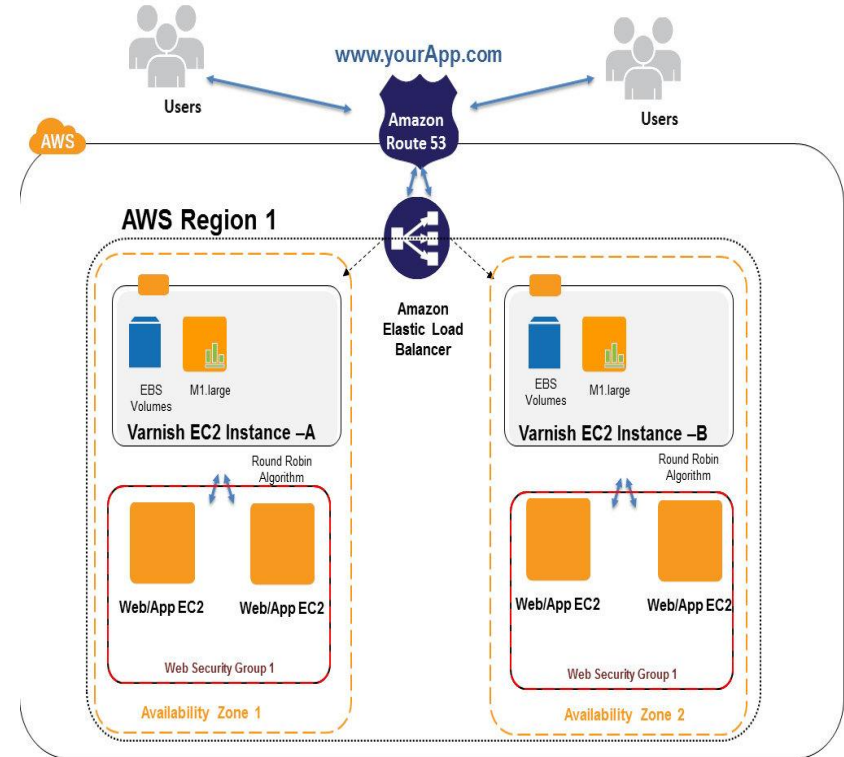# AWS Solution Architect

What is AWS?

- Amazon Web Services (AWS) is a secure cloud services platform, offering compute power, database storage, content delivery and other functionality to help businesses scale and grow.

- The AWS Cloud provides a broad set of infrastructure services, such as computing power, storage options, networking and databases, delivered as a utility: on-demand, available in seconds, with pay-as-you-go pricing.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

AWS Regions

AWS Regions

- GovCloud (ITAR Compliance)
- US West (Oregon)
- US West (Northern California)
- US East (Northern Virginia)
- EU (Ireland)
- China (Beijing)
- Asia Pacific (Tokyo)
- Asia Pacific (Singapore)
- South America (Sao Paulo)
- Asia Pacific (Sydney)

- AWS Regions (10)
- AWS Edge Locations (52)

amazon web services

www.yourApp.com

Users — Amazon Route 53 — Users

AWS

AWS Region 1

Amazon Elastic Load Balancer

Availability Zone 1

EBS Volumes / M1.large
Varnish EC2 Instance –A

Round Robin Algorithm

Web/App EC2   Web/App EC2

Web Security Group 1

Availability Zone 2

EBS Volumes / M1.large
Varnish EC2 Instance –B

Round Robin Algorithm

Web/App EC2   Web/App EC2

Web Security Group 1

AWS Global Presence 16 Regions | 42 AZ

- Region
  - A region is geographical location of the AWS data centre
- Availability Zone
  - Each availability zone is isolated location within the same region

All communication between regions is across the public Internet. Data transfer between regions is charged at the Internet data transfer rate for both the sending and the receiving instance.

AWS Registration

- To use AWS services register with AWS
  - URL : https://aws.amazon.com
- AWS Pricing
  - Pricing varies depending on the services used

EC2 – Elastic Compute Cloud

- EC2 is a compute server
- You have choice to run windows and Linux virtual machines
- Instance type can vary based on the requirement

EC2 Instance configuration

- T2 Instance are used in dev environment where there is less usage of CPU
- M3 Instance used for data processing tasks that require additional memory, for running backend servers on SAP
- C3/C4 Instance type used for high performance front-end ,web servers
- G2 Instance type used for Game streaming, Video streaming, 3d application streaming
- HS1 instance type used for parallel systems

EC2 Instance configuration

- ## EC2 Instance Metadata and User Data

  - *Instance metadata* is data about your instance that you can use to configure or manage the running instance. Instance metadata is divided into categories.

  - You can also use instance metadata to access *user data* that you specified when launching your instance.

  - For example, you can specify parameters for configuring your instance, or attach a simple script. You can also use this data to build more generic AMIs that can be modified by configuration files supplied at launch time. For example, if you run web servers for various small businesses, they can all use the same AMI and retrieve their content from the Amazon S3 bucket you specify in the user data at launch. To add a new customer at any time, simply create a bucket for the customer, add their content, and launch your AMI. If you launch more than one instance at the same time, the user data is available to all instances in that reservation.

EC2 Instance configuration

- **EC2 Instance Metadata and User Data -***Retrieving Instance Metadata*
  - *Because your instance metadata is available from your running instance, you do not need to use the Amazon EC2 console or the AWS CLI. This can be helpful when you're writing scripts to run from your instance. For example, you can access the public hostname of your instance from instance metadata to manage a connection.*
  - *To view all categories of instance metadata from within a running instance, use the following URI:*
  - *http://169.254.169.254/latest/meta-data/*
  - *Note that you are not billed for HTTP requests used to retrieve instance metadata and user data.*
  - *You can use a tool such as cURL, or if your instance supports it, the GET command; for example:*
  - *[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/*
  - *[ec2-user ~]$ GET http://169.254.169.254/latest/meta-data/*
  - curl http://169.254.169.254/latest/meta-data/public-hostname

EC2 Instance configuration

- **Running Commands on Your Linux Instance at Launch- EC2 Bootstrapping**

- When you launch an instance in Amazon EC2, you have the option of passing user data to the instance that can be used to perform common automated configuration tasks and even run scripts after the instance starts.

- You can pass two types of user data to Amazon EC2: shell scripts and cloud-init directives. You can also pass this data into the launch wizard as plain text, as a file (this is useful for launching instances using the command line tools), or as base64-encoded text (for API calls).

EC2 Instance configuration

- **Running Commands on Your Linux Instance at Launch- EC2 Bootstraping**

```
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
usermod -a -G apache ec2-user
```

- **To access - http://my.public.dns.amazonaws.com**
- **To view and update - /action/instance-setting/View-Change user data**

11

Demo

- Create EC2 instance

EBS – Elastic Block Store

- Provide persistent block level storage volumes
- Can attach it to EC2 Instance
- In easy words, think of EBS volumes like a D drive of your VM

EBS – Elastic Block Store

- **EBS General Purpose (SSD)**
  - For small to med size data
  - Charges $0.10/GB/month

- **EBS Provisioned IOPS (SSD)**
  - For large relational data
  - Charges $0.125/GB/month

Demo

- Create volumes
- Attaching volume to the instance

Amazon EFS

- Scalable, reliable, and elastic file storage for the AWS Cloud

- Amazon Elastic File System (Amazon EFS) provides simple, scalable, elastic file storage for use with AWS Cloud services and on-premises resources.

- Amazon EFS is built to elastically scale on demand without disrupting applications, growing and shrinking automatically as you add and remove files, so your applications have the storage they need, when they need it.

- It is designed to provide massively parallel shared access to thousands of Amazon EC2 instances, enabling your applications to achieve high levels of aggregate throughput and IOPS that scale as a file system grows, with consistent low latencies.

Amazon EFS

- As a regional service, Amazon EFS is designed for high availability and durability storing data redundantly across multiple Availability Zones.

- With these capabilities, Amazon EFS is well suited to support a broad spectrum of use cases, including web serving and content management, enterprise applications, media and entertainment processing workflows, home directories, database backups, developer tools, container storage, and big data analytics workloads.

Amazon EFS

- **How it works**
  - When mounted on Amazon EC2 instances, an Amazon EFS file system provides a standard file system interface and file system access semantics.
  - Multiple Amazon EC2 instances can access an Amazon EFS file system at the same time, allowing Amazon EFS to provide a common data source for workloads and applications running on more than one Amazon EC2 instance.
  - You can mount your Amazon EFS file systems on your on-premises datacenter servers when connected to your Amazon VPC with AWS Direct Connect. You can mount your EFS file systems on on-premises servers to migrate data sets to EFS, enable cloud bursting scenarios, or backup your on-premises data to EFS.
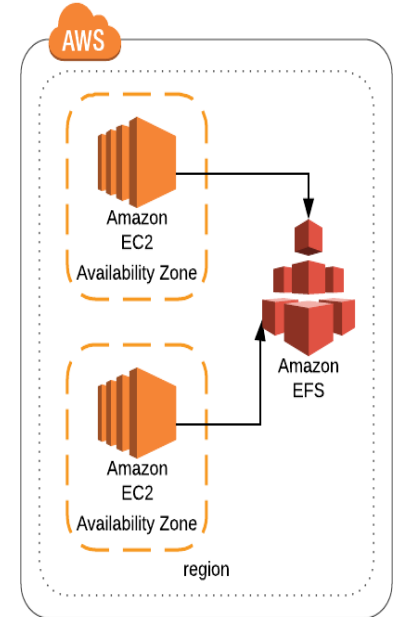
Amazon EFS

- Benefits

  - SIMPLE -Amazon EFS has a simple web services interface that allows you to create and configure file systems quickly and easily. The service manages all the file storage infrastructure for you, avoiding the complexity of deploying, patching, and maintaining complex file system deployments.

  - SHARED FILE STORAGE -Multiple Amazon EC2 instances and on-premises servers can simultaneously access an Amazon EFS file system, so applications that scale beyond a single instance can access a file system. Amazon EC2 instances running in multiple Availability Zones (AZs) within the same region can access the file system, so that many users can access and share a common data source.
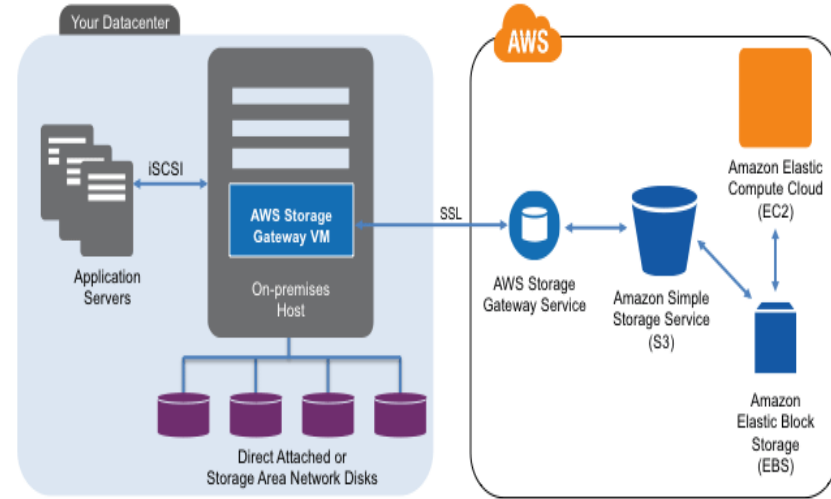
Amazon EFS

- SEAMLESS INTEGRATION - Amazon EFS provides a file system interface and file system access semantics (such as strong data consistency and file locking). Amazon EC2 instances mount Amazon EFS file systems via the NFSv4 protocol, using standard operating system mount commands. You can also mount Amazon EFS file systems on your on-premises datacenter servers via the NFSv4 protocol when connected to your Amazon VPC with AWS Direct Connect.

- AUTOMATICALLY SCALES - Amazon EFS automatically and instantly scales your file system storage capacity up or down as you add or remove files without disrupting your applications, giving you just the storage you need while also reducing time-consuming administration tasks.

- SCALEABLE PERFORMANCE - Amazon EFS is designed to provide the throughput, IOPS, and low latency needed for a broad range of workloads. With Amazon EFS, throughput and IOPS scale as a file system grows, and file operations are delivered with consistent, low latencies.

Amazon EFS

- LOW COST - Amazon EFS provides the capacity you need, when you need it, without having to provision storage in advance. You pay for what you use, with no minimum commitments or up-front fees.

- HIGHLY AVAILABLE AND DURABLE -Amazon EFS is designed to be highly available and durable. Each Amazon EFS file system object (i.e. directory, file, and link) is redundantly stored across multiple Availability Zones.

- OPTIMIZED TRANSFERS -Amazon EFS provides a simple and secure way to move data from existing on-premises or in-cloud file systems, with the ability to copy files and directories at speeds up to 5x faster than standard Linux copy tools.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

AWS Storage Gateway

- Hybrid cloud storage with local caching
- AWS Storage Gateway is a hybrid storage service that enables your on-premises applications to seamlessly use AWS cloud storage.
- You can use the service for backup and archiving, disaster recovery, cloud data processing, storage tiering, and migration.
- Your applications connect to the service through a virtual machine or hardware gateway appliance using standard storage protocols, such as NFS, SMB and iSCSI.

AWS Storage Gateway

- The gateway connects to AWS storage services, such as Amazon S3, Amazon Glacier, and Amazon EBS, providing storage for files, volumes, and virtual tapes in AWS.

- The service includes a highly-optimized data transfer mechanism, with bandwidth management, automated network resilience, and efficient data transfer, along with a local cache for low-latency on-premises access to your most active data.
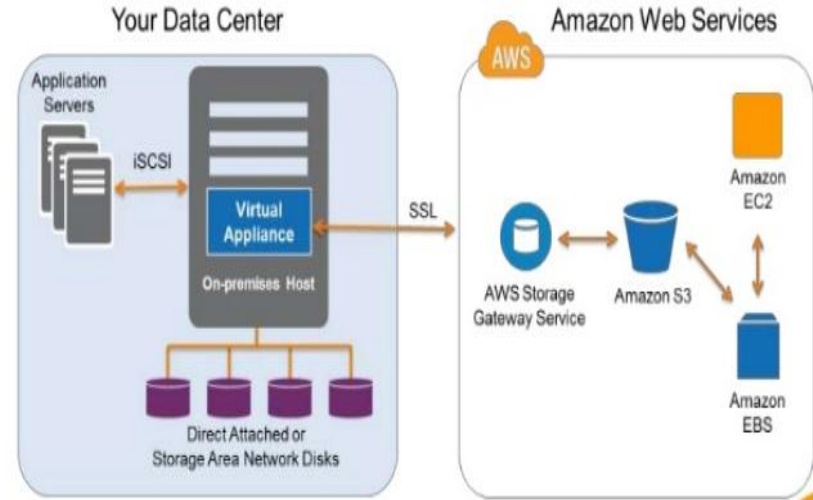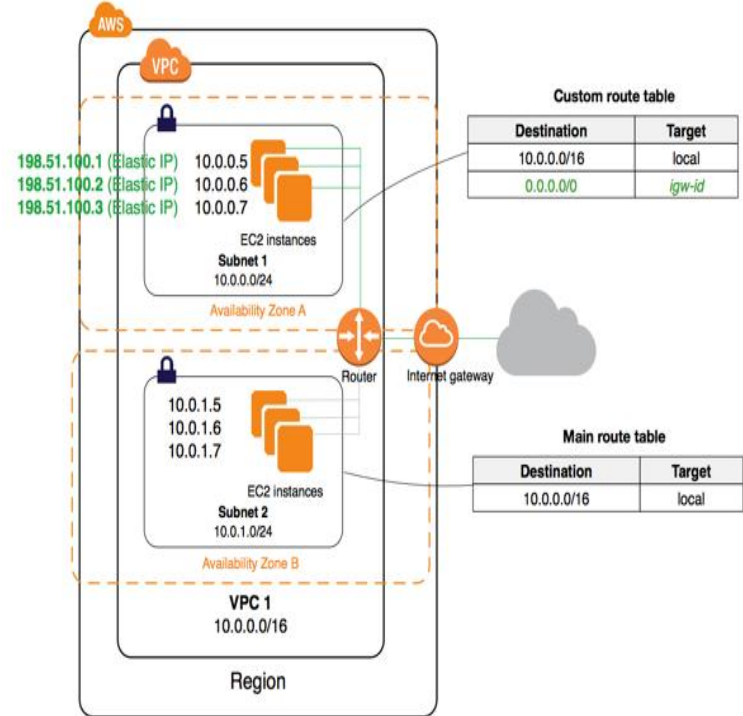
VPC

- A *virtual private cloud* (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch your AWS resources, such as Amazon EC2 instances, into your VPC.

- You can specify an IP address range for the VPC, add subnets, associate security groups, and configure route tables.
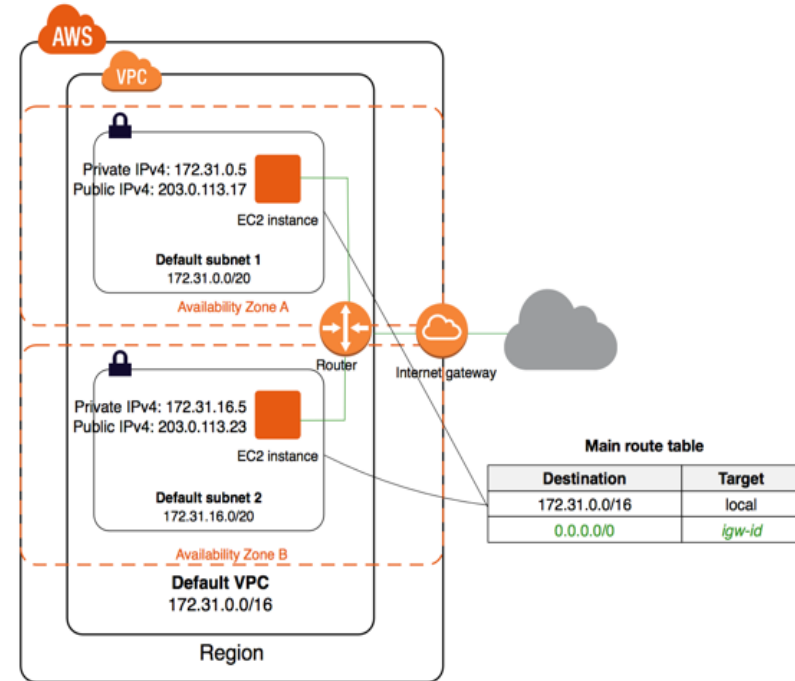
VPC

- A *subnet* is a range of IP addresses in your VPC. You can launch AWS resources into a specified subnet. Use a public subnet for resources that must be connected to the internet, and a private subnet for resources that won't be connected to the internet.

- To protect the AWS resources in each subnet, you can use multiple layers of security, including security groups and network access control lists (ACL).
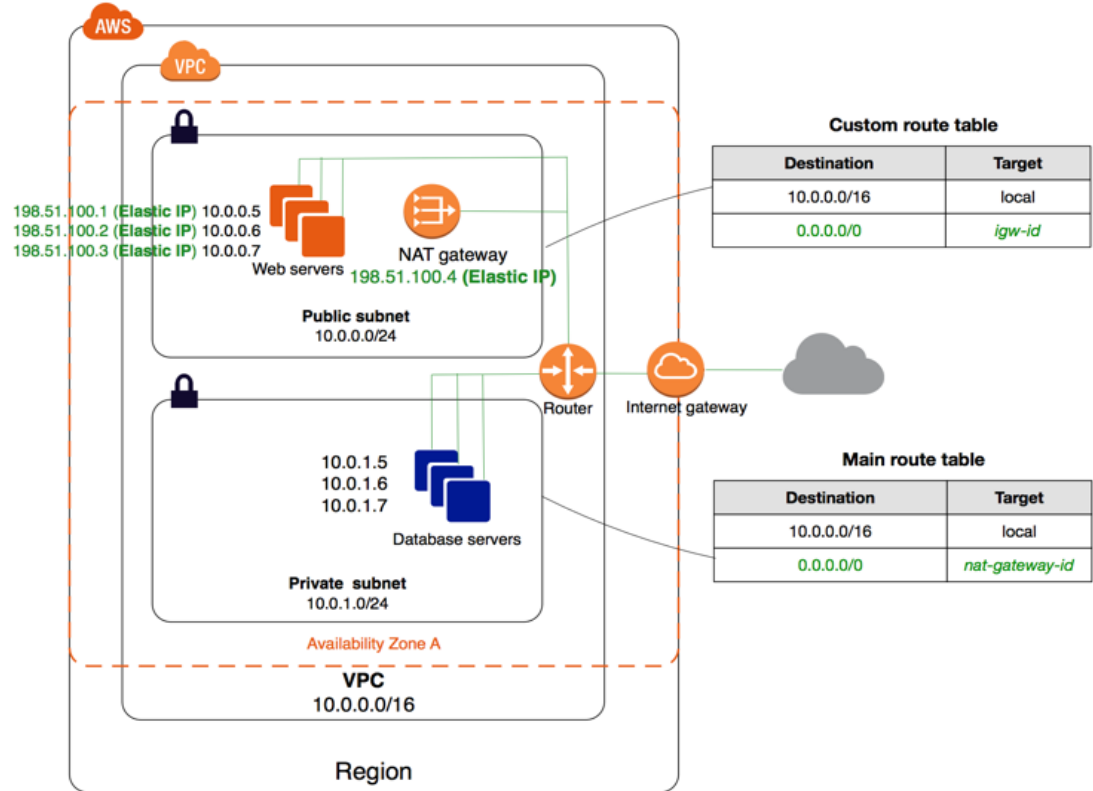

Custom route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-id

Main route table

Destination	Target
10.0.0.0/16	local

VPC peering

- A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses.

- Instances in either VPC can communicate with each other as if they are within the same network. You can create a VPC peering connection between your own VPCs, or with a VPC in another AWS account.

- The VPCs can be in different regions (also known as an inter-region VPC peering connection)..
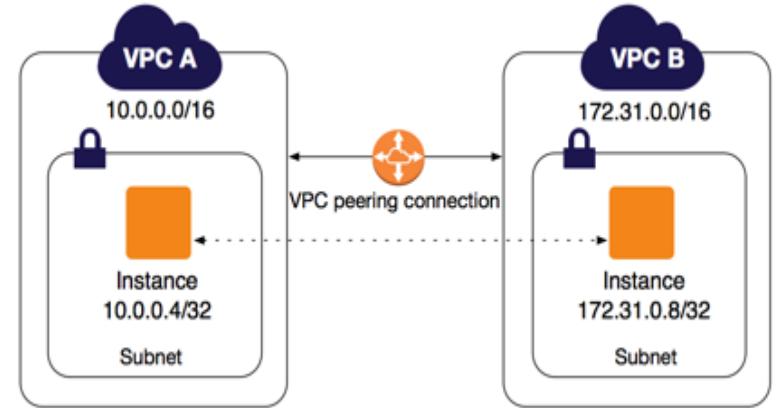
NAT Gateways

- Network address translation (NAT) gateway is used to enable instances in a private subnet to connect to the internet or other AWS services, but prevent the internet from initiating a connection with those instances.

198.51.100.1 (**Elastic IP**) 10.0.0.5
198.51.100.2 (**Elastic IP**) 10.0.0.6
198.51.100.3 (**Elastic IP**) 10.0.0.7

Web servers

NAT gateway 198.51.100.4 (**Elastic IP**)

Public subnet 10.0.0.0/24

Custom route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	*igw-id*

Router   Internet gateway

10.0.1.5
10.0.1.6
10.0.1.7

Database servers

Private subnet 10.0.1.0/24

Availability Zone A

VPC 10.0.0.0/16

Region

Main route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	*nat-gateway-id*

ELB- Elastic Load Balancer

- Elastic Load balancer automatically distributes incoming application traffic across multiple Amazon EC2 instances in the cloud
- Achieve higher levels of fault tolerance for your applications by using Elastic Load Balancing to automatically route traffic across multiple instances and multiple Availability Zones
- Additionally, Elastic Load Balancing offers integration with Auto Scaling to ensure that you have back-end capacity to meet varying levels of traffic without requiring minimal intervention
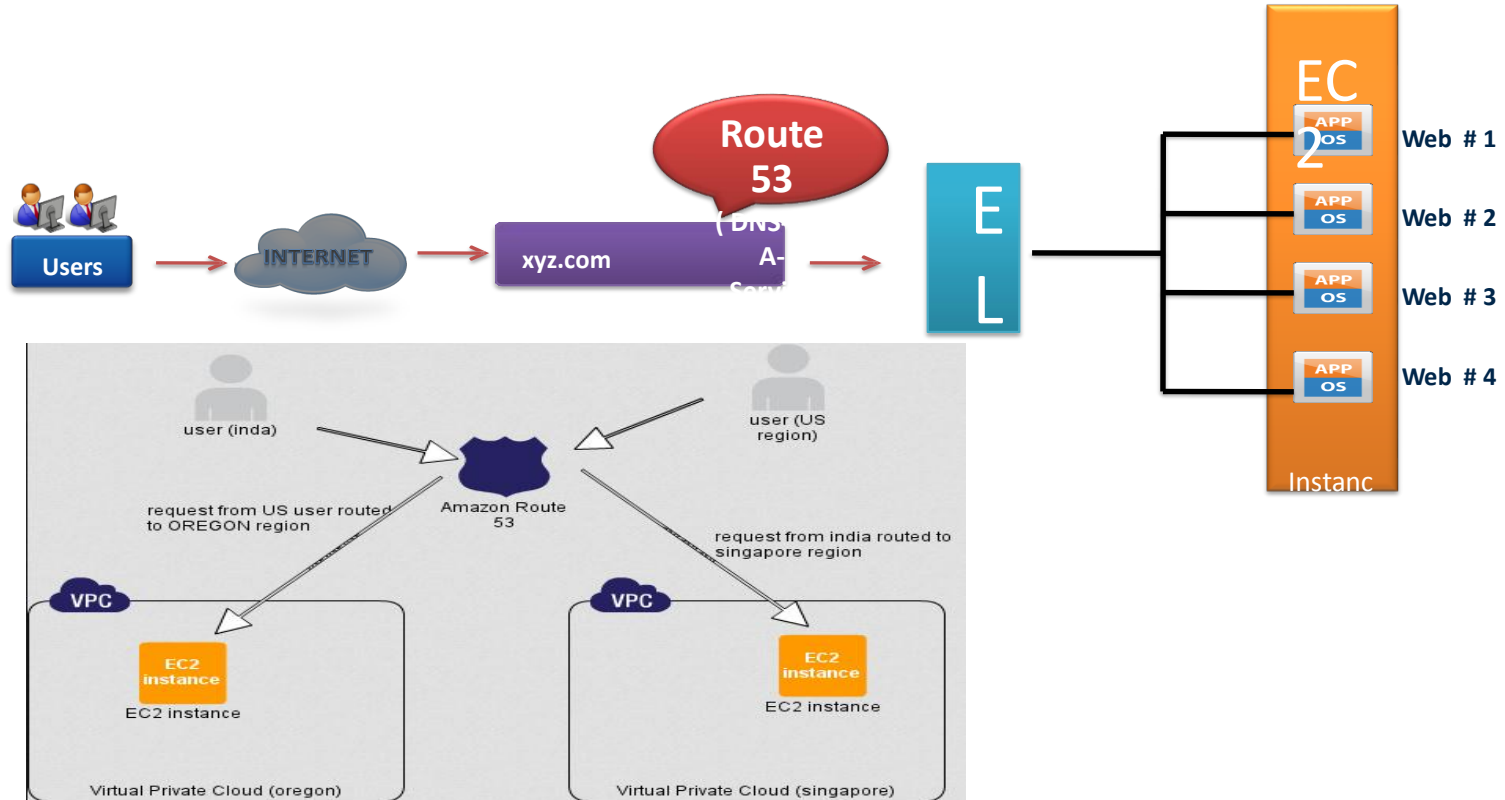
ELB- Elastic Load Balancer
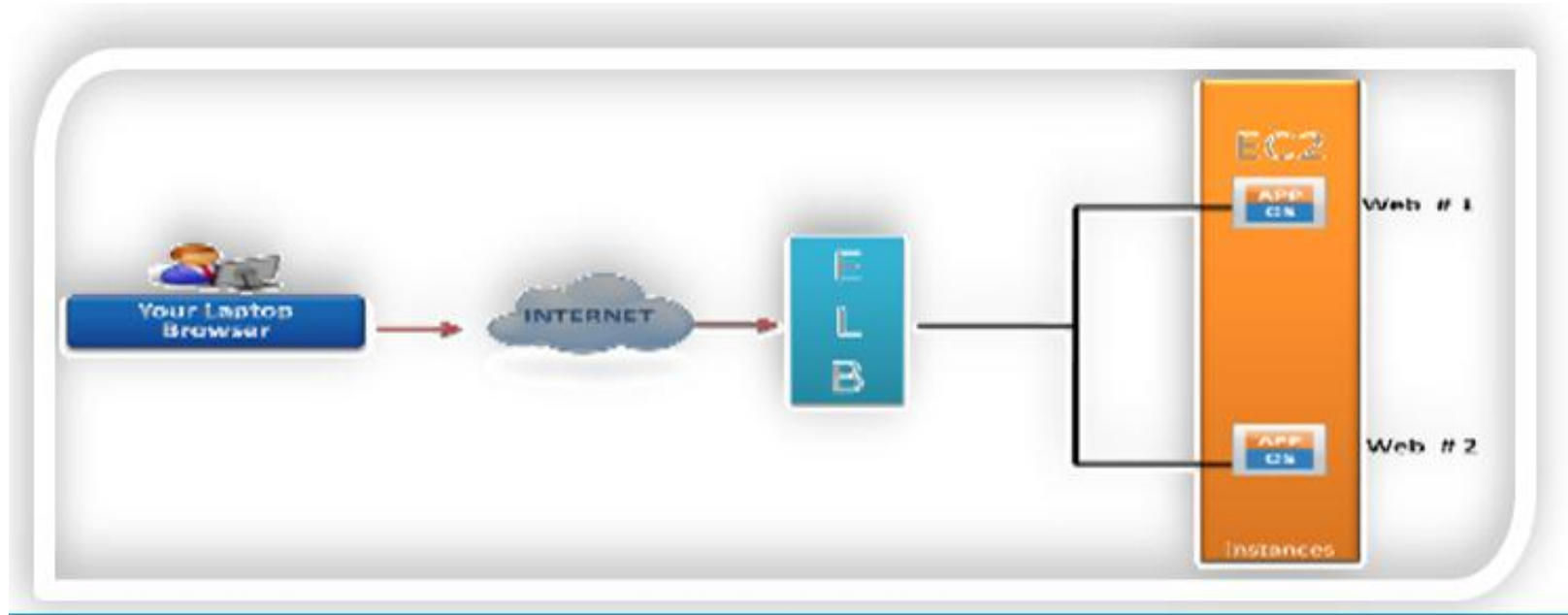
Demo

- Configuring Load Balancer

Route-53

- Elastic Amazon Route-53 is a highly available and scalable cloud Domain Name System (DNS) web service

- It is designed to route end users to Internet applications by translating names like www.example.com in to numeric IP addresses like 192.0.2.1 that computers use to connect to each other

- Amazon Route-53 effectively connects user requests to infrastructure running in AWS – such as Amazon EC2 instances, ELB, etc

- The DNS port number is 53 and that is how the name Route53 was coined by AWS for the global DNS service

Route-53

Users → INTERNET → xyz.com → Route 53 (DNS A-Serv) → E L → EC2 Instanc

- Web # 1
- Web # 2
- Web # 3
- Web # 4

user (inda)

request from US user routed to OREGON region

Amazon Route 53

user (US region)

request from india routed to singapore region

VPC

EC2 instance

EC2 instance

VPC

EC2 instance

EC2 instance

Virtual Private Cloud (oregon)

Virtual Private Cloud (singapore)

IAM

- AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources.

- You use IAM to control who is authenticated (signed in) and authorized (has permissions) to use resources.

- When you first create an AWS account, you begin with a single sign-in identity that has complete access to all AWS services and resources in the account.
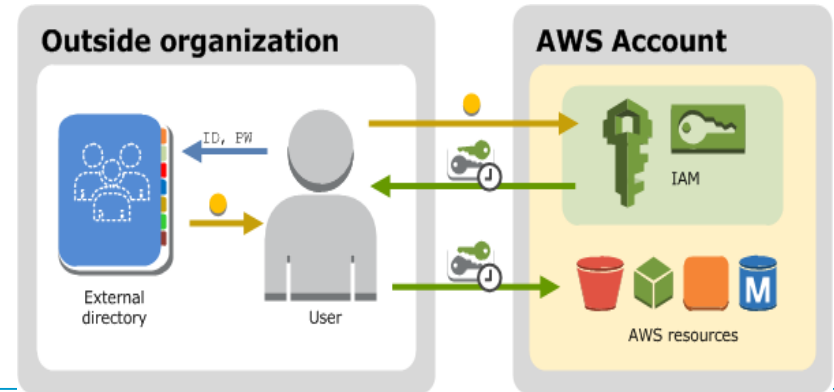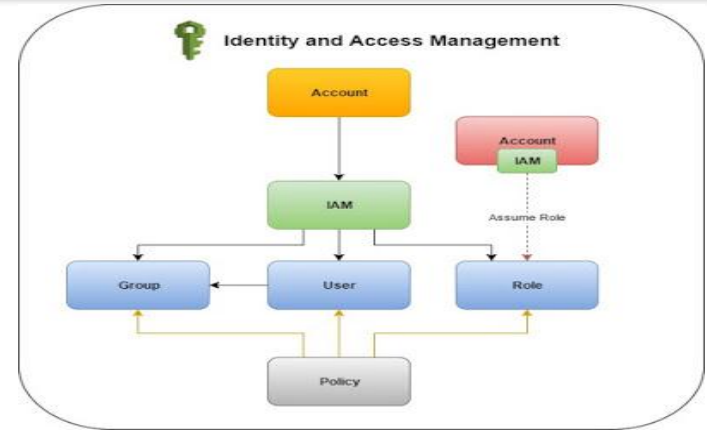
IAM

- This identity is called the AWS account *root user* and is accessed by signing in with the email address and password that you used to create the account.

- It is strongly recommend that you do not use the root user for your everyday tasks, even the administrative ones.
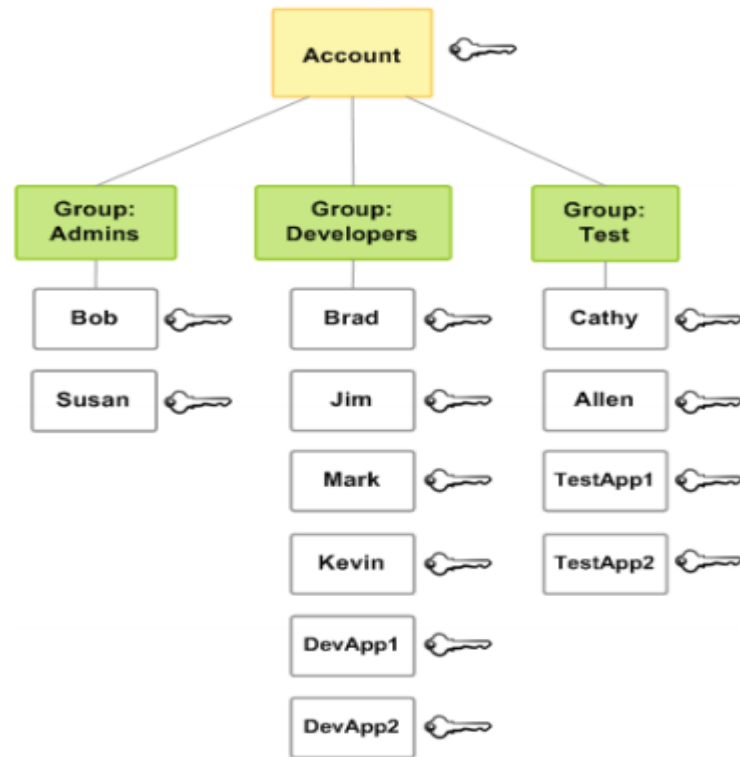
IAM

- You manage access in AWS by creating policies and attaching them to IAM identities or AWS resources.
- A policy is an object in AWS that, when associated with an entity or resource, defines their permissions. AWS evaluates these policies when a principal, such as a user, makes a request.
- Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents.
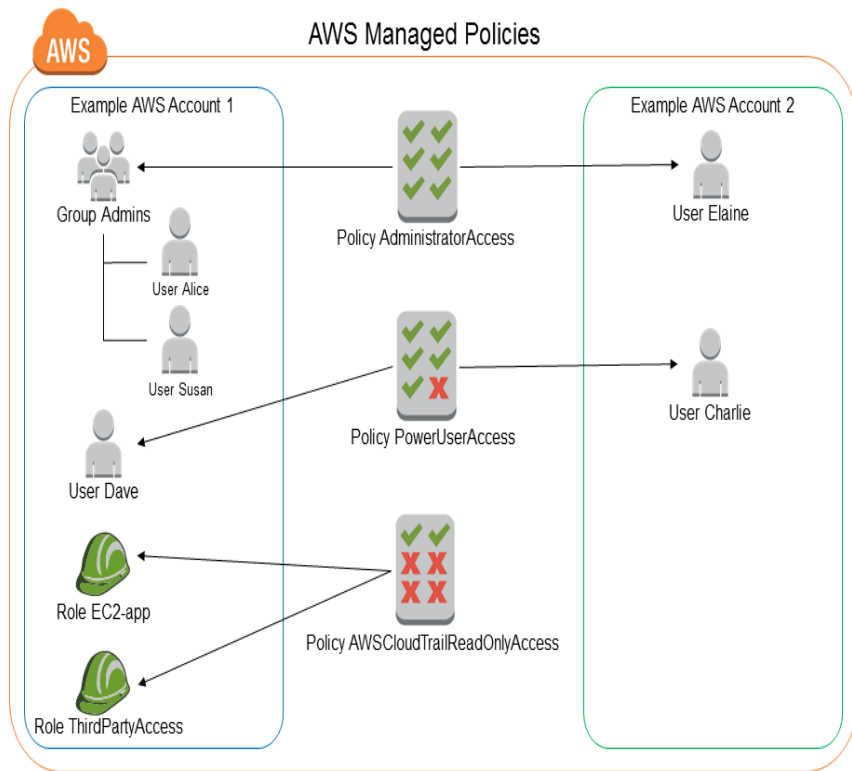
IAM

- IAM policies define permissions for an action regardless of the method that you use to perform the operation.

- When you create an IAM user, you can set up the user to allow console or programmatic access.

- The IAM user can sign in to the console using a user name and password. Or they can use access keys to work with the CLI or API.

IAM: Policy Assignment
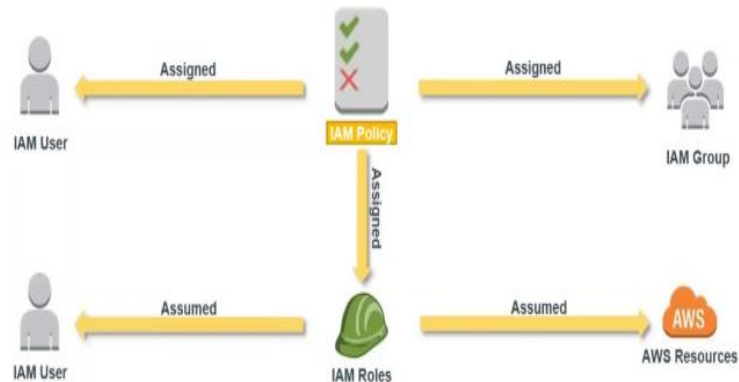
S3- Simple Storage Service

- Companies today need the ability to simply and securely collect, store, and analyze their data at a massive scale.

- Amazon S3 is object storage built to store and retrieve any amount of data from anywhere – web sites and mobile apps, corporate applications, and data from IoT sensors or devices.

- It is designed to deliver 99.999999999% durability, and stores data for millions of applications used by market leaders in every industry.

- S3 provides comprehensive security and compliance capabilities that meet even the most stringent regulatory requirements.

- It gives customers flexibility in the way they manage data for cost optimization, access control, and compliance.

S3- Simple Storage Service

- Amazon S3 is easy to use object storage, with a simple web service interface to store and retrieve any amount of data from anywhere on the web.

- You can store static files docs, ppt ,xls , pdf ,audio and video files any other files

- Amazon S3 provides cost-effective object storage for a wide variety of use cases including backup and recovery, big data analytics, content distribution

S3- Simple Storage Service

- S3 uses buckets to store files. Buckets act as a container for storage
- By default all buckets are private, you need to make it public to access
- Each developer account has limit of 100 buckets
- By default three replicas of file are made in same region
- Replication across region is allowed but it comes with additional charges. The charges are based on the region

S3- Simple Storage Service

Users

File
operations
Upload
Download
Share
Delete

INTERNET

https

https://console.aws.amazon.com/s3/home?region=ap-southeast-2
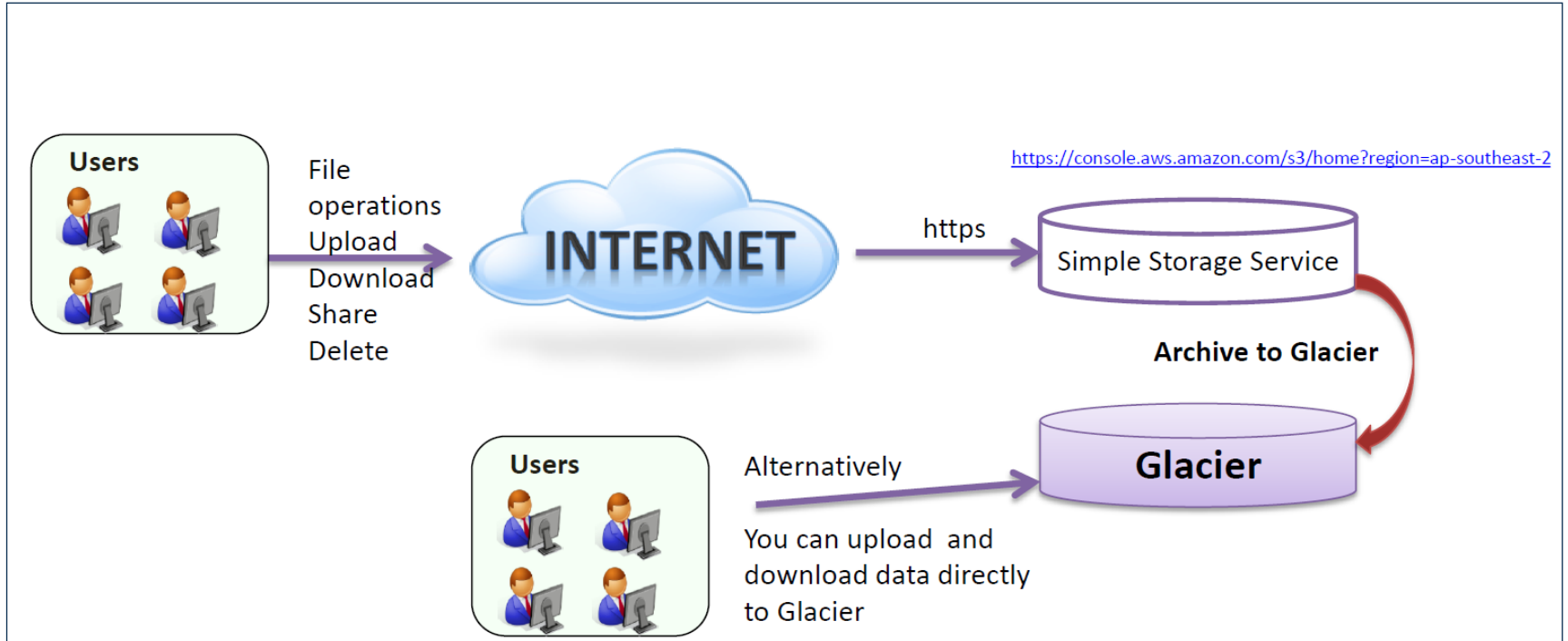
Simple Storage Service

Demo

- Creating S3 bucket
- Upload, download files
- Delete bucket
- Static website configuration

Glacier

- Amazon Glacier is a secure, durable, and extremely low-cost cloud storage service for data archiving and long-term backup
- Through Amazon S3 lifecycle policies, you can optimize your storage costs by moving infrequently accessed objects from Amazon S3 to Amazon Glacier (or vice-versa)
- For long term retention of data, Customers can reliably store large or small amounts of data for as little as $0.007 per gigabyte per month, a significant savings compared to storing on S3
- Amazon Glacier supports data transfer over SSL and automatically encrypts your data at rest
- A single archive can be as large as 40 terabytes

Glacier

Users

File operations
Upload
Download
Share
Delete

INTERNET

https://console.aws.amazon.com/s3/home?region=ap-southeast-2

https

Simple Storage Service

Archive to Glacier

Glacier

Users

Alternatively

You can upload and download data directly to Glacier

Snowball

- With Snowball, you don't need to write any code or purchase any hardware to transfer your data.
- Simply create a job in the AWS Management Console ("Console") and a Snowball device will be automatically shipped to you.
- Once it arrives, attach the device to your local network, download and run the Snowball Client ("Client") to establish a connection, and then use the Client to select the file directories that you want to transfer to the device.
- The Client will then encrypt and transfer the files to the device at high speed.
- Once the transfer is complete and the device is ready to be returned, the E Ink shipping label will automatically update and you can track the job status via Amazon Simple Notification Service (SNS), text messages, or directly in the Console.

AWS Lambda

- 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.

AWS Lambda

- AWS Lambda supports (currently Node.js, Java, C#, Go and Python).

- You can use AWS Lambda to run your code in response to events, such as changes to data in an Amazon S3 bucket or an Amazon DynamoDB table; to run your code in response to HTTP requests using Amazon API Gateway; or invoke your code using API calls made using AWS SDKs.

- With these capabilities, you can use Lambda to easily build data processing triggers for AWS services like Amazon S3 and AmazonDynamoDB or create your own back end that operates at AWS scale, performance, and security.

AWS Lambda

- **When Should I Use AWS Lambda?**

  - When using AWS Lambda, you are responsible only for your code.

  - AWS Lambda manages the compute fleet that offers a balance of memory, CPU, network, and other resources. This is in exchange for flexibility, which means you cannot log in to compute instances, or customize the operating system or language runtime. These constraints enable AWS Lambda to perform operational and administrative activities on your behalf, including provisioning capacity, monitoring fleet health, applying security patches, deploying your code, and monitoring and logging your Lambda functions.

API Gateway

- Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale.

- With a few clicks in the AWS Management Console, you can create an API that acts as a "front door" for applications to access data, business logic, or functionality from your back-end services, such as workloads running on Amazon Elastic Compute Cloud (Amazon EC2), code running on AWS Lambda, or any web application.

- Amazon API Gateway handles all the tasks involved in accepting and processing up to hundreds of thousands of concurrent API calls, including traffic management, authorization and access control, monitoring, and API version management. Amazon API Gateway has no minimum fees or startup costs. You pay only for the API calls you receive and the amount of data transferred out.

API Gateway

- Benefits

  - LOW COST AND EFFICIENT -With Amazon API Gateway, you pay only for calls made to your APIs and data transfer out. There are no minimum fees or upfront commitments

  - PERFORMANCE AT ANY SCALE - With Amazon CloudFront integration, API Gateway allows you to take advantage of the worldwide network of edge locations to provide your end users with the lowest possible latency for API requests and responses. Amazon API Gateway also helps you manage traffic through throttling, so that back-end operations can withstand traffic spikes. Additionally, Amazon API Gateway helps you improve the performance of your APIs by caching the output of API calls to avoid calling your backend systems unnecessarily.

  - EASILY MONITOR API ACTIVITY -After your API is deployed, Amazon API Gateway provides you with a dashboard to visually monitor calls to your services using Amazon CloudWatch, so you see performance metrics and information on API calls, data latency, and error rates.

  - RUN YOUR APIS WITHOUT SERVERS-Amazon API Gateway tightly integrates with AWS Lambda to allow you to create completely serverless APIs.AWS Lambda runs your code on a high-availability compute infrastructure and performs all the heavy lifting and administration of your compute resources.

API Gateway

- Benefits

  - CREATE RESTFUL API ENDPOINTS FOR EXISTING SERVICES -With Amazon API Gateway, you can create modern resource based APIs, and then use the dynamic and flexible data transformation capabilities to generate the requests in the language your target services expect. API Gateway also helps you protect your existing services by setting throttling rules to avoid overwhelming your back-end infrastructure during unpredictable traffic spikes.

AWS Elastic Beanstalk

- AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

- You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. At the same time, you retain full control over the AWS resources powering your application and can access the underlying resources at any time.

- There is no additional charge for Elastic Beanstalk - you pay only for the AWS resources needed to store and run your applications.

AWS Elastic Beanstalk

- Benefits
  - FAST AND SIMPLE TO BEGIN-Elastic Beanstalk is the fastest and simplest way to deploy your application on AWS. You simply use the AWS Management Console, a Git repository, or an integrated development environment (IDE) such as Eclipse or Visual Studio to upload your application, and Elastic Beanstalk automatically handles the deployment details of capacity provisioning, load balancing, auto-scaling, and application health monitoring. Within minutes, your application will be ready to use without any infrastructure or resource configuration work on your part.
  - DEVELOPER PRODUCTIVITY-Elastic Beanstalk provisions and operates the infrastructure and manages the application stack (platform) for you, so you don't have to spend the time or develop the expertise. It will also keep the underlying platform running your application up-to-date with the latest patches and updates. Instead, you can focus on writing code rather than spending time managing and configuring servers, databases, load balancers, firewalls, and networks

AWS Elastic Beanstalk

- Benefits

  - COMPLETE RESOURCE CONTROL-You have the freedom to select the AWS resources, such as Amazon EC2 instance type, that are optimal for your application. Additionally, Elastic Beanstalk lets you "open the hood" and retain full control over the AWS resources powering your application. If you decide you want to take over some (or all) of the elements of your infrastructure, you can do so seamlessly by using Elastic Beanstalk's management capabilities.

  - IMPOSSIBLE TO OUTGROW- Elastic Beanstalk automatically scales your application up and down based on your application's specific need using easily adjustable Auto Scaling settings. For example, you can use CPU utilization metrics to trigger Auto Scaling actions. With Elastic Beanstalk, your application can handle peaks in workload or traffic while minimizing your costs.
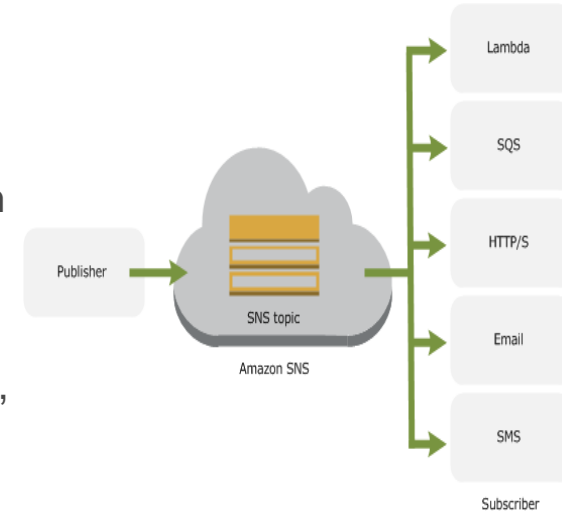
Amazon Simple Notification Service (SNS)

- Amazon Simple Notification Service (SNS) is a flexible, fully managed pub/sub messaging and mobile notifications service for coordinating the delivery of messages to subscribing endpoints and clients.
- With SNS you can fan-out messages to a large number of subscribers, including distributed systems and services, and mobile devices. It is easy to set up, operate, and reliably send notifications to all your endpoints – at any scale.
- You can get started using SNS in a matter of minutes using the AWS Management Console, AWS Command Line Interface, or using the AWS SDK with just three simple APIs. SNS eliminates the complexity and overhead associated with managing and operating dedicated messaging software and infrastructure.

Amazon Simple Notification Service (SNS)

- **Amazon SNS Pub/Sub Messaging**
  - Amazon SNS is a fully managed pub/sub messaging service that makes it easy to decouple and scale microservices, distributed systems, and serverless applications.
  - With SNS, you can use topics to decouple message publishers from subscribers, fan-out messages to multiple recipients at once, and eliminate polling in your applications.
  - SNS supports a variety of subscription types, allowing you to push messages directly to Amazon Simple Queue Service (SQS) queues, AWS Lambda functions, and HTTP endpoints. AWS services, such as Amazon EC2, Amazon S3 and Amazon CloudWatch, can publish messages to your SNS topics to trigger event-driven computing and workflows.
  - SNS works with SQS to provide a powerful messaging solution for building cloud applications that are fault tolerant and easy to scale.

Amazon Simple Queue Service (SQS)

- Amazon Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications.

- SQS eliminates the complexity and overhead associated with managing and operating message oriented middleware, and empowers developers to focus on differentiating work.

- Using SQS, you can send, store, and receive messages between software components at any volume, without losing messages or requiring other services to be available. Get started with SQS in minutes using the AWS console, Command Line Interface or SDK of your choice, and three simple commands.

- SQS offers two types of message queues. Standard queues offer maximum throughput, best-effort ordering, and at-least-once delivery. SQS FIFO queues are designed to guarantee that messages are processed exactly once, in the exact order that they are sent.

Amazon CloudWatch- Complete Visibility of Your Cloud Resources and Applications

- Amazon CloudWatch is a monitoring and management service built for developers, system operators, site reliability engineers (SRE), and IT managers.

- CloudWatch provides you with data and actionable insights to monitor your applications, understand and respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health.
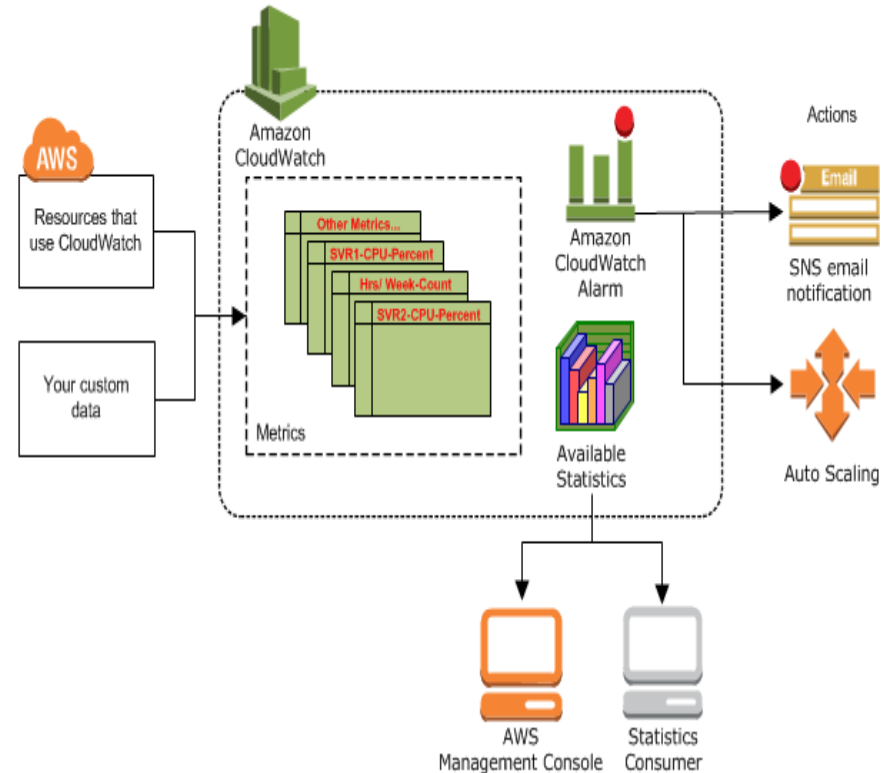
Amazon CloudWatch- Complete Visibility of Your Cloud Resources and Applications

- CloudWatch collects monitoring and operational data in the form of logs, metrics, and events, providing you with a unified view of AWS resources, applications and services that run on AWS, and on-premises servers.

- You can use CloudWatch to set high resolution alarms, visualize logs and metrics side by side, take automated actions, troubleshoot issues, and discover insights to optimize your applications, and ensure they are running smoothly.
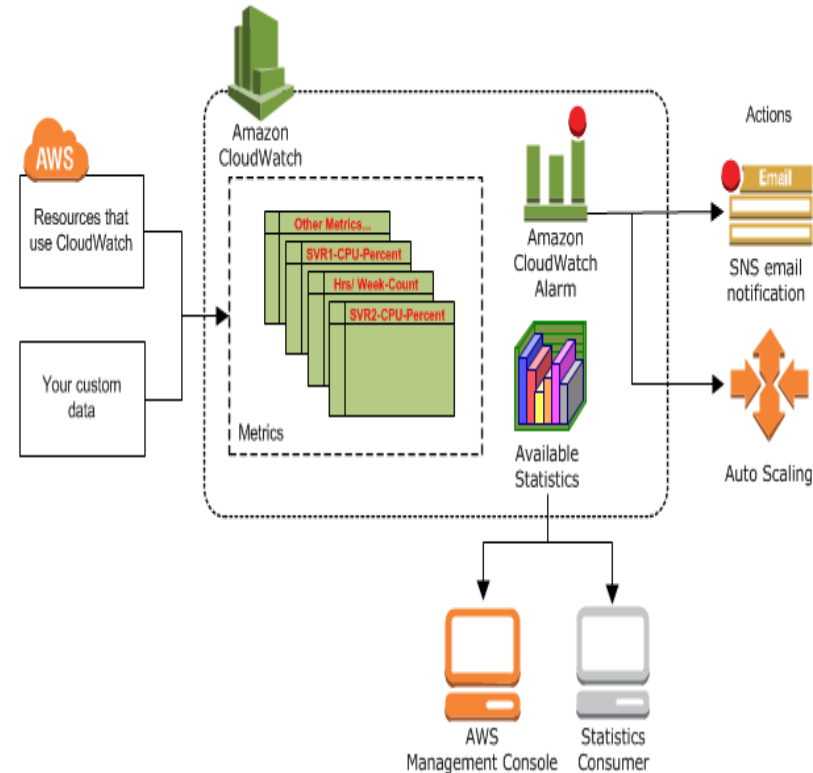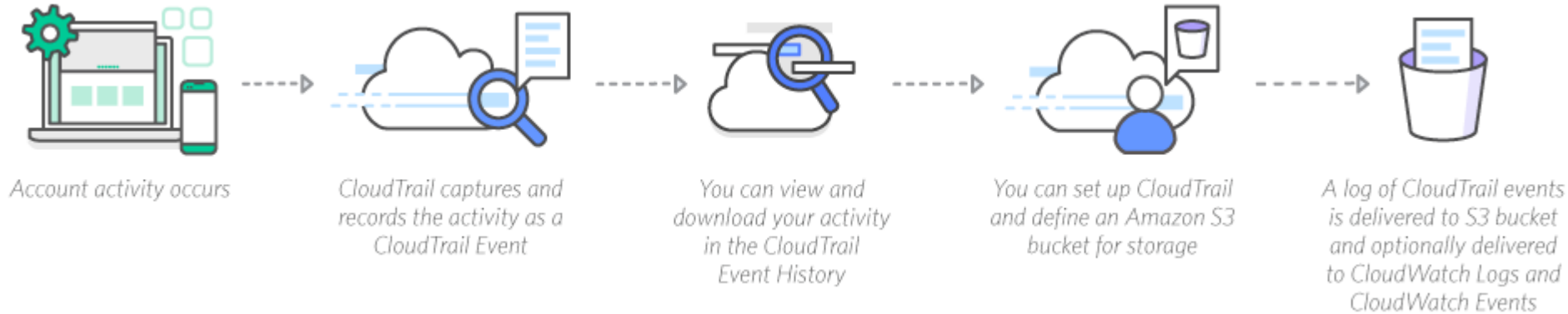
AWS CloudTrail - Track user activity and API usage

- AWS CloudTrail is a service that enables governance, compliance, operational auditing, and risk auditing of your AWS account.

- With CloudTrail, you can log, continuously monitor, and retain account activity related to actions across your AWS infrastructure.

- CloudTrail provides event history of your AWS account activity, including actions taken through the AWS Management Console, AWS SDKs, command line tools, and other AWS services.

- This event history simplifies security analysis, resource change tracking, and troubleshooting.

AWS CloudTrail -
Track user activity and API usage

- How it works

Account activity occurs

CloudTrail captures and records the activity as a CloudTrail Event

You can view and download your activity in the CloudTrail Event History

You can set up CloudTrail and define an Amazon S3 bucket for storage

A log of CloudTrail events is delivered to S3 bucket and optionally delivered to CloudWatch Logs and CloudWatch Events

What is RDS?

- RDS (Relational Database Service) : Amazon Relational Database Service (Amazon RDS) is a web service that makes it easy to set up, operate, and scale a relational database in the cloud .Started in 2009

RDS

- Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud
- It provides cost-efficient and resizable capacity while managing time-consuming database administration tasks, freeing you up to focus on your applications and business
- Amazon RDS provides you six familiar database engines to choose from

Amazon RDS Database Engines

RDS-Benefits

- No need to launch EC2 instance
- No need to install and manage database servers
- No need to setup replication (primary database on instance A and secondary database on instance B)
- Even minor upgrade of your database version are taken care by AWS

AWS DynamoDB

- Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability.

- You can use Amazon DynamoDB to create a database table that can store and retrieve any amount of data, and serve any level of request traffic.

- Amazon DynamoDB automatically spreads the data and traffic for the table over a sufficient number of servers to handle the request capacity specified by the customer and the amount of data stored, while maintaining consistent and fast performance.

- Amazon DynamoDB provides on-demand backup capability. It allows you to create full backups of your tables for long-term retention and archival for regulatory compliance needs.

Amazon ElastiCache-Managed, Redis or Memcached-compatible in-memory data store.

- Amazon ElastiCache offers fully managed Redis and Memcached.
- Seamlessly deploy, run, and scale popular open source compatible in-memory data stores.
- Build data-intensive apps or improve the performance of your existing apps by retrieving data from high throughput and low latency in-memory data stores.
- Amazon ElastiCache is a popular choice for Gaming, Ad-Tech, Financial Services, Healthcare, and IoT apps.

Internet-scale applications
Real-time apps in Gaming, Ride Hailing, Media Streaming, Dating, and Social media need fast data access
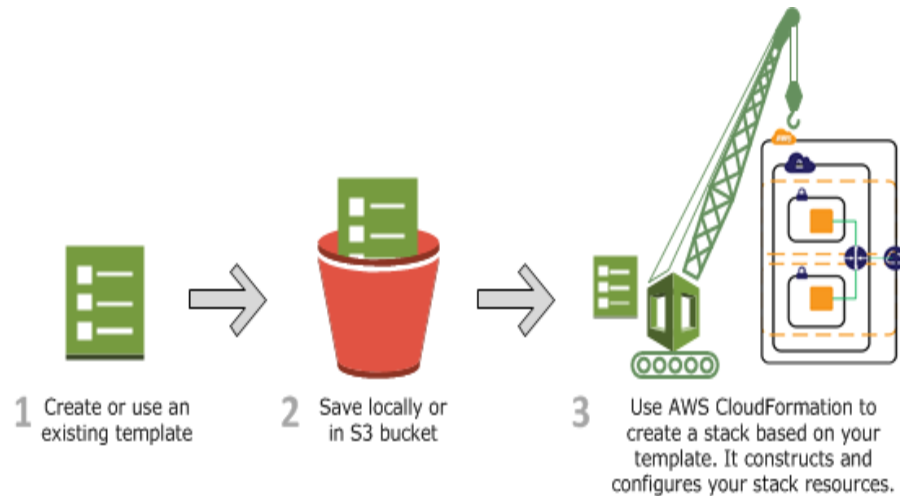
Amazon ElastiCache
Blazing fast in-memory data store for use as a database, cache, message broker, and queue. Store ephemeral data in-memory for sub-millisecond response

Use cases
Real-time transactions, chat, BI and analytics, session store, gaming leaderboards, and cache

AWS CloudFormation

- AWS CloudFormation is a service that helps you model and set up your Amazon Web Services resources so that you can spend less time managing those resources and more time focusing on your applications that run in AWS.

- You create a template that describes all the AWS resources that you want (like Amazon EC2 instances or Amazon RDS DB instances), and AWS CloudFormation takes care of provisioning and configuring those resources for you.

- You don't need to individually create and configure AWS resources and figure out what's dependent on what; AWS CloudFormation handles all of that.

1 Create or use an existing template

2 Save locally or in S3 bucket

3 Use AWS CloudFormation to create a stack based on your template. It constructs and configures your stack resources.

AWS CloudFormation

- The following scenarios demonstrate how AWS CloudFormation can help.

  - **Simplify Infrastructure Management –**You can create or modify an existing AWS CloudFormation template. A template describes all of your resources and their properties. When you use that template to create an AWS CloudFormation stack, AWS CloudFormation provisions the Auto Scaling group, load balancer, and database for you. After the stack has been successfully created, your AWS resources are up and running. You can delete the stack just as easily, which deletes all the resources in the stack. By using AWS CloudFormation, you easily manage a collection of resources as a single unit.

  - **Quickly Replicate Your Infrastructure -**If your application requires additional availability, you might replicate it in multiple regions so that if one region becomes unavailable, your users can still use your application in other regions.

AWS CloudFormation

- The following scenarios demonstrate how AWS CloudFormation can help.

  - **Easily Control and Track Changes to Your Infrastructure-** When you provision your infrastructure with AWS CloudFormation, the AWS CloudFormation template describes exactly what resources are provisioned and their settings. Because these templates are text files, you simply track differences in your templates to track changes to your infrastructure, similar to the way developers control revisions to source code. For example, you can use a version control system with your templates so that you know exactly what changes were made, who made them, and when. If at any point you need to reverse changes to your infrastructure, you can use a previous version of your template.
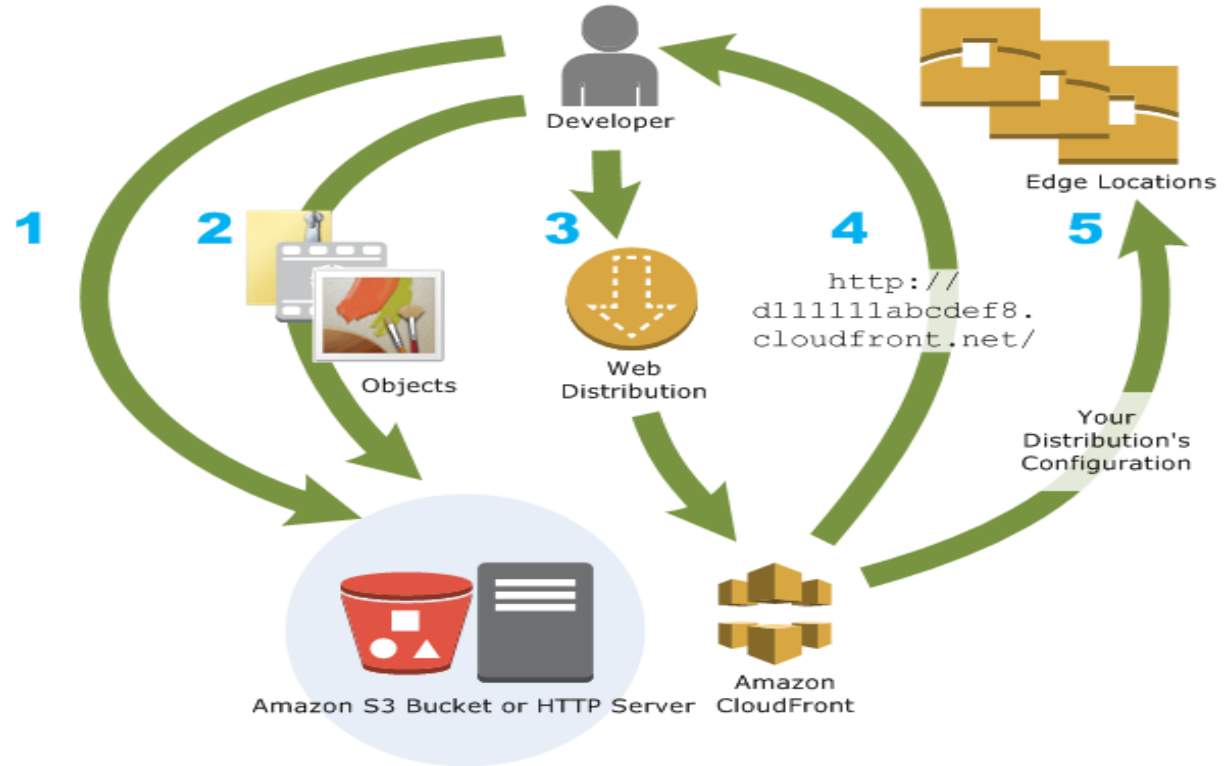
Amazon CloudFront

- 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 user 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 MediaPackage channel, or an HTTP server (for example, a web server) that you have identified as the source for the definitive version of your content.

Amazon CloudFront

- CloudFront speeds up the distribution of your content by routing each user request through the AWS backbone network to the edge location that can best serve your content.

- You also get increased reliability and availability because copies of your files (also known as *objects)* are now held (or cached) in multiple edge locations around the world.
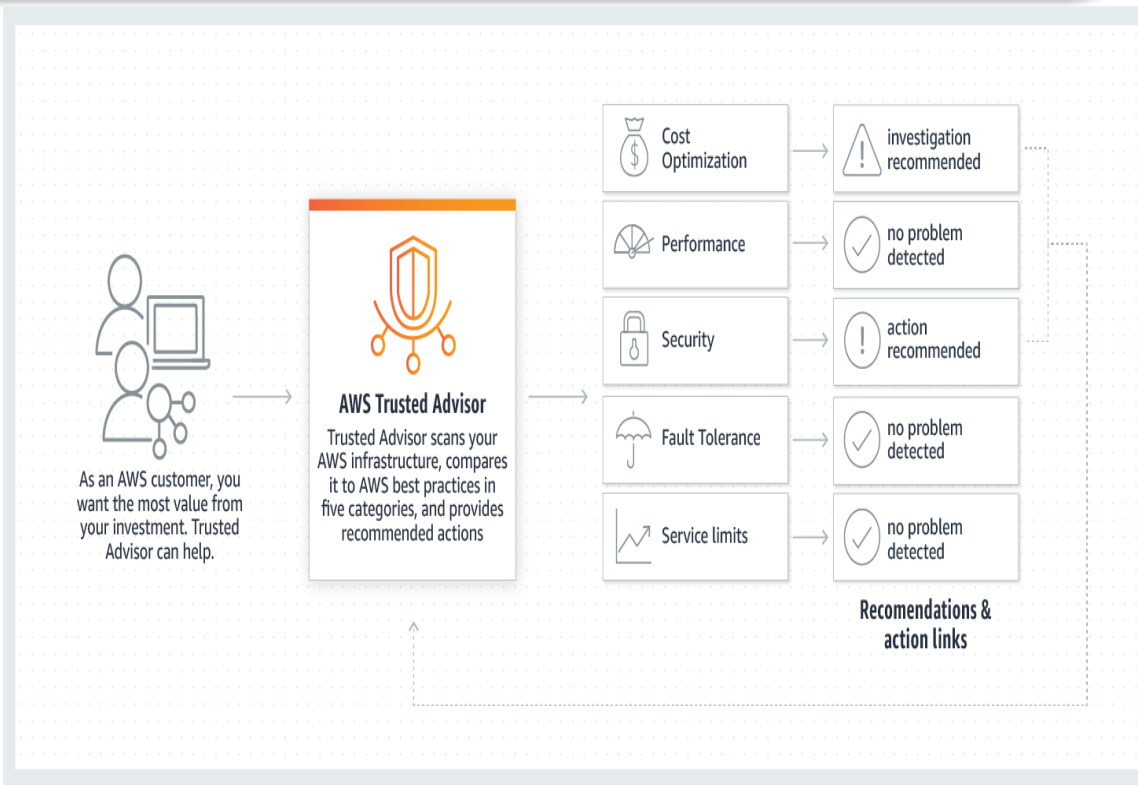
Amazon CloudFront

- **How You Set Up CloudFront to Deliver Content**

Developer

Objects

Web Distribution

http://d111111abcdef8.cloudfront.net/

Your Distribution's Configuration

Edge Locations

Amazon S3 Bucket or HTTP Server

Amazon CloudFront

AWS Trusted Advisor

- An online resource to help you reduce cost, increase performance, and improve security by optimizing your AWS environment,

- Trusted Advisor provides real time guidance to help you provision your resources following AWS best practices.

As an AWS customer, you want the most value from your investment. Trusted Advisor can help.

AWS Trusted Advisor

Trusted Advisor scans your AWS infrastructure, compares it to AWS best practices in five categories, and provides recommended actions

Cost Optimization — investigation recommended

Performance — no problem detected

Security — action recommended

Fault Tolerance — no problem detected

Service limits — no problem detected

Recomendations & action links

Amazon Simple Workflow Service (SWF)

- Amazon SWF helps developers build, run, and scale background jobs that have parallel or sequential steps.

- You can think of Amazon SWF as a fully-managed state tracker and task coordinator in the Cloud.

- AWS Flow Framework application consists of three basic components: *workflow starters*, *workflow workers*, and *activity workers*. One or more host applications are responsible for registering the workers (workflow and activity) with Amazon SWF, starting the workers, and handling cleanup.

- The workers handle the mechanics of executing the workflow and may be implemented on several hosts.
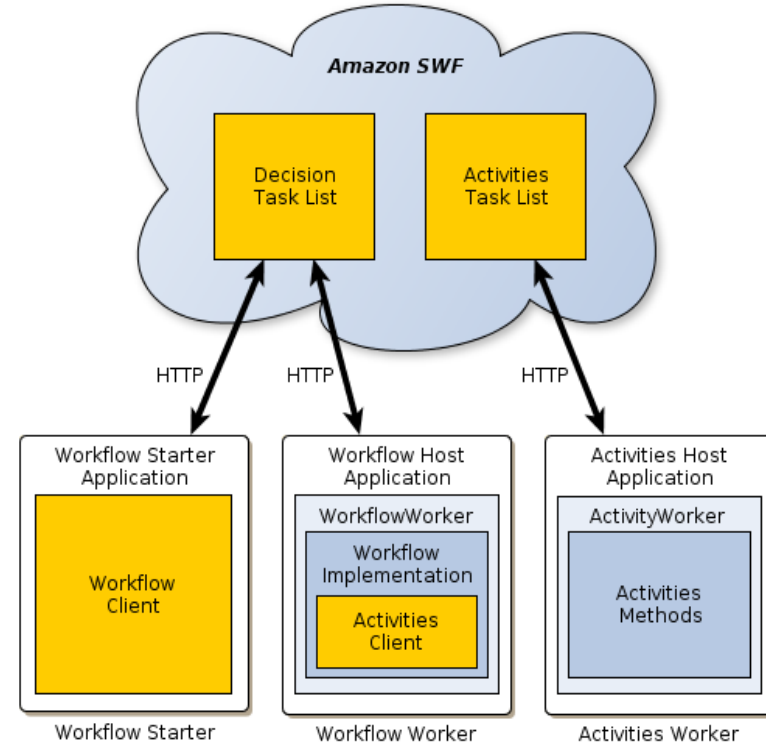
AWS Direct Connect

- AWS Direct Connect is a cloud service solution that makes it easy to establish a dedicated network connection from your premises to AWS.

- Using AWS Direct Connect, you can establish private connectivity between AWS and your datacenter, office, or colocation environment, which in many cases can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than Internet-based connections.

- AWS Direct Connect lets you establish a dedicated network connection between your network and one of the AWS Direct Connect locations.

- Using industry standard 802.1q VLANs, this dedicated connection can be partitioned into multiple virtual interfaces.

AWS Direct Connect

- This allows you to use the same connection to access public resources such as objects stored in Amazon S3 using public IP address space, and private resources such as Amazon EC2 instances running within an Amazon Virtual Private Cloud (VPC) using private IP space, while maintaining network separation between the public and private environments. Virtual interfaces can be reconfigured at any time to meet your changing needs.
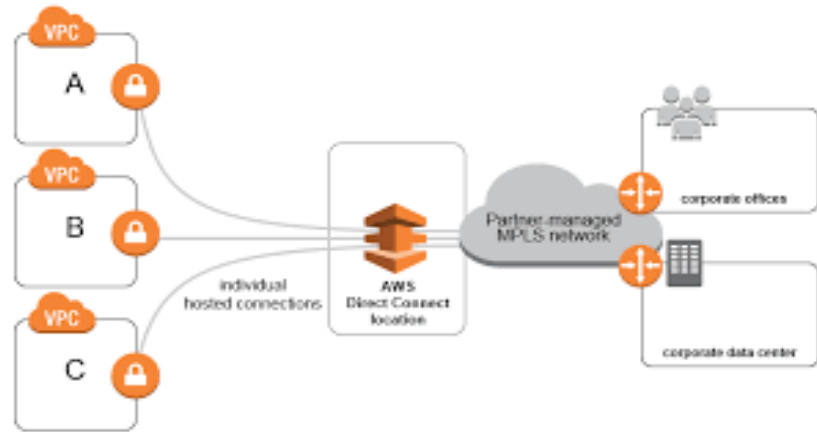
Multiprotocol Label Switching (**MPLS**) is a type of data-carrying technique for high-performance telecommunications **networks**. **MPLS** directs data from one **network** node to the next based on short path labels rather than long **network** addresses, avoiding complex lookups in a routing table.

AWS Direct Connect
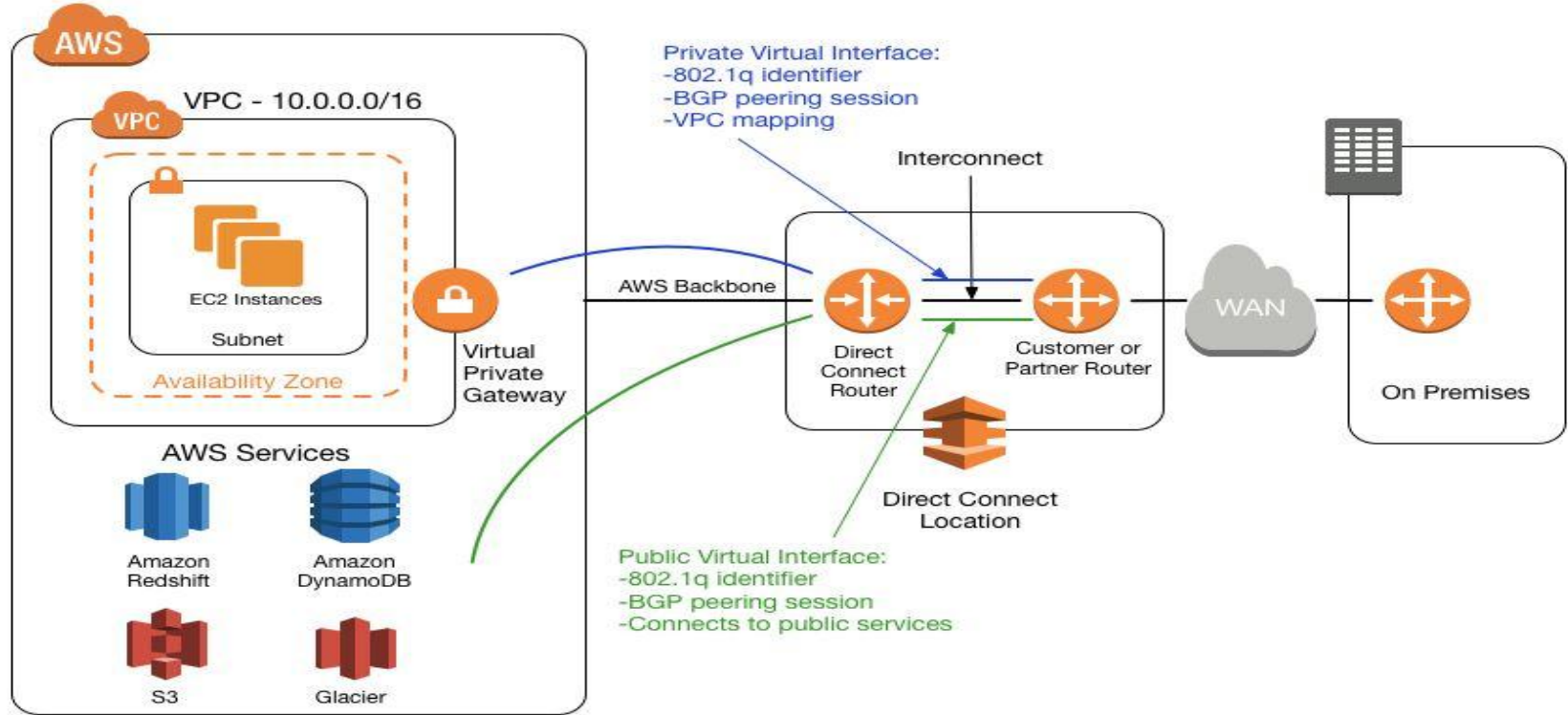
Summary

- We learned various services and understood when to use which service.