

# Cloud computing 101

---

## Cloud Computing Basics

---

### Basics

- elastic service, variable expense, no huge capital expense
- No need for estimating infrastructure resources
- No need for maintenance of data centers and hardwares
- Easy deployment

### Types of cloud computing

1. Infrastructure as a Service
2. Platform as a Service
3. Application as a Service

### Hybrid deployment

Connecting resources that are cloud based and non-cloud based(on-premises)

---

## AWS Analytics Services

---

### 1. Elastic Search (ES)

- Free Opensource analytic engine and *full-text search engine*
- Used for enabling *search functionalities* for applications
- Data is stored as documents ( same as rows in database )
- Fields as columns
- Can be used to detect anomalies
- as it is coupled with kabana, so we can query and visualize data behaviours, real time analytics
- ES can integrate with DynamoDB also

### 2. Elastic MapReduce (EMR)

- cluster management framework/platform
- simplifies big data running frameworks like Hadoop, apache spark

- can also be used to move and transform large amounts of data between other aws storage services like S3, dynamoDB
- Storage types:
  1. HDFS (Hadoop Distributed File System)
  2. EMRFS (HDFS or Amazon s3 as file system in cluster)
  3. Local File System
- Uses yarn as default Cluster Resource Management
- Data Processing Frameworks(Hadoop MapReduce, Tex, Spark)
- Supports applications such as Hive, Pig, Spark, Tensorflow
- enables multiple clusters to access data same data

### **3. Amazon kinesis (AK)**

- platform for streaming data on aws
- managed alternative to Apache Kafka
- Great for 'real-time' big data
- can be used to process and analyze high volume/high velocity data from various sources
- data is automatically replicated synchronously to 3 AZ
- Contains three services:

#### **1. AK Firehose:**

- low latency streaming ingest

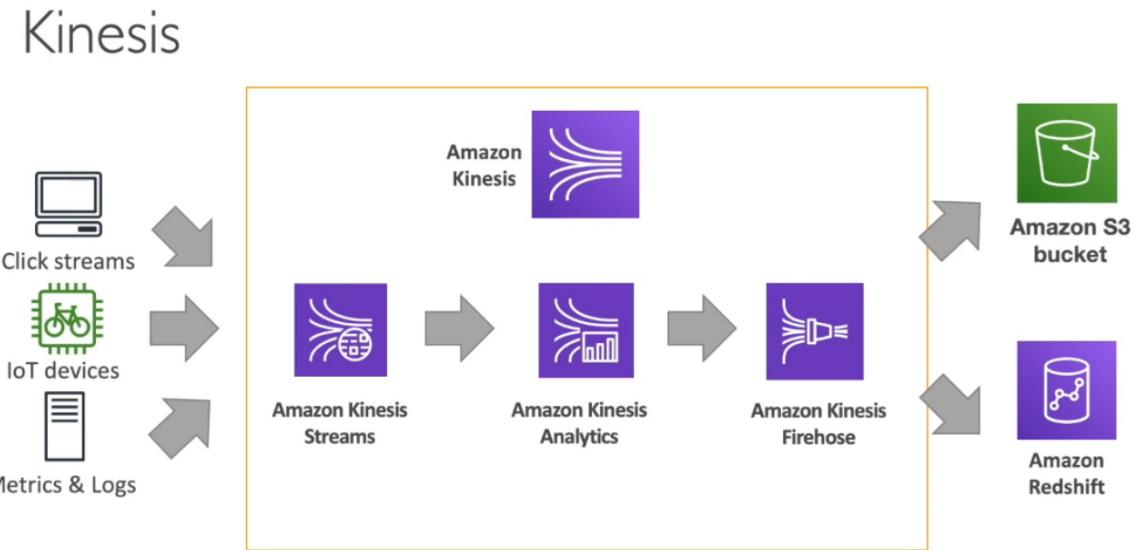
#### **2. AK Analytics**

perform real-time analytics on streams using SQL

#### **3. AK Streams**

load streams into S3, Redshift, ElasticSearch & Splunk

- Architecture of kinesis:

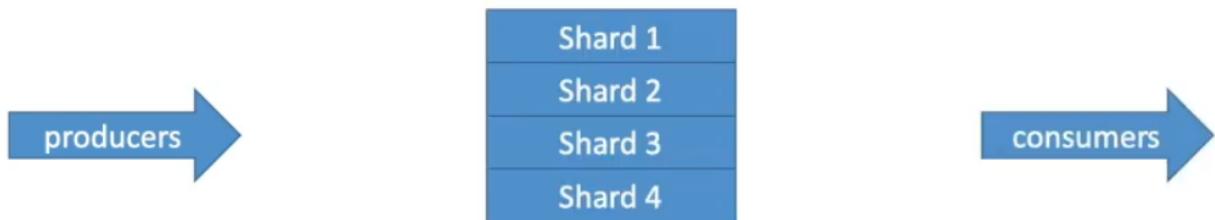


- **Streams:**

- on stream is divided into ordered shards/partitions
- data is retained only upto 24 hrs (unless extended to 7 days)
- Can reprocess/ replay data
- Real-time processing
- once data is inserted in Kinesis, it can't be deleted

- **Shards:**

- Records are ordered per shard
- billing is per shard provisioned
- no. of shards can evolve over time



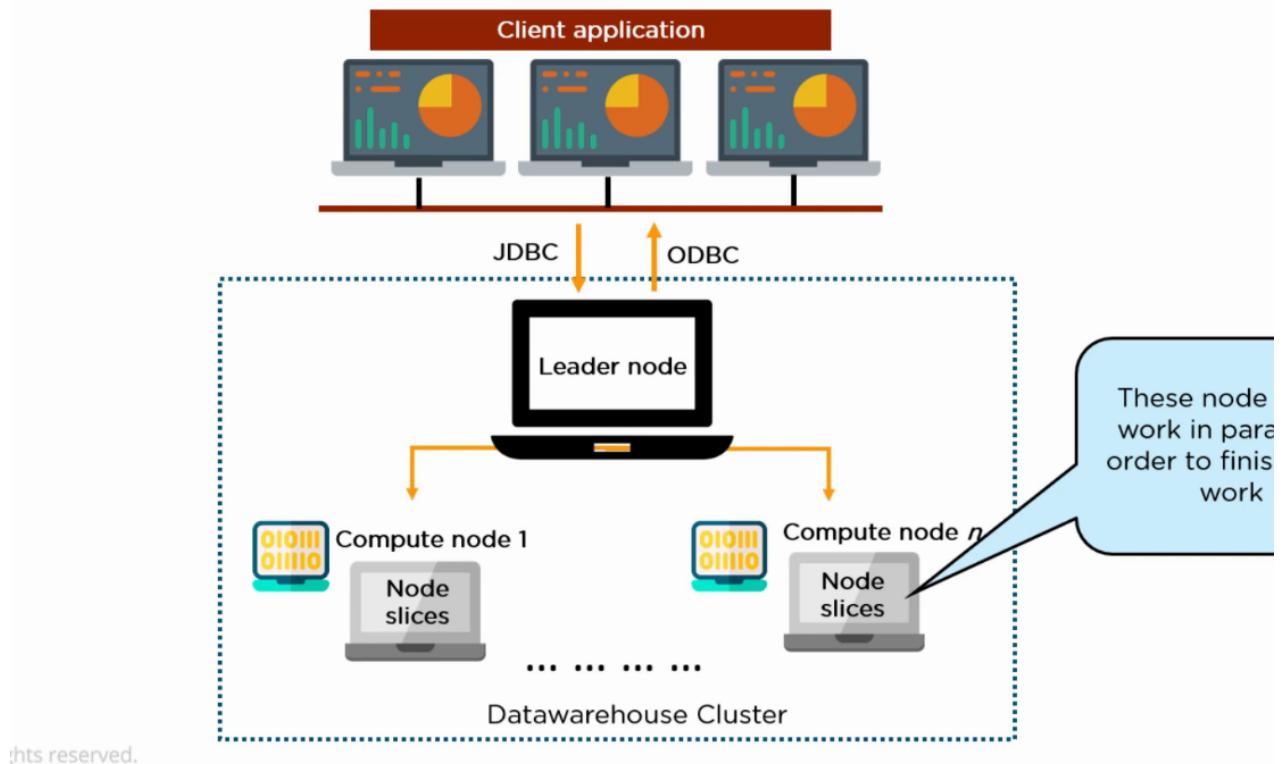
- **Limitations**

- Producer: limited to 1MB/s(1000msg/s) at write Per shard
- Consumer: 2MB/s read limit per shard across all consumers, 5 API calls/sec per shard across all consumers
- 24 hour unextended data retention period

## 4. Amazon Redshift

- data warehouse service in the cloud
- fast, scalable
- column based database designed for OLAP
- redshift contains a collection of computing resources called 'Nodes'
- groups of nodes is called cluster
- Limitation: not highly available in various zones

## Architecture of Amazon Redshift



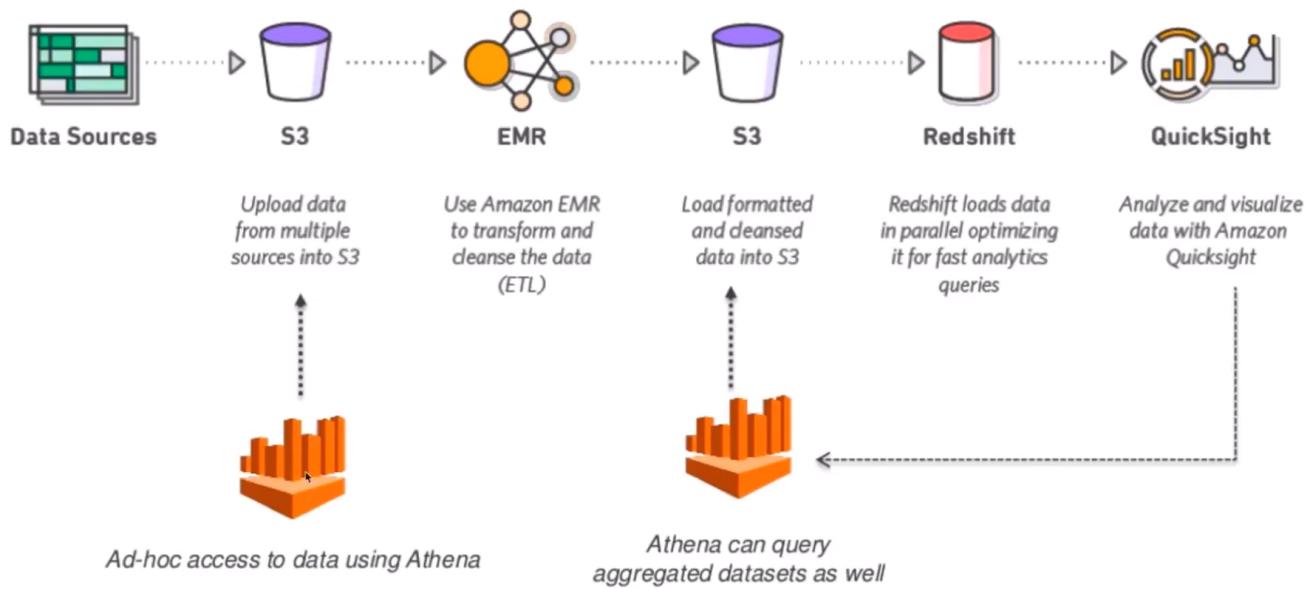
### 5. Amazon Athena

- interactive query service
- not an ETL tool
- pay per query(data scanned)
- used to analyze data (unstructured, semi-structured, structured) directly in S3
- supports data formats like CSV, JSON, Avro, columnar data (Parquet, ORC)
- uses standard SQL, ANSI SQL compliant
- it is serverless, ie no management, spin-up time, administration required
- queries are run parallelly
- built on top of Apache Presto
- works with Hive like meta store like AWS Glue
- *Cost Optimizing Methods:*

- use compression
  - use columnar formats(Parquet)
  - use partitions
- 

## Workflow of Redshift + EMR + Athena

### Redshift + EMR + Athena



## Amazon Compute Services

### Compute Service:

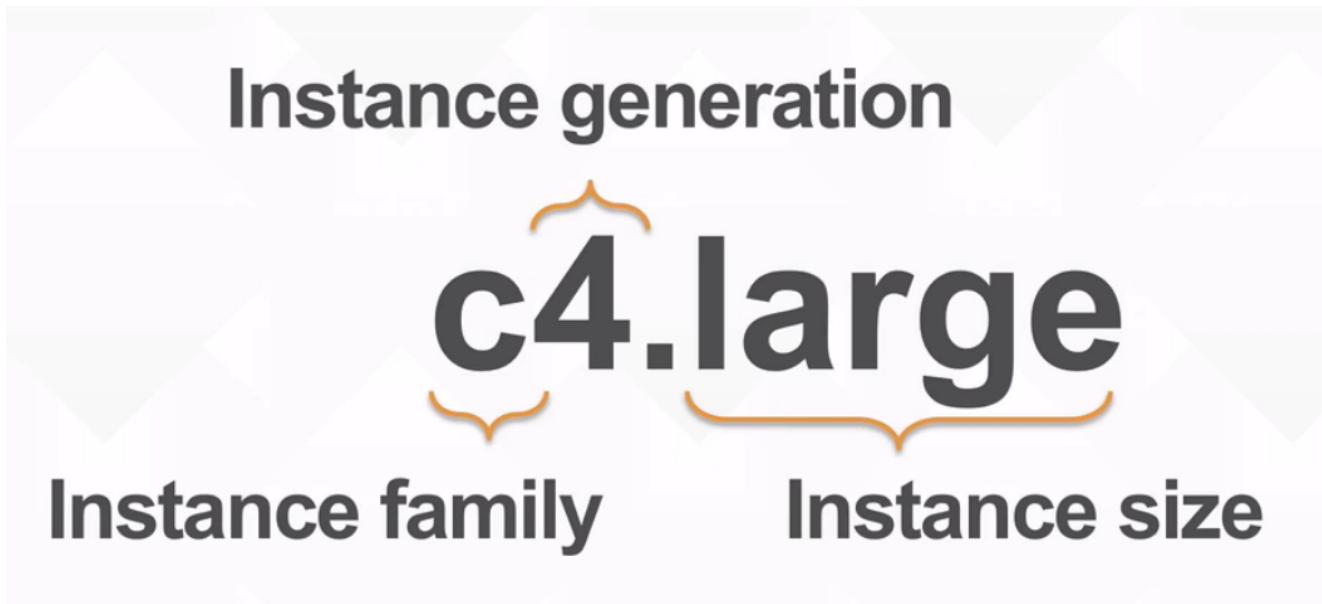
- Main processing power (brain) required by apps and systems to carry out computational tasks via series of instructions
- Closely related to server components as CPUs and RAMs

### 1. Amazon EC2

- Elastic virtual servers running on the cloud
- easy access to elastic server capacity
- Instance choices:
  1. CPU
  2. Memory
  3. Storage
- EC2 Instance families:

- General Purpose: M1, M3, T2, T3a, T3, A1
- Compute Optimized: C1, CC2, C3, C4, C5, C5a, C6g
- Memory Optimized: M2, CR1, R3, R5a, R5, R6g
- Accelerated Computing: F1, G3, G4, Inf1, P2, P4
- Dense Storage: HS1, D2, I3en, I3
- I/O Optimized: HI1, I2
- GPU: CS1, G2
- Micro: T1, T2

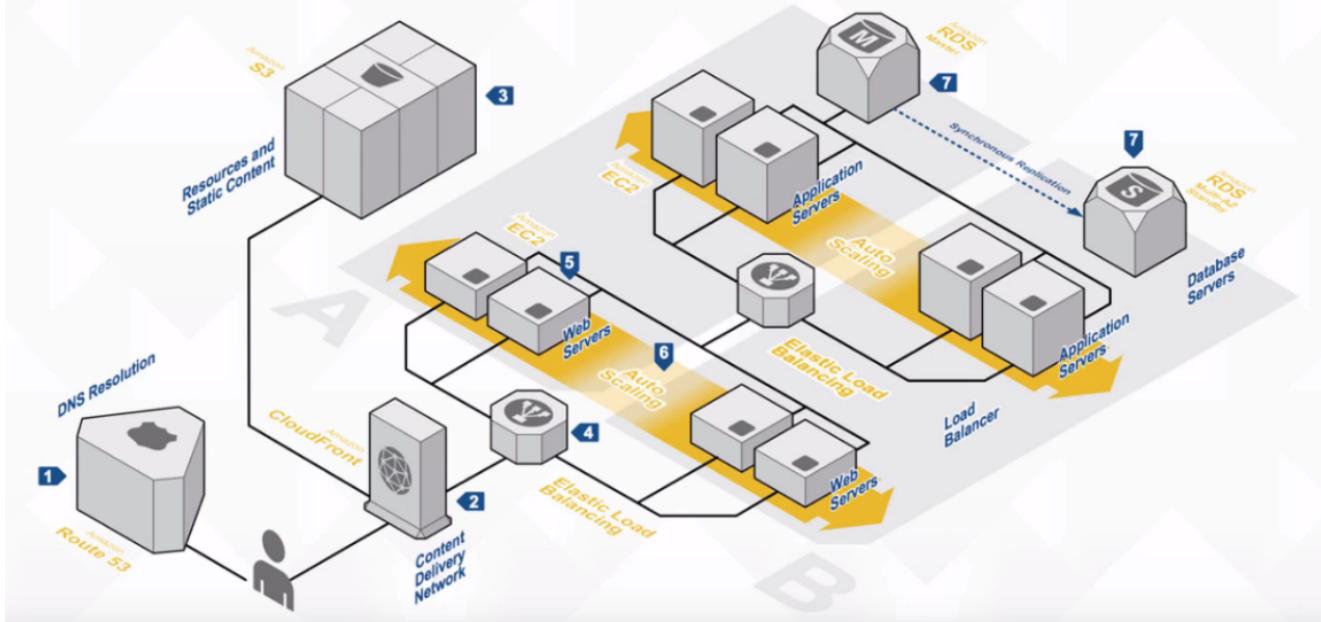
- Naming convention:



- Networking performance factor:
  - Highest performance in highest (8x large) instances
  - performance increases with increase in instance size
- Storages:
  - Locally attached (instance storage)
  - EBS General purpose (SSD)
  - EBS Provisioned IOPS (SSD)
  - EBS Magnetic volumes (Low cost)
  - S3/ Glacier

- 3-Tier Web App Architecture

## Example 3-tier Web App architecture



- Security:

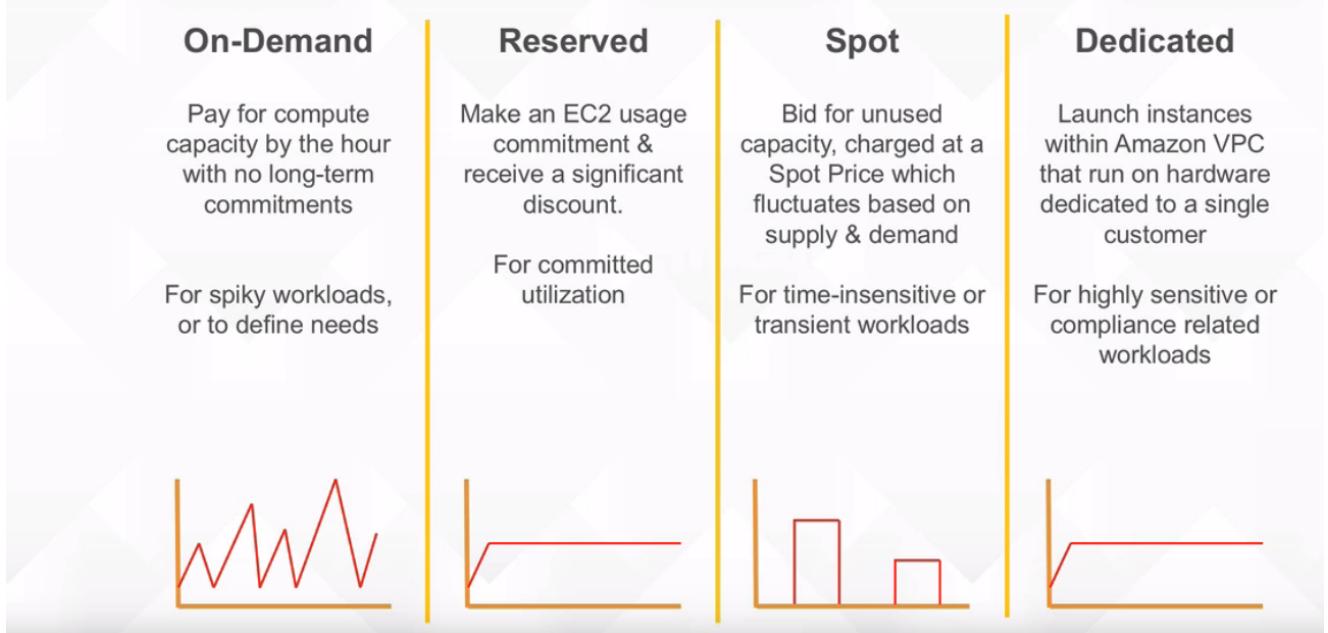
- Secured Premises
- Secured access
- Built-in firewalls
- Unique users
- Multi-factor authentication
- Private subnets
- Encrypted data storage
- Direct Connect

- **Amazon Virtual Private Cloud (VPC)**

- new instances of EC2 use VPC as a default
- provide logically isolated section of AWS cloud
- full control of how and when to internet access by instances
- connect to local (on-premises) network via hardware VPN

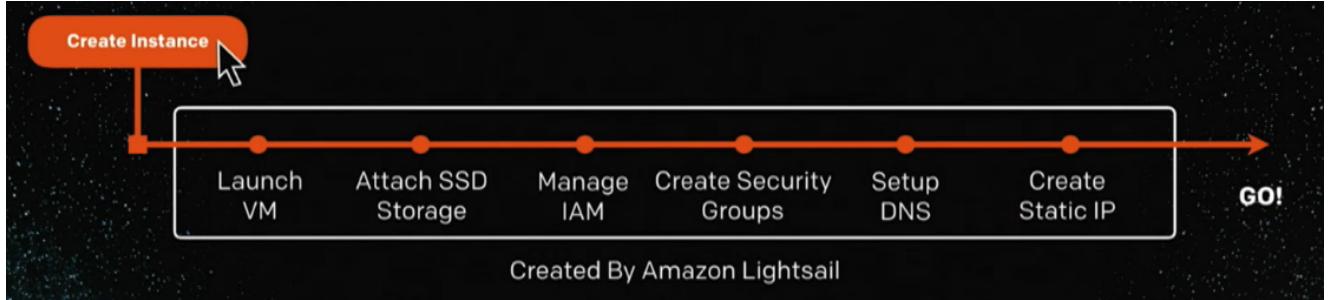
- Cost options to fit workloads:

## Purchase options that fit your workloads



## 2. Amazon Lightsail

- Easiest way to create and use VPS among AWS services
- PaaS



- Steps inside the margin handled automatically by lightsail
- Mostly suitable for developers, freelancers, hobbyists
- More focus on project than on installing and setting-up software/frameworks
- contains base OSs, stacks (LAMP, LEMP), apps (wordpress, drupal)

- Difference between EC2 and Lightsail:

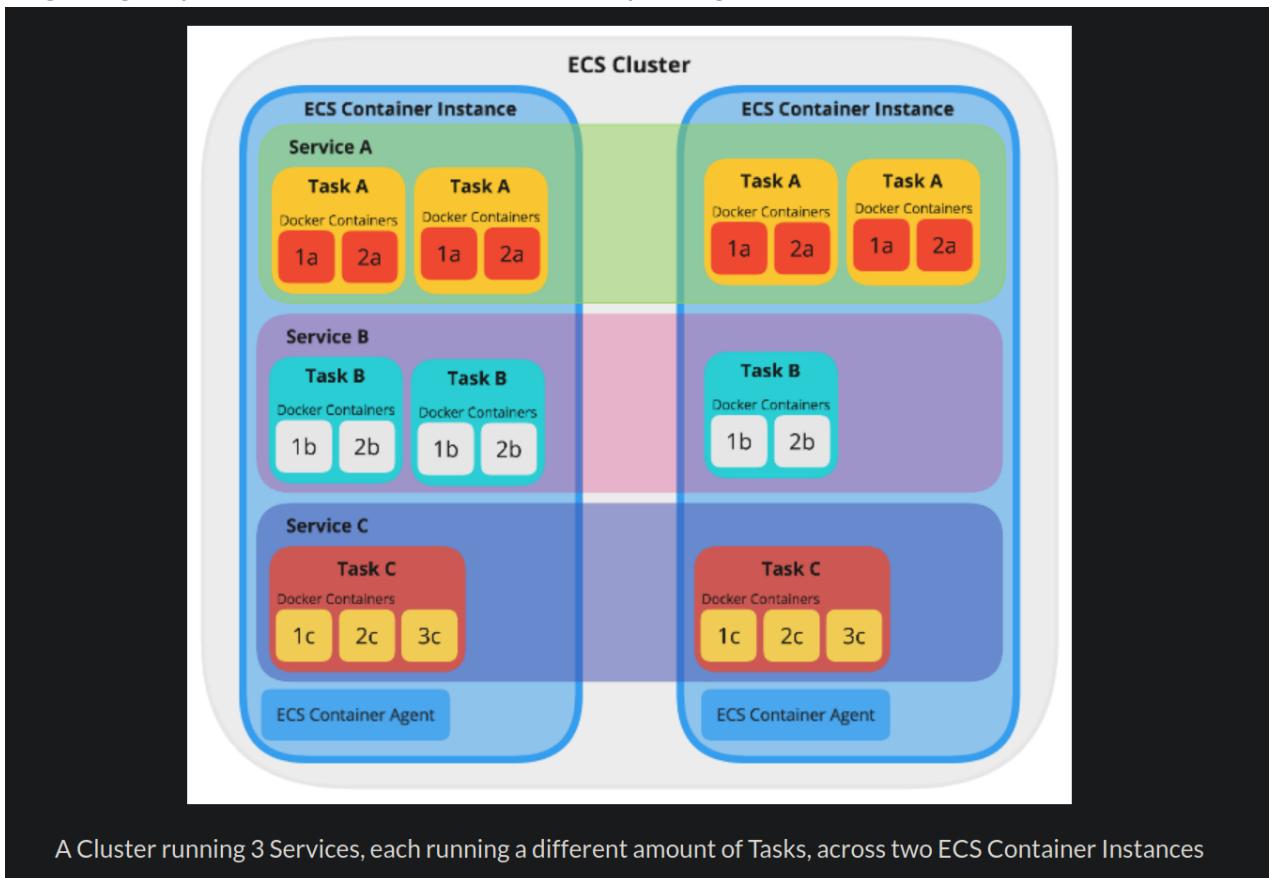
	Amazon Lightsail	Amazon EC2
<b>Pricing</b>	Lightsail is much less costlier than EC2 instances. Various outbound costs are included in lightsail instances, such as, each lightsail instance includes 5 TB of internet data transfer allowance.	Outbound costs are not included in EC2 instances. An enterprise using EC2 instances may have to pay \$90 per TB of outbound data transfer.
<b>Suitability</b>	Lightsail might not be ideal for applications that need highly configurable environment or high CPU performances. Lightsail would be perfect for small developers.	For applications that require high CPU performances or highly configurable environment, Amazon EC2 would be the most appropriate choice. EC2 is meant for huge organizations
<b>API integration</b>	There is no particular web service interface for Lightsail. Amazon Lightsail instances are not accessible directly from the AWS management console.	Amazon EC2 instances are virtual machines with Amazon's web service interface.

#### **Note:-**

- Load balancing: efficiently distributing incoming network traffic across a group of backend servers(server farm, server pool)
- Load balancer:
  - acts as traffic cop for client requests
  - maximizes speed and capacity utilization
  - ensure no one server is overworked
  - redirect and manage traffic when one or more servers fail/join

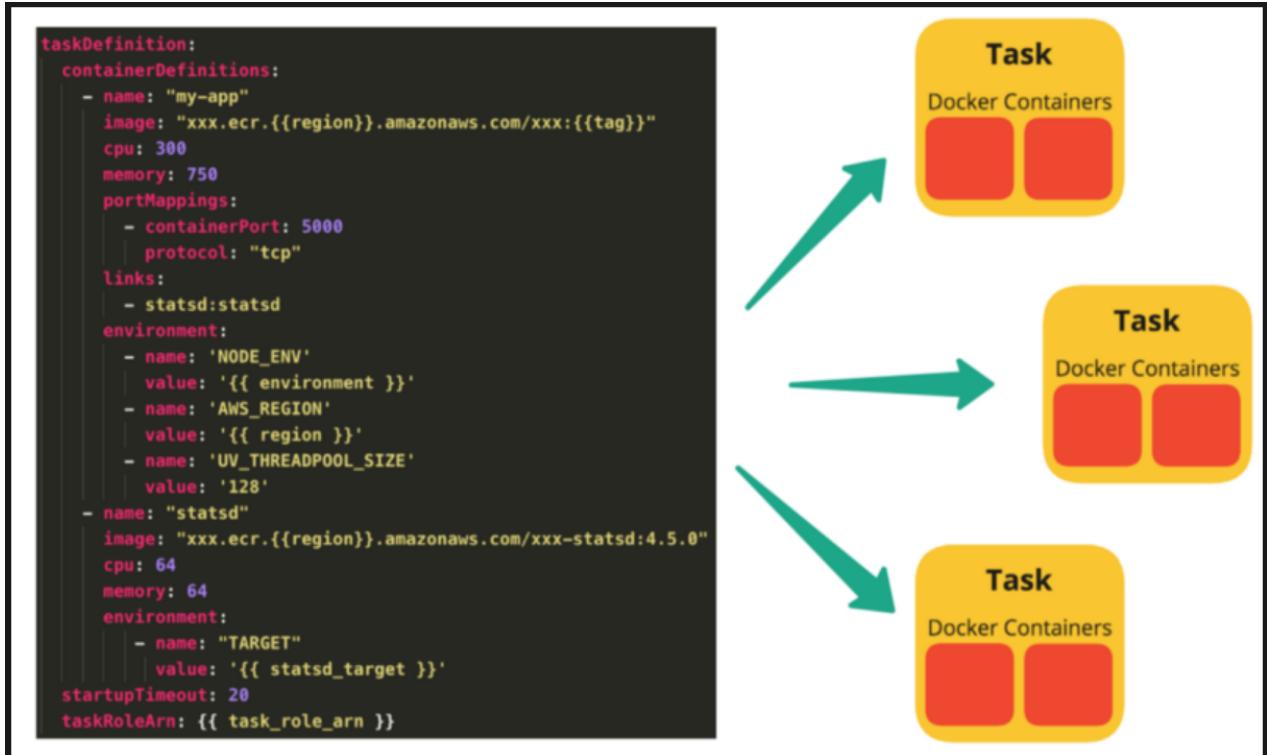
### **3. Amazon Elastic Container Service(ECS)**

- Container management service
- runs containers on a cluster of EC2 virtual machine instances
- registry for storing, managing, and deploying docker container images
- **ECS Cluster:**
  - group of ECS container instances
  - Logical group of amazon EC2 instances for placing containers onto



- **Task Definition:**
  - like a blueprint describing which docker containers to run
  - define which images to use, what CPU and memory to allocate
  - env variables, ports to expose
  - interaction between containers
- **Task:**
  - instance of a task definition
  - Group of containers that make up an application

- multiple tasks can be created by one Task Definition as demanded



- Service:**

- define min and max tasks from one task definition to run at any given time
- Define autoscaling, load balancing

## 4. Amazon Lambda

- compute service for running code without provisioning/managing servers
- pay for compute time consumed (per 100ms)
- just write working code and upload it to lambda
- automatic scaling based on size of workflow
- usecases - image conversion (into thumbnails), file compression etc

## AWS Database Services

### 1. DynamoDB:

- fast flexible NoSQL database
- always stored on SSD
- suited for read heavy applications
- Measuring capacity:  
1 row of data 4kb in size = 1 RCU  
1 row of data 1kb in size = 1 WCU

- does not use SQL queries
- queries only to indexed data (resulting very fast speed 1-9 ms)
- Trigger or procedures:
  - Small snippets of SQL or code executed when data changes
  - option to use a stream of changes
  - canstreams can be processed by a Lambda function

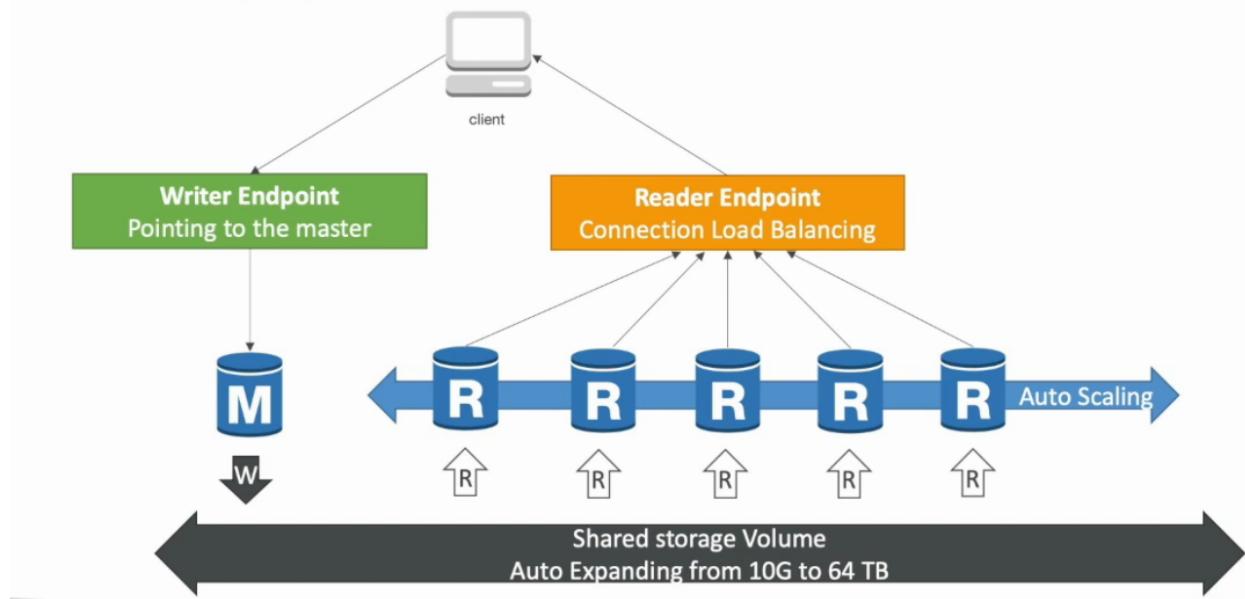
## 2. Relational Database Service (RDS):

- Relational cloud DB that supports major db services like aurora, mysql, mariaDB, postgreSQL

## 3. Amazon Aurora

- Proprietary AWS technology
- AWS cloud optimized, resulting 3x-5x faster than PSQL, mySQL
- aurora storage automatically grows in increments of 10 GB upto 64TB
- MySQL and PostgreSQL compatible

### Aurora DB Cluster



## AWS Developer Tools

### 1. Amazon CodeCommit:

- source control service for hosting private Git repos in the cloud
- store codes, binaries, images and libraries

## 2. Amazon CodePipeline

- CI/CD service
- fast application and infrastructure updates
- CodePipeline builds, tests and deploys code every time there is code change based on release process models
- code changes go through a pre-defined set of standardized steps/actions automatically
- can enable manual confirmation on every steps

## 3. Amazon CodeBuild:

- CI service
- compiles source code, runs tests and produces deployable software packages

## 4. Amazon CodeDeploy:

- CD service to EC2 instances
- automate deployments
- eliminates error-prone manual operations
- maximize app availability

---

# AWS Management tools

---

## 1. Amazon CloudFormation:

- service that helps to model and set up Amazon Web Services resources
- you can create template for easy setup using JSON
- JSON doc contains resources for AWS stack setup

## 2. Amazon CloudWatch:

- Monitoring service for cloud resources and running apps
- collect and track metrics about app performance health and react accordingly
- monitors resources such as EBS, EC2, ELB, S3
- 2 types of monitoring:

### 1. Basic Monitoring

- Free
- polls every 5 mins
- 10 metrics

- 5Gb of data ingestion and 5GB of data storage

## 2. Detailed Monitoring:

- chargeable
- polls every minute
- charged per instance per month
- Get visual outputs/hypervision
- create events for certain situations
- get monitor data/logs using cloudWatch agents on EC2
- can set alarms based on maxed-out resources usages, shut down unused/idle resources

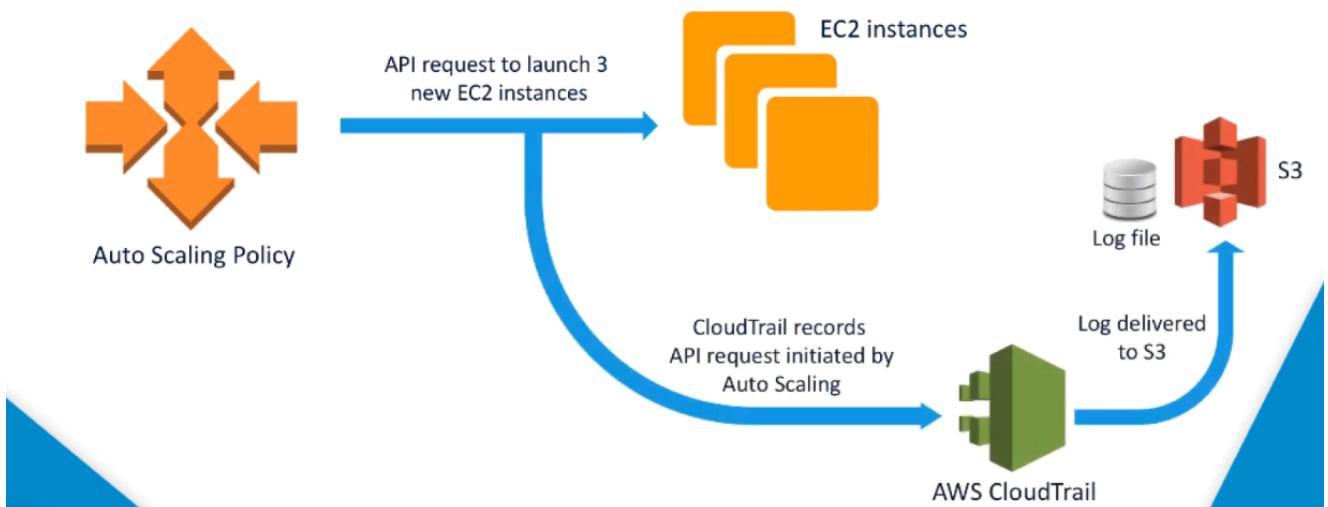
## 3. Amazon System Manager:

- gives visibility and control of AWS infrastructure
- provides a unified user interface for viewing operational data from AWS services
- allows automation of operational tasks accross resources

## 4. Amazon CloudTrail:

- Service that records and tracks all API requests in AWS account
- requests can be initiated from SDKs, AWS CLIs, AWS management console, other services
- Example:

## Example



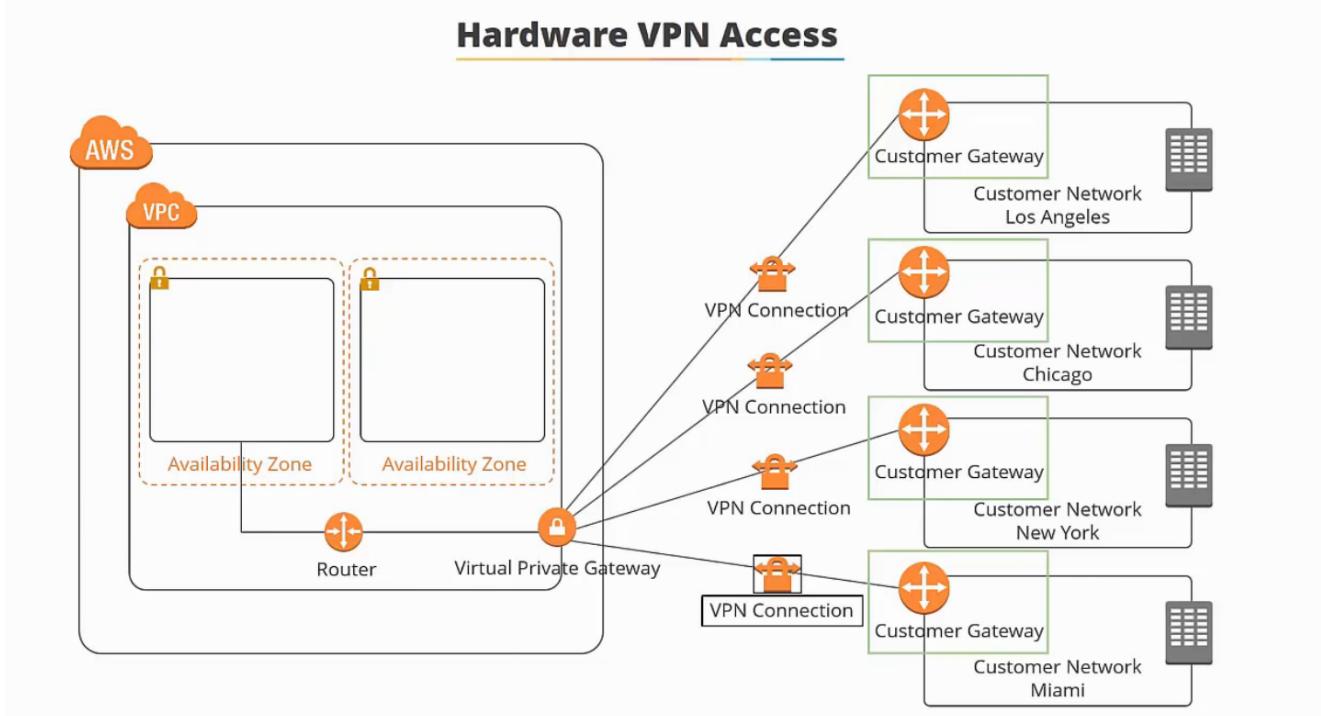
- logs are delivered and stored within S3, info like
  1. identification of the caller
  2. timestamp of request
  3. source IP address

- new log files created every 5 mins
- log files can be stored as long as required
- Effective for security analysis
- track changes to AWS infrastructure

## AWS Networking and Content Delivery Services

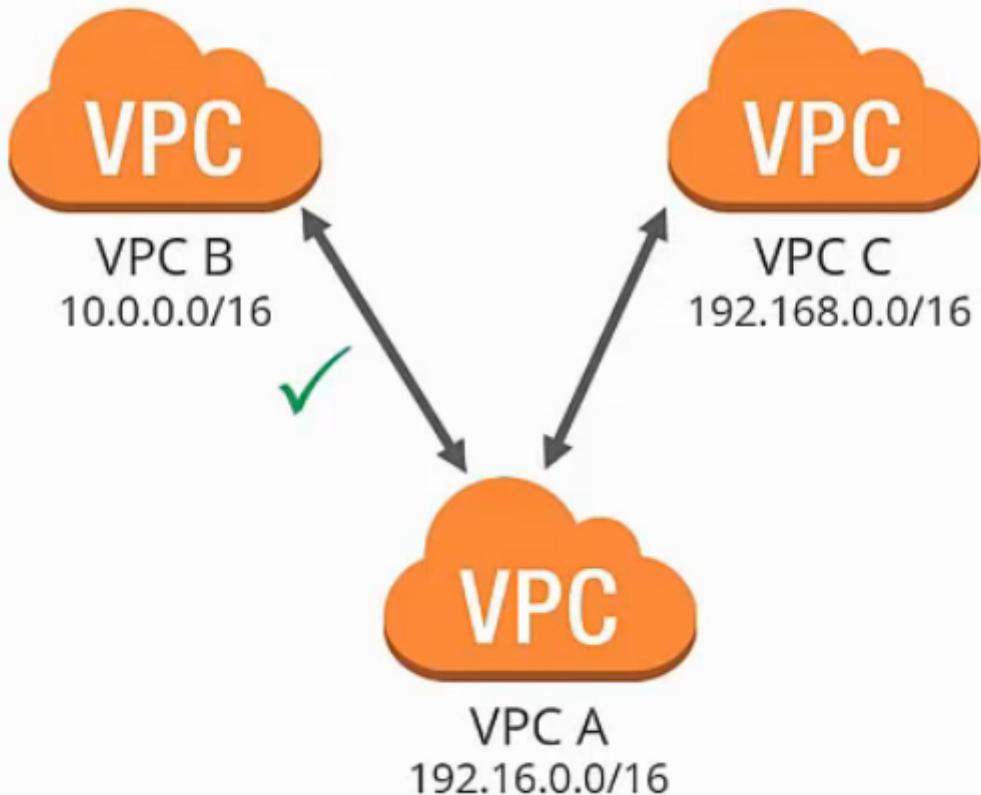
### 1. Amazon Virtual Private cloud (VPC):

- isolated region in the AWS cloud resources
- Each VPC is logically isolated from other networks in the AWS cloud
- each amazon account comes with a default VPC
- with a custom VPC, sets of IPs, subnets, gateways can be defined
- VPC can be connected to existing datacenter via hardware VPN access



- **VPC Peering:** create connection between two VPCs in same region

## VPC Peering



### 2. Amazon Direct Connect(DX):

- data travels through completely isolated tunnel from on-premise network to AWS
- dedicated network connection
- alternative (to internet) network service to connect customer's on-premise sites to AWS
- data can be exchanged through private network connection between AWS and on-premise datacenter or corporate network
- reduce cost, increase bandwidth, consistent network experience
- data is transferred through 3rd party companies (DX partner) like equinox, coresight

### 3. Amazon Route 53:

- DNS services for AWS
- health-checking services to route traffic to healthy endpoints

- create, update and manage public DNS records

## 4. Amazon Elastic Load Balancing(ELB):

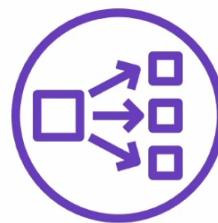
- Distribute and manage incoming app traffic/requests across multiple targets
- help manage and control flow and requests evenly across targeted resource groups
- targets could be fleet of EC2 instances, Lambda functions, IP addressed or containers
- automatically detect failure and redirect to remaining resources
- three types:
  1. Application load balancer
  2. Network load balancer
  3. classic load balancer

### Load Balancer Types



#### Application Load Balancer

Flexible feature set for your web applications running the HTTP or HTTPS protocols  
Operates at the request level  
Advanced routing, TLS termination and visibility features targeted at application architectures



#### Network Load Balancer

Ultra-high performance while maintaining very low latencies  
Operates at the connection level, routing traffic to targets within your VPC  
Handles millions of requests per second

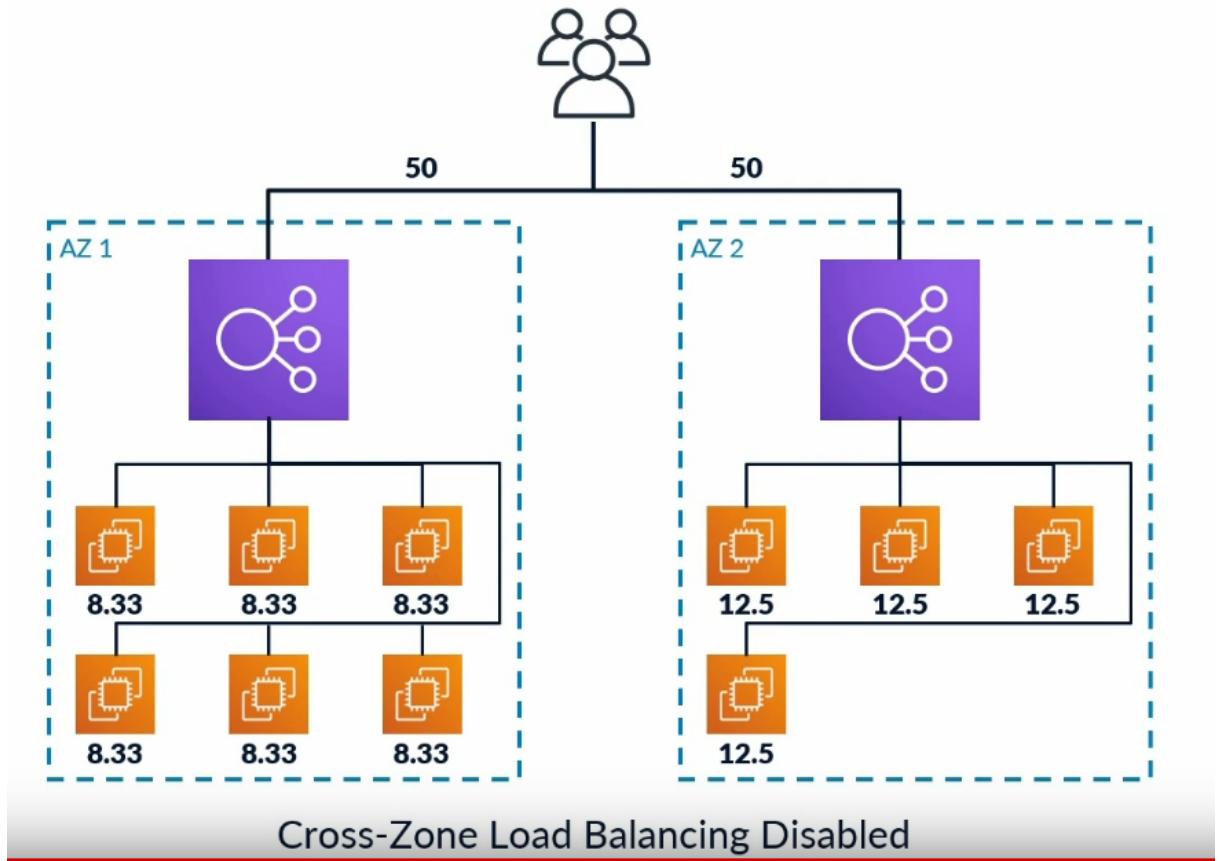


#### Classic Load Balancer

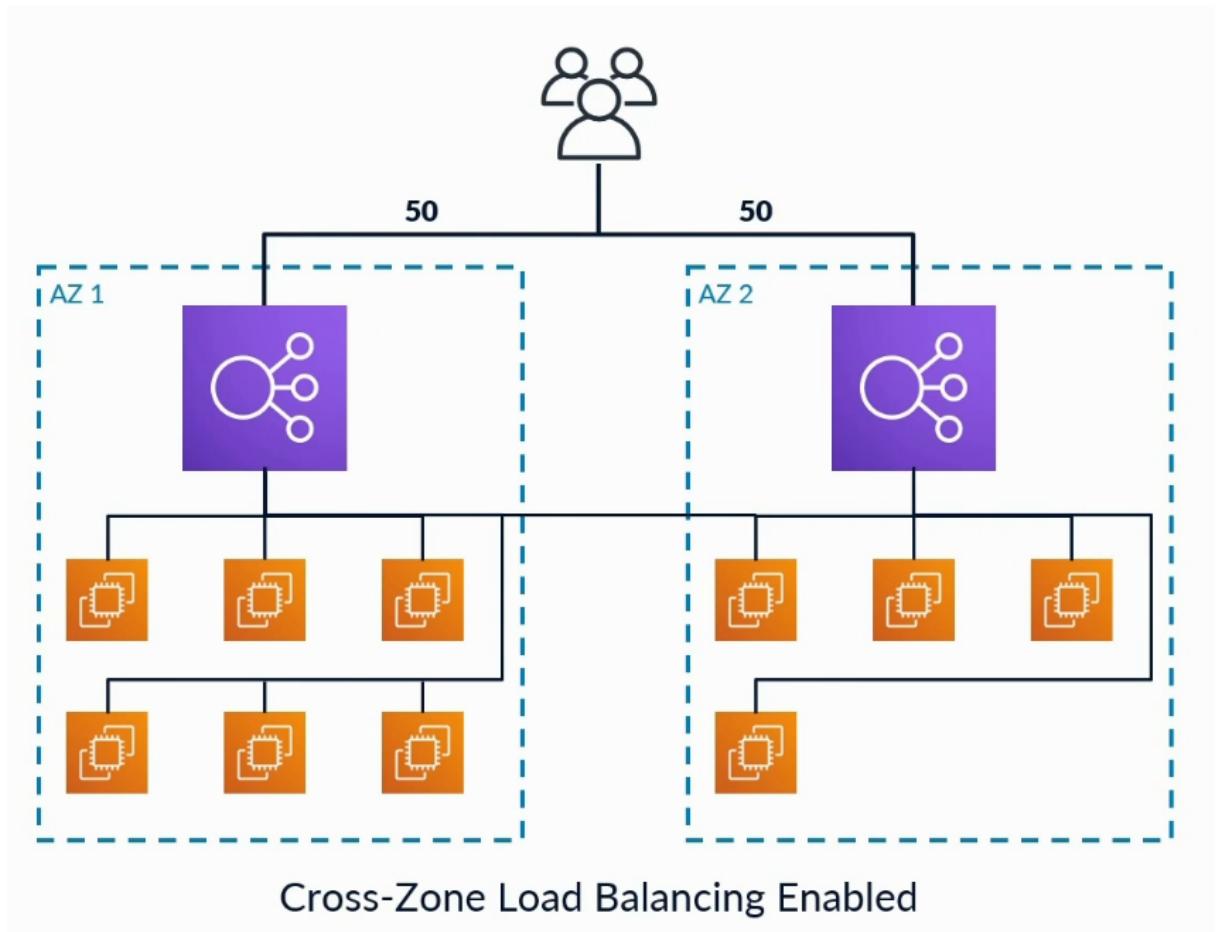
Used for applications that were built in the existing EC2 Classic environment  
Operates at both the connection and request level

- ELB components:
  1. **Listeners:** how to handle inbound requests
  2. **target groups:** which group of resources to route requests to
  3. **rules:** associated with listeners as to how requests get routed
  4. **health checks:** check status of resource in target group and redirect if failed
  5. **Internet-Facing ELB:** allow ELB to serve incoming requests from external outside internet
  6. **Internal ELB** handle and route requests from within internal components of VPC
  7. **node:** used by ELB to receive and distribute traffic to target groups, an AZ must contain a node to participate in successful load balancing
  8. **Cross-Zone Load Balancing:** for distribution between multiple AZs

- Without CZLB:



- With CZLB enabled:



- ELB can contain >1 listeners, each listener can contain >1 rules, each rule can contain >1 condition, all conditions in a rule equal a single action

**IF**

- ✓ Source IP is 10.0.1.0/24
- ✓ Http request method is PUT

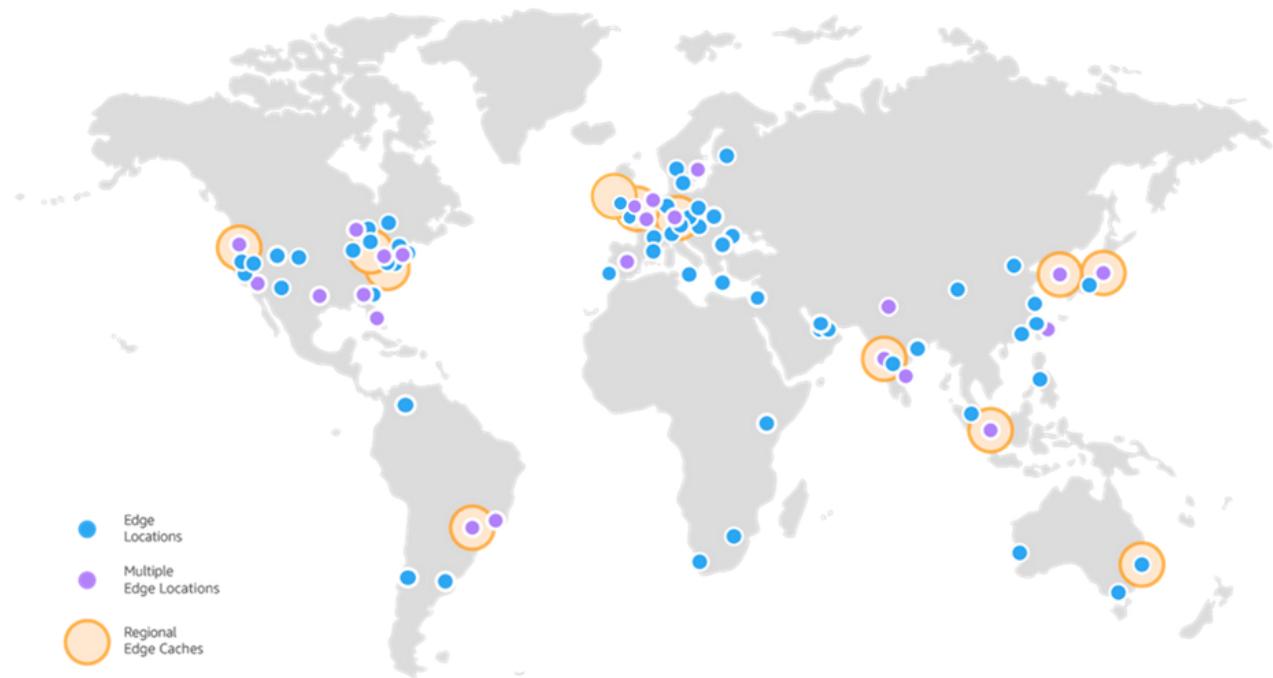
**THEN**

Forward to Group1

- The **IF** statement resembles the conditions
- The **THEN** statement acts as the action if all the conditions are met

## 5. Amazon CloudFront:

- Content delivery network service CDN
- CDN is a geographically distributed network of proxy servers and data centers that provide high availability and performance across multiple areas
- speeds up distribution of static and dynamic web content like html, js, images etc
- worldwide network of data centers called edge locations
- users are routed to most nearest/low latency edge location
- if requested content is not in nearest edge location, it is brought there and served



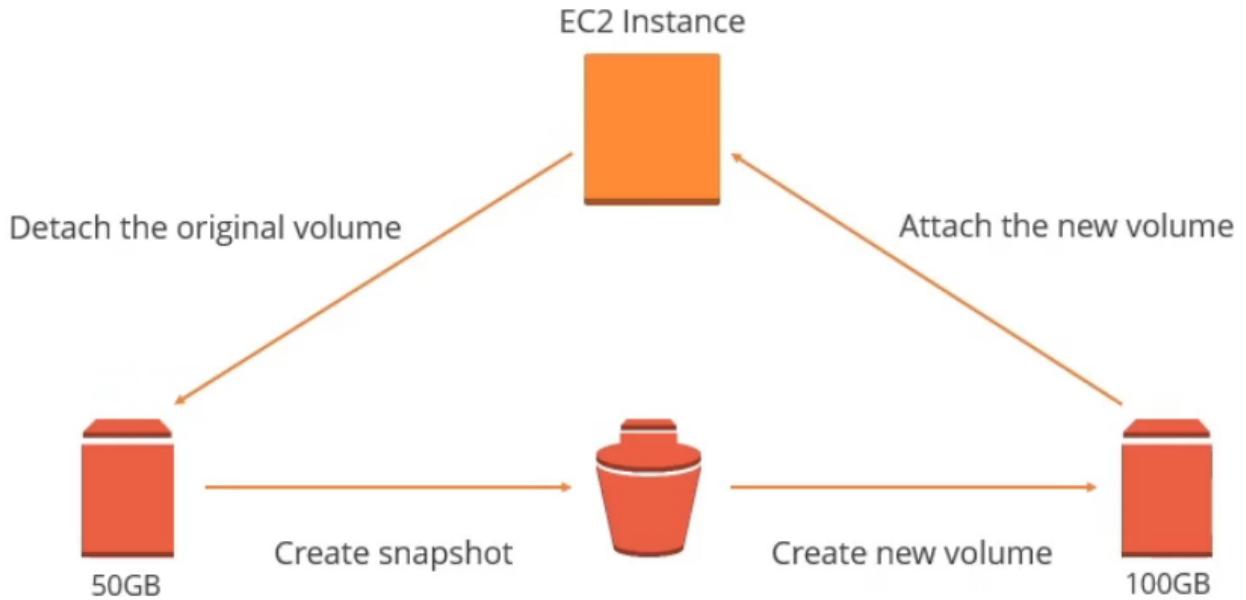
## Amazon Storage Service

### 1. Amazon Elastic Block Storage (EBS):

- disk volumes that are attached to EC2 instances

```
aws ec2 create-volume --size 80 --availability-zone us-east-1a --volume-type gp2
```

- AZ and region can be defined on creation
- EBS volumes can be upscaled as follows:



- Block Storage:**
    - traditional type
    - data written in raw blocks accessed by servers using Storage Area Network(SAN)
    - data is broken up into blocks and stored with a unique identifier separately in the most efficient places
    - provides low latency
    - provides flexibility which makes it ideal for containers
    - for high performance apps
    - highly redundant
    - boot volumes for VMwares
- File Storage:**
    - network attached storage, any server on the network can access data
    - highly scalable

- simultaneous read/write from multiple users
- **Storage Area Networks:**
  - block storages are deployed using SAN
  - it is a computer network that provides access to block data storage
  - server can attach to a SAN using a data network connection to access a block
  - can be connected with multiple servers or multiple storage arrays
- **Snapshots:**
  - an image of the EBS volume at a certain timestamp that can be stored as backup
  - can be used to create a duplicate/replicate EBS volume

## 2. Amazon Elastic File System:

- file storage for EC2 instances
- file system for content management systems and web serving apps
- store and serve info for a range of applications like websites, online publications
- can allow file system to be mounted across multiple regions and instances of EC2

### Elastic File System

- Replicated across AZs in a region
- Could be mounted to on premise server as well (over VPN or DirectConnect)
- Could be mounted to multiple EC2 at the same time
- No sizing to be done (unlike EBS)

## 3. Amazon Simple Storage Service (S3):

- cloud storage
- data is redundantly stored for durability

- S3 standards:

	STANDARD	STANDARD - IA	GLACIER	RRS
Durability	99.99999999%	99.99999999%	99.99999999%	99.99%
Availability	99.99%	99.99%	N/A	99.99%
Min Storage Duration	N/A	30 days	90 days	N/A
Retrieval Fee	N/A	Per GB retrieved	Per GB retrieved	N/A
First Byte Latency	milliseconds	milliseconds	4 hours	milliseconds

#### 4. Amazon Glacier:

- very cheap for storing backup data and archives for a long time(years)
- very high response time (4hrs)
- not for frequent access
- very durable and reliable over long time