About AWS, Why AWS

AWS Management Console Introduction

Launching First EC2 Instance (Virtual Machine) (Elastic Compute Cloud)

How to take remote of that Instance using various methods?

Benefits of Cloud Computing

Linux Operating System (Amazon Linux similar to RHEL)

4 Sessions to complete Linux OS

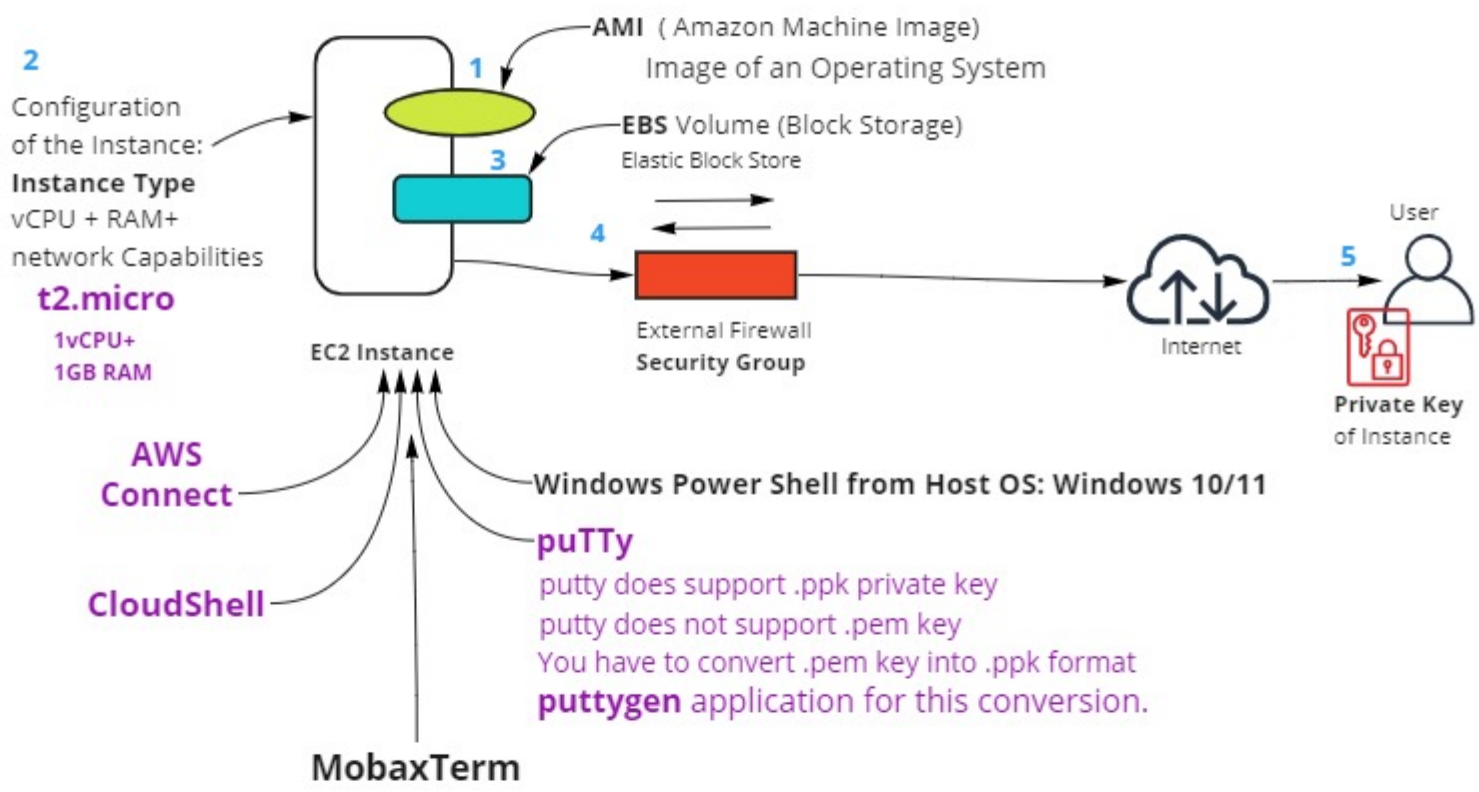
Virtualization
Virtual Machine
Containerization
Container Vs Virtual Machine

AWS Journey
AWS Global Infrastructure (Regions, AZs, Edge Locations, Local & Wavelength Zone)
AWS List of Services

for Introduction Launching First EC2 Instance (Virtual Machine) (Elastic Compute Cloud)

Launching First Virtual Machine in AWS

Methods to Connect EC2 Instance



AWS Free Tier Account = 750 Hrs/mo for EC2 instance

1 Instance = Entire one month is free for an instance

2 instances = 15 days

10 Instance = 75 Hrs

Instance States
- Start
- Stop
- Terminate

```
ssh -i AIntelKey04Dec.pem ec2-user@34.227.21.198
```

Command Keyname Username Public IP

```
ls -l AIntelKey04Dec.pem
sudo chmod 400 AIntelKey04Dec.pem
ls -l AIntelKey04Dec.pem
ssh -i AIntelKey04Dec.pem ec2-user@34.227.21.198
```

Launching Windows 2019 Base Server

t2.micro

1 vCPU, 1GB RAM

FREE

c4.xlarge

4 vCPU & 7.5 GB RAM

Chargeable

EC2 Service (Elastic Compute Cloud)

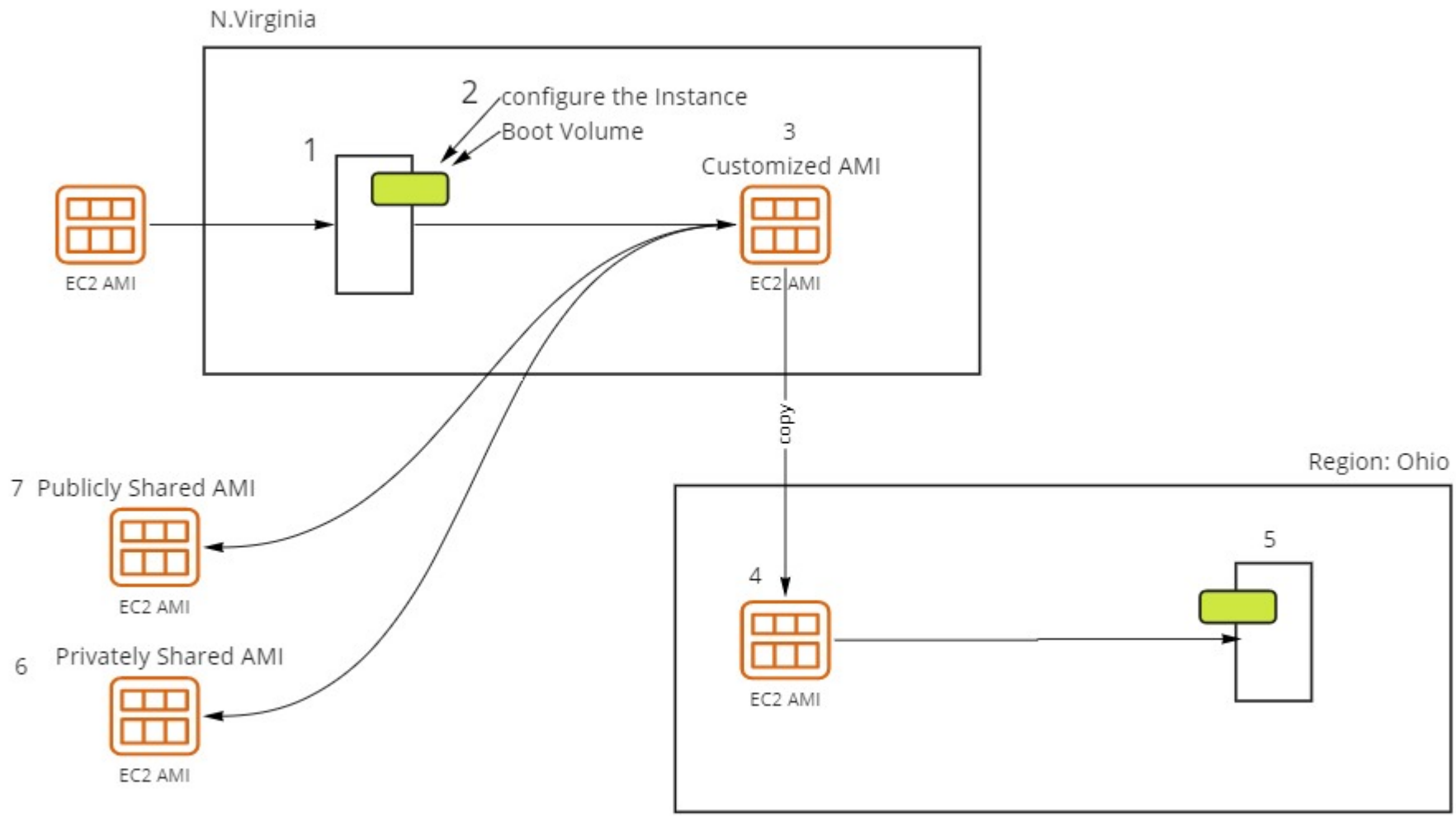
AMI (Amazon Machine Image)
Instance Type (Configuration of Instance)
EBS (Elastic Block Store): <-- Just Like a Hard Disk for the Instance
Key Pair
EIP (Elastic IP Address) Static Fixed Public IP Address
Bootstrapping

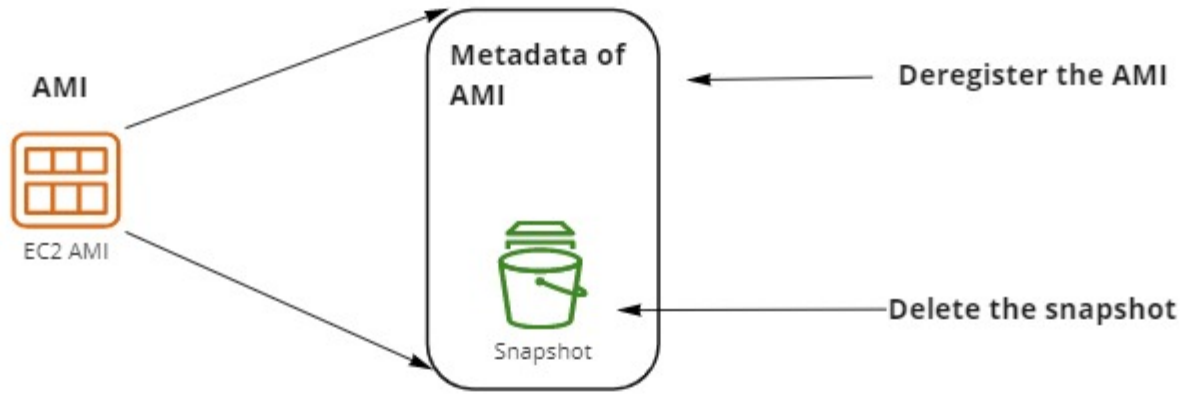
Security Group (External Firewall)
Snapshot
Bastion Host (Jump Server)

AMI (Amazon Machine Image)

AMI Contains: Operating System+Configuration of OS+Applications+User Data

LAB





Instance Type (Configuration of an Instance)

vCPU, RAM, and Network Capabilities

t2.micro free for AWS FREE Tier A/C

1vCPU & 1GB RAM

Xeon Class of Intel Microprocessors/Server Architecture

General Purpose

Compute Optimized

Memory Optimized

Accelerated Computing

Storage Optimized

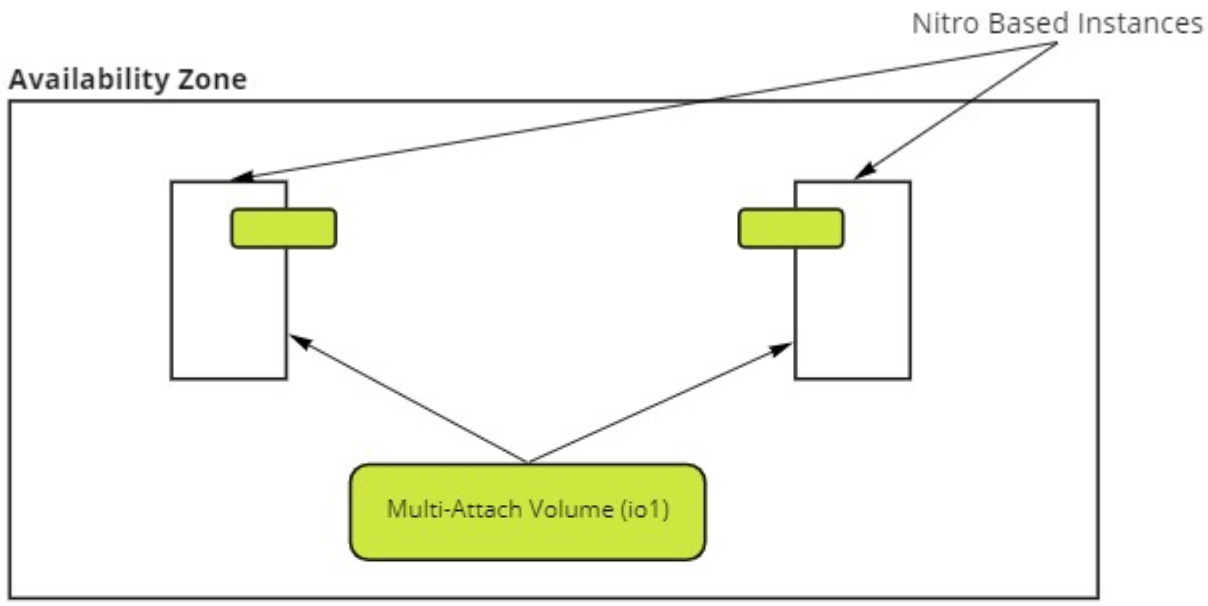
EBS (Elastic Block Storage)

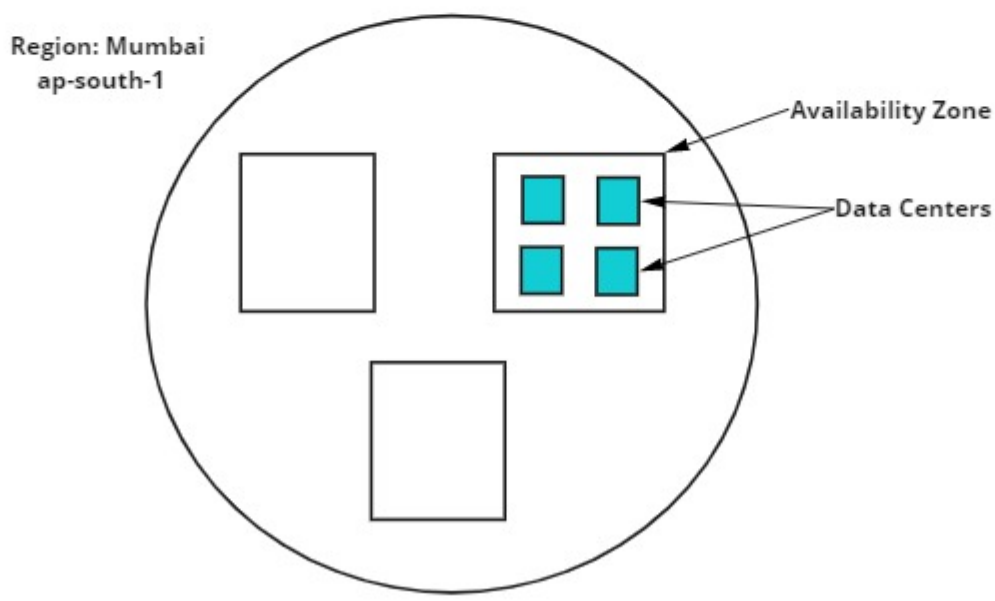
SSDs and HDDs

These EBS Volumes behaves like Hard disks in your VMs(Instances)

Amazon EBS Volume Types

	SSD					HDD		Previous Generation
	General Purpose SSD		Provisioned IOPS SSD			Throughput Optimized HDD	Cold HDD	Magnetic
Volume type	gp3	gp2	io2 Block Express ‡	io2	io1	st1	sc1	standard
Durability	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)	99.999% durability (0.001% annual failure rate)		99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)	
Use cases	Low-latency interactive apps Development and test environments		Workloads that require sub-millisecond latency, and sustained IOPS performance or more than 64,000 IOPS or 1,000 MiB/s of throughput	Workloads that require sustained IOPS performance or more than 16,000 IOPS I/O-intensive database workloads		Big Data Data Warehouse Log Processing	Throughput-oriented storage for data that is infrequently accessed Scenarios where the lowest storage cost is important	Workloads where data is infrequently accessed
Volume size	1 GiB - 16 TiB		4 GiB - 64 TiB	4 GiB - 16 TiB		125 GiB - 16 TiB	125 GiB - 16 TiB	1 GiB-1 TiB
Max IOPS per volume (16 KiB I/O)	3 IOPs/GB 16,000		2,56,000	Up to 50 IOPs per GB 64,000 †		500	250	40–200
Max throughput per volume	1,000 MiB/s	250 MiB/s	4,000 MiB/s	1,000 MiB/s †		500 MiB/s	250 MiB/s	40–90 MiB/s
Amazon EBS Multi-attach	Not supported		Not supported	Supported		Not supported	Not supported	Not supported
Boot volume	Supported					Not supported	Not supported	Supported

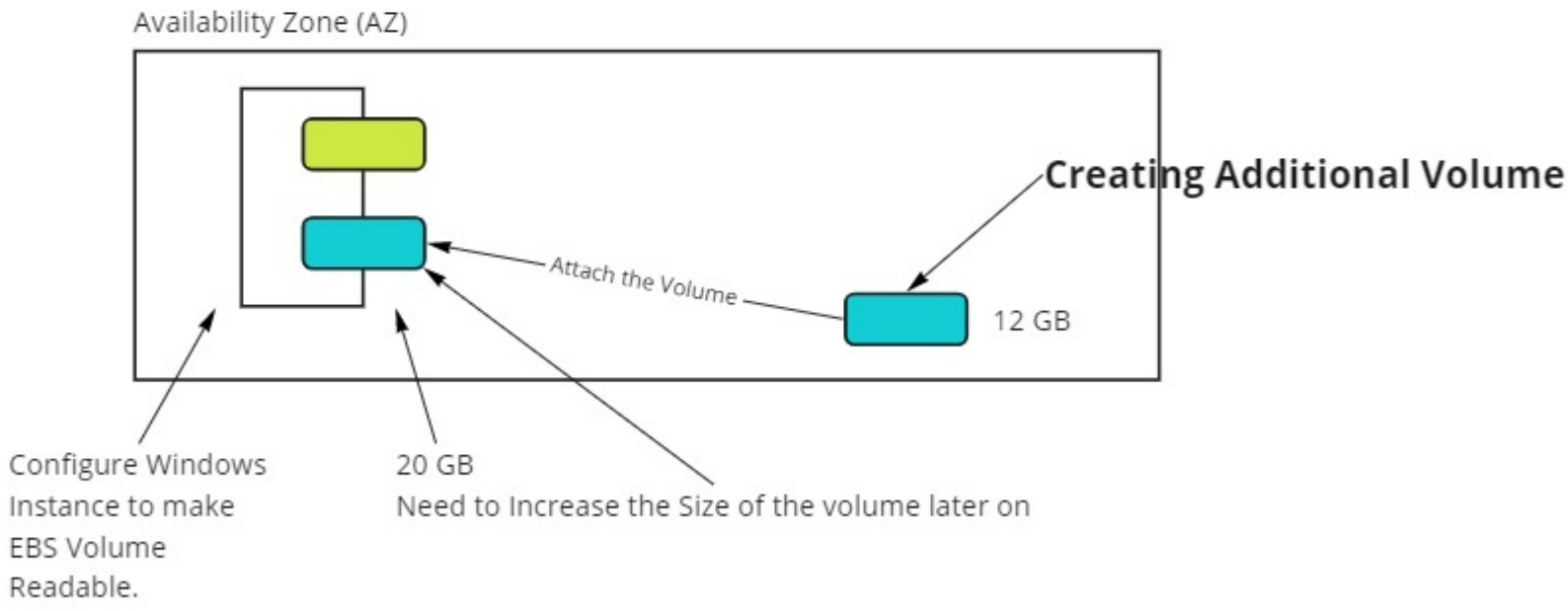




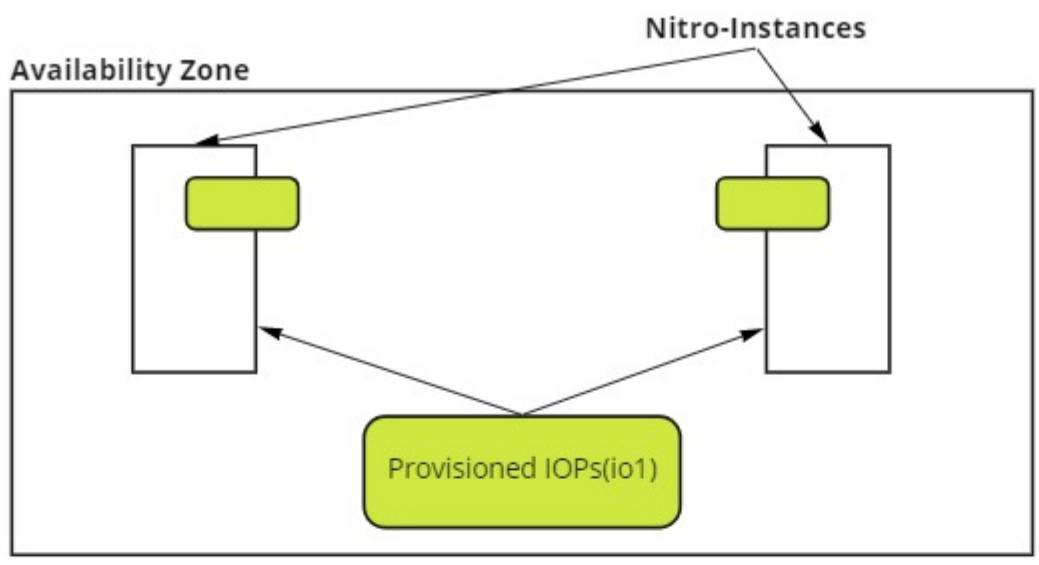
The following virtualized instances are built on the Nitro System:

- **General purpose**: M5, M5a, M5ad, M5d, M5dn, M5n, M5zn, M6a, M6g, M6gd, M6i, M6id, T3, T3a, T4g
- **Compute optimized**: C5, C5a, C5ad, C5d, C5n, C6a, C6g, C6gd, C6gn, C6i, C6id , Hpc6a
- **Memory optimized**: R5, R5a, R5ad, R5b, R5d, R5dn, R5n, R6a, R6g,R6gd, R6i, R6id, `u-3tb1.56xlarge`, `u-6tb1.56xlarge`, `u-6tb1.112xlarge`, `u-9tb1.112xlarge`, `u-12tb1.112xlarge`, X2gd, X2idn, X2iedn, X2iezn, z1d
- **Storage optimized**: D3, D3en, I3en, I4i , Im4gn , Is4gen
- **Accelerated computing**: DL1, G4, G4ad, G5, G5g, Inf1, `p3dn.24xlarge`, P4 , VT1

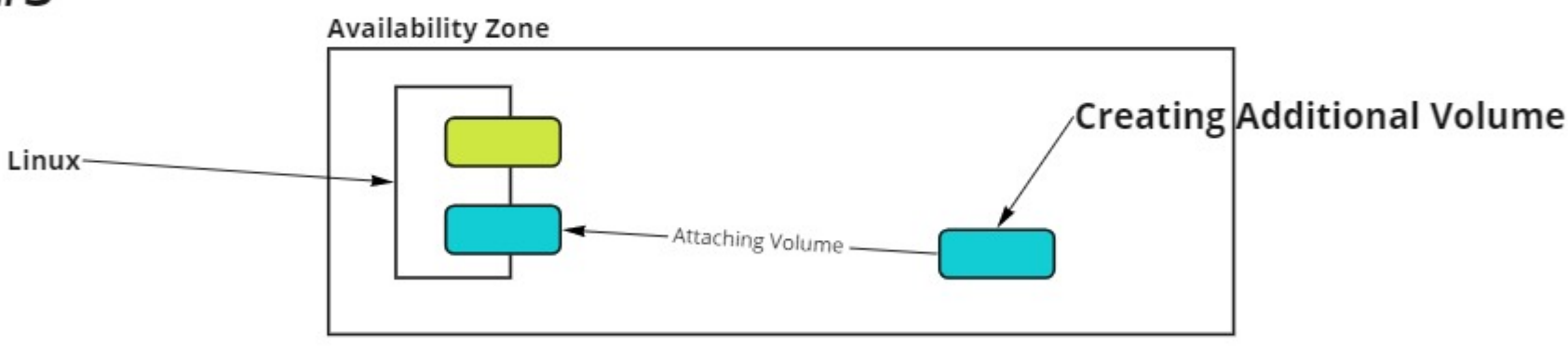
LAB #1 Attaching an additional EBS Volume with Windows Server Instance



LAB#2



LAB #3




```
[root@ip-172-31-2-20 dd1]# history
    1  lsblk
    2  fdisk /dev/xvdf
    3  lsblk
    4  mkfs.xfs /dev/xvdf1
    5  mkdir /mnt/dd1
    6  mount /dev/xvdf1 /mnt/dd1
    7  lsblk
    8  cd /mnt/dd1
    9  ll
```

EC2 Key Pair:

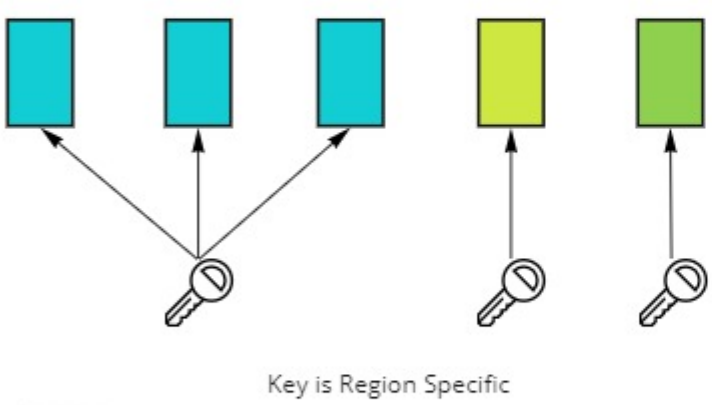
Public Key Cryptography

Public Key

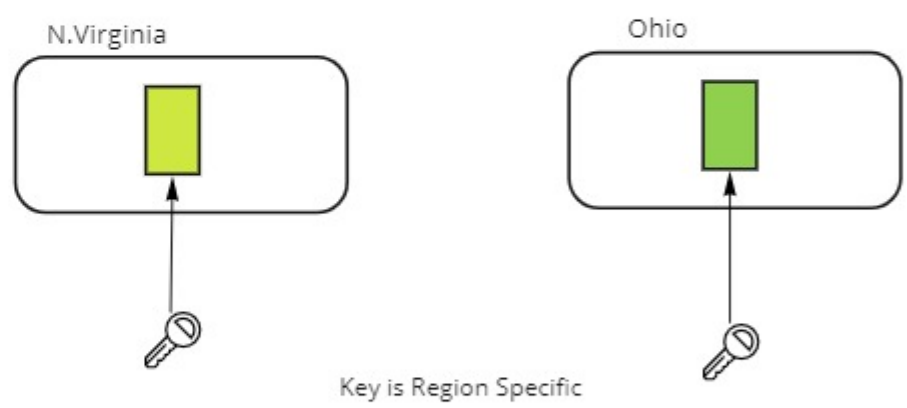
It is used to encrypt the information, and it belongs to AWS

Private Key

It is used to decrypt the encrypted information, and we download it.



Key is Region Specific



Key is Region Specific

You can also use third party tools to create and use your own key pairs

AWS generated key used **2048** bit and ***SSH-2 RSA*** Algorithm

AWS account can have upto 5000 keys pairs per region

finds keys both in .pem or .ppk format

.pem is used for OpenSSH tools

.ppk is used to **puTTy**

Important: you can also launch an Instance without a key pair

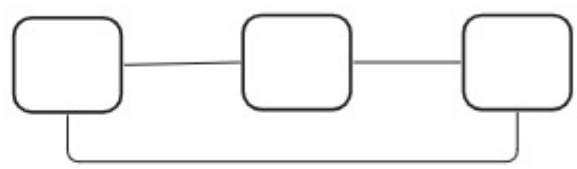
Public Instances

Public Subnet

Machines are accessible from the Internet,
In such cases, it is recommended to use keys with the instance.

Private Instances

Private Subnets



These machines are not reachable from the Internet, and we are transferring data very frequently between them.

Elastic IP Address

EIP is a Fixed or static public IPv4 address

Chargeable but in AWS Free Tier A/c One EIP is Free

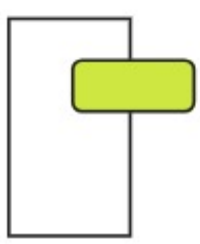
Max 5 EIPs can be allocated to your AWS Account Region

EIPs are needed for DNS (Domain Name System) reverse entry or EIP is required for NAT(Network address translation) Gateway
EIPs are also needed to Global Accelerators
if you need fixed or non changeable IP address in that case EIP will be used.

By default VMs (instances) will have dynamic public IP address

To provide public Static IP (EIP) is a two step process.

1. Allocate an EIP to your AWS Account
2. Associate the EIP with your EC2 Instance or Global Accelerator or NAT G/W



Dynamic IP
3.234.255.3

44.203.102.97

Procedure/steps to dissassociate the EIP from the Instance

1. Dissassociate the EIP from the Instance
2. Release the EIP from you AWS Account.

Bootstrapping

Bootstrapping is a method to configure Instance at launch time using script.

LAB: Configure simple Linux Apache Server using Shell Script

```
#!/bin/bash
sudo su -
yum install httpd -y
systemctl start httpd
systemctl enable httpd
cd /var/www/html
echo "This is my Bootstrapp Server" > index.html
```

Next Weekend
Security Group (External Firewall)
Snapshot
Bastion Host (Jump Server)
EFS Network File System (NFS)
ELB (Elastic Load Balancer)

Security Group: External Firewall to be attached with EC2 Instance

Its a bunch of firewall rules
You would write rules in Security Group to allow or restrict traffic

You can connect multiple security groups with one instance.

Max 5 security groups can be connected with one instance

Max 2500 security groups can be created per region/VPC

You find rules written in Security Group are **permissive** in nature, it means you cannot create rules that deny access.
Security Group are **stateful in nature**. In stateful, when you send a request from your instance, acknowledgement traffic for that request is allowed.

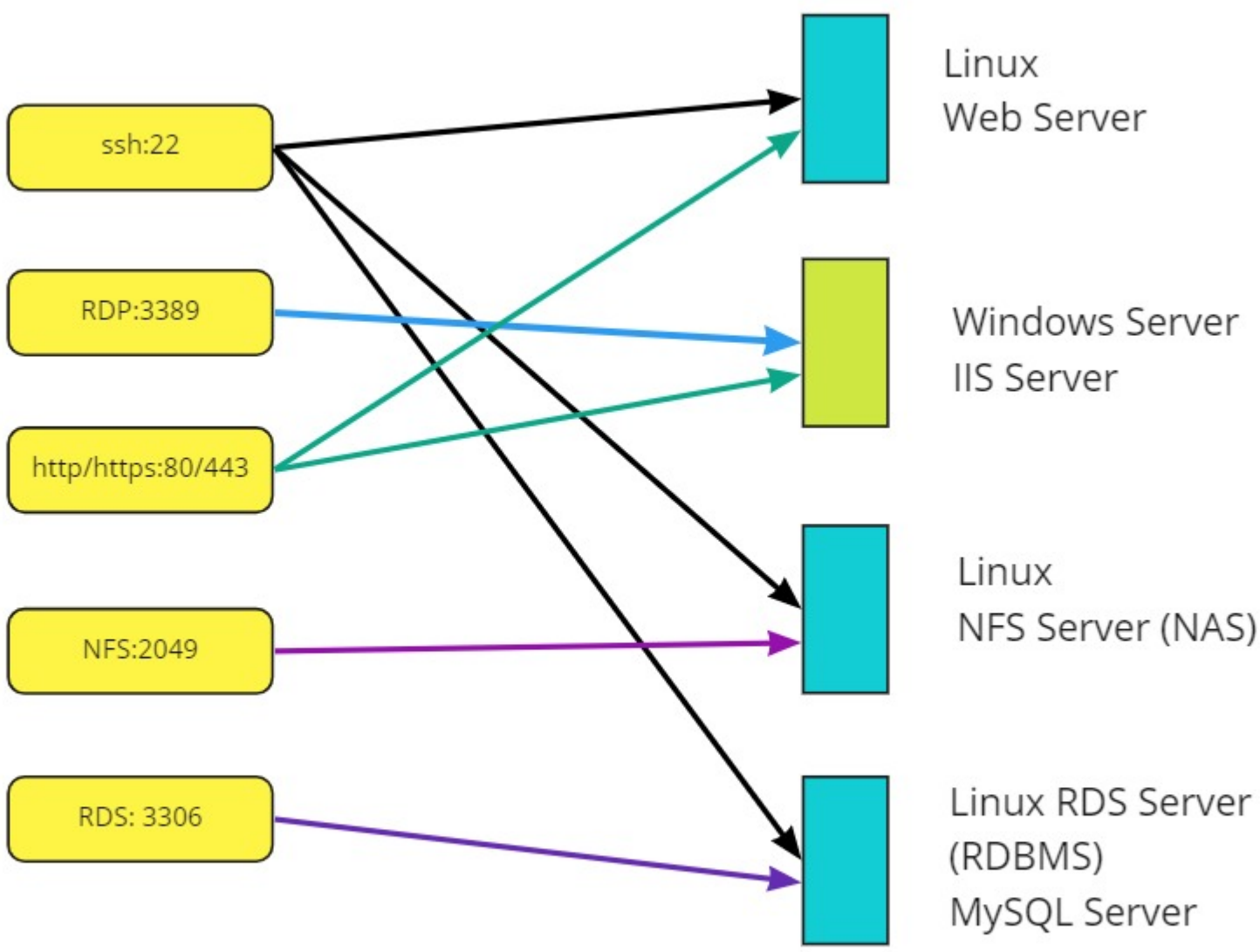
Security Group Sections

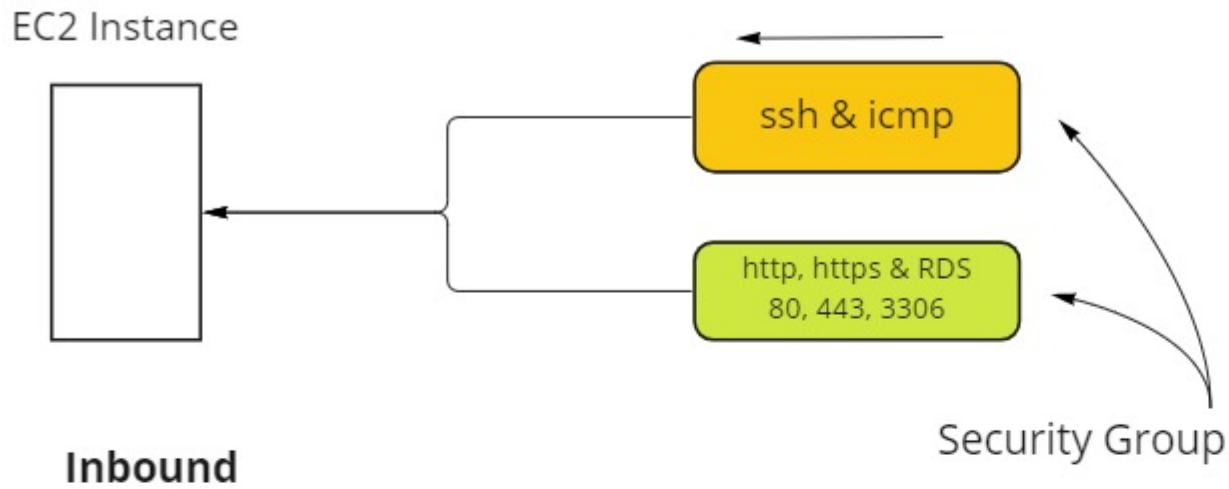
Inbound: to filter incoming traffic

Max 60 rules can be written in Inbound
It does filter the traffic on the basis of **Protocol**, **Port Number** and **IP address/NID**
By default, in inbound, ***all traffic is denied***

Outbound: to filter outgoing traffic

Max 60 rules can be written in Outbound
It does filter the traffic on the basis of Protocol, Port Number and IP address/NID
By Default, in outbound, **all traffic is allowed**





Inbound

In inbound by default all ports are closed.

need to allow, ssh, http, https, rds, icmp

ssh & icmp

http, https and RDS

ssh ==> secure shell (used to take remote of Linux Instance)

Under TCP/IP we have the following protocols

TCP

UDP

ICMP

Logical Range of TCP/IP portocol 1-65535

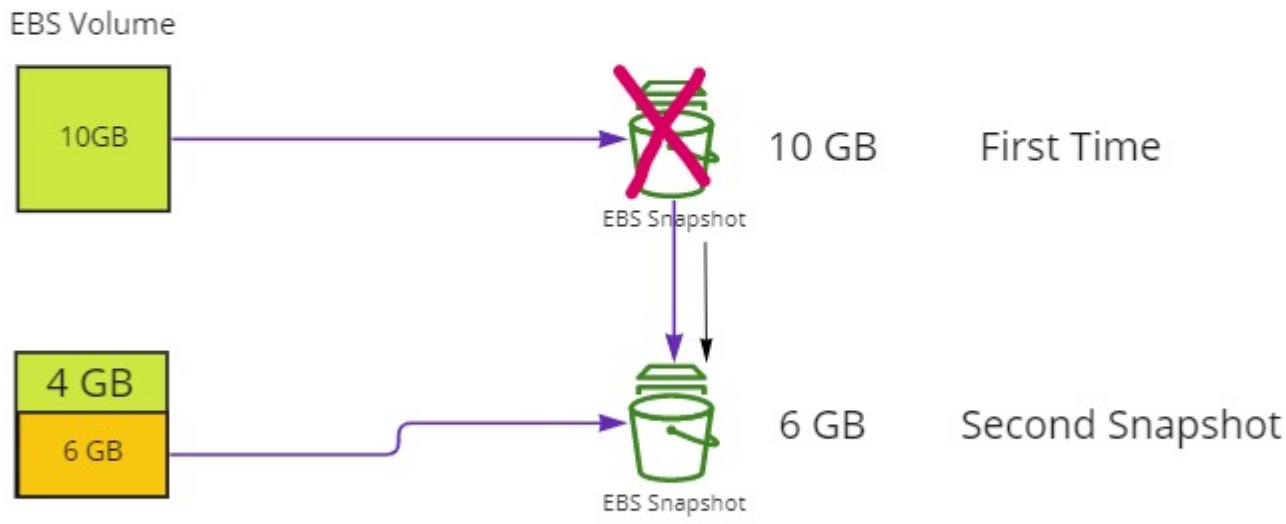
Snapshot

Snapshot is a backup and recovery method for EBS volume

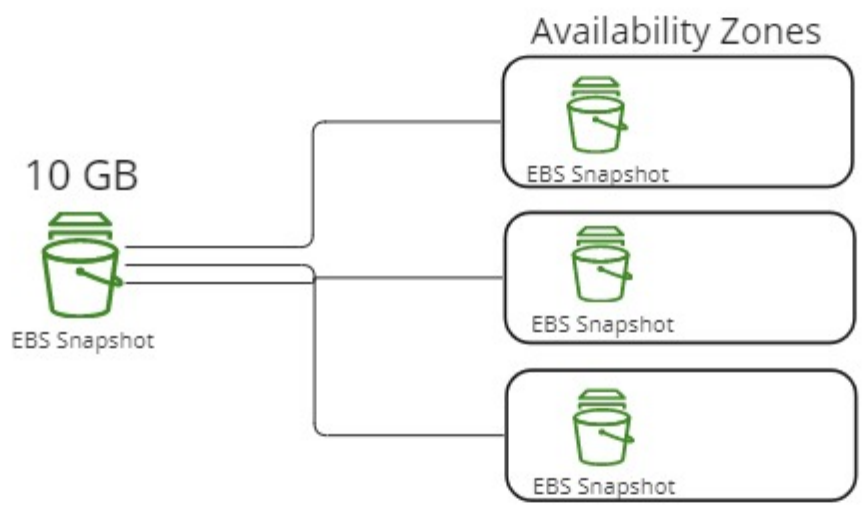
The snapshot is a ***point in time*** backup of an EBS Volume

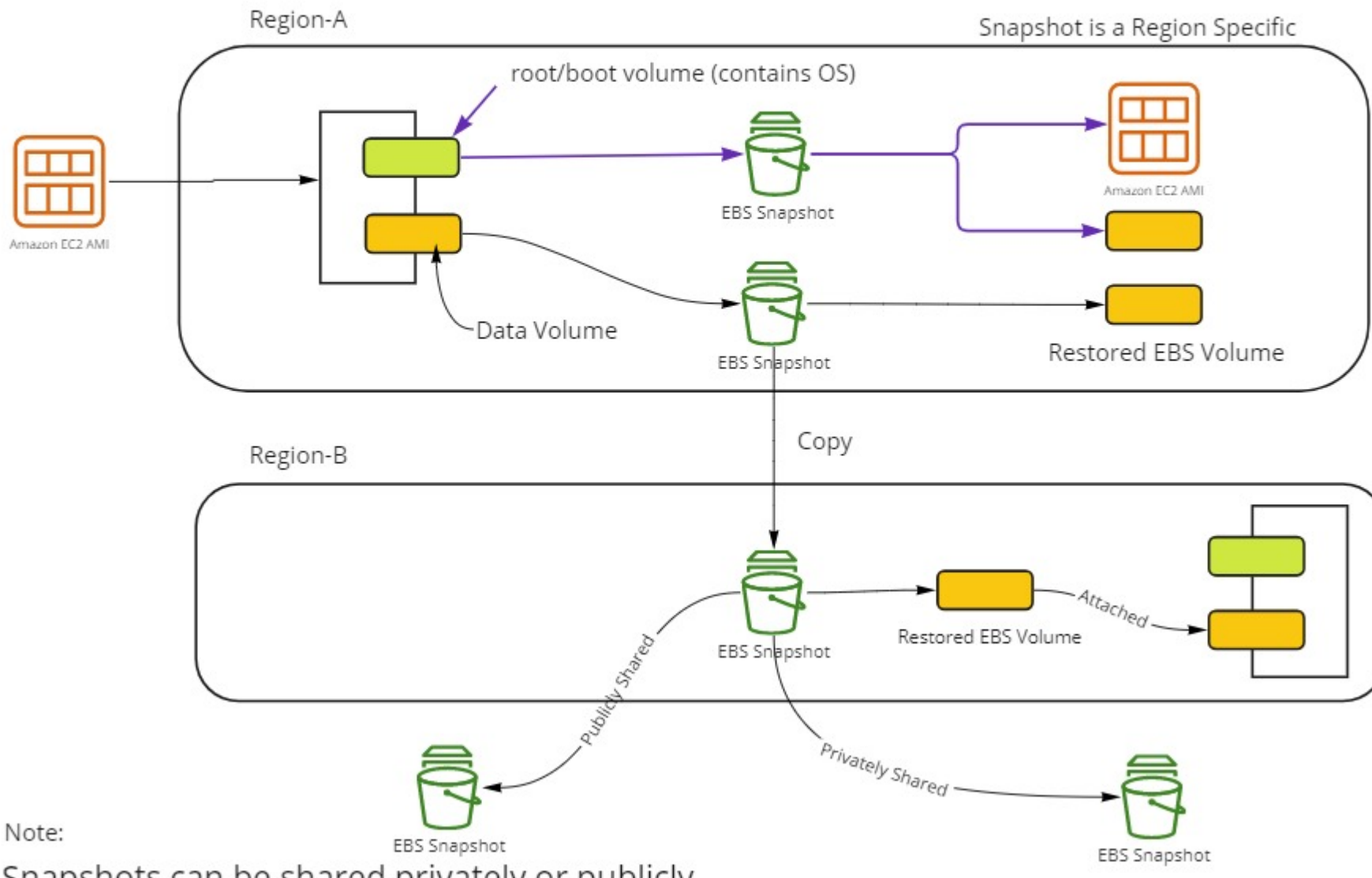
EBS Snapshots are incremental and cost effective solution

if multiple backups are taken of a volume, they are incremental



Snapshots are stored in ***S3 storage space***

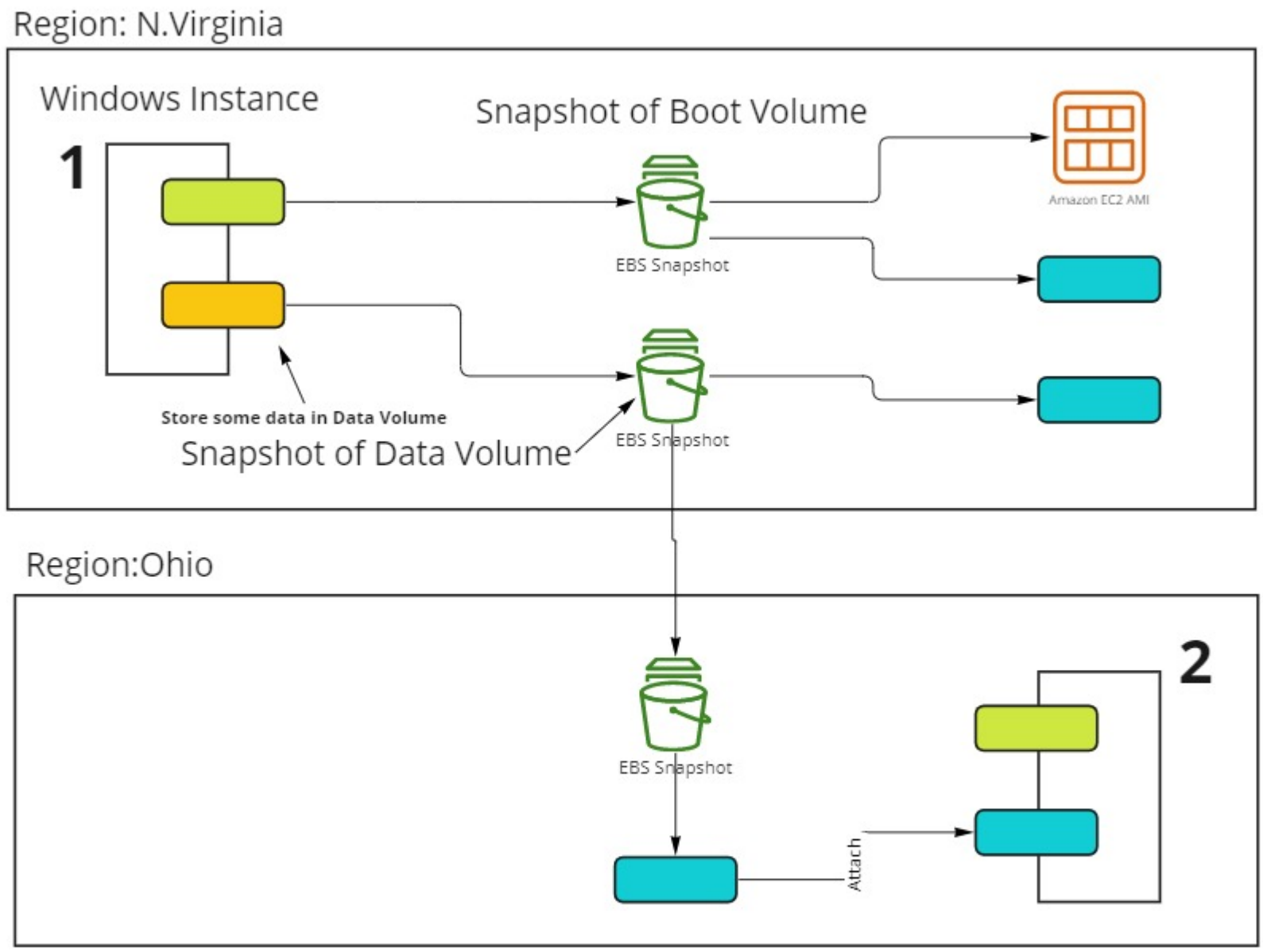


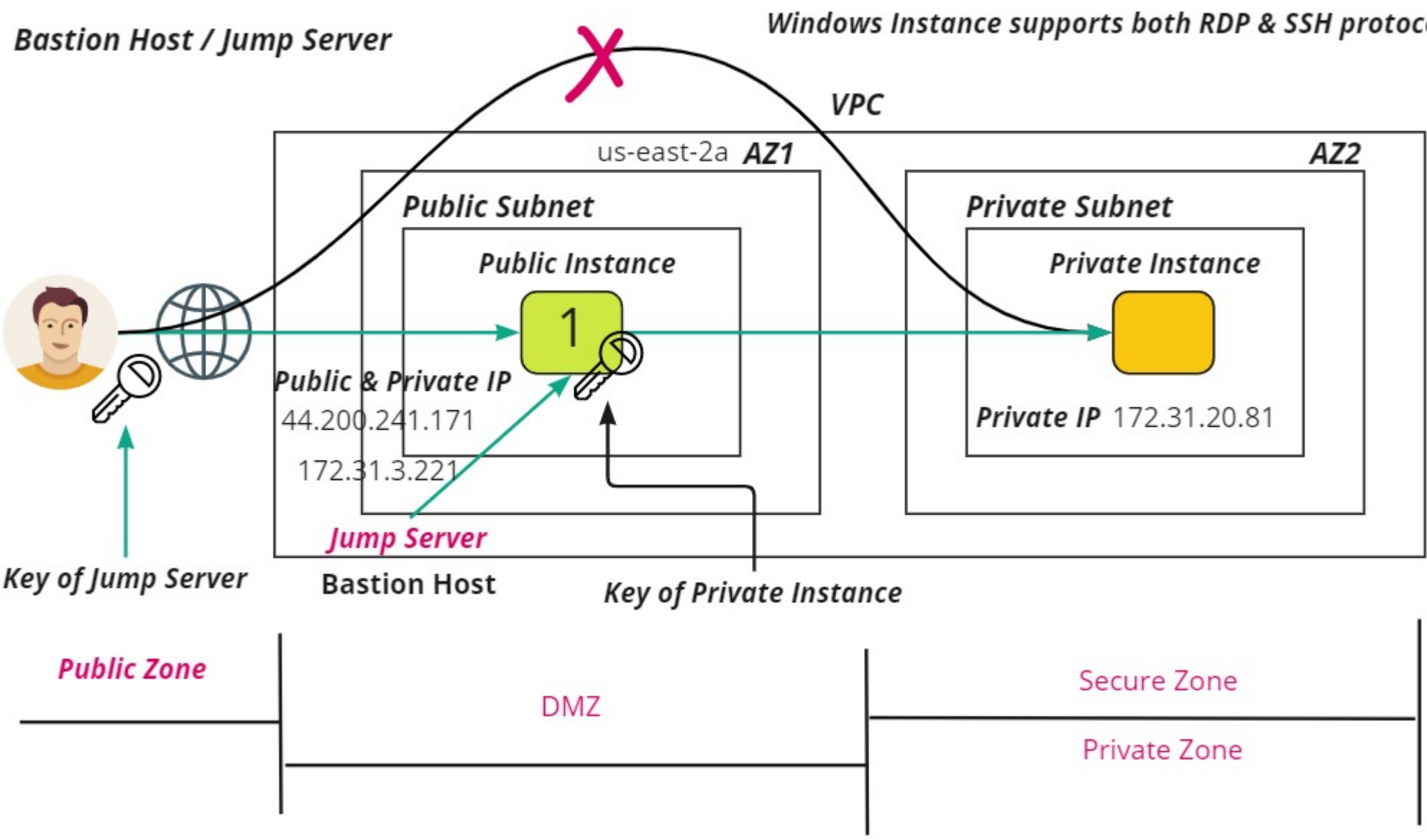


Note:

Snapshots can be shared privately or publicly

LAB:


Region: N.Virginia
Windows Instance
1
Snapshot of Boot Volume
EBS Snapshot
Amazon EC2 AMI
Store some data in Data Volume
Snapshot of Data Volume
EBS Snapshot
Region:Ohio
EBS Snapshot
2
Attach


Bastion Host / Jump Server
Windows Instance supports both RDP & SSH protocol
VPC
us-east-2a AZ1
AZ2
Public Subnet
Public Instance
Private Subnet
Private Instance
1
Public & Private IP
44.200.241.171
172.31.3.221
Private IP 172.31.20.81
Jump Server
Key of Jump Server
Bastion Host
Key of Private Instance
Public Zone
DMZ
Secure Zone
Private Zone

Storages in AWS

Block Storage

EBS

(EBS can be used as a disk with Instance)

File System Storage

EFS

Elastic File System

(Its a common Storage for multiple Instances)

Object Storage

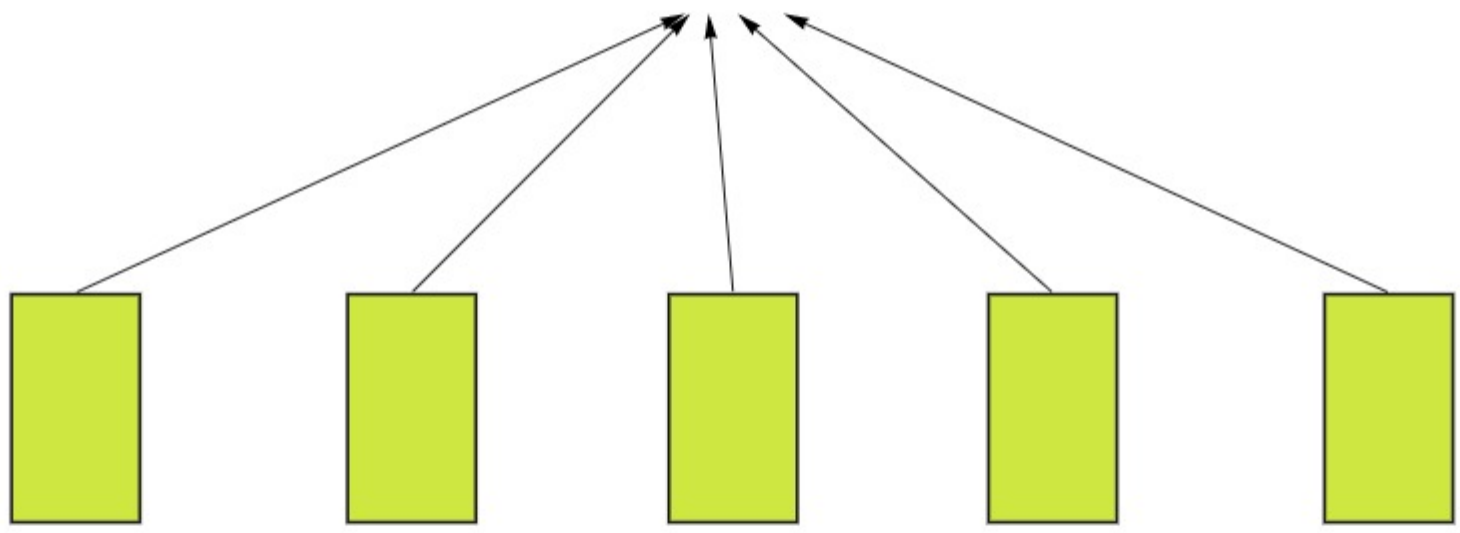
S3

Simple Storage Service

NAS (Network Attached Storage)

In Physical Infra

Expensive
Limited Space
Complex to Manage
Not Auto Scalable

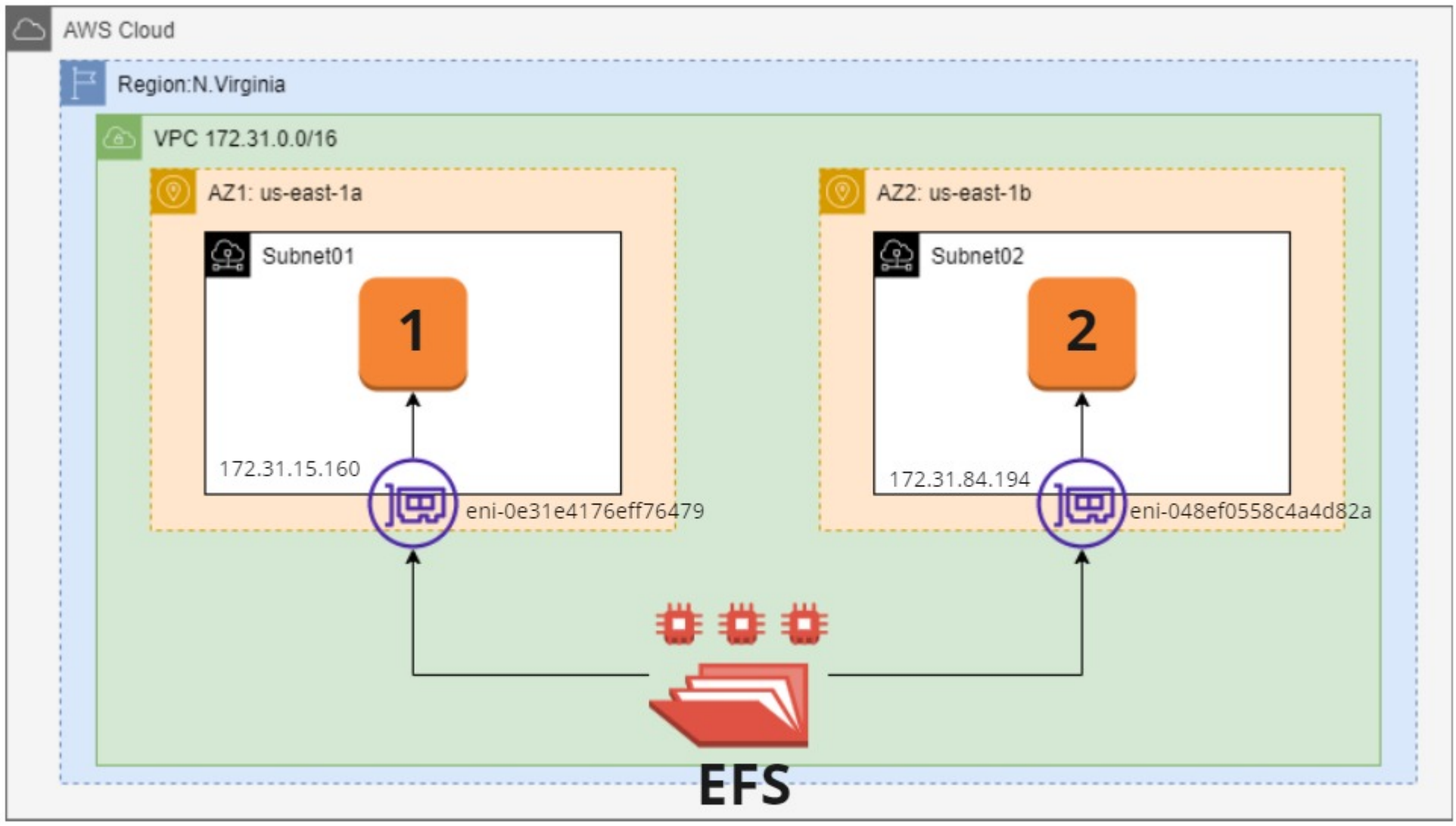


NFS (Storage on Physical Infra)

Expensive
Limited Space
Complex to Manage
Not Auto Scalable

EFS

You will pay only for used space
It can grow up to PetaBytes
Easy to Manage
It is highly scalable



Can't we use EFS with Windows OS:
in AWS, EFS is recommended for Linux machines only. EFS uses NFSv4

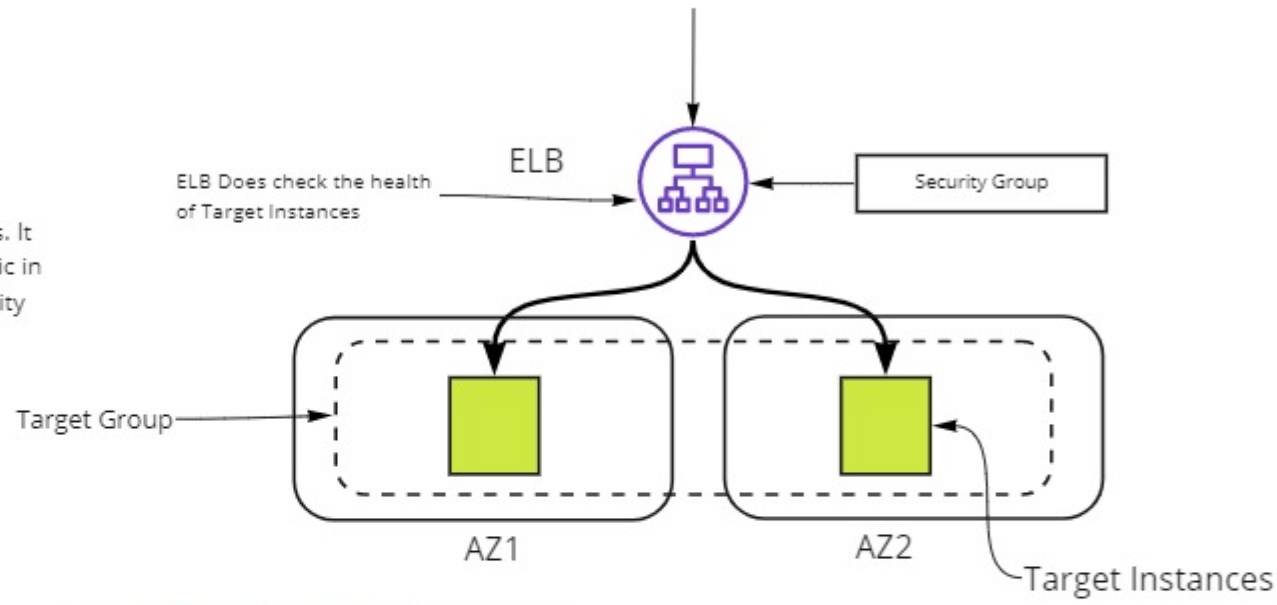
Common Storage for Windows Instances is FSx, FSx uses SMB protocol.
FSx does support to Active Directory of Windows Operating System.

Introduction with AWS ELB

Sanjay Sharma

Elastic Load Balancer

Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, Lambda functions, and virtual appliances. It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones.



Types of ELB in AWS

ALB
Application Load Balancer

80/443 with http/https
it works on Layer 7
Application Layer

NLB
Network Load Balancer

it does support all logical ports (1-65535), it works on layer 4, Transport Layer

~~Classic Load Balancer~~

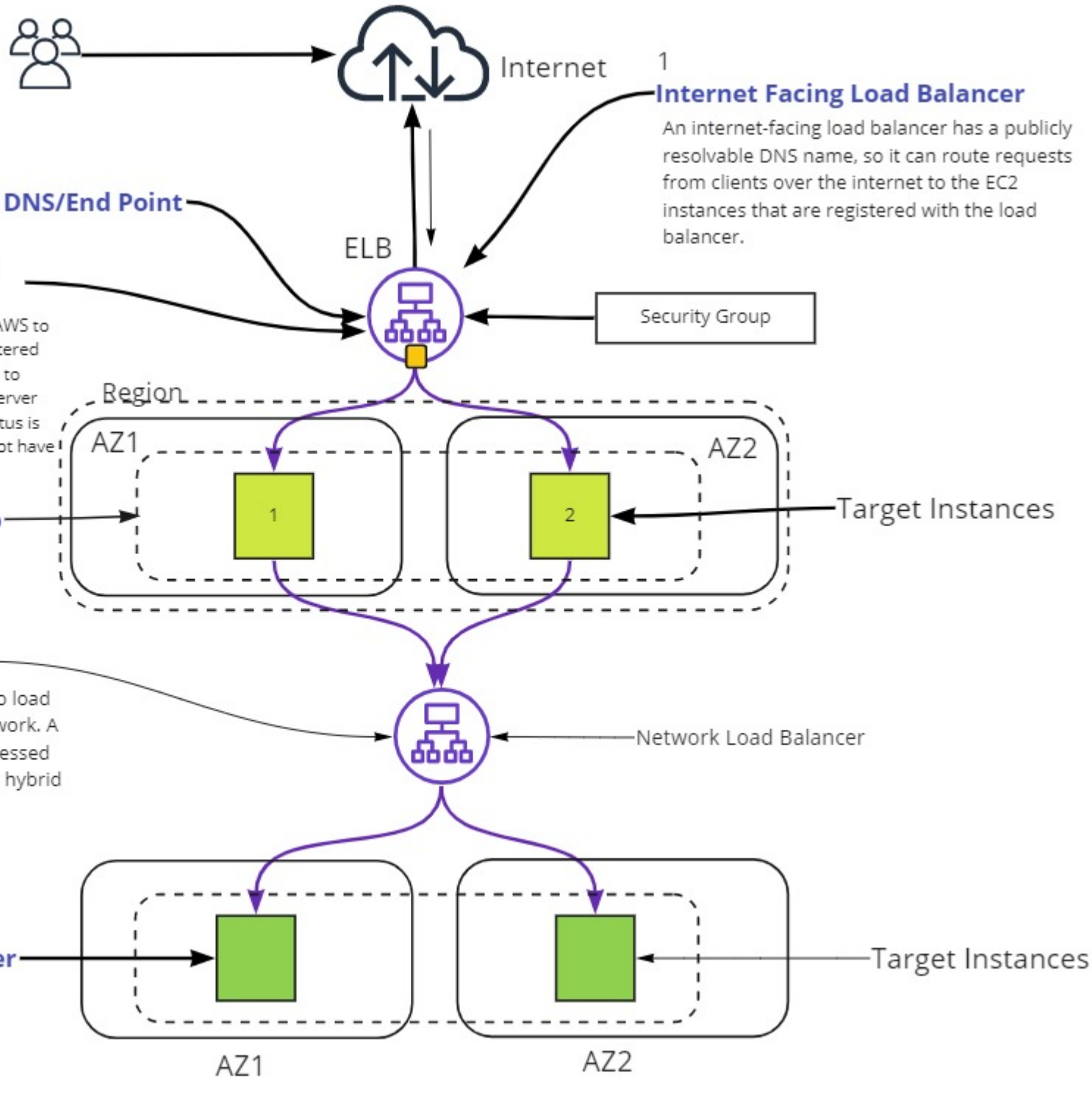
~~Pervious Generation Load Balancer~~
It does support both Layer4 and Layer 7

GWLB
Gateway Load Balancer

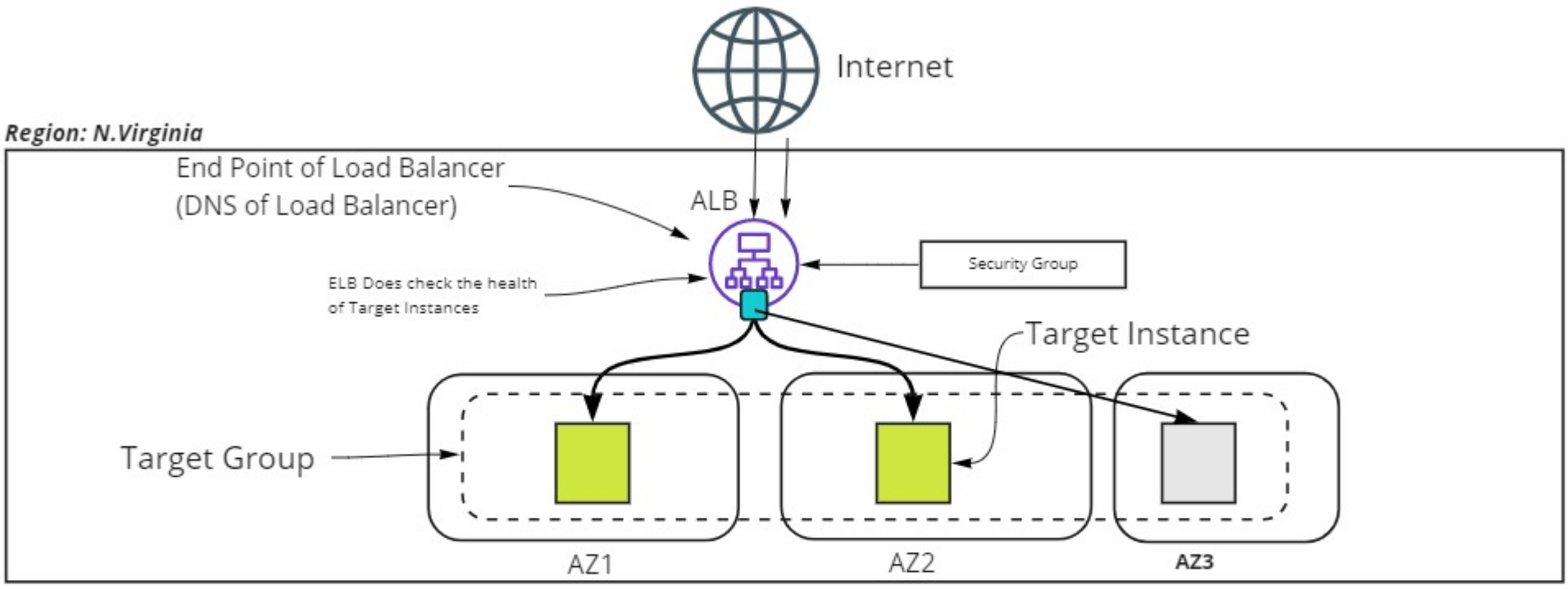
Gateway Load Balancer makes it easy to deploy, scale, and manage your third-party virtual appliances. It gives you one gateway for distributing traffic across multiple virtual appliances, while scaling them up, or down, based on demand.

Elastic Load Balancer

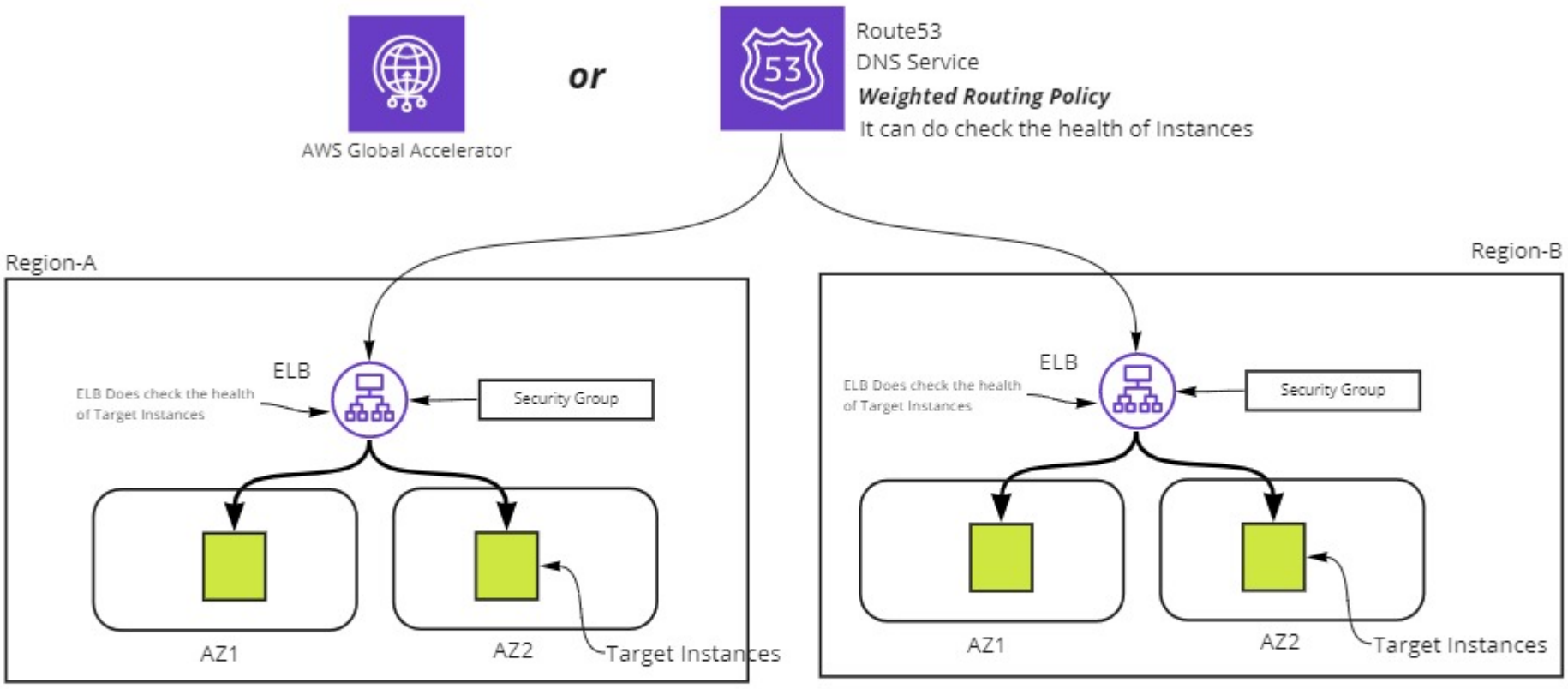
Type of ELB (Architectural Point of View)



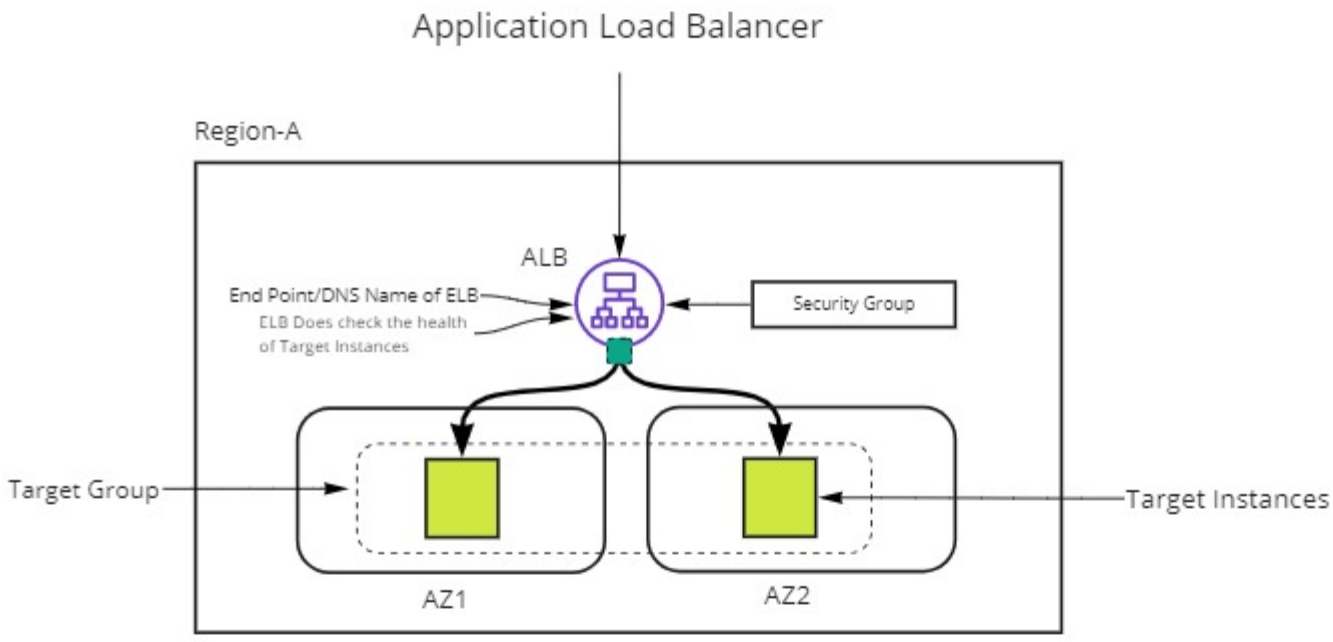
Application Load Balancer



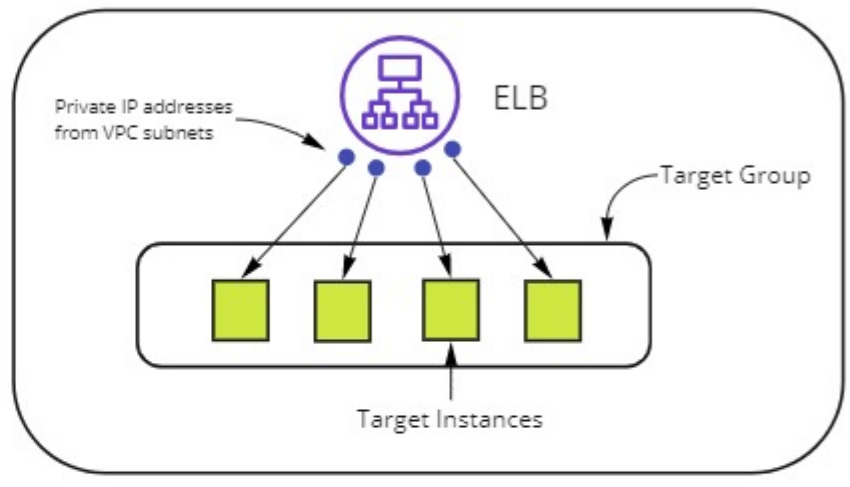
Cross Region Load Balancing



LAB:



Important Facts about ELB:



Traffic Distribution

Continuous Health Check

ELB can be Public or Private

Target Instances must be in diff AZs

Algorithm by default: ***Round Robin***

Every ELB can be accessed using its end-point (DNS name of ELB)

But if ELB is used with ***Global Accelerator***, GA will/can use EIPs

End Point of ELB and also be mapped with a Domain Name using Route53 to open website using a Domain Name.

ELB can also be integrated with Auto Scaling to manage traffic load at backend.

ELB is highly available and Scalable

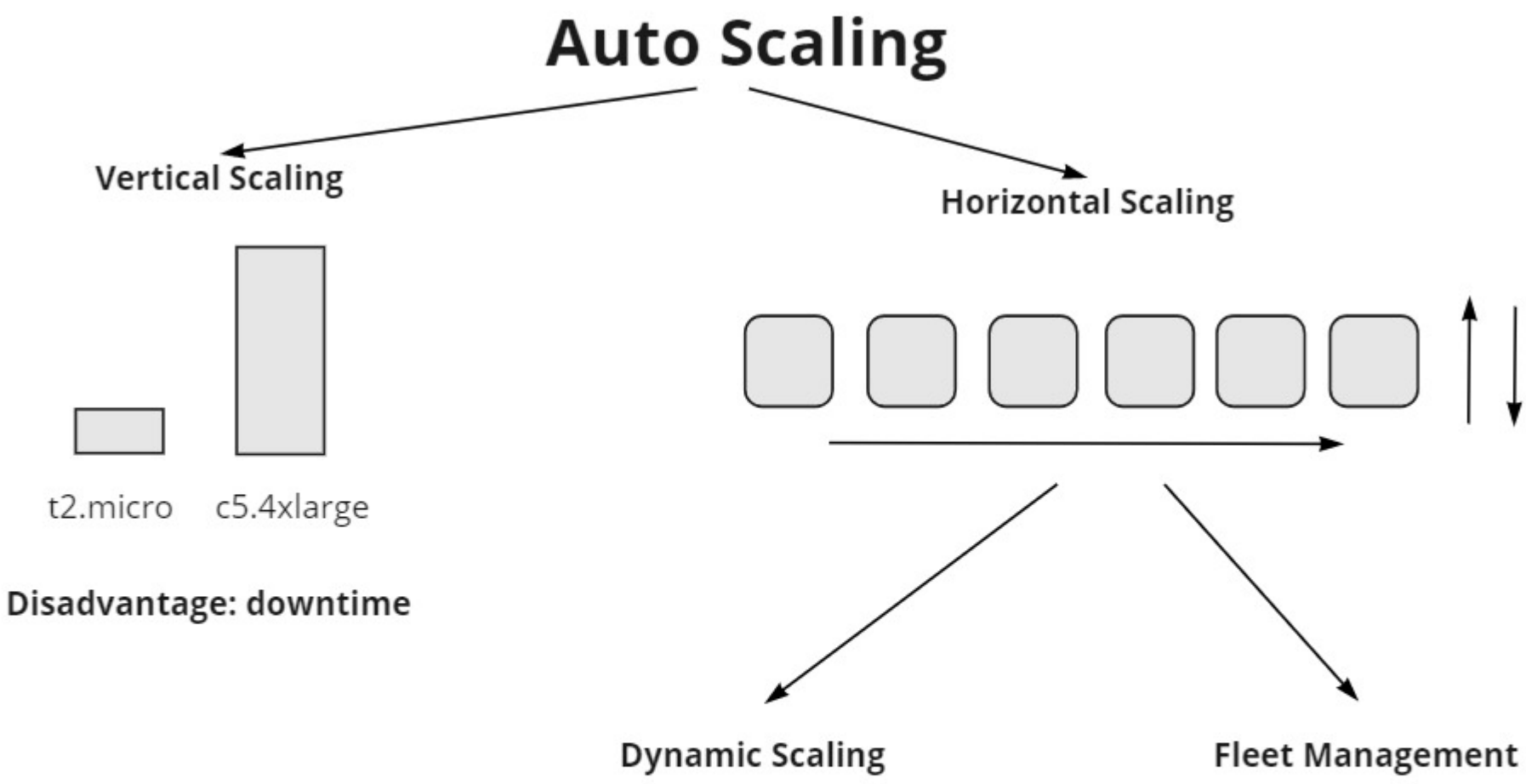
ELB will connect with Security Group in order to filter traffic

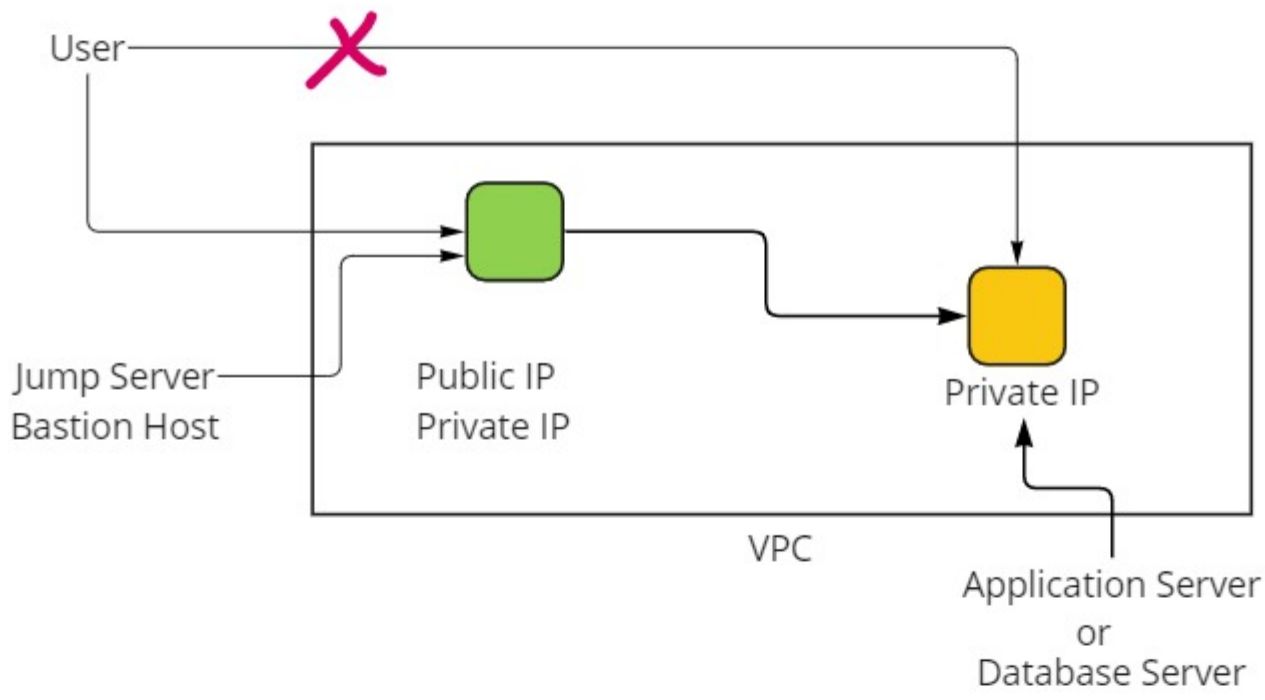
ELB can also be connected with WAF (***Web Application Firewall***)

Internal ELB uses Private IP address to distribute the load within the VPC

We can set ***SSL Certificate*** to allow/configure https traffic

Instances
IP addresses >> ENI
Lambda Functions
Application Load Balancer



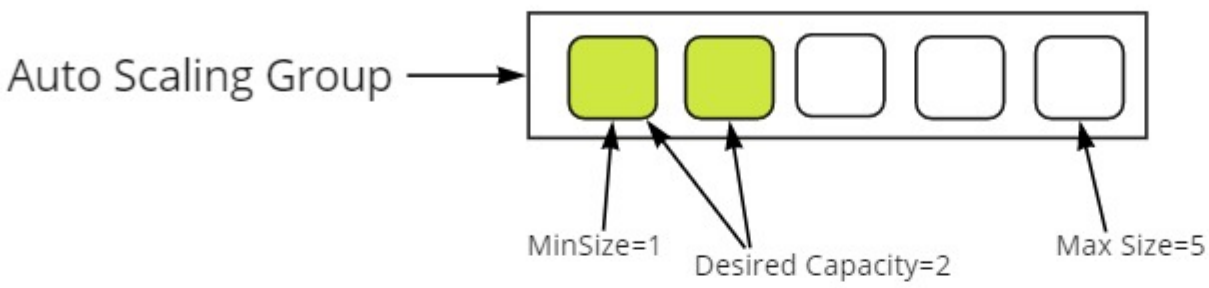


AWS Backup

Schedule backup of EFS, EBS, S3 etc.

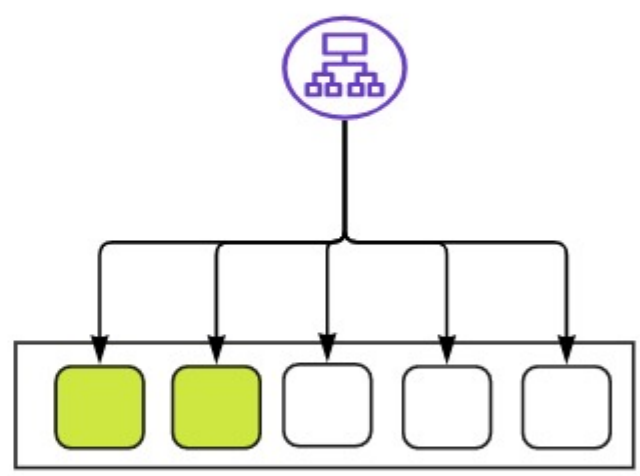
YOu can take immediate backup of resources

Auto Scaling



Benefits of Auto Scaling:

- Improved Fault Tolerance
- Improved Availability
- Improved Cost Management



Auto Scaling Configuration

Launch Configuration Auto Scaling Uses a launch configuration to launch instance

AMI ID, Instance Type, Key Pair, Security Group and EBS Volume Configuration

Auto Scaling Group ASG

in Auto Scaling Group you will declare Min, Max and Desired Capacity and Condition based on metrics to decide number of instances.

Metrics used in scaling policy: Target Tracking Scaling Policy

Metrics to use
- Avg CPU Utilization
- Network In
- Network Out
- Number of Hits on target Instances of Load Balancer

if Avg CPU Utilization of an instance > 35% then additional instance will join the ASG

Current Utilization of CPU = 99.0

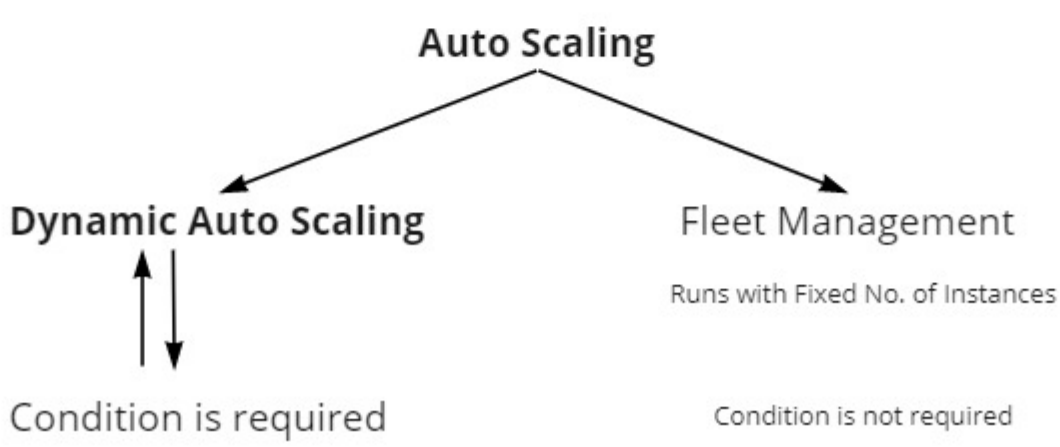
Expected Avg CPU Util > 35%

Number of Instances will join the group $\frac{\text{Current Utilization of CPU = 99.0}}{\text{Expected Avg CPU Util > 35\%}}$

99/35 = Approx 2.7/2.8

Number of Instances = 3

Current Number of Instance = 1 + 3 = 4



VPC (Virtual Private Cloud)

No. of classes in IP addressing
Subnetting
Network ID 192.168.10.0/24
You are asked to split this network into 4 equal Network
What slash notation you will take (Subnet Mask)
Calculate First and Last IP of each range

IP Addressing

IPv4
32 Bit Address
4 Octet
172.31.0.0

IPv6
128 Bit

Default IPv4 Table

Class	Range	Subnet Mask	Slash Notation
A	**0-127**	255.0.0.0	/8
B	128-191	255.255.0.0	/16
C	192-223	255.255.255.0	/24
D	224-239	N/A	N/A
E	240-255	N/A	N/A

Commercially → A, B, C
Multicasting → D
R&D → E

Class C

NID — HID
192.168.10.0
C
255.255.255.0
/24

Network ID → 192.168.10.0
Reserved IPs within AWS → 192.168.10.1, 192.168.10.2, 192.168.10.3
192.168.10.4
..
192.168.10.254
BroadCast IP → 192.168.10.255

for AWS
we are excluding 5 IPs from each range
Usable IPs= 256-5 =251

NID → **192.168.10**.0 /24

/24 → 255.255.255.0

11111111.11111111.11111111.**00000000**
Fixed

No of 1's in Forth Octet = x = 0
No. of 0's in Last Octet = y =8

Number of Networks = 2^x= 2^0 = 1
Number of Hosts/Network = 2^y=2^8= 256

Example of Class C

NID → **192.168.10**.0

/25 → 255.255.255.128

11111111.11111111.11111111.**10000000**

Fixed

No of 1's in Forth Octet = x = 1
No. of 0's in Last Octet = y =7

Number of Networks = 2^x= 2^1= **2**

Number of Hosts/Network = 2^y=2^7= 128

192.168.10.0/24

	/25	/25
NID	192.168.10.0	192.168.10.128
AWS Excluded	192.168.10.1	192.168.10.129
	192.168.10.2	192.168.10.130
	192.168.10.3	192.168.10.131
AWS Usable 128-5=123	192.168.10.4	192.168.10.132
	..	..
	..	..
	192.168.10.126	192.168.10.254
Broadcast IP	192.168.10.127	192.168.10.255

Subnet Table for Class C

Slash Notation	Subnet	Networks	Hosts/Net
/24	255.255.255.0	01	256
/25	255.255.255.128	02	128
/26	255.255.255.192	**04**	064
/27	255.255.255.224	08	032
/28	255.255.255.240	16	016

172.16.0.0 /16

172.16.0.0 /24

172.16.10.0 /20

CIDR

10.0.0.0 /8
10.0.0.0 /12
10.0.0.0 /16
10.0.0.0 /24

Virtual Private Cloud Example of Infrastructure as a Service (IaaS)

VPC is a logically isolated Network within a Region.

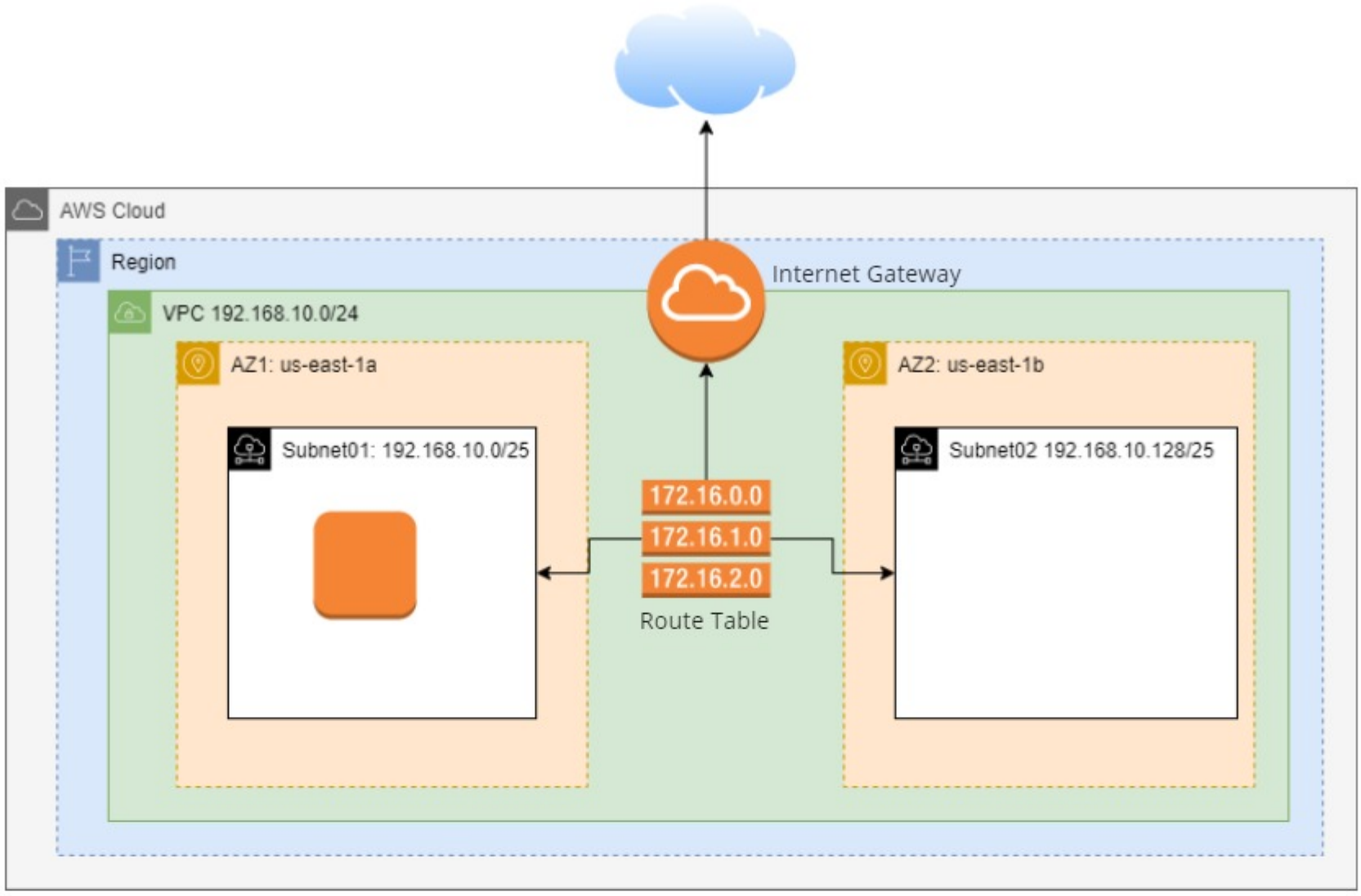
In VPCs we can only use Private IP address ranges

Private IP Address Ranges	
A	10.0.0.0 - 10.255.255.255
B	172.16.0.0 - 172.31.255.255
C	192.168.0.0 - 192.168.255.255

VPC's NID =CIDR= **192.168.10.0 /24**

Subnet01 = 192.168.10.0 /25
Subnet02 = 192.168.10.128 /25

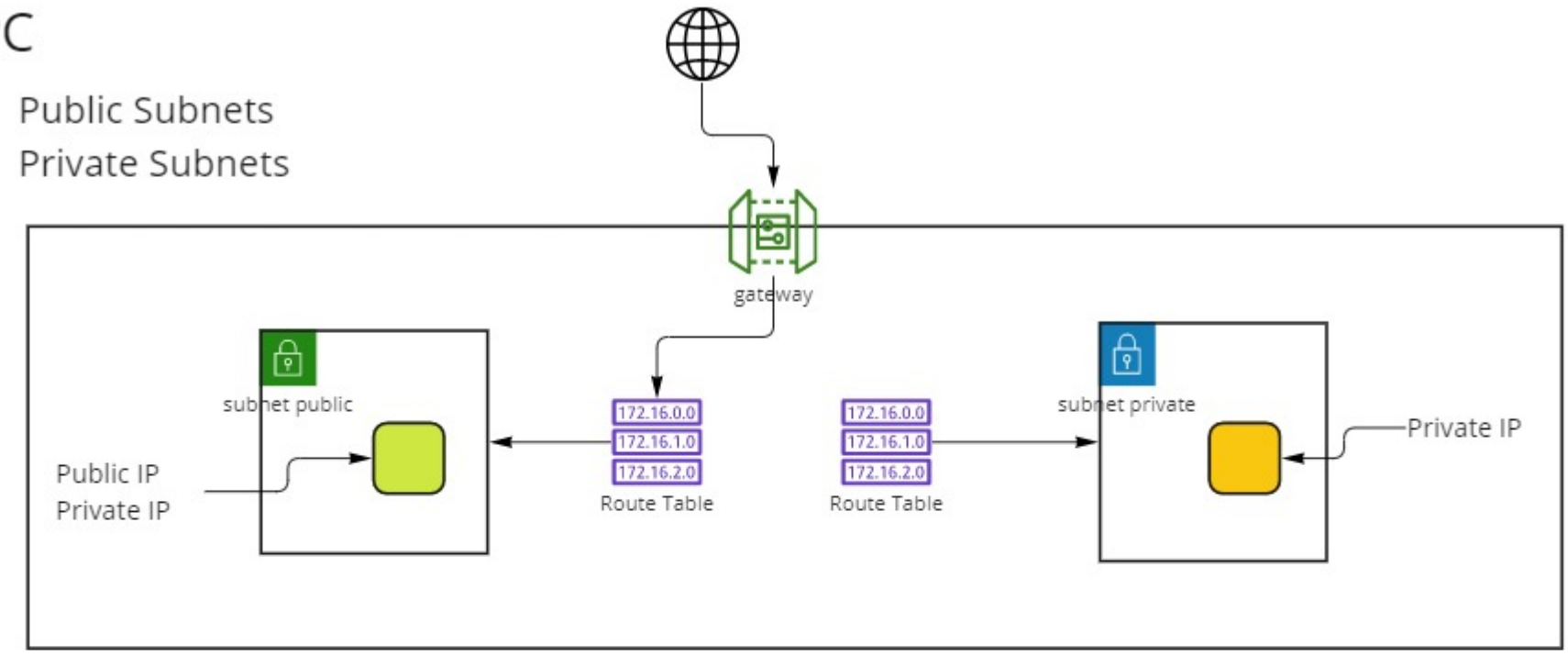
VPC contains Internet Gateway → It allows your VPC to connect with the Internet
Route Table
Subnet



VPC (Public & Private Subnets)
NAT Gateway
Peering Connection
Transit Gateway
S3

VPC

Public Subnets
Private Subnets

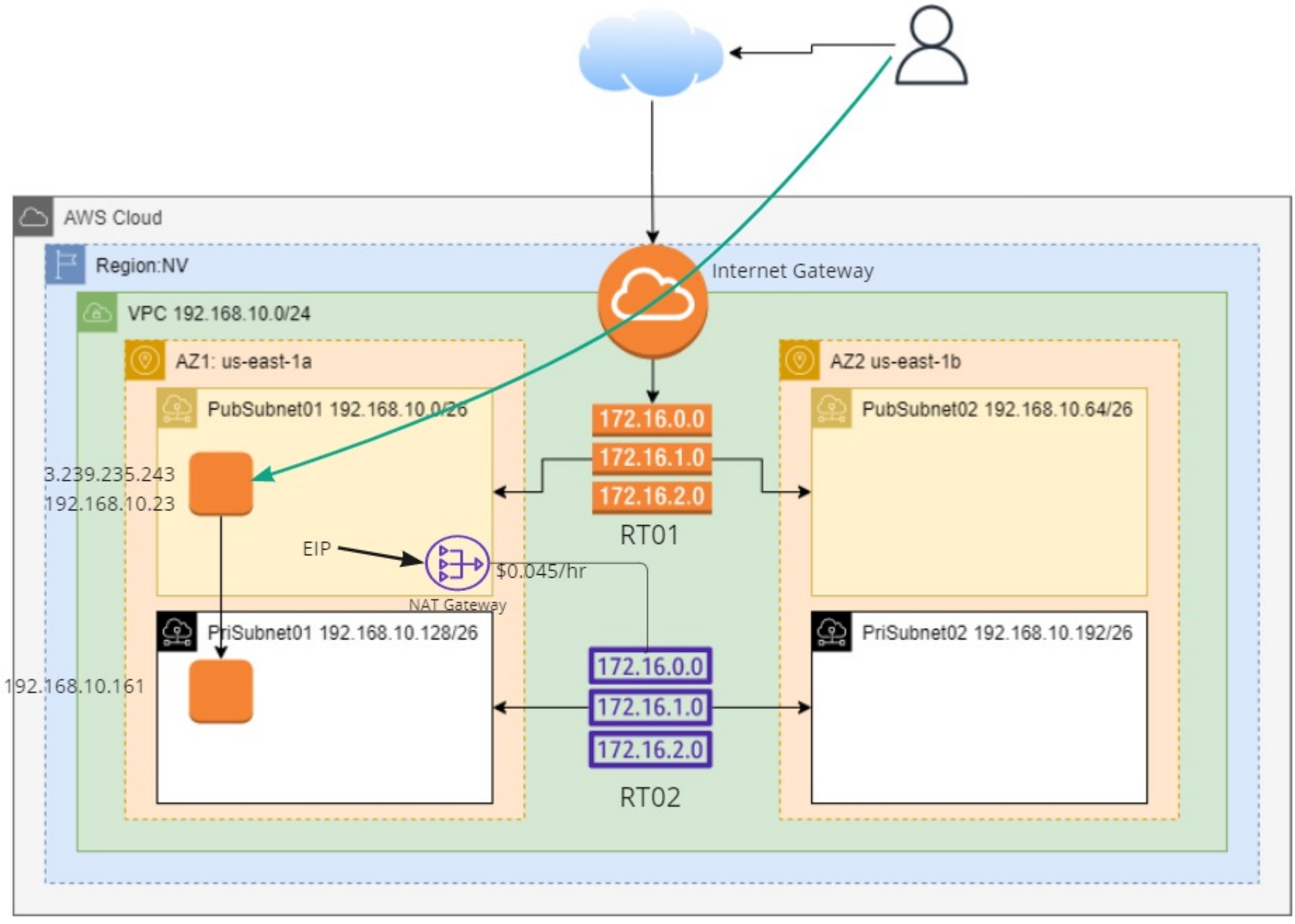


VPC [CIDR= 192.168.10.0/24]

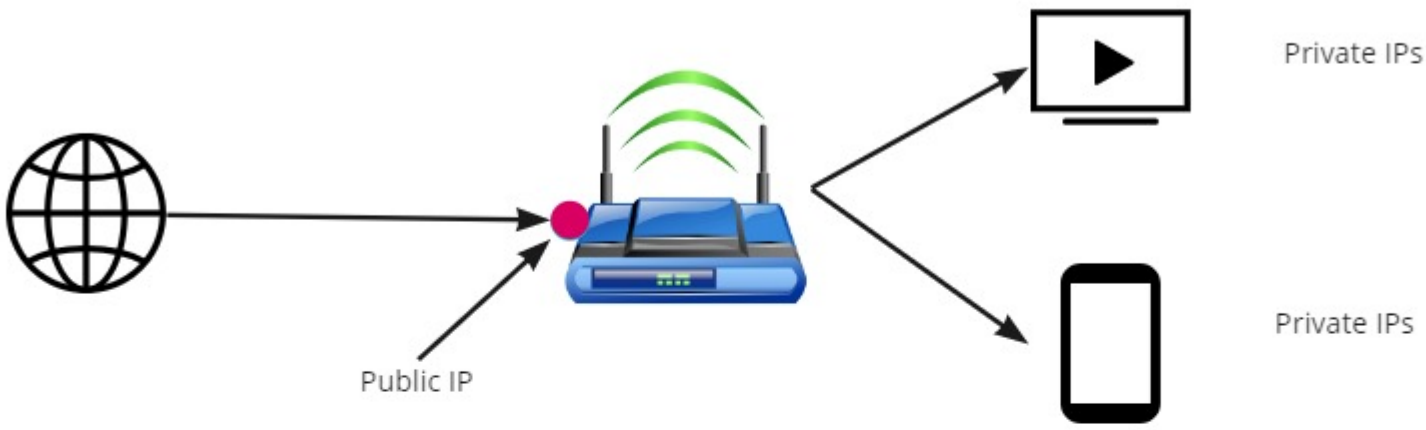
Public Subnet01 = 192.168.10.0 /26	us-east-1a
Public Subnet02 = 192.168.10.64 /26	us-east-1b
Private Subnet01 = 192.168.10.128 /26	us-east-1a
Private Subnet02 = 192.168.10.192 /26	us-east-1b

RouteTable01

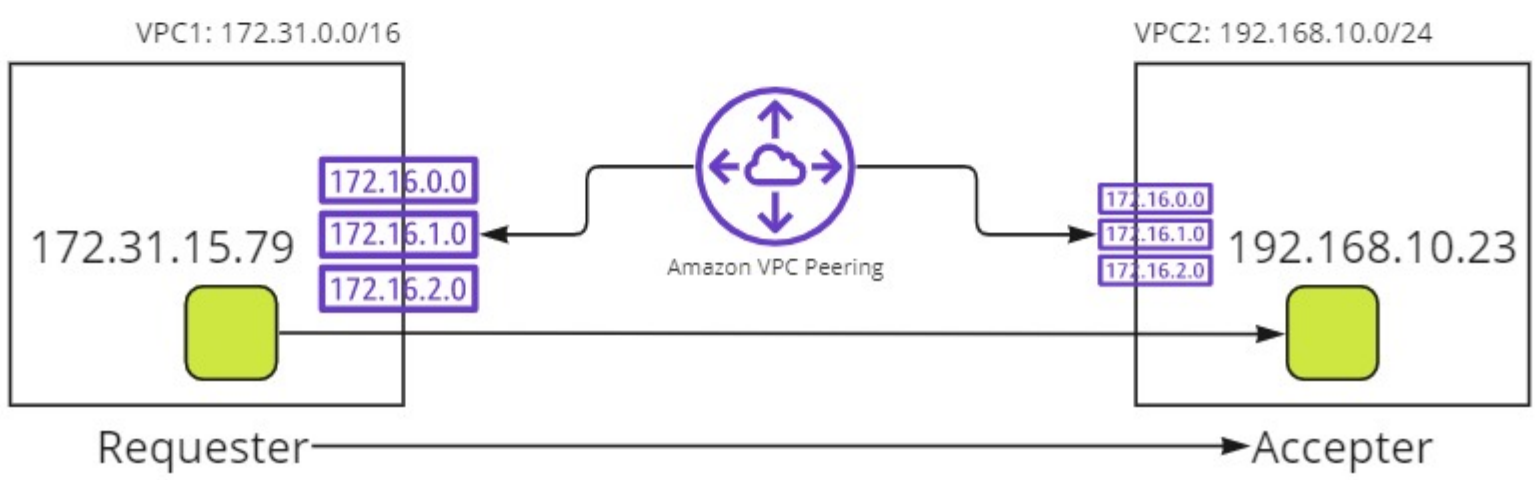
RouteTable02

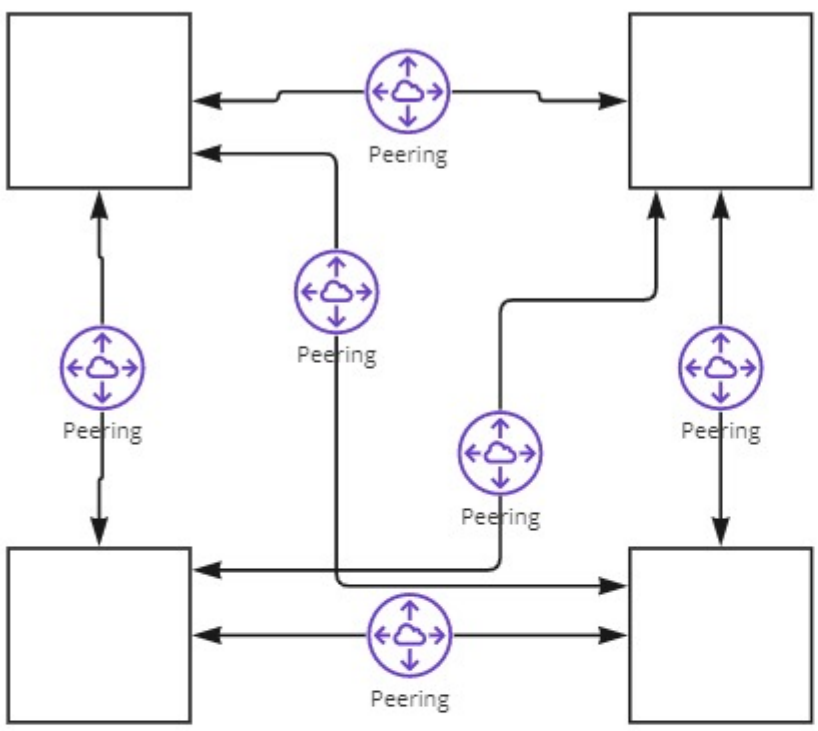


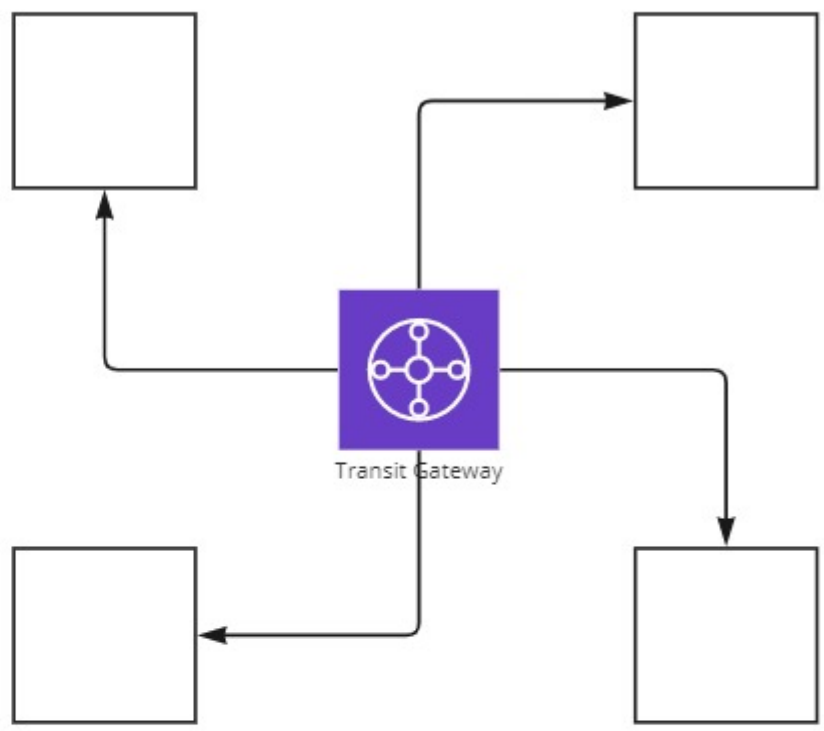
Wi-Fi Route => NAT Device



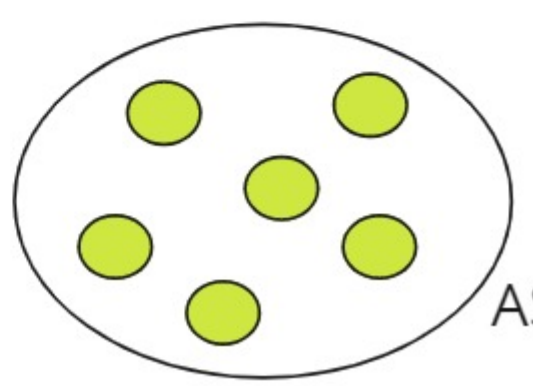
Peering Connection







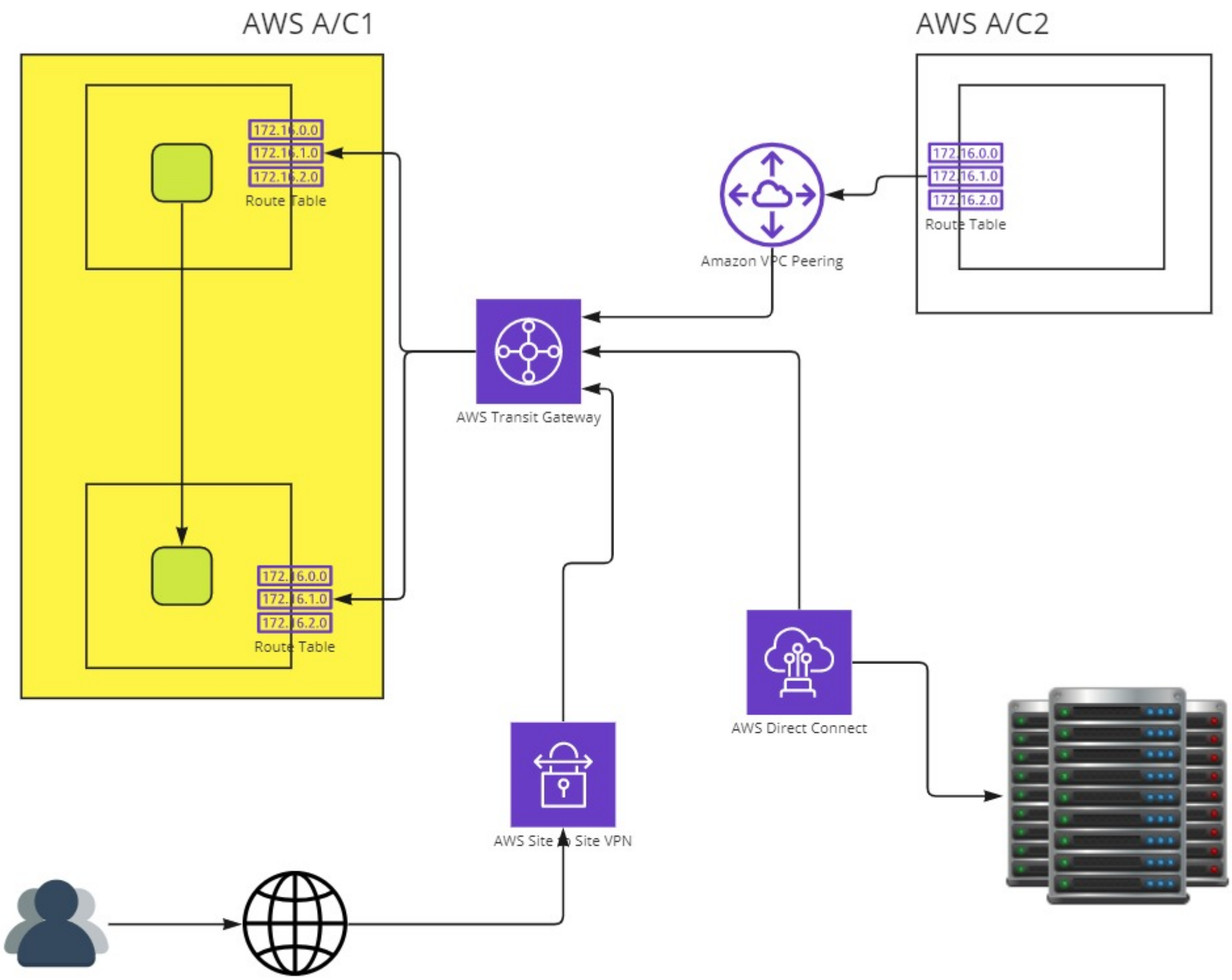
Autonomous System Number



Assignment: Connectivity between VPCs in different Regions

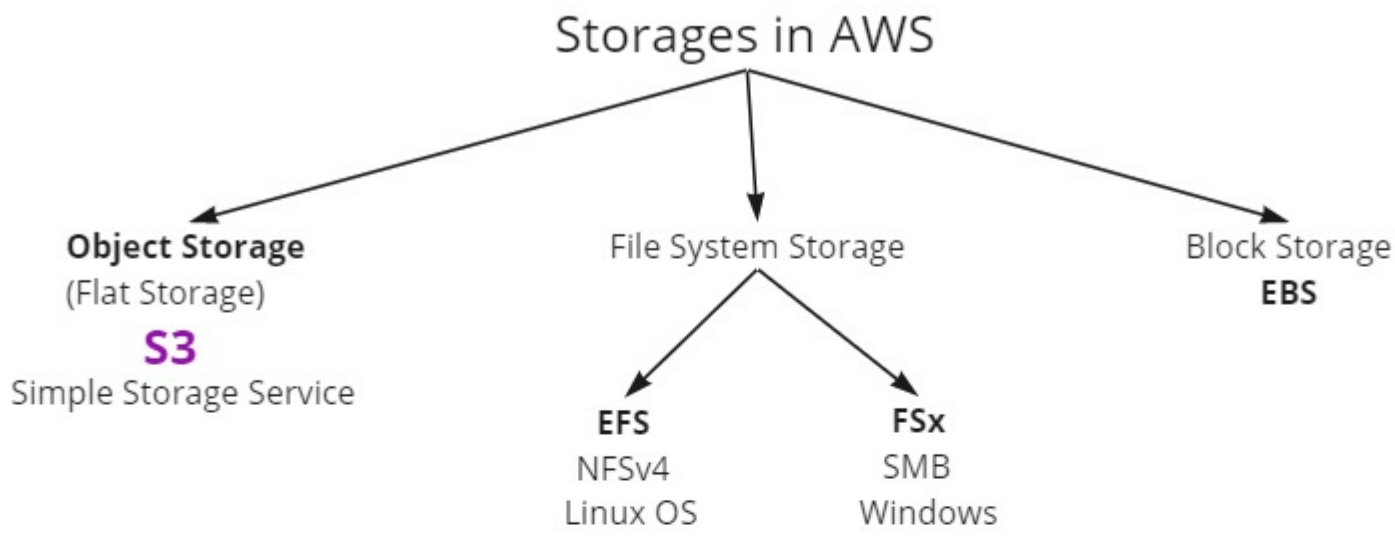
VPC, Data Center, access of VPC over the Internet, need to connect multiple VPC togather

Transit Gateway



Pricing

Price per AWS Transit Gateway attachment ($)	$0.05
Price per GB of data processed ($)	$0.02



Understanding AWS S3 (Simple Storage Service)

Sanjay Sharma

Object Storage
Flat Storage
Storage Accessible over the Internet

S3 is a Global Service in AWS
S3 is Object Storage or Flat Storage
In S3 you can store any amount of data
S3 Objects are accessible over the Internet

AWS S3

S3 Bucket is Region specific
Every bucket name is ***Globally unique*** within AWS Global Infrastructure
Every bucket has unlimited capacity
Buckets can not be linked as disk with EC2 instances directly
Buckets can be Public or Private, By default buckets are private
Bucket is Replicated on Multiple AZs by default

Bucket is a kind of container for objects (files)

Bucket

Buckets can be used for:

- Static Web Site Hosting
- Versioning
- Replication (Same & Cross Region)
- Object Life Cycle Management
- Data Encryption
- Server Access Logging
- Transfer Acceleration
- Event Notification

Folder (data)
data
Every object (file) stored in S3 will have it's Key name
Every object (file) will have its URL to access it from the Internet
Object
myfile.txt
Key of the object----- data/myfile.txt
Nested Folders
any number of nested folders can be created
Files are Objects in S3

S3 Quota or Limitation:

By default, you can create up to ***100 buckets*** in each of your AWS accounts. This limit can be increased up to ***1000*** Bucket on demand.

In AWS Free Tier A/C you have 5GB Storage space Free of Cost in S3.

By default, bucket is replicated in >= 3 Availability Zone to provide data availability

Availability Zone
Availability Zone
Availability Zone

S3 Object Storage Class

Amazon S3 offers a range of storage classes designed for different use cases or to save cost.

- *Standard*
- *Intelligent-Tiering*
- *Standard -IA (Infrequent Access)*
- *One Zone - IA*
- *Glacier Instance Retrieval*
- *Glacier Flexible Retrieval*
- *Glacier Deep Archival*

LAB:

S3
1. *Creating Bucket*
2. *Uploading Files (Objects)*
3. *Sharing Objects*
4. *Access Objects over the Internet*
5. Deleting S3 Bucket

Storage Class
Web Site Hosting

In S3 5GB Space is Free of Cost in Free Tier

Static Web Site Hosting

Using S3 bucket we can host a static website

You have not to provision any server.

Pre-requisite to host a static website: Template of Website is required

S3 (Simple Storage Service)

- **S3 Storage Classes**
- **Versioning**
- **Disaster Recovery using CRR**
- **Replication**
- **Life Cycle Management**

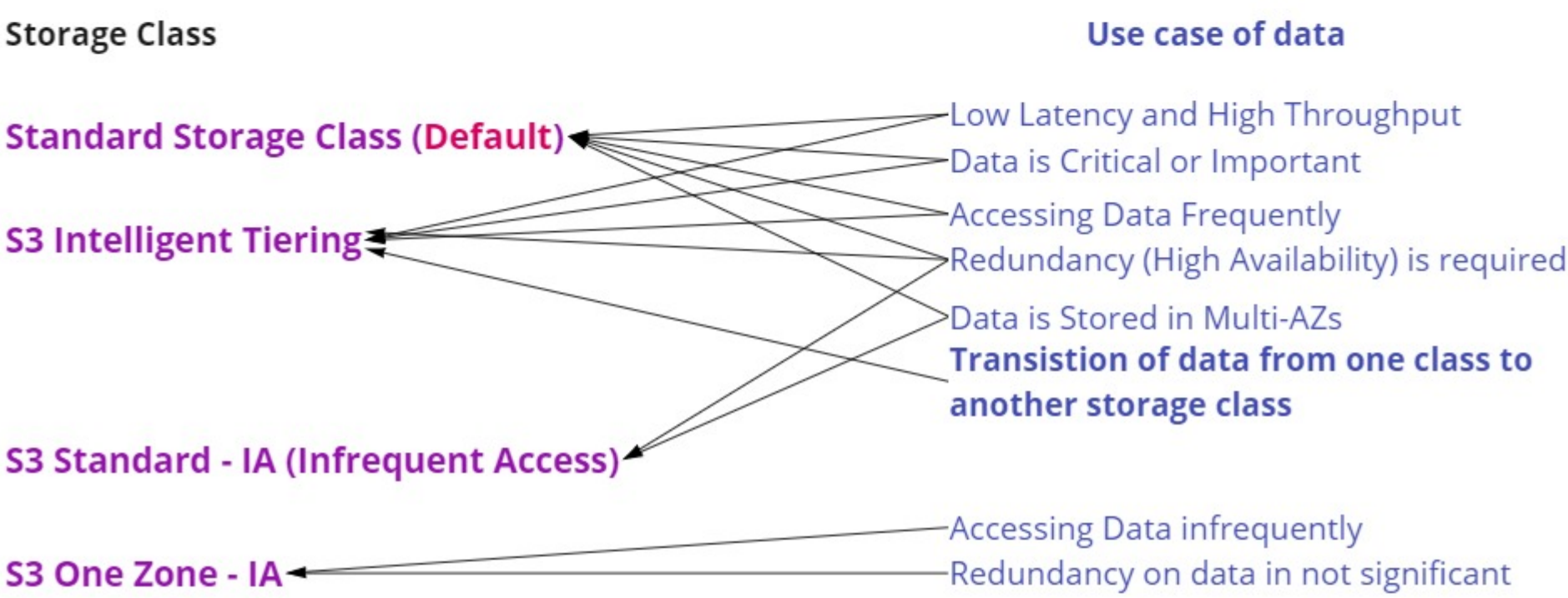
Data Base Services

S3 Storage Classes

On the basis of use pattern we select storage class to store data in S3, which provide data accessibility, resiliency and cost.

Types of Storage Classes

Standard Storage Class (Default)
Intelligent Tiering
Standard IA (Infrequent Access)
One Zone IA
Glacier Instant Retrieval
Glacier flexible Retrieval
Glacier Deep Archival



In One zone, data is stored in single AZ
while in Standard-IA data is stored in multiple AZs

S3 Pricing

Standard Storage Class (Default)	$**0.023** per GB/mo
S3 Intelligent Tiering	$0.023 per GB/mo
S3 Standard - IA (Infrequent Access)	$0.0125 per GB/mo
S3 One Zone - IA	$0.01 per GB
Glacier Instant Retrieval	$0.004 per GB
S3 Glacier Flexible Retrieval	$0.0036 per GB

for free tier eligible 5 GB data storage in S3 is Free of cost

logs, snapshots, application data, others

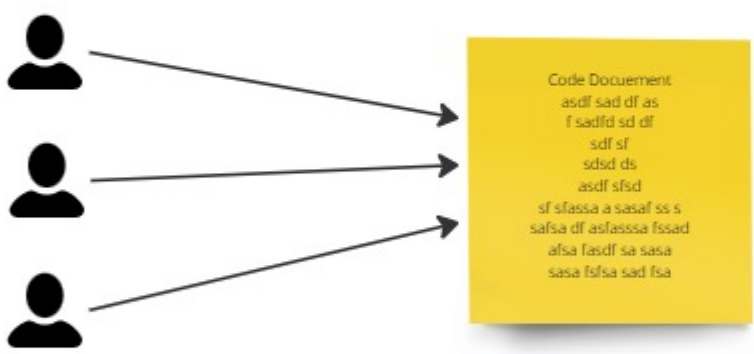
Versioning

To maintain multiple variants on an object is called versioning

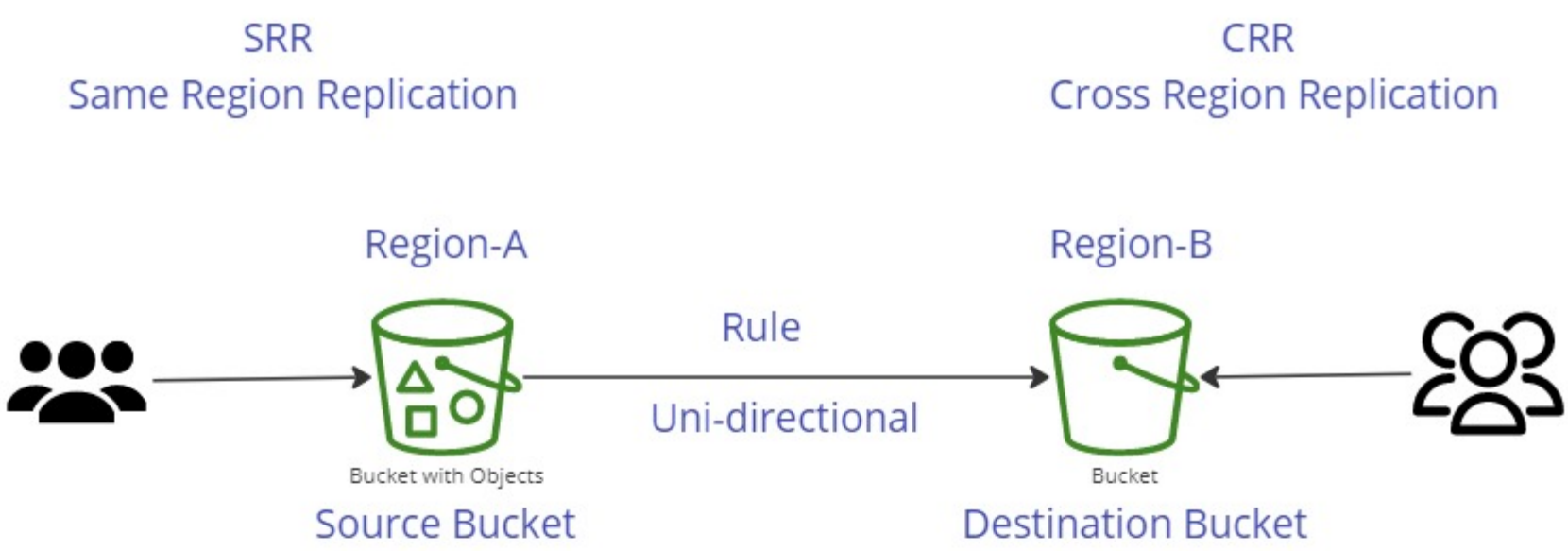
Versioning is disabled by default on buckets

Once you enable it, can't be disabled, it can only be suspended.

if the object is deleted accidently, can be restored because of versioning



Replication



Prerequisite for replication

Both buckets must be in different regions

Versioning on the both buckets must be allowed

Why use Replication

DR

During the transit you can store data from one storage class to another

Maintain object copies under different ownership

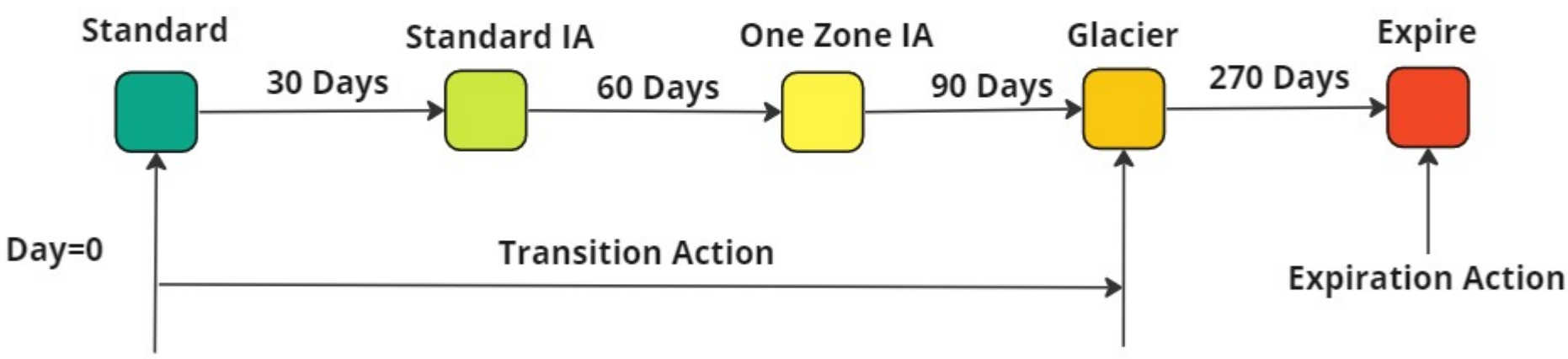
If you want to minimize latency

or you need to increase operational efficiency

Life Cycle Management

Two lifecycle actions:

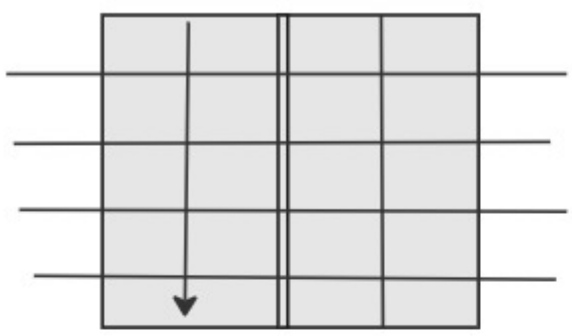
1.Transistion Action
2.Expiration Action



Current Version of Data ← Fresh data/just uploaded

non Current Version of data ← Existing Data

Multipart Upload



RDS

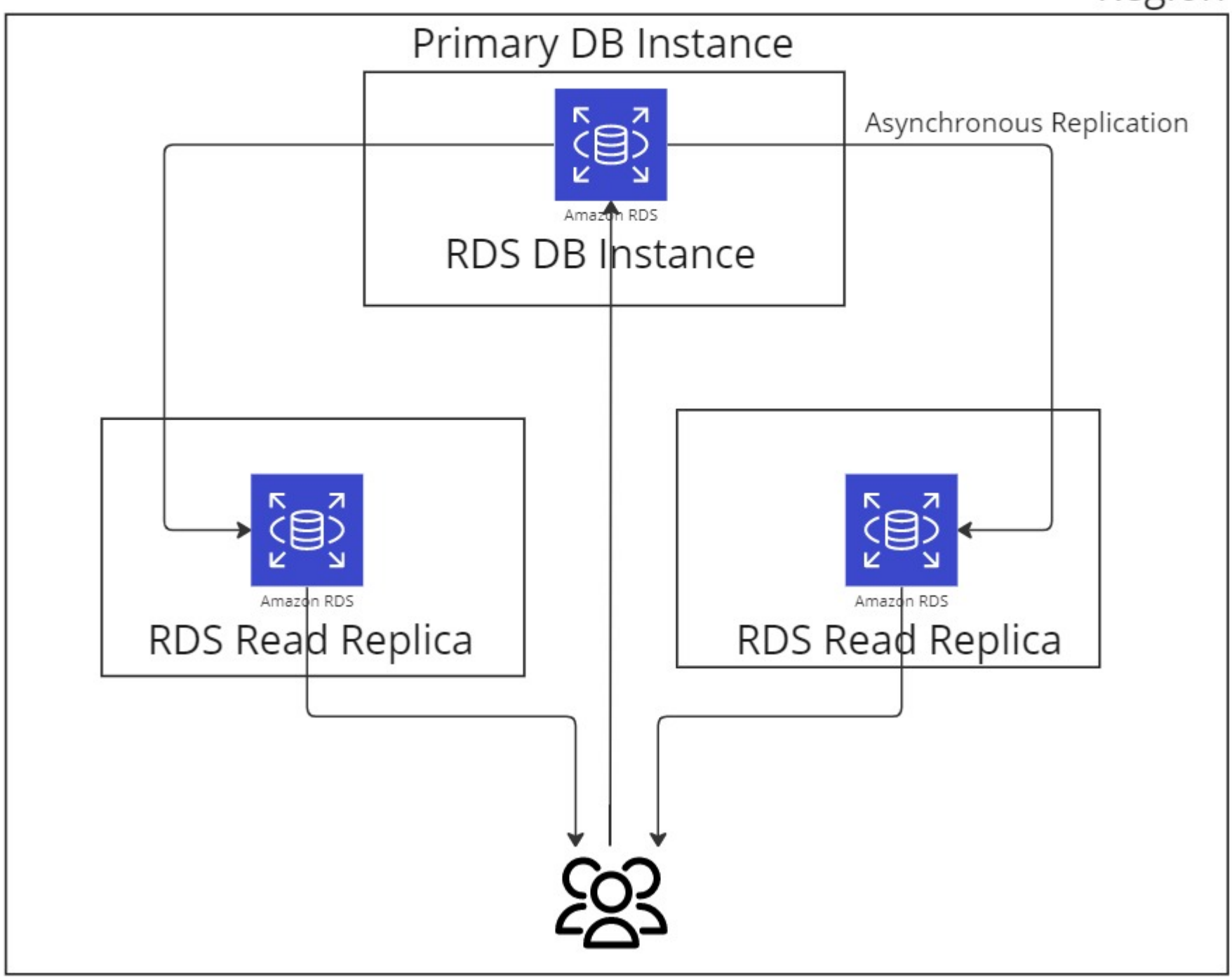
DB Instance

Snapshots

Restore DB Instance from Snapshot

Read Replica

DynamoDB



DynamoDB

Unstructured and semi structured Database

No SQL Database use alternate models for data management such as key-value pairs or document storage

Examples of Unstructured database:

Weather Data
Employees Emails
Surveillance Data IOT
Stock Market Data
Gaming Data

DynamoDB is fully Managed, multiregion, mulimaster, durable database with built in security

It can handle 10 trillion requests per day
20 million requests per second

DynamoDB is a Serverless Service

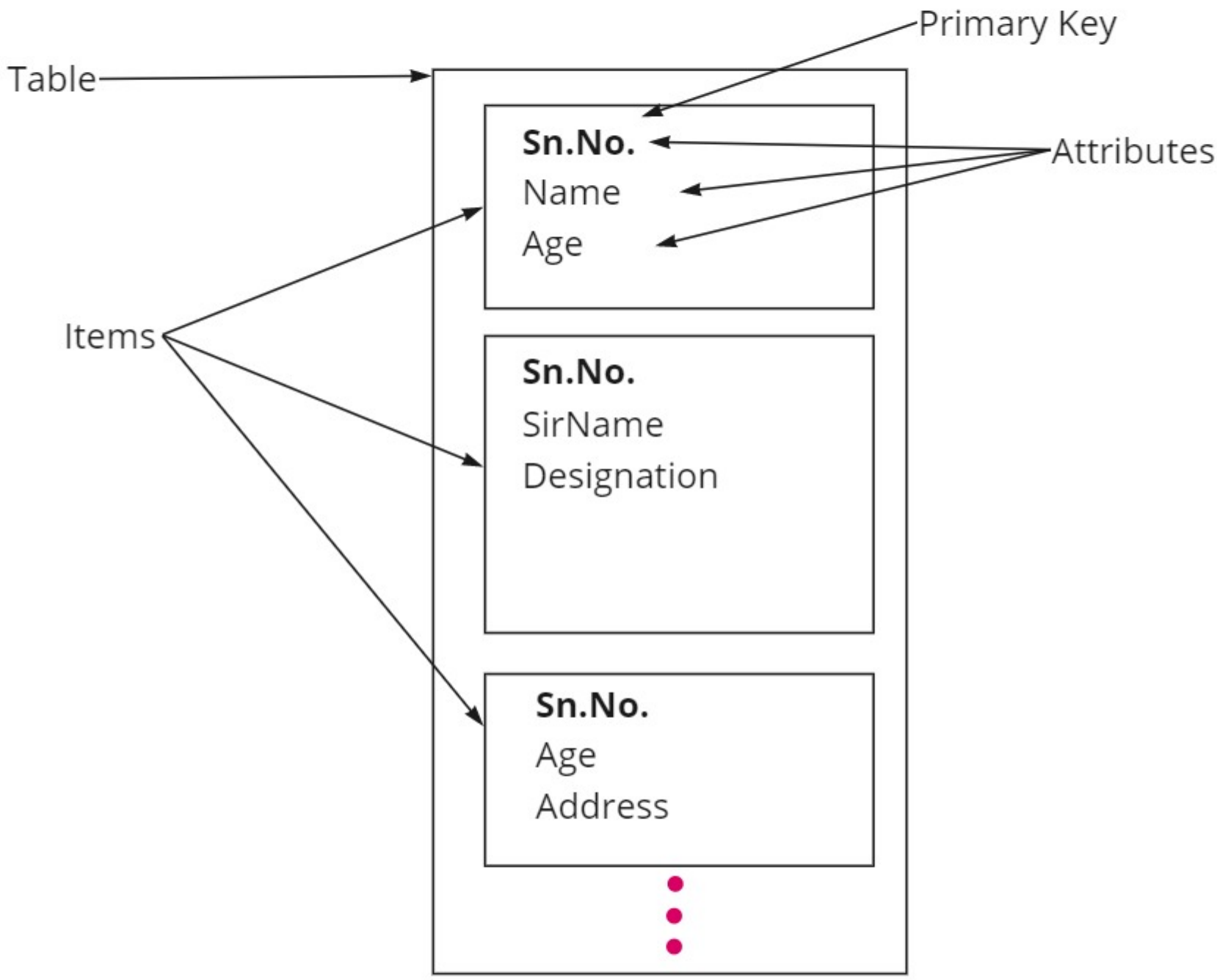
You can store any amount of data in DynamoDB Tables

You can scale up or scale down your **tables throughput** capacity with any downtime.

DynamoDB provides on-demand backup capabilities
Backup of tables is possible
Max retention period is 35 days
it delete expire items from tables automatically .
TTL allows you to define a per-item timestamp.

High Availability and Durability

Tables are replicated in muli-AZs in the Region
Where to use DynamoDB
DynamoDB Table Structure



DynamoDB supports two different kind of Primary Keys

Partition Key

Partition Key and Sort Key (Composite Primary Key)

S.No.
Country

DynamoDB Features:

It allows rapid replication of your data among multiple AZs in a Region.

Read/Write Capacity Mode

1. **On-Demand**
2. **Provisioned (default, free tier eligible)**

On-Demand

it is good if you create new tables with unknown workloads.
it is good if you have unpredictable application traffic
it is good if you prefer the ease of paying for only what you use.

Provisioned

it is good if you have predictable traffic
it is good if you run applications whose traffic is consistent and ramps gradually
it is good if you can forecast capacity requirement to control costs.

DynamoDB Read Capacity Unit (RCU)

Eventually Consistent (Fast and cheaper) (1 RCU = 8 KB/sec)

Strongly consistent (little slow but expensive)(1 RCU = 4 KB/sec)

Transactional (ACID: Atomicity, Consistency, isolation and durability)

DynamoDB Write Capacity Unit (WCU) (1 WCU= 1 KB/Sec)

Pricing: it charges for reading, writing and for storing data.

DynamoDB Limits

Quota of 256 tables per AWS Region

Size: no practical limit

AWS DynamoDB

25GB

Free of Cost

IAM (Identity and Access Management)

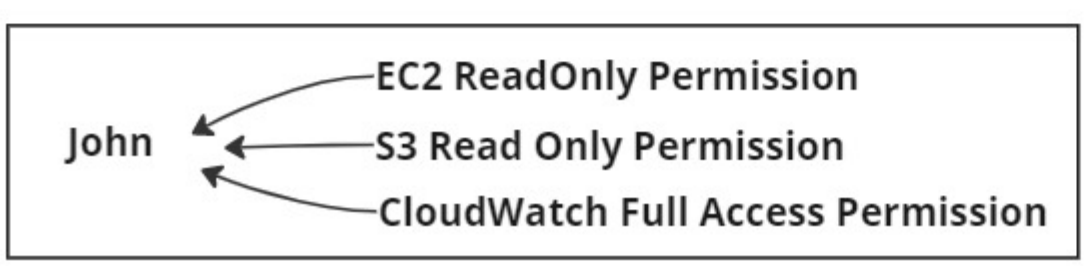
Users, Groups, Roles and Policies

Users

AWS Management Console Access Users → UI
Username & Passwords

Programmatic Access Users

Username, Access Key and Secret Access Key
recommended for AWS CLI, APIs, SDKs and other dev. tools



Number of Max Permission=10

AWS CLI

How to install AWS CLI in your host machine?

Programmatic Access User

↓

AWS CLI

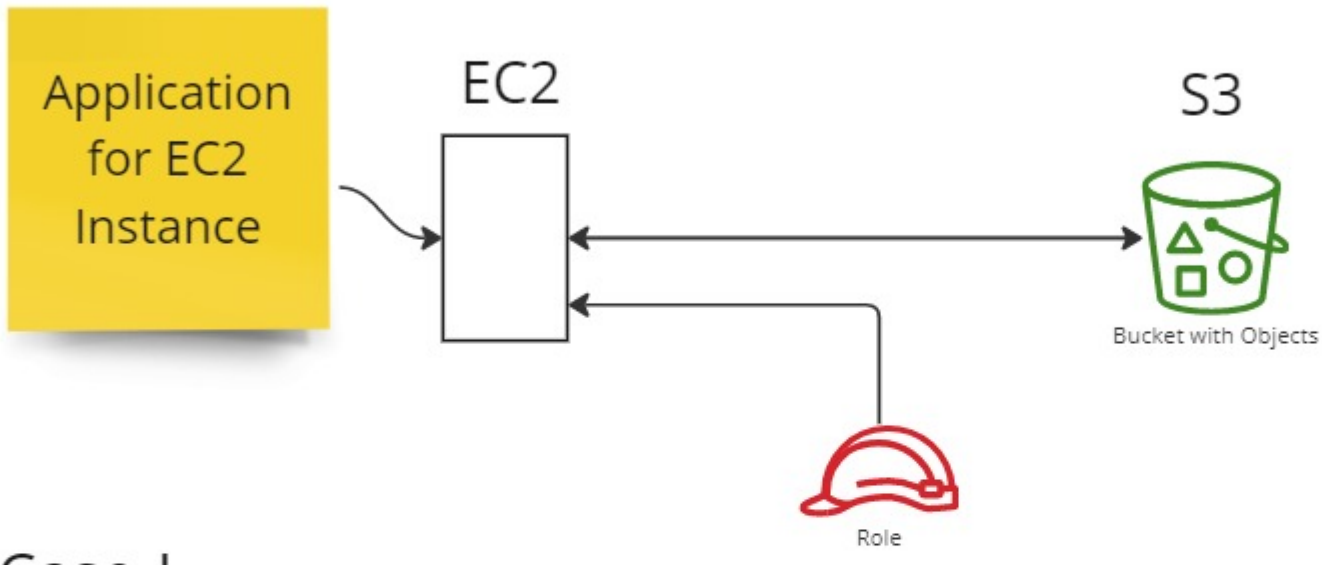
aws ec2 run-instances --image-id ami-xxxxxxxx --count 1 --instance-type t2.micro --key-name MyKeyPair --security-group-ids sg-903004f8 --subnet-id subnet-6e7f829e

aws ec2 run-instances --image-id ami-026b57f3c383c2eec --count 1 --instance-type t2.micro --key-name MyKey08Oct --security-group-ids sg-0191124cc71ae8a6a --subnet-id subnet-0cd2381d2ad7a0321

Roles
Groups
Permissions

IAM Group

Group is a collection of IAM users



Case-I

You have a user with S3 permission

Case-II

You will create a Role for EC2 Service with S3 Permission and the Role will attach with EC2 Instance.

Systems Manager: To manage multiple instances simultanously

Systems Manager

Role Connected with SSM Permission
System Manager's Agent must be installed on these instances

Policy

We need to create a policy for EC2 service with very limited access to just see few EC2 resources.

Policy

I have the Code in JSON format

Tool: Visual Editor
Visual Editor will write code for you

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": "ec2:*",
         "Resource": "*",
         "Condition": {
            "DateGreaterThan": {"aws:CurrentTime": "2022-10-15T00:00:00Z"},
            "DateLessThan": {"aws:CurrentTime": "2022-10-18T23:59:59Z"}
         }
      }
   ]
}
```

Route53
CloudFormation
SNS
SQS
Lambda
Elastic Beanstalk
CloudWatch
CloudTrail
Migration

Route53

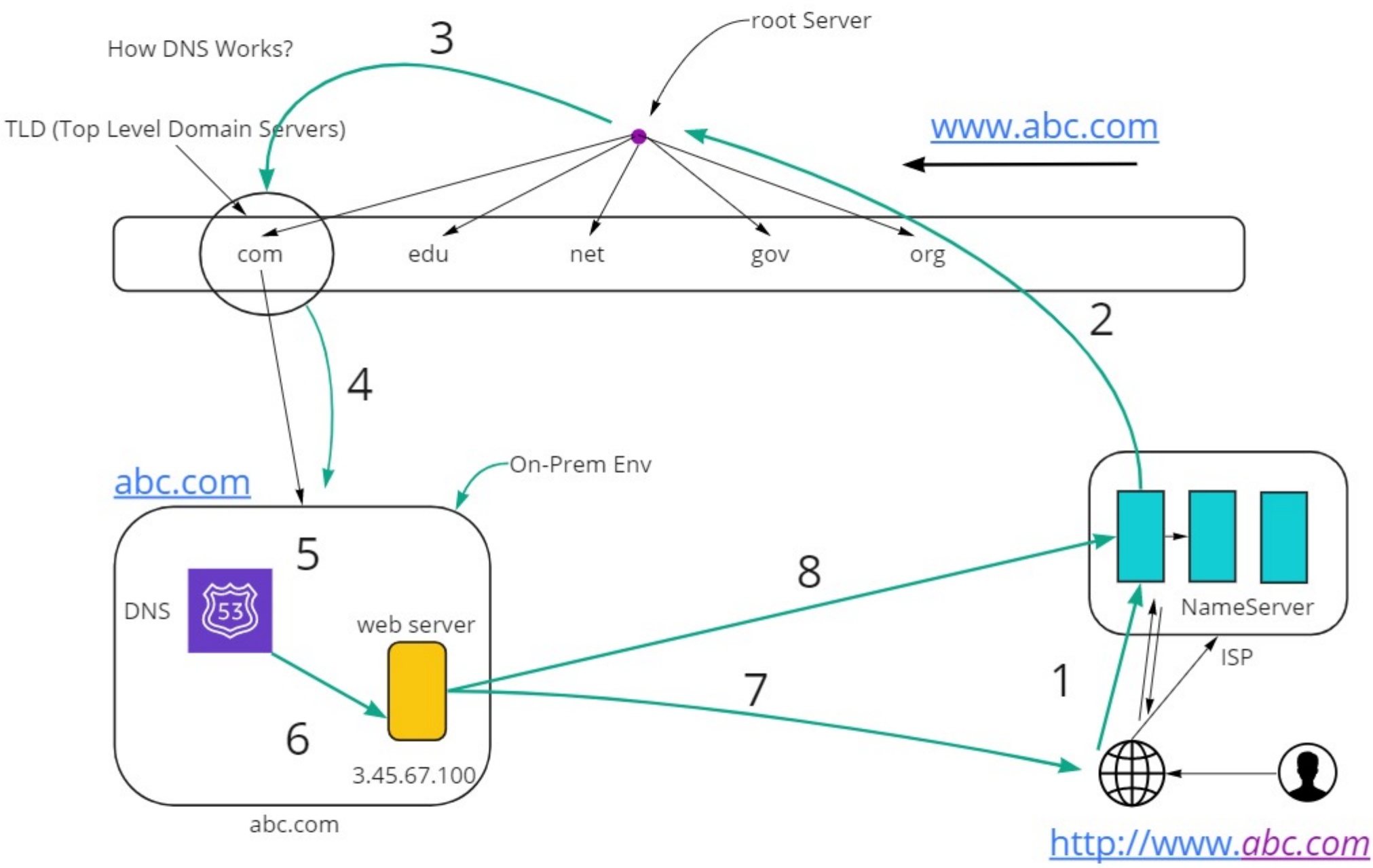
Route53 it is a DNS service in AWS  Serverless Service

Route53 is a DNS Service in AWS

Its a Serverless Service

1. ***Register Domain Names***
2. ***To route Internet Traffic to the resources for Domain***
3. ***Check the Health of the resource***
4. ***Hosted Zone Configuration***

$0.5 per domain per mo
per hosted zone/mo



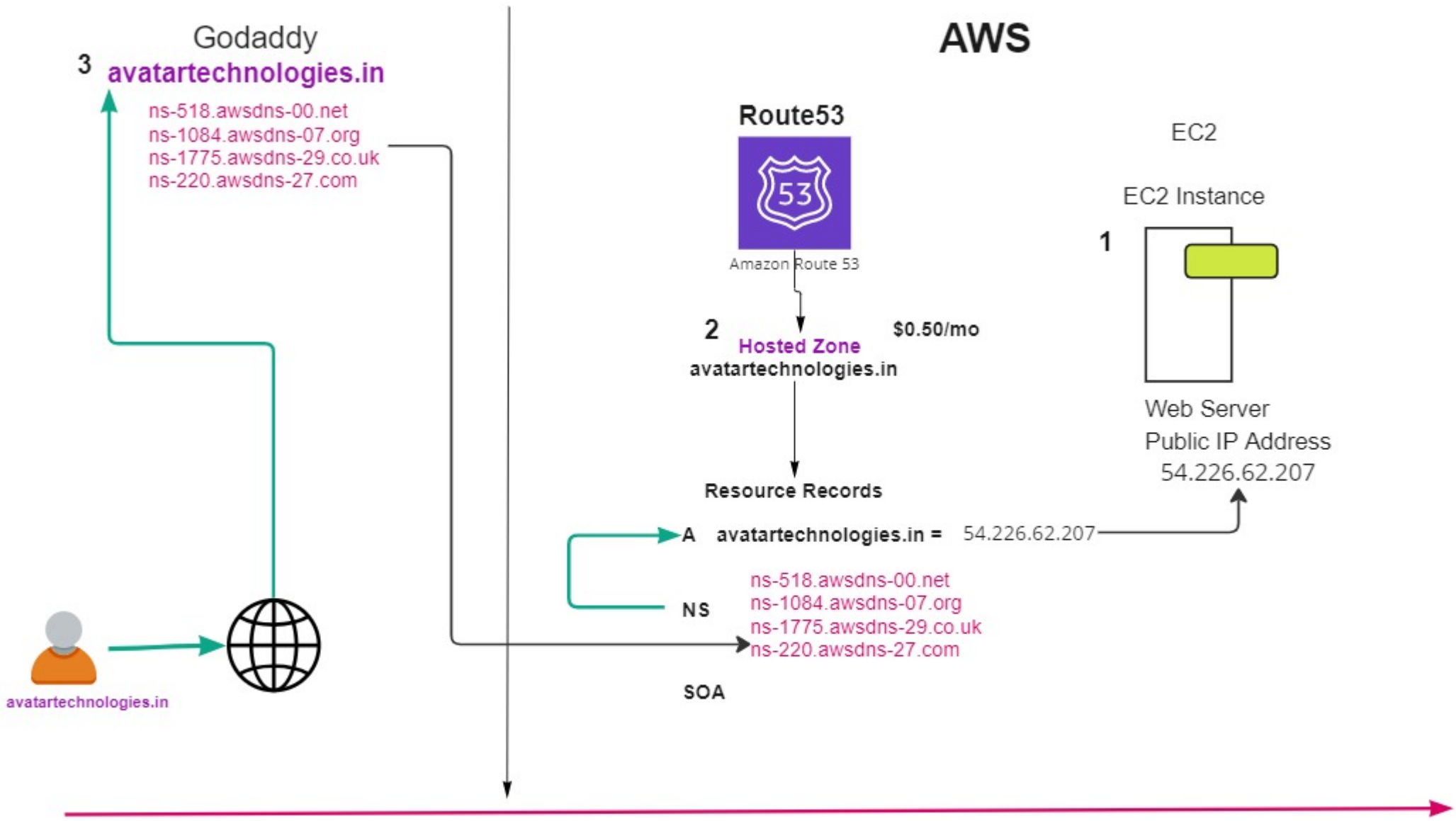
DNS is containing Records of Resources

Number of Resource Records (RR)

A = > Hostname/Domain = IPv4 address of the website
AAAA => Hostname/Domain = IPv6 Address of the Website
CNAME => Canonical name (Alias Name) => another name of the domain
NS => NameServer => it is connected with ISPs/Domain name Provider Name Server
SOA ==> Start of Authority ==> providing some important parameters such as admin email ID, TTL, DNS Ver Name etc.
MX ==> Mail Exchange Record ==> it is for mail server, it routes your traffic to the mail server in your infrastructure.

Hosted zone: a set of records for a particular domain name

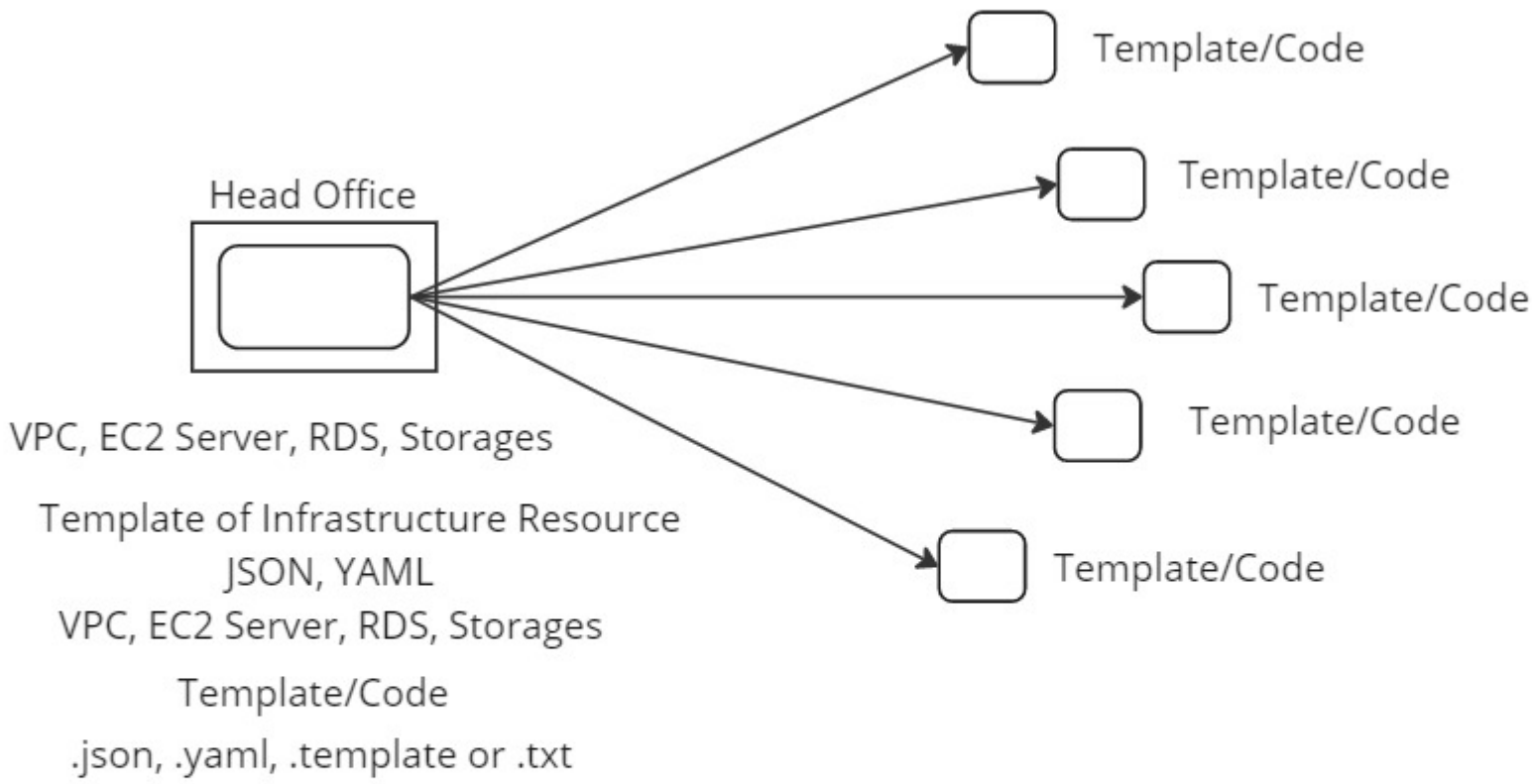
Route53: You can manage multiple hosted zone, for every hosted zone you need to pay $0.5/mo
One zone pertaining to one domain name.



Routing Policies

1. **Simple routing policy** – Use for a single resource that performs a given function for your domain, for example, a web server that serves content for the example.com website. You can use simple routing to create records in a private hosted zone.
2. **Failover routing policy** – Use when you want to configure active-passive failover. You can use failover routing to create records in a private hosted zone.
3. **Geolocation routing policy** – Use when you want to route traffic based on the location of your users. You can use geolocation routing to create records in a private hosted zone.
4. **Latency routing policy** – Use when you have resources in multiple AWS Regions and you want to route traffic to the region that provides the best latency. You can use latency routing to create records in a private hosted zone.
5. **IP-based routing policy** – Use when you want to route traffic based on the location of your users, and have the IP addresses that the traffic originates from.
6. **Multivalue answer routing policy** – Use when you want Route 53 to respond to DNS queries with up to eight healthy records selected at random. You can use multivalue answer routing to create records in a private hosted zone.
7. **Weighted routing policy** – Use to route traffic to multiple resources in proportions that you specify. You can use weighted routing to create records in a private hosted zone.

CloudFormation ⟶ Infrastructure as Code



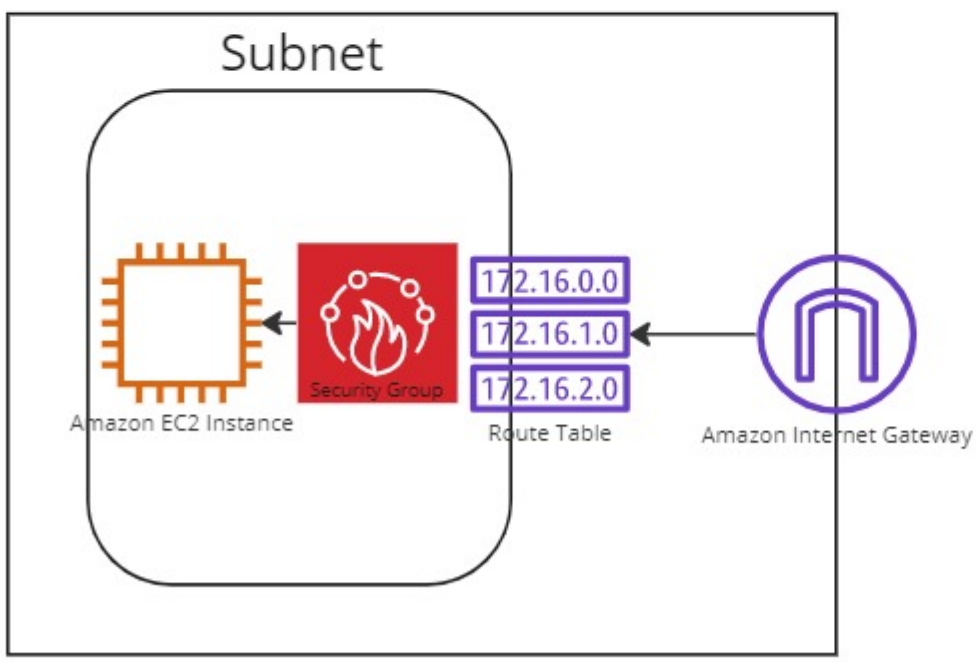
Stack:

CloudFormation Designer Tool ⟶ Write a Template for Infrastructure

aws cloudformation template to create s3 bucket

```
{
  "Resources": {
    "S3Bucket": {
      "Type": "AWS::S3::Bucket",
      "DeletionPolicy": "Retain",
      "Properties": {
        "BucketName": "intel.16oct.in"
      }
    }
  }
}
```

VPC

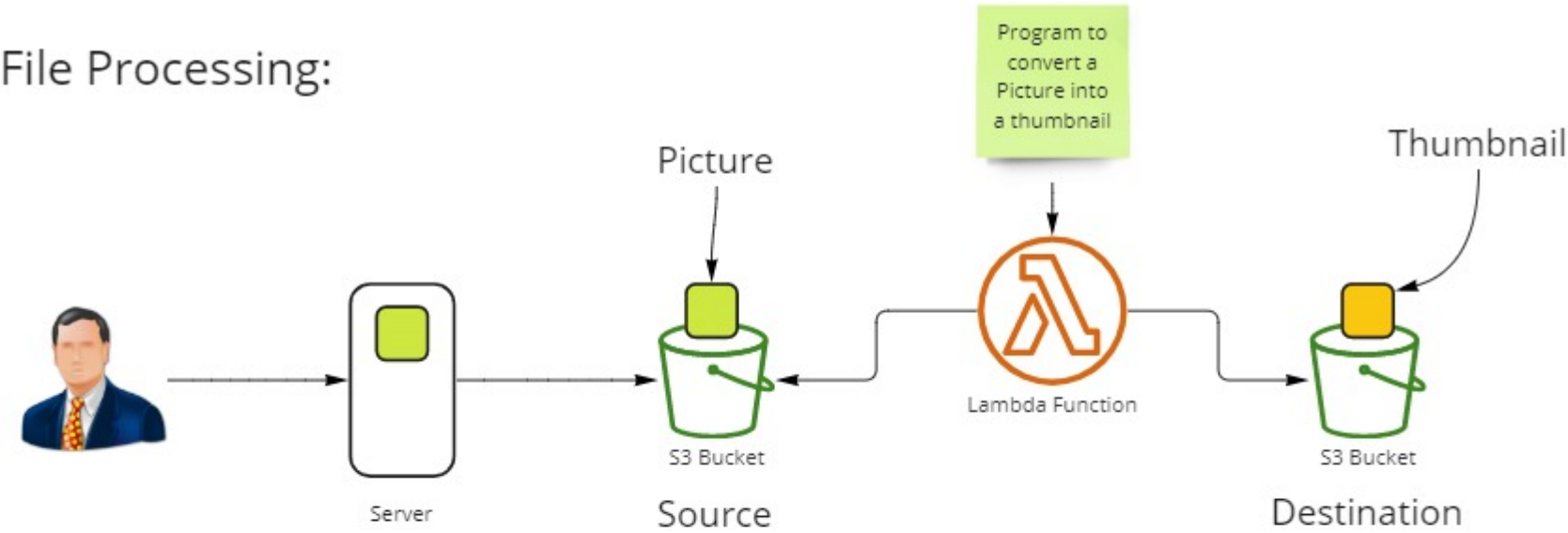


Where Lambda can be used?

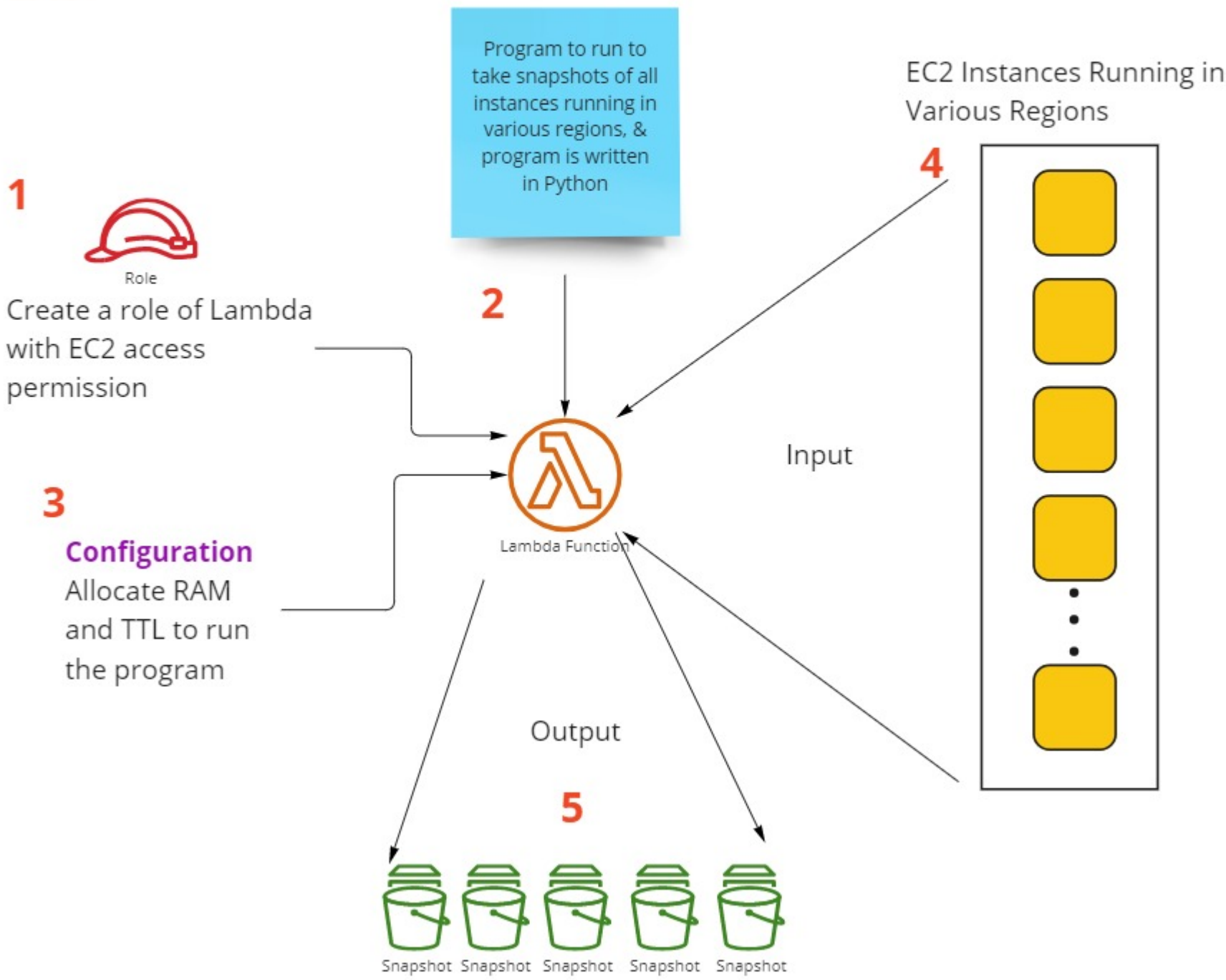
Lambda functions and triggers are the core components of building applications on AWS Lambda. A Lambda function is the code and runtime that process events, while a trigger is the AWS service or application that invokes the function. To illustrate, consider the following scenarios:

1. ***File processing*** – Suppose you have a photo sharing application. People use your application to upload photos, and the application stores these user photos in an Amazon S3 bucket. Then, your application creates a thumbnail version of each user's photos and displays them on the user's profile page. In this scenario, you may choose to create a Lambda function that creates a thumbnail automatically. Amazon S3 is one of the supported AWS event sources that can publish object-created events and invoke your Lambda function. Your Lambda function code can read the photo object from the S3 bucket, create a thumbnail version, and then save it in another S3 bucket.
2. ***Data and analytics*** – Suppose you are building an analytics application and storing raw data in a DynamoDB table. When you write, update, or delete items in a table, DynamoDB streams can publish item update events to a stream associated with the table. In this case, the event data provides the item key, event name (such as insert, update, and delete), and other relevant details. You can write a Lambda function to generate custom metrics by aggregating raw data.
3. ***Websites*** – Suppose you are creating a website and you want to host the backend logic on Lambda. You can invoke your Lambda function over HTTP using Amazon API Gateway as the HTTP endpoint. Now, your web client can invoke the API, and then API Gateway can route the request to Lambda.
4. ***Mobile applications*** – Suppose you have a custom mobile application that produces events. You can create a Lambda function to process events published by your custom application. For example, you can configure a Lambda function to process the clicks within your custom mobile application.
5. ***Orchestration*** (Task Administration)

File Processing:



LAB:



Duration: 11129.08 ms

Billed Duration: 11130 ms

Memory Size: 512 MB

Max Memory Used: 94 MB

Memory (MB)	Price per 1ms
128	$0.0000000021
512	$0.0000000083

Total Charges for the program for a month = 30 x11130 ms x $0.0000000021 = $0.00070119

EC2 Instance = $25- $30

Lambda Function

Serverless Service to run code
It runs code in a serverless manner without provisioning any Instance

Serverless service has some characteristics

1. No Server management
2. Flexible Scaling
3. No Idle Capacity
4. High Availability

Serverless Applications have several components and layers:

Compute
API
Storage (we have ephemeral storage (temp storage))
Interprocess Messaging
Orchestration

Lambda provide compute resources to run program.

Use of memory(RAM) and CPU Power are proportionate in Lambda

→ ✓ RAM ↑ → CPU ↑

Execution	TTL
30 Seconds	40 sec

Min RAM =128 MB
Max RAM = 10 GB

→ Lambda can also provide TTL (Time to Live), time in which Lambda Function should be completed. ***TTL is the max amount of time in which program must be completed.***

Eg: if you allocate some TTL = 40 seconds, it means program must be completed within 40 seconds, otherwise lambda function will die.

AWS Lambda will not charge for the TTL, lambda will charge only to execution time.

Let's say if TTL = 40 sec, and program takes 10 sec to complete, 10 sec is execution time and Lambda will charge for 10 sec, not for 40 sec.

Pricing in Lambda depends on two factors:

1. Amount of RAM is consumed to run the Program, not allocated RAM

   *eg: if you allocate 512 MB RAM, and program takes only **128 MB** to execute the program, you will pay only for 128 MB RAM, not for 512 MB.*

2 *Lambda charges in milliseconds not in seconds*

eg: you consume 10 sec to run the program, it means you will pay for 10000 ms

1 second = 1000 milliseconds

Elastic Beanstalk PaaS

Application Deployment

Auto Deployment of Application

Prerequisite: You must have an application to deploy

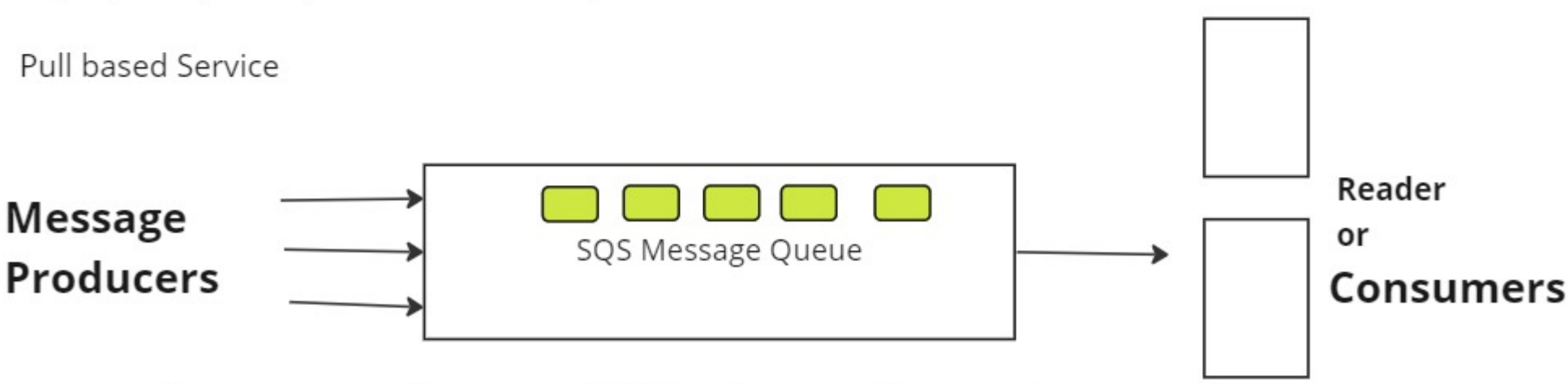
SNS
SQS
OpsWorks
CloudWatch
CloudTrail
Migration

Simple Notification Service (SNS)

SNS service is PUSH based System

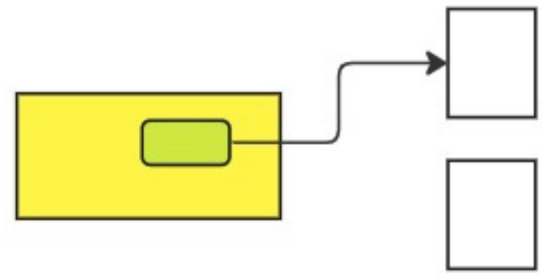
SQS (Simple Queue Service) -- Pull based service

Pull based Service



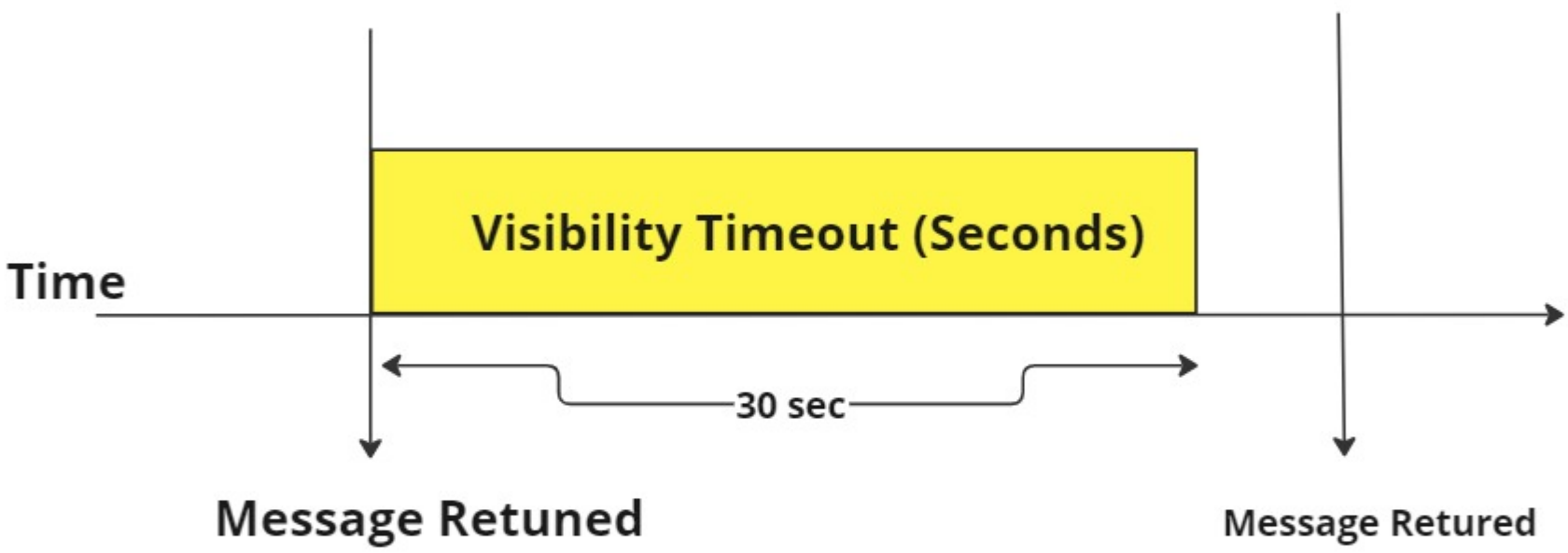
it can contain upto 256 KB in any format json, xml etc.

Standard Queues (Default)
FIFO Queues



SQS Visibility Timeout

Receive Message Request



Default Visibility Timeout=30sec
Visibility Timeout can be changed
Max Visibility Timeout can be 12 Hours
But messages can be kept in queues from 1 min to 14 days, and default retention period is 4 days.

https://aws.amazon.com/getting-started/hands-on/send-fanout-event-notifications/?ref=gsrchandson

CloudWatch & CloudTrail

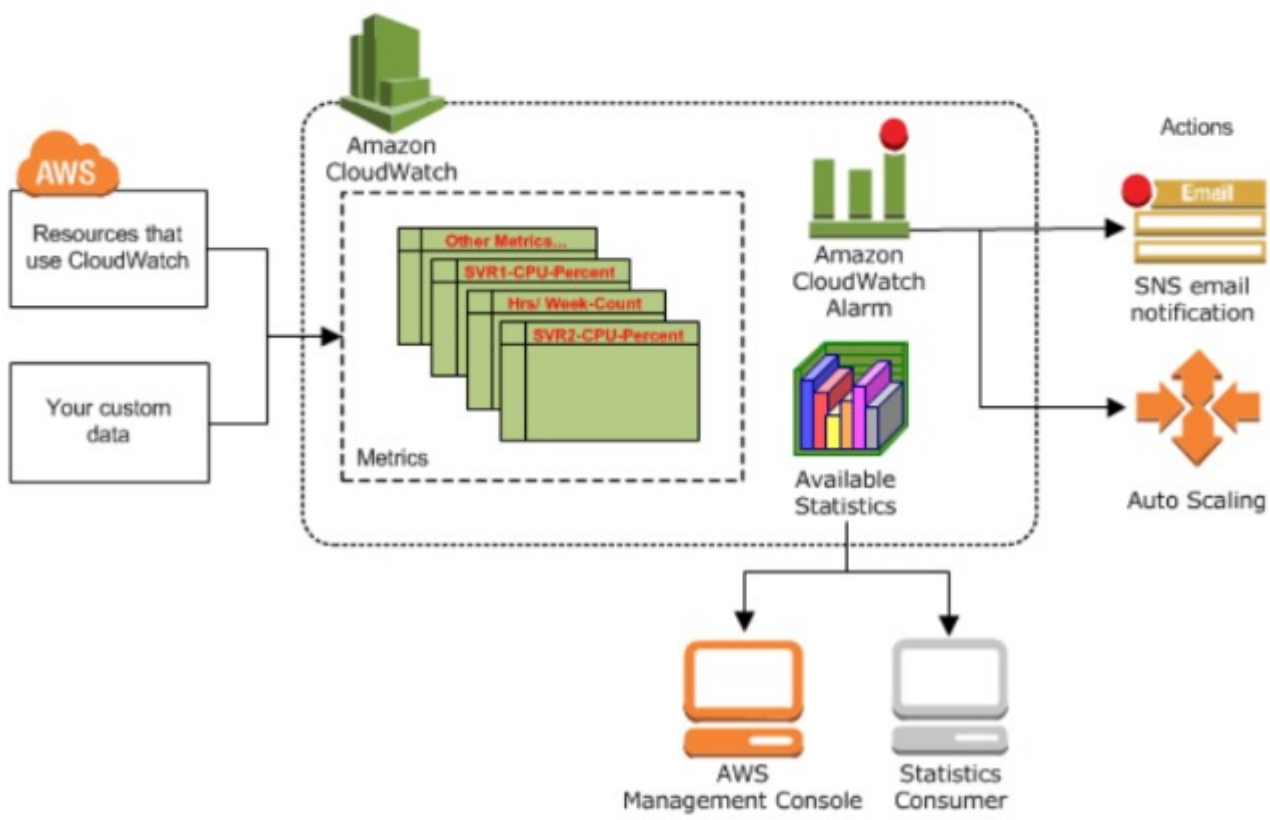
Monitoring Service

Basic Monitoring

Its Free, Polls data in every 5 minutes, work with limited metrics, 5 GB ingested data

Enhanced Monitoring

Its a chargeable, charge per instance per month, it can polls data in every one minute, large number of metrics.



Metric: It is known as sensors on which Cloudwatch is monitoring services

AWS EC2 Metrics:

- CPUUtilization
- DiskReadOps
- DiskWriteOps
- NetworkIn
- NetworkOut
- NetworkPacketIn etc.

AWS Billing and Cost Metrics

- Estimated Charges
- ServiceName
- LinkedAccount etc

AWS EBS Metrics

- EBSReadOps
- EBSWriteOPs
- EBSReadBytes
- EBSWriteBytes etc

Alarm

if CPUUtilization >=60%, the instance must be stopped

Benefits of CloudWatch Monitoring

- AWS Resource Monitoring
- AWS EC2 Instance Monitoring
- Setting Alarms and Actions
- Viewing Graphical Representation
- Viewing Statistics
- Monitoring and Storing Logs

Lab: Using CloudWatch to create a Bill Alarm by enabling estimated charges for AWS Account

Set an alarm if current AWS Account Charges are >= $10, you should receive notification on your registered email ID with SNS.

Enable Billing Alerts
Create a Billing Alarm
Check the Alarm Status
Delete the Billing Alarm

OpsWork
Migration

https://explore.skillbuilder.aws/learn

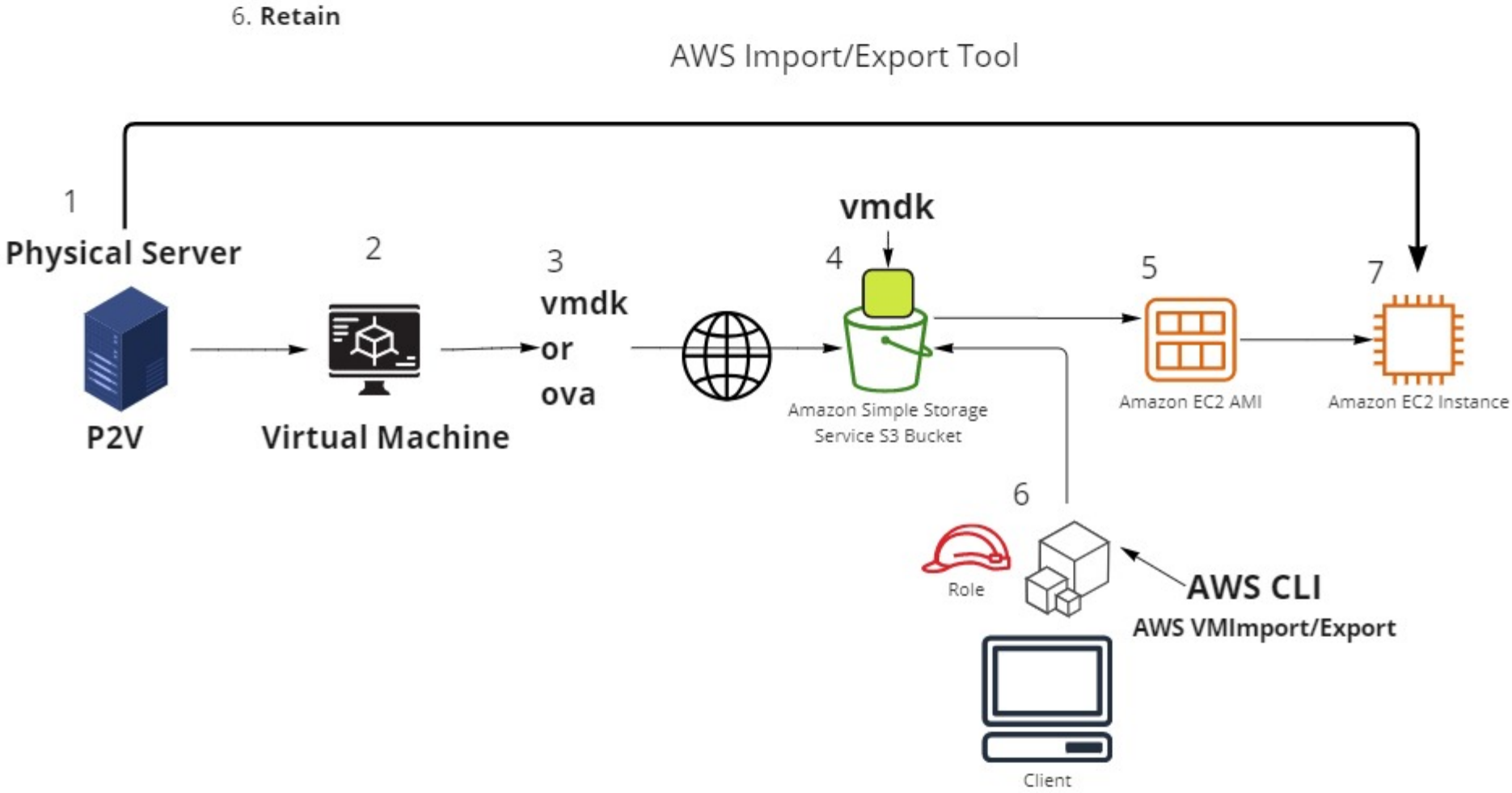
Migration:

Assess the workload
Mobilize your workload
Migrate and Modernize

Strategy behind migration:

6 R's Strategy

1. **Rehost (lift and Shift)**
2. **Re-Platform (lift, tinker and shift)**
3. **Re-Factor/Re-Architect**
4. **Re-Purchase**
5. **Retire**
6. **Retain**



Command to create Role

aws iam create-role --role-name vmimport --assume-role-policy-document "file://trust-policy.json"

Import an image

aws ec2 import-image --description "My server disks" --disk-containers "file://containers.json"

Monitor an import image task

aws ec2 describe-import-image-tasks --import-task-ids ***import-ami-1234567890abcdef0***

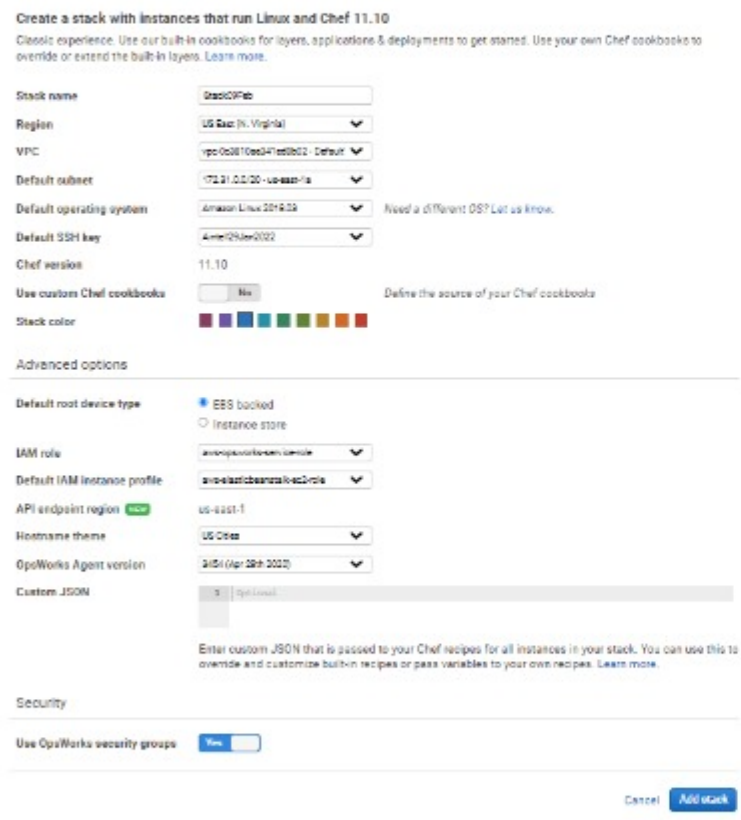
Link of Migration (VMImport/Export Documents)
https://docs.aws.amazon.com/vm-import/latest/userguide/vmimport-image-import.html

Sanjay Sharma

OpsWorks

AWS OpsWorks is a configuration management service that provides managed instances of Chef and Puppet. Chef and Puppet are automation platforms that allow you to use code to automate the configurations of your servers. OpsWorks lets you use Chef and Puppet to automate how servers are configured, deployed, and managed across your Amazon EC2 instances or on-premises compute environments.

1. *AWS OpsWorks Stacks*
2. *First Stack*
3. *Create a stack with instances that run Linux and Chef 11.10*



github sample for opsworks

https://github.com/aws-samples/opsworks-demo-php-simple-app.git

opsworks-demo-php-simple-app

Pre-Requisites

IAM Role

aws-opsworks-service-role

Policy name

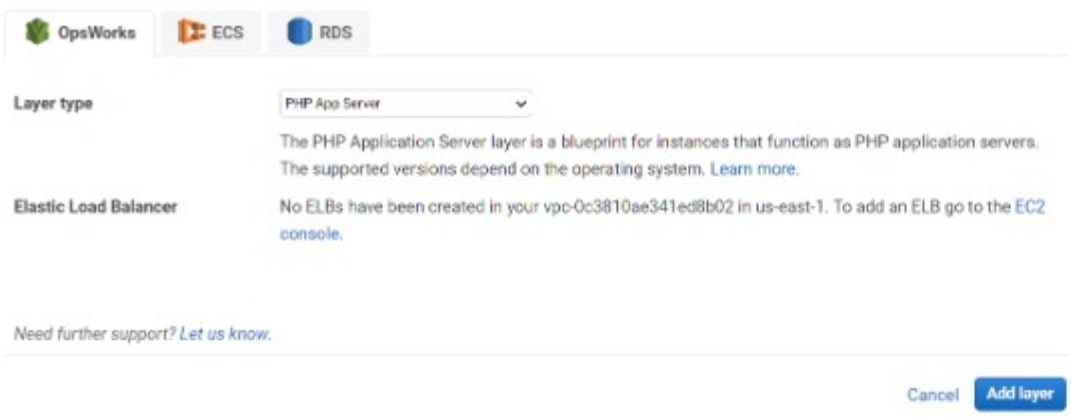
- AmazonRDSFullAccess
- AmazonEC2FullAccess
- IAMFullAccess
- ElasticLoadBalancingFullAccess
- CloudWatchActionsEC2Access

Trusted entities

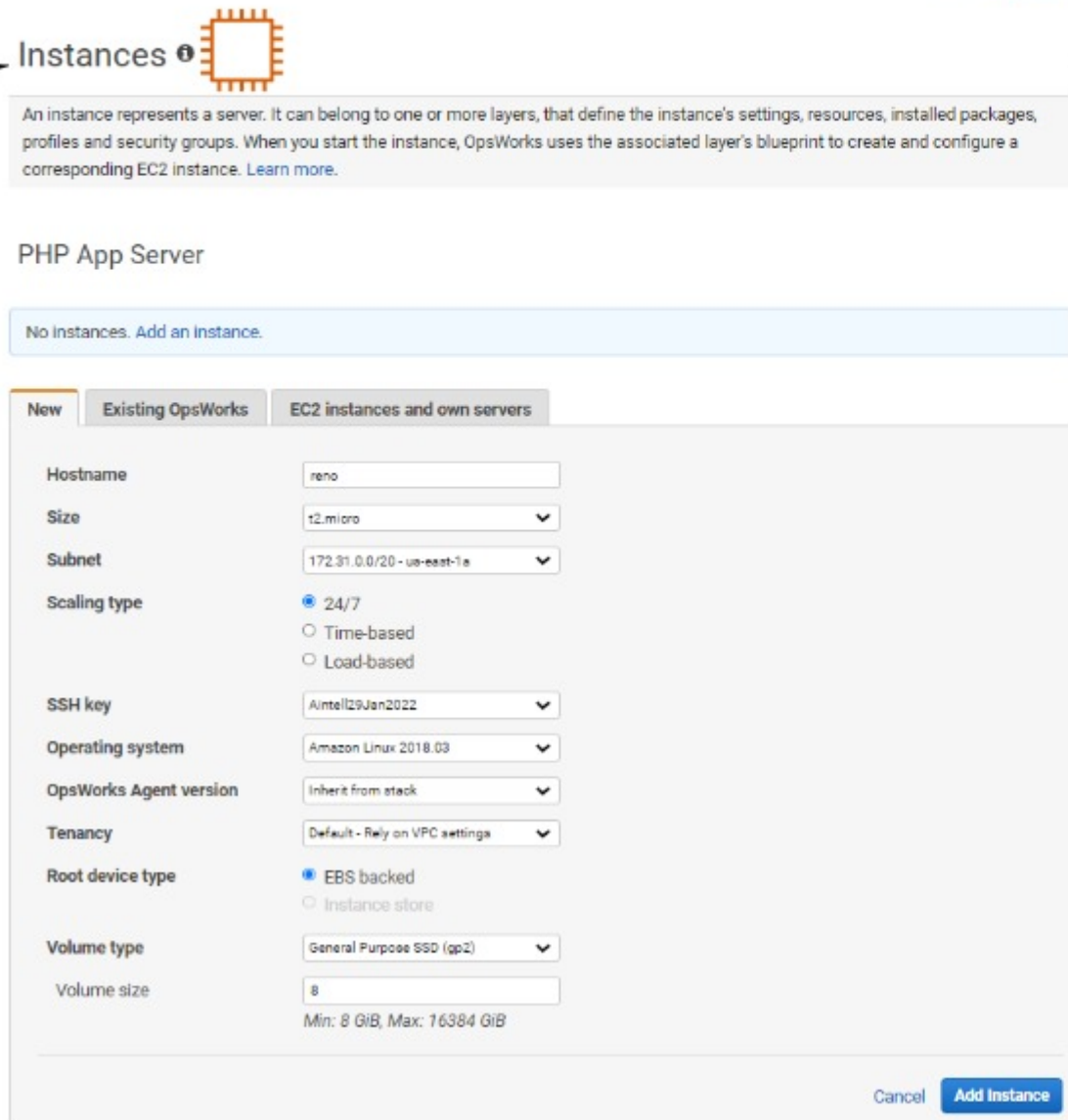
Entities that can assume this role under specified conditions.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "opsworks.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

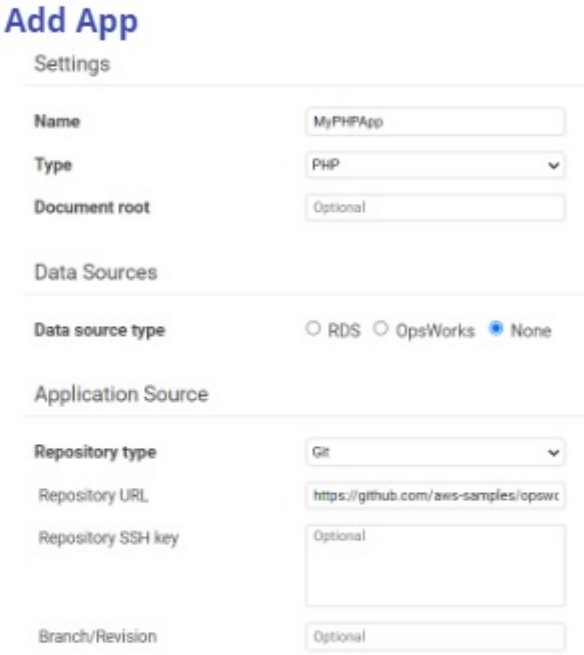
4. **Layers** — *A layer is a blueprint for a set of Amazon EC2 instances. It specifies the instance's settings, associated resources, installed packages, profiles, and security groups. You can also add recipes to lifecycle events of your instances, for example: to set up, deploy, configure your instances, or discover your resources.*



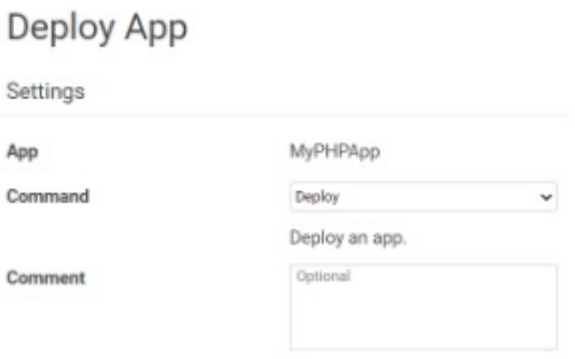
5. **Instances**



6. **Add App**



7. **Deploy App**



https://github.com/aws-samples/opsworks-linux-demo-cookbook-nodejs.git

README.md

AWS OpsWorks Demo Cookbooks

These are the cookbooks used for the OpsWorks demo for Linux.

Modifying and using these cookbooks in AWS OpsWorks

The following steps will help you get started with your own AWS OpsWorks custom application based on the demo app:

- Fork the repository and make modifications, add new cookbooks, etc.
- Run `berks package` to create a `cookbooks-*.tar.gz` file on your computer.
- Upload the generated `cookbooks-*.tar.gz` to an Amazon S3 bucket or a web server.
- On the AWS OpsWorks Stack settings toggle Use custom Chef cookbooks and set the Repository type and URL.
- On the OpsWorks Stack create a custom layer.
- Add `nodejs_demo` as recipe in `setup` or add your own custom recpies.
- Create an app, using e.g. https://github.com/awslabs/opsworks-windows-demo-nodejs.git as a source.
- Add and start an instance on the Layer that was created.
- Go to http://public IP or DNS name of your instance/

CloudWatch:

It's a monitoring Service

Types of Monitoring

Basic Monitoring

It's Free
Polls Data/Collects Data in every 5 Minutes
Works on limited metrics
5 GB data ingestion
5 GB of data storage

Detailed Monitoring/Enhanced Monitoring

Not Free
Chargeable
Charges are per instance/mo
Polls data/Fetches data in every one minute that can further be reduced.

Metric: can be known as sensors/parameters on which Cloudwatch is monitoring services.

Some services also used with Cloudwatch:

SNS (Simple Notification Service)

Cloudwatch is well integrated with EC2 Service such as Autoscaling.

CloudTrail: if cloudtrail logging is turned on, Cloudwatch writes log files to the S3 Bucket, that stored logs can be analyzed further with Athena or similar service.

CloudTrail also used to see/read AWS cloud events by users/roles/services.
By default you can see last 90 days events.

AWS IAM: can be managed: authentication and authorization

LAB :.

CloudWatch: DashBoard
The Alarm configuration, and can trigger some actions on Resources
Monitoring EC2 Instance using Cloudwatch metric to monitor Value.
Using Cloudwatch to create a bill alarm by enabling estimated Charges.