developer, you can create APIs for use in your own client applications (apps). Or you can make your APIs available to third-party app developers. For more information, see [Who Uses API Gateway?](https://docs.aws.amazon.com/apigateway/latest/developerguide/api-gateway-overview-developer-experience.html#apigateway-who-uses-api-gateway).

API Gateway creates REST APIs that:

* Are HTTP-based.
* Adhere to the [REST](https://en.wikipedia.org/wiki/Representational_state_transfer) protocol, which enables stateless client-server communication.
* Implement standard HTTP methods such as GET, POST, PUT, PATCH and DELETE

**Security**: Access Resource policies rules ( ip,Users,groups), Cognito identiy pool based on resource policies. API certificates. Keys for access for legitimate users.

AWS Lambda: High availability scaling for server less app( Elasticity)

**Usage**: AWS Lambda is a compute service that lets you run code without provisioning or managing servers. AWS Lambda executes your code only when needed and scales automatically, from a few requests per day to thousands per second. You pay only for the compute time you consume - there is no charge when your code is not running. With AWS Lambda, you can run code for virtually any type of application or backend service - all with zero administration. AWS Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, code monitoring and logging.

Security: Lambda triggers- Users configured in triggrers will be able to execute lamba function, Lamba roles to access configuration variables in database using KMS or IAM roles.

Integration with API gateway

**Data Tier:**

Oracle,MSSQL,Redis,memcached

**Usage**:

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

**Security :** DB security group (VPC,IP,port level lambda function) policies can be written for sec groups. Having RDS as private end points

**WAF**: Web application firewall

**Usage**: AWS WAF is a web application firewall that lets you monitor the HTTP and HTTPS requests that are forwarded to an Amazon API Gateway API, Amazon CloudFront or an Application Load Balancer. AWS WAF also lets you control access to your content. Based on conditions that you specify, such as the IP addresses that requests originate from or the values of query strings, API Gateway, CloudFront or an Application Load Balancer responds to requests either with the requested content or with an HTTP 403 status code (Forbidden). You also can configure CloudFront to return a custom error page when a request is blocked.

**Security**: Provides layer 7 capabilities, from the url to IP based rule policies (Access/Deny) to our resources.

**Three Tier App Architecture**

AWS S3

AWS

LAMBDA

With ALB targets

RDS

API

AWS CDN

APAP

WAF

**Presentation Layer Application Layer Data Layer**

**Monitoring enabled**

Load Balancer (IGW)

**Mode**

RDS

App server

Web Server