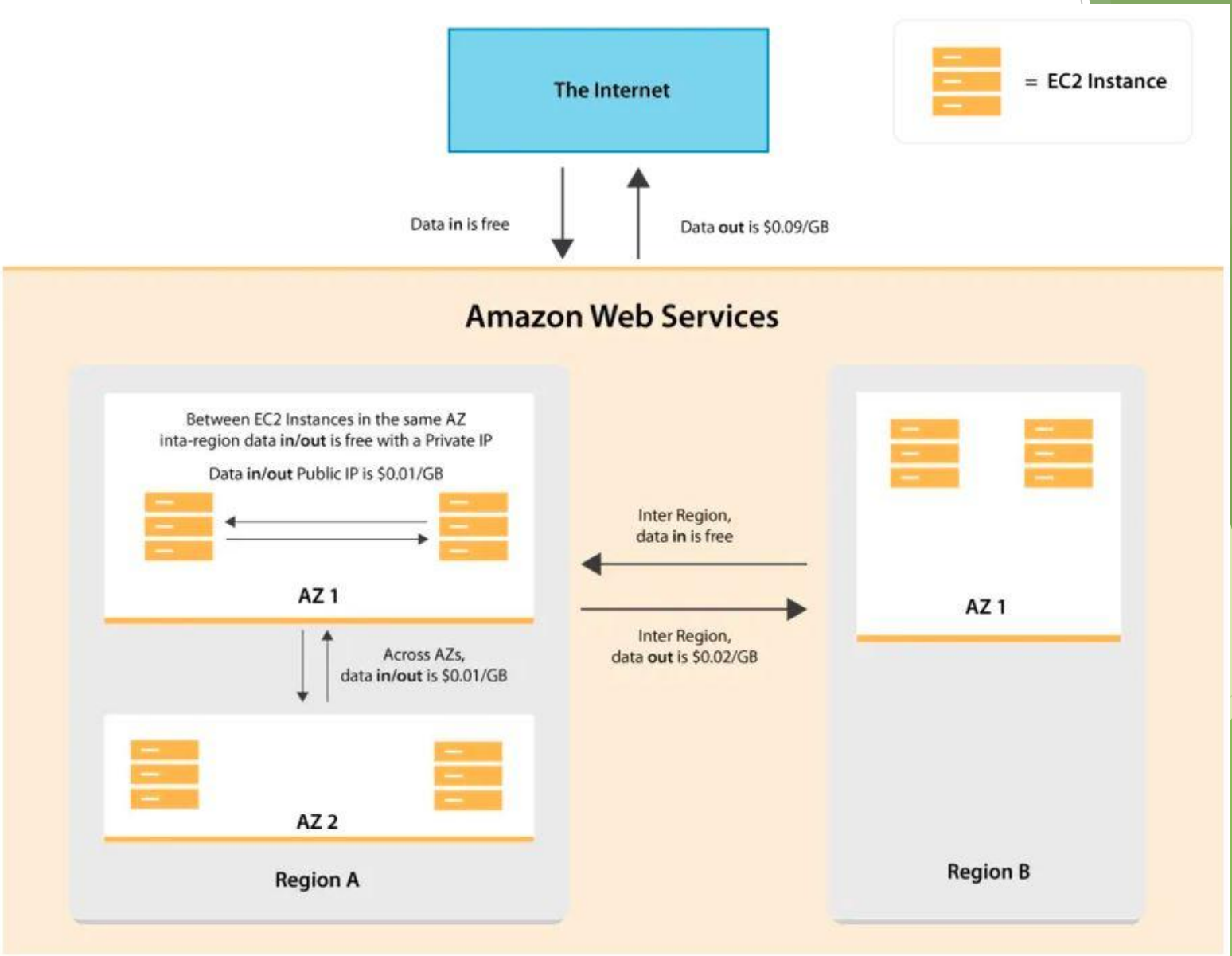
AWS DATA TRANSFER RATE



AWS DATA TRANSFER RATE

Service	Data transfer in	Data transfer to different AZ in region	Data transfer out to other regions	Data transfer out to internet	Data transfer out to CloudFront
Amazon EC2 (Includes EBS)		•	•	•	
Amazon EKS (Based off EC2 clusters)		•	•	•	
Amazon S3			•	•	
Amazon CloudFront				•	
Amazon RDS		•	•	•	
Amazon DynamoDB			•	•	
Amazon Aurora			•	•	
Amazon Glacier			•	•	
AWS Snowball			•	•	
Amazon CloudSearch				•	
Amazon SNS				•	
Amazon SQS				•	

Payment For AWS services

- Pay For What You Use(Pay As You Go)
- Reserve And Save Money
- Pay Less By Using More
- Free Usage Tier

Instance Classification

1. On-Demand Instances
2. Spot Instances
3. Reserved Instances
4. Scheduled reserved Instances
5. Dedicated Instances
6. Dedicated Host

On-Demand Instances

- Pay As You Go
- Pay For What You Use
- No Offer
- Fixed Rate By the Hour/Second
- No Long Term Commitment

Spot Instance

- Excess capacity/Unused Instances
- Drop Prices of ec2 On-Demand Instances
- Offer Upto 90% Off Compared To On-Demand instances
- Prices Will Go Up And Down (Updating Every 5 minutes)
- Request Needed

Working On Spot-Instance

- Requirement Submission Needed
- If Available,Get Instance With An Offer

  Ex:

Instance Type	Spot Price	Offer
C3.Large	.0068/hr	75%*

*Actual Price may be Different

- The Hourly Price For Spot Instance Is Called Spot Price
- Set Your Maximum Price/Hour Against Spot Price
- If Max Price Exceeds Spot Price And If Capacity Available,Amazon Fulfill Your Request

Spot Instance

- If Spot Price Exceeds Maximum Price,The Instance will be Terminated By Amazon
- Give 2 min Interruption Notice Before Termination, Charging For The Partial Hour
- Terminated By User, Full Hour Will Be Charged
- Two type Spot Request
  1. One Time Request
  2. Persistent Request

Spot Instance

Defined Duration

- 30% to 50% Reduction
- 1 To 6 Hour Workload
- Fixed Rate For The Spot Instance Until The Instance Terminated
- If Don’t Specify Max Price, Max Price Will Be The On-Demand price

Spot Instance

Strategy

- Running Application With Min Instances And If Opportunity Arises, Supplement Them With Spot Instances
- Run Spot Instance In a Specified Duration(Business Time)
- Good For Testing And Development Servers
- Not a Good Choice For Production Servers (Sensitive Workloads Or databases)

Reserved Instance

- Payment Done In advance
- Upto 72% discount
- Term commitment
    1. One year
    2. Three year
- Payment Options
    1. All upfront
    2. Partial Upfront
    3. No upfront

Reserved Instances

All Upfront

- Full Payment Is Made At The Start Of The Term
- One Single Payment
- No Other Cost Or Additional Hourly Charges

Partial Upfront

- A Portion Of The Cost Being Paid
- Remaining Hours In The Term Are Billed At a Discounted Hourly rate

Reserved Instances

No Upfront

- You Are Billed a Discounted Hourly Rate For Every Hour With In The Term

Offering Classes

1. Standard
2. Convertible

Standard

- Upto 75% Discount
- Customers Have The Flexibility To Change The Availability Zone, The Instance Size, And Networking

Reserved Instances

Convertible

- Upto 54% Discount
- Additional Flexibility To Use Different Instance Families, Operating Systems, Or Tenancies Over The Reserved Instance Term

Reserved Instance limits

- limit To Purchase The number Of Reserved Instances Per Month. For Each Region You Can Purchase 20 Regional Reserved Instances Per Month Plus An Additional 20 Zonal Reserved Instances Per Month For Each Availability Zone.

Scheduled Instances

- Schedule An Instance On a Monthly,Weekly Daily Basis
- Specify The Starting Date,Time And Duration
- For One Year Term Only
- Pay For The Scheduled Time Even If You Are Not Using It
- Good Choice For Workloads That Do Not Run Continuously, But Do Run On a Regular Schedule
- You Can Use Scheduled Instances For An Application That Runs During Business Hours Or For Batch Processing That Runs At The End Of The Week
- The minimum required utilization is 1,200 hours per year and You can purchase a Scheduled Instance up to three months in advance.

Dedicated Instances

- Using A Seperate Hardware That Will Be Used By a Single Customer
- it is physically isolated at the host hardware level from instances that belong to other AWS accounts
- Dedicated Instances Share Hardware With Other Instance From The Same AWS Account That Are Not Dedicated Instances
- Upon Restart Or Stop,It Will Change Its Hardware
- On-Demand,Reserved Instance 1&3 Year Standard And Convertible(70%),Spot Instance(90%)
- Hourly Pricing For Each Instance
- Regional Fee $2 Per Hour

- https://aws.amazon.com/ec2/pricing/dedicated-instances/#:~:text=Dedicated%20Instances%20are%20Amazon%20EC2,belong%20to%20other%20AWS%20accounts.

Dedicated Host

- Using A Dedicated Hardware For Your Own Usage
- It Is A Dedicated Physical Server
- Upon Restart And Stop, It Will Not Change Its Hardware
- Upto 70% Offer
- Dedicated Host Cost Between $1 To $2 Per Hour, whereas Dedicated Instances Cost Just Few Cents Per Hour
- Can Apply BYOL
  - https://aws.amazon.com/blogs/aws/now-available-ec2-dedicated-hosts/
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/dedicated-hosts-overview.html

Dedicated Host

Instance Placement Control

- You Can Place Or Launch Instances On To a Dedicated Host Or Amazon Will Launch For You
- Controlling Instance Placement Is a Flexibility For You

Visibility Of Socket And Physical Core

- You Have The Visibility Of Number Of Socket And Cores Being Used By a Particular Instance

Affinity

- It Allows You To Specify Which Dedicated Host Will Run An Instance After i Has Been Stopped Or Restarted
- It Gives You Additional Confidence That The Instance Is Running On The Same Physical Server

EC2 Bare Metal

- You ,As a User, Would Only Have Access To The Guest OS And The Management Interface Used To Create The VM. You Would Not Have Direct Access To Physical Hardware
- On The Other Hand With Bare Metal Server You Would Have Full Access To The Underlying Architecture
- Physically Dedicated Server
- Your Applications With Direct Access To The Processor And Memory Resources Of The Underlying Server.
- Some Specialised Workload Required Direct Access To The Bare Metal Infrastructure. Legacy Workload Not Supported In The Virtual Environment And Licensing Is Restricted In Tier 1 Business Applications

https://aws.amazon.com/blogs/aws/new-amazon-ec2-bare-metal-instances-with-direct-access-to-hardware/

EC2 Bare Metal Links

- https://aws.amazon.com/about-aws/whats-new/2019/02/introducing-five-new-amazon-ec2-bare-metal-instances/

- https://aws.amazon.com/ec2/instance-types/m5/

EC2(Elastic Compute Cloud)

EC2(Elastic Compute Cloud)

- One Of The Most Widely Used AWS Service,It Is A Server We Call It An Instance
- With In Minute We Obtain And Boot A New Server Instance
- If The Computing Environment Changes We Can Scale Up Or Scale Down Instance
- As Per The Ec2 Sla 99.999 Availability There Giving In Each Zone

Ami

- Amazon Machine Image
- It Uses Your Virtual Machine

EC2(Elastic Compute Cloud)

Documentation

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html

Lab

create a Amazon EC2 instance running Amazon Linux 2

- https://docs.aws.amazon.com/efs/latest/ug/gs-step-one-create-ec2-resources.html

EC2(KEY)

- Pem File For Windows ,Ppk (For Linux)
- Ec2 Uses Public Key Cryptography To Encrypt And Decrypt Login Information
- Can Have Up To 5,000 Key Pairs Per Region
- Amazon Ec2 Stores The Public Key Only, And You Store The Private Key(RSA)
- Anyone Who Possesses Your Private Key Can Decrypt Your Login Information, So It's Important That You Store Your Private Keys In A Secure Place
- Amazon Ec2 Doesn't Keep A Copy Of Your Private Key, There Is No Way To Recover A Private Key If You Lose It

EC2(KEY) Links

.

Manage user accounts on your Linux instance

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/managing-users.html

Windows Key

- https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2-key-pairs.html

Linux Key

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html#having-ec2-create-your-key-pair

EC2(Security Groups)

- security group acts as a virtual firewall for your instance to control incoming and outgoing traffic
- All Inbound Rules Are Blocked and All Outbound Rules Are Opened by Default
- When You Make A Change In Rules On A Security Group That Change Takes Effect Immediately
- You can add and Delete sg Rule(Inbound & Outbound)
- But If You Create An Inbound Rules, An Outbound Rule Created Automatically
- If You Allow Http Inbound, It's Automatically Allow Http Out As Well.So It's Stateful

EC2(Security Groups)

- You Cannot Actually Block An Individual Port Or Individual Ip Addresses Using Security Group
- When You Create A Security Group Everything Blocked By Default
- You Have To Specify Allow Rules, Deny Rules Not Permitted
- By default, you can apply up to five security groups to a an ec2 instance
- Actions-> Networking-> Change Security Group-> Add Another Security Group
- You Can Have Any Number Of Ec2 Instances In A Security

EC2(Security Groups)

https://docs.aws.amazon.com/vpc/latest/userguide/VPC_SecurityGroups.html

Documentation

- https://docs.aws.amazon.com/vpc/latest/userguide/VPC_SecurityGroups.html

Lab

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/working-with-security-groups.html#creating-security-group

Amazon EC2 security groups for Linux instances

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html

EC2(IP Addressing)

Ip Addresses

- Private And Public Addresses Are Used
- Can Be Used Ipv4 And Ipv6

Create EC2 Instances Steps

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html

AWS Storage Types

1. Elastic Block Storage(EBS)
2. Elastic File Storage(EFS)
3. Amazone File Storage(FSx)
4. Simple Storage Service(S3)

EBS(Elastic Block Storage)

- It Is Elastic Block Storage
- It Offers 99.999% Availability Of Data
- Automatically Replicated Within The Availability Zone
- It Uses Block Level Storage
- You Can Configure Multiple Volumes On The Same Instance
- But Each Volume Can Be Attached To Only One Instance At A Time
- We Can Create Filesystem On Top Of Each Volume
- Can Be Attached Like A Hard Drive

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-volume-types.html

EBS(Elastic Block Storage)

Two Types Of EBS

1. SSD Backed Storage
2. HDD Backed Storage

SSD Backed Storage

- For Transactional Workloads Such Us Database

Two Types Of SSD

1. Provisioned IOPS
2. General Purpose

EBS(SSD-Backed)

Provisioned IOPS

- High Performance And Latency Sensitive Transactional Workloads.

General Purpose

- This Balances Price And Performance For Variety Of Transactional Data

EBS(SSD-Backed)

	General Purpose SSD	Provisioned IOPS SSD	
Volume type	gp2	io2	io1
Durability	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)
Use cases	• Boot volumes • Low-latency interactive apps • Development and test environments	• Workloads that require sustained IOPS performance or more than 16,000 IOPS or 250 MiB/s of throughput per volume • I/O-intensive database workloads	
Volume size	1 GiB - 16 TiB	4 GiB - 16 TiB	
Max IOPS per volume (16 KiB I/O)	16,000 *	64,000 †	
Max throughput per volume	250 MiB/s *	1,000 MiB/s †	
Amazon EBS Multi-attach	Not supported	Not Supported	Supported

EBS(HDD-Backed)

- Throughput Intensive Workloads Such As Mapreduce(Hadoop) And Log Processing

Hadoop

- Manages Big Data Processing And Storage For Big Data Applications Runs In Clustered Environment

MAP

- Set Of Data And Convert It Into Another Set Of Data And Stores In A HDFS (Hadoop File System)

Reduce

- It Will Process The Converted Data And Produce A New Set Of Data

EBS(HDD-Backed)

Three Types Of EBS

1. Throughput Optimized
2. Cold HDD
3. Magnetic

Throughput Optimized

- For Frequently Accessed ,Throughput Intensive Workload

Cold HDD

- Lowest Cost,Less Frequently Accessed

Magnetic(Previous Generation Volume)

- Data Infrequently Accessed
- 16GB-1TB
- Max IOPS/Volume-40-200
- Max Throughput -800MB/Second

EBS(HDD-Backed)

Throughput

- Can Be Affected By IOPS and Packet Size

Throughput=IOPS*[Block Size]

=3000*8KB

=24000 KB/Second

=24MB/Second

Database Throughput=Transaction/Second

Bandwidth

- Total Possible Speed Of Data Along The Network

	Solid State Drives (SSD)		Hard Disk Drives (HDD)	
Volume Type	EBS Provisioned IOPS SSD (io1)	EBS General Purpose SSD (gp2)*	Throughput Optimized HDD (st1)	Cold HDD (sc1)
Short Description	Highest performance SSD volume designed for latency-sensitive transactional workloads	General Purpose SSD volume that balances price performance for a wide variety of transactional workloads	Low cost HDD volume designed for frequently accessed, throughput intensive workloads	Lowest cost HDD volume designed for less frequently accessed workloads
Use Cases	I/O-intensive NoSQL and relational databases	Boot volumes, low-latency interactive apps, dev & test	Big data, data warehouses, log processing	Colder data requiring fewer scans per day
API Name	io1	gp2	st1	sc1
Volume Size	4 GB - 16 TB	1 GB - 16 TB	500 GB - 16 TB	500 GB - 16 TB
Max IOPS**/Volume	64,000	16,000	500	250
Max Throughput***/Volume	1,000 MB/s	250 MB/s	500 MB/s	250 MB/s
Max IOPS/Instance	80,000	80,000	80,000	80,000
Max Throughput/Instance	2,375 MB/s	2,375 MB/s	2,375 MB/s	2,375 MB/s
Price	$0.125/GB-month $0.065/provisioned IOPS	$0.10/GB-month	$0.045/GB-month	$0.025/GB-month
Dominant Performance Attribute	IOPS	IOPS	MB/s	MB/s

LAB

- Create an Amazon EBS volume
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-creating-volume.html
- Attach an Amazon EBS volume to an instance
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-attaching-volume.html
- Make an Amazon EBS volume available for use on Linux
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-using-volumes.html
- Make an Amazon EBS volume available for use on Windows
  - https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ebs-using-volumes.html
- Extend a Linux file system after resizing a volume
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/recognize-expanded-volume-linux.html

LAB

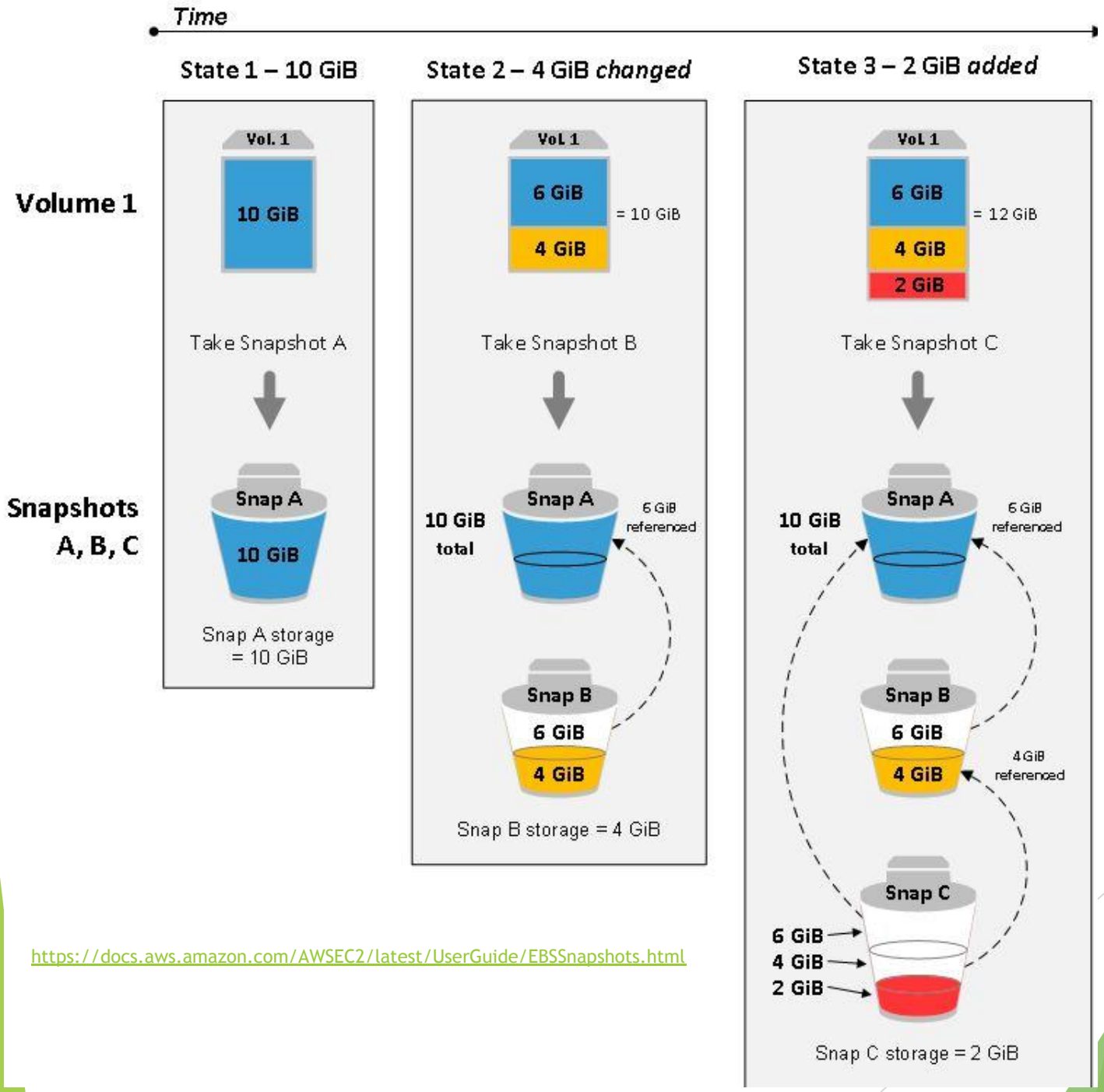
- Modify an EBS volume using Elastic Volumes
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/requesting-ebs-volume-modifications.html
- attach an available EBS volume to one or more of your instances that is in the same Availability Zone as the volume.
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-attaching-volume.html
- blog
  - https://aws.amazon.com/blogs/storage/clustered-storage-simplified-gfs2-on-amazon-ebs-multi-attach-enabled-volumes/

Snapshot(EBS)

- Point In Time Copy Of Your An Amazon Ebs Volume
- Backup Copy Of An Ebs Volume Is Saved In S3
- Photograph Of A Disc
- Snapshots Are Incremental
- You Can Create A Snapshot From Ebs Volume
- Snapshot Of Encrypted Volumes Are Automatically Encrypted
- You Can Create An Ebs Volume Using Snapshot
- Volume That You Created From Encrypted Snapshot Is Also Encrypted

Snapshot(EBS)

- You Can Create An Encrypted Volume Using Unencrypted Snapshot
- When You Create A Snapshot Of Root Volume, It Is Better To Stop The Instance
- But You Can Take A Snapshot Of A Running Instance
- If Hibernation Is Enabled,You Cannot Take A Snapshot
- Ami Is The Snapshot Of An Ebs Volume With An Operating System On It
- You Can Create An Ami And Volume Using A Snapshot
- Deleting A Volume Does Not Automatically Delete The Associated Snapshots



https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSSnapshots.html

Amazon Data Lifecycle Manager(EBS)

- You can use Amazon Data Lifecycle Manager to automate the creation, retention, and deletion of EBS snapshots and EBS-backed AMIs.
- When you automate snapshot and AMI management, it helps you to:
- Protect valuable data by enforcing a regular backup schedule.
- Create standardized AMIs that can be refreshed at regular intervals
- Retain backups as required by auditors or internal compliance.
- Reduce storage costs by deleting outdated backups.
- Create disaster recovery backup policies that back up data to isolated accounts.

Amazon Data Lifecycle Manager LAB

Automate snapshot lifecycles

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/snapshot-ami-policy.html

Documents

- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/snapshot-lifecycle.html

Snapshot(EBS)

To Move An Ec2 Volume From One Az To Another

- Take A Snapshot ,
- Create An Ami From It
- And Use That Ami To Launch An Ec2 Instance In New Availability Zone

To Move An Ec2 Volume From One Region To Another

- Take A Snapshot Of That Volume,
- Create An Ami From It,
- Copy That Ami From One Region To Another
- And Use That Copied Ami To Launch An Ec2 Instance In The New Region

LAB

- Create Amazon EBS snapshots
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-creating-snapshot.html
- Copy an Amazon EBS snapshot
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-copy-snapshot.html
- Share an Amazon EBS snapshot
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-modifying-snapshot-permissions.html
- Create An Ami From Instance
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html
- Create a Linux AMI from a snapshot
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html
- Amazon EBS pricing
  - https://aws.amazon.com/ebs/pricing/

Snapshot(EBS)

Two Types Of EBS

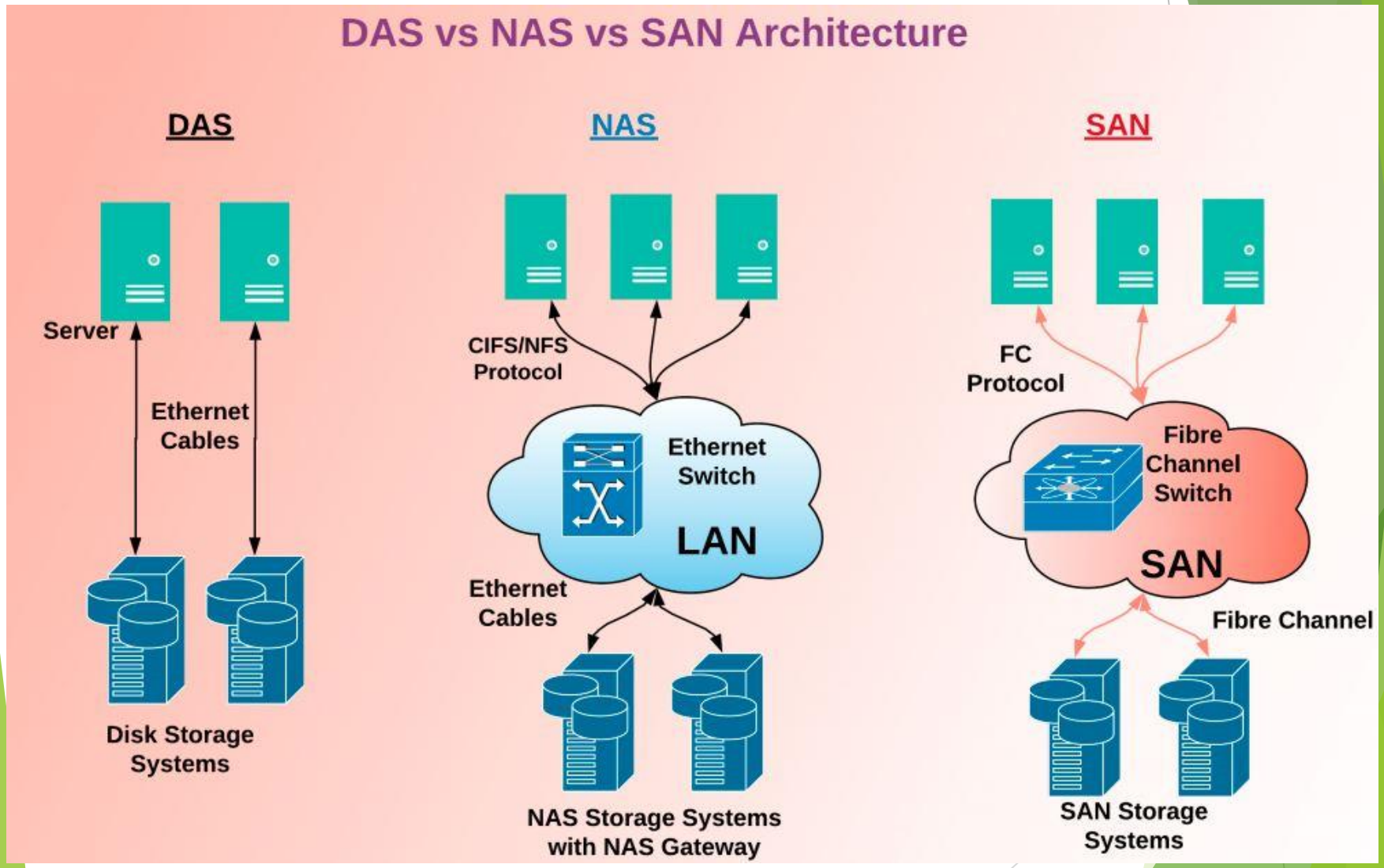
1.Instance Volume Snapshot

- Multiple Volume
- For All Ebs Volume Is Attached To A Single Instance

2.Single Volume

- Can Take Snapshot Of Single Volume Alone

Instance Store Volume



DAS(Direct Attached Storage)

- DAS Is A Block Device From A Disk Which Is Physically [Directly] Attached To The Host Machine.
- You Must Place A Filesystem Upon It Before It Can Be Used.
- Simply Save Files As Usual Like Any Other Operating System
- Storage Is Connected To One Computer And Not Accessible To Other Systems
- Communicating Using Protocols Like IDE, SCSI, SATA, etc.

SAN(Storage Area Network)

- San Is A Block Device Which Is Delivered Over The Network.
- Like Das You Must Still Place A Filesystem Upon It Before Using It.(San File System)
- Communication Using Protocols Like Fibre Channel, iSCSI, FoE, etc

.

NAS(Network Attached Storage)

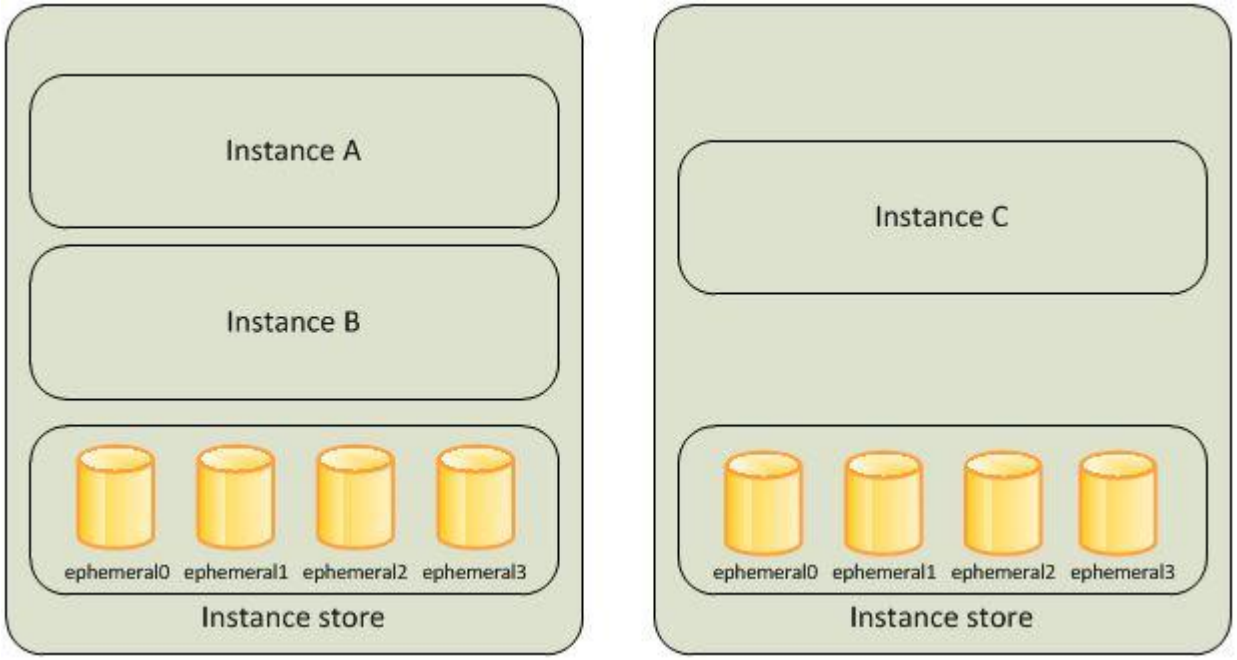
- Nas Is A File Level Storage System Delivered Over The Network.
- It Is Ready To Mount And Use
- Communicating Using Protocols Like NFS, CIFS/SMB etc.

Instance Store Volume

- It Is A Disk That Is Physically Attached To Virtualization Host.So It Can Give Massive Iops At Low Latency
- Ebs Storage Is Storage On A Remote Network Connected to San And May Be Competing For I/O With Thousands Of Other Instances
- An Instance Store Volume Provides Temporary Block-level Storage For Your Instance.So It Is Ephemeral Storage
- If The Underlying Host Fails,Stops, Terminations Due To Hardware Failures ,You Will Lose All The Data
- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/InstanceStorage.html

Instance Store Volume

-



Instance Store Volume

- You Can't add an Instance Store Volume After You Launch The Instance
- It Cannot Be Stopped
- It Can Only Reboot Or Terminate
- It Is Useful For Applications That Focus On Processing Data Rather Than Storing Data
- It Is Perfect For Temporary Data That Changes Often Or If Needed ,Replicate The Data
- Instance-store Is Over 5x Faster Than EBS-SSD For Uncached Reads
- Instance-store And EBS-SSD Are Equivalent For Cached Reads
- Instance-store Is Over 10x Faster Than EBS-SSD For Writes

Enhanced Networking

Provides Higher I/O Performance And Lower Cpu Utilization When Compared To Traditional Virtualized Network Interfaces. Enhanced Networking Provides Higher Bandwidth, Higher Packet Per Second (Pps) Performance, And Consistently Lower Inter-instance Latencies. There Is No Additional Charge For Using Enhanced Networking

Instance store volume LAB

- Create an AMI from an instance store-backed Amazon Linux instance
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-instance-store.html
- Add instance store volumes to an instance
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/add-instance-store-volumes.html#adding-instance-storage-instance
- Add instance store volumes to an AMI
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/add-instance-store-volumes.html

AMI(Amazon Machine Image)

What Happens When You Launch An Instance

- When You Launch An Instance, The Root Device Volume Contains A Boot Image That Is Used To Boot The Instance.We Can Launch An Ec2 Instance By Using That Image

What Is Ami

- Ami Is An Image That Is Used To Boot An Ec2 Instance
- It Can Contains Os(That Is Configured In A Specified Way), Set Of Applications And Services
- An Ami Contains All The Information Necessary To Start Up And Run The Software
- An Ami Is A Template For The Root Device Volume Of An Instance

AMI(Amazon Machine Image)

- Ami With Encrypted Volume Cannot Be Made Public
- Ami Is Regional Resource And Can Share It To Another Region

You Can Select Ami Based On

- Regions
- Operating System
- Architecture
- Launch Permission
- Storage For The Root Device

AMI(Amazon Machine Image)

Regions

- Mumbai,Oregon,Virginia Etc

Operating System

- Linux,Windows

Architecture

- 32bit Or 64bit)

Launch Permission

- Public -grand Permission To All AWS Users
- Explicit -grant Permission To Specific AWS Users
- Implicit -grand Implicit Launch Permission To An Ami(Since You Created It)

AMI(Amazon Machine Image)

Shared Ami

- Shared Ami-developers Change The Ami And They The Will Allow To Use Others
- Use This Ami At Your Own Risk
- Amazon Is Not Responsible For It

Paid Ami

- You Can Purchase An AMI From a Developer
- It Is Available In The Marketplace

AMI LAB

- Create a Linux AMI from an instance

&

- Create a Linux AMI from a snapshot
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html

- move my EC2 instance to another subnet, Availability Zone, or VPC
  - https://aws.amazon.com/premiumsupport/knowledge-center/move-ec2-instance/

- create and copy an Amazon Machine Image (AMI) from one AWS Region to another
  - https://aws.amazon.com/premiumsupport/knowledge-center/copy-ami-region/

AMI(Amazon Machine Image)

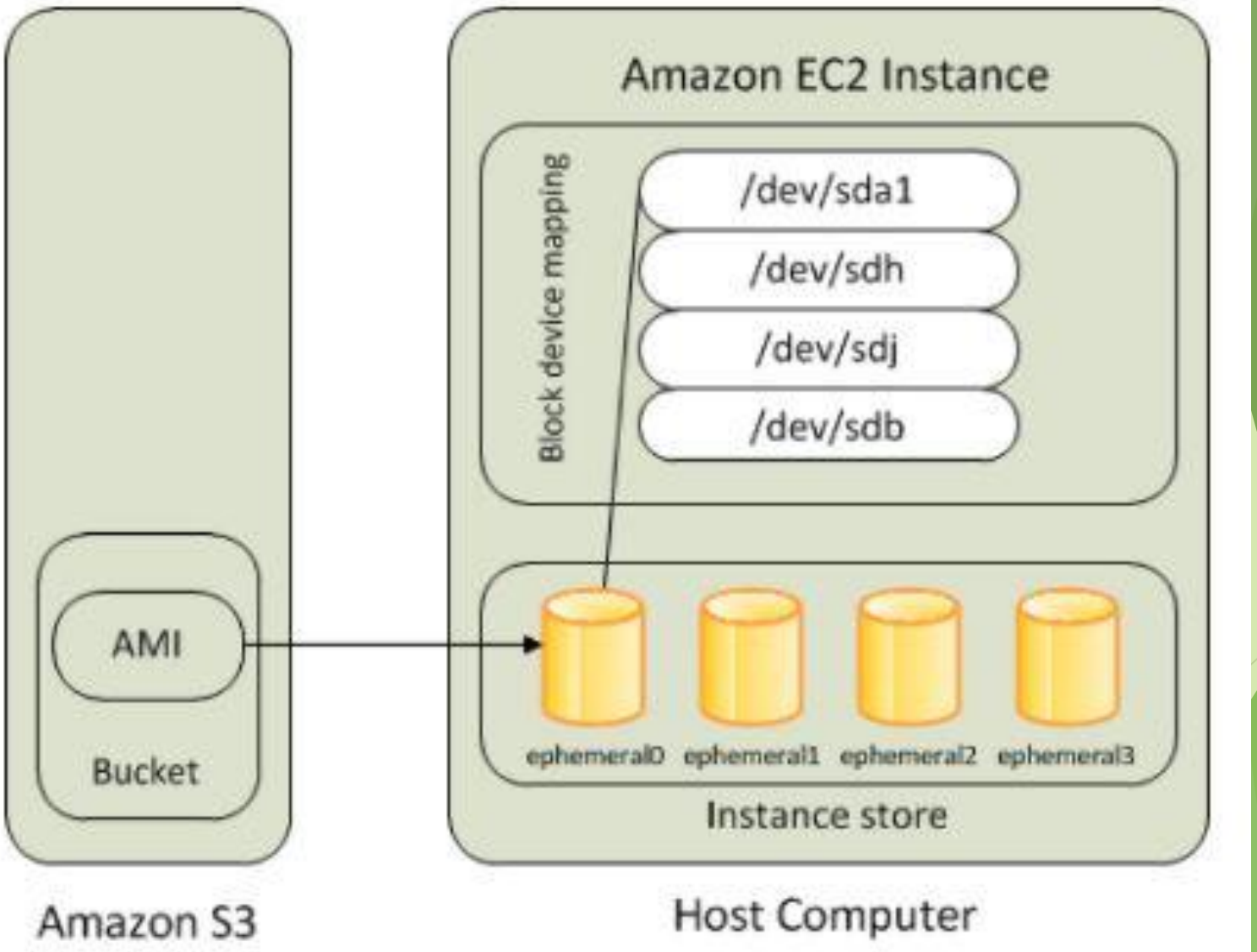
Storage For The Root Device

1. Instance Store
2. EBS Volume

Instance Store

- AMI Is Stored In The Instance Store
- When The Instance Terminate, All The Data Will Be Lost
- It Is An Ephemeral Storage
- If The Underlying Host Fails,All The Data Will Be Lost
- https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/RootDeviceStorage.html

Instance Store Volume



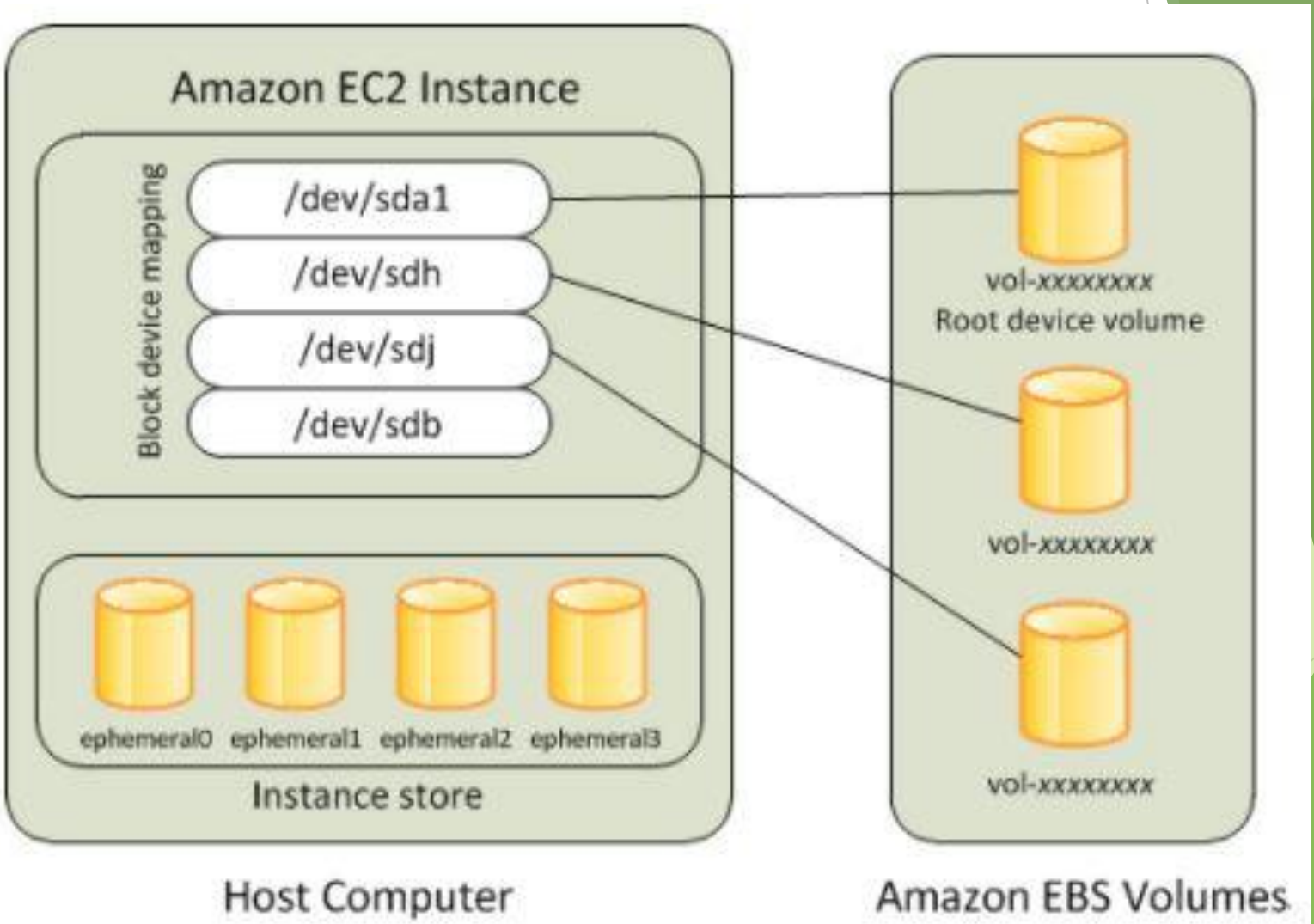
AMI(Amazon Machine Image)

Storage For The Root Device

EBS Volume

- AMI Is Stored In The EBS Volume
- Which Means The Root Device Of The Instance Which Have The AMI Created From A Snapshot Of EBS
- By Default ,Root Volume Is Deleted And Attached Volume Will Not Be Deleted

EBS Volume


Amazon EC2 Instance
Block device mapping
/dev/sda1
/dev/sdh
/dev/sdj
/dev/sdb
ephemeral0
ephemeral1
ephemeral2
ephemeral3
Instance store
Host Computer
vol-xxxxxxxx
Root device volume
vol-xxxxxxxx
vol-xxxxxxxx
Amazon EBS Volumes

EBS Volume vs Instance Store

Characteristic	Amazon EBS-backed AMI	Amazon instance store-backed AMI
Boot time for an instance	Usually less than 1 minute	Usually less than 5 minutes
Size limit for a root device	16 TiB	10 GiB
Root device volume	Amazon EBS volume	Instance store volume
Data persistence	By default, the root volume is deleted when the instance terminates.* Data on any other Amazon EBS volumes persists after instance termination by default.	Data on any instance store volumes persists only during the life of the instance.
Modifications	The instance type, kernel, RAM disk, and user data can be changed while the instance is stopped.	Instance attributes are fixed for the life of an instance.
Charges	You're charged for instance usage, Amazon EBS volume usage, and storing your AMI as an Amazon EBS snapshot.	You're charged for instance usage and storing your AMI in Amazon S3.
AMI creation/bundling	Uses a single command/call	Requires installation and use of AMI tools
Stopped state	Can be in a stopped state. Even when the instance is stopped and not running, the root volume is persisted in Amazon EBS	Cannot be in stopped state; instances are running or terminated

https://aws.amazon.com/blogs/aws/amazon-elastic-file-system-shared-file-storage-for-amazon-ec2/

Elastic File System(EFS)

- It Is Is A Storage Service Of AWS
- It Is Growing And Shrinking While Adding or Removing Files
- It Is For Ec2 Instances And On-premise Servers
- Thousands Of Ec2 Instances Can Concurrently Connect To EFS At A Time(EBS Can Connect To One Ec2 Instance)
- It Uses NFS-v4.(File Sharing Protocol)
- It Is Like NAS(Data Organised Two Directories And Subdirectories)
- It Is Supporting Current Generation Linux Machines
- It Can Scale Up To Petabytes(1000TB=1 PetaBytes)

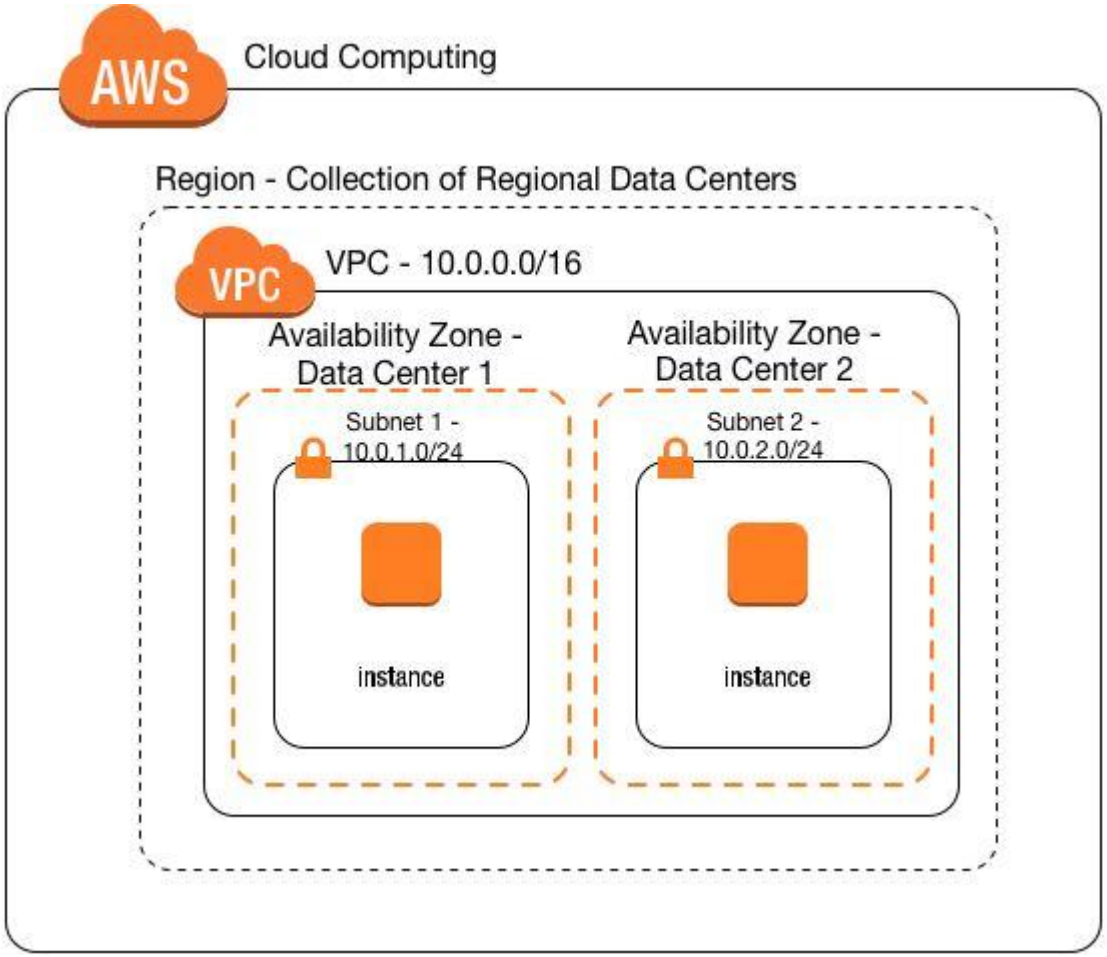
Elastic File System(EFS)

- Amazon EFS Is A Regional Service Storing Data Within And Across Multiple Availability Zones (AZs) For High Availability And Durability
- Only One VPC Can Mount To EFS At A Time
- It Support Cross Region Ec2 Connection And On-premise Server Connection
- For Cross-region Connection,It Uses Inter Region VPC Peering
- For On-premise Connection,It Uses Vpn Connection Or AWS Direct Connect

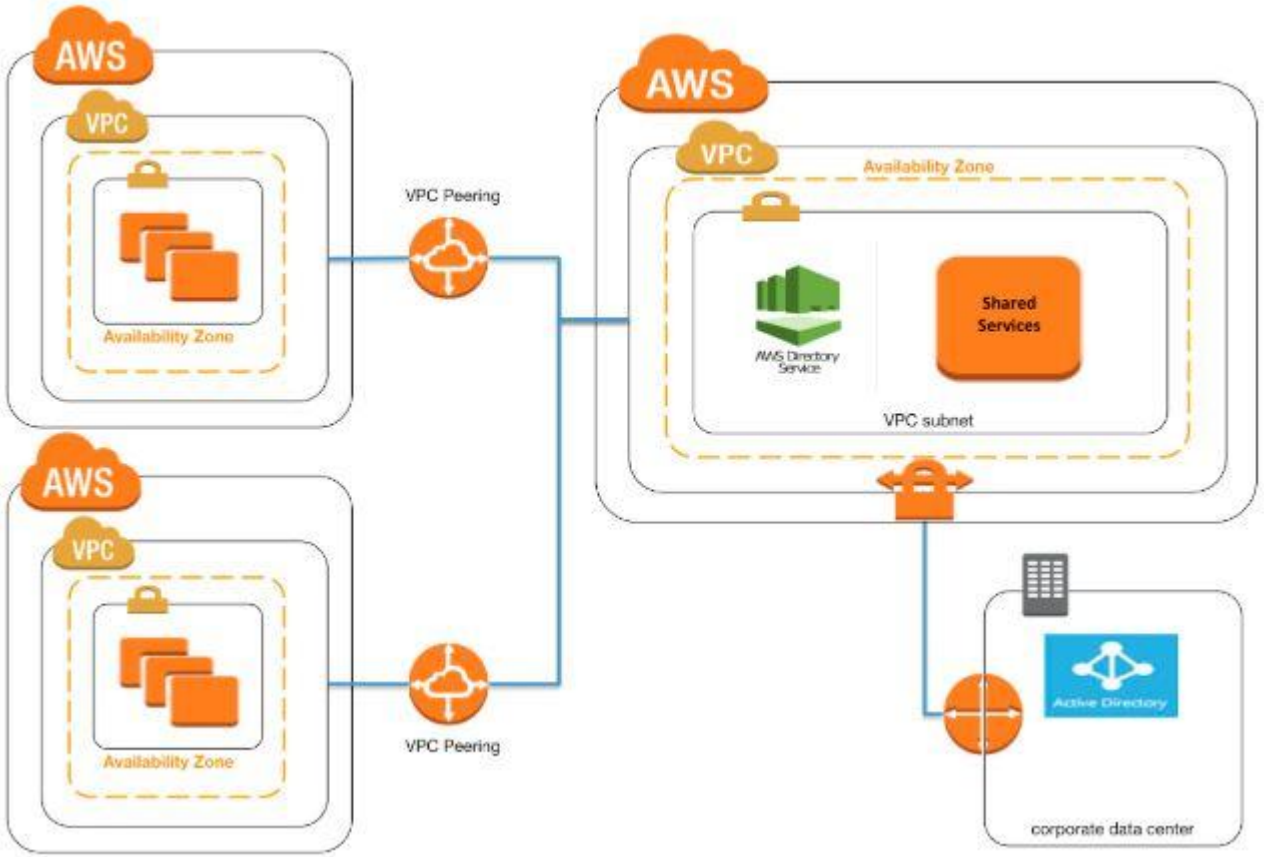
Use Case

- You Can Move Data Between On Premise And EFS.Then You Can Backup On Premise Data To EFS Or Migrate Data To EFS.So You Can Assign Workload To AWS

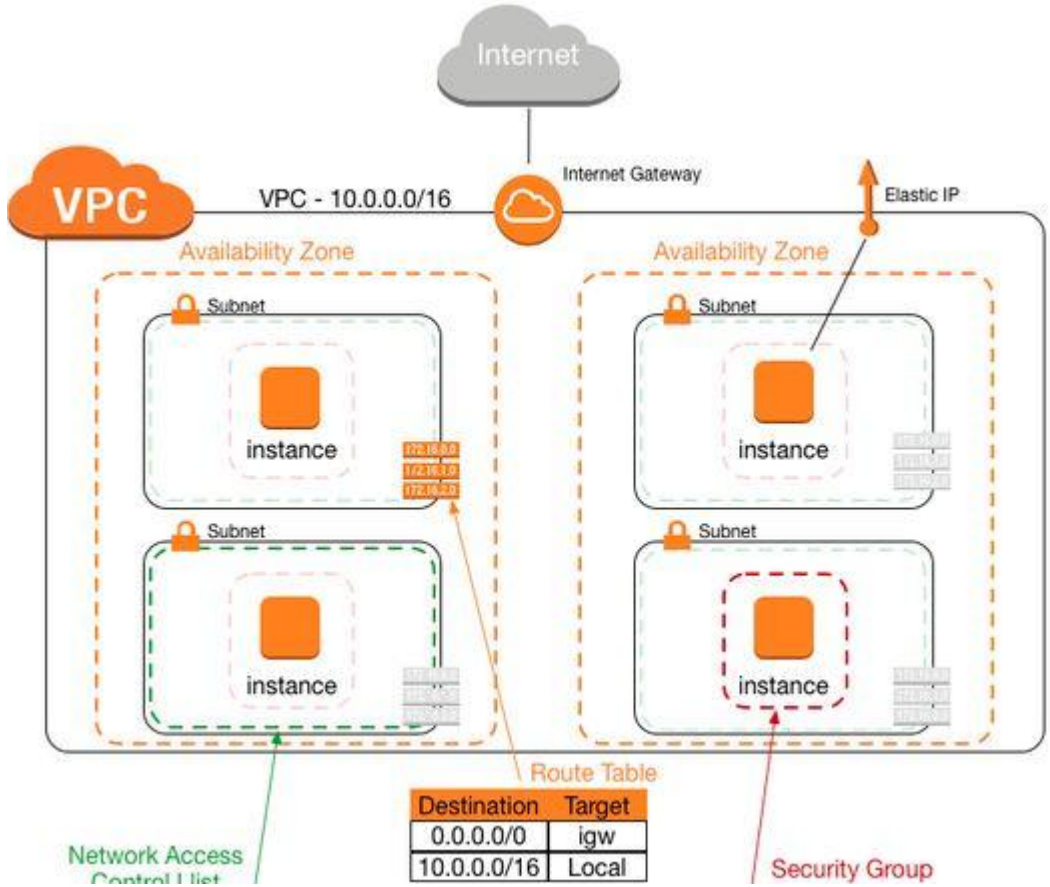
Elastic File System(EFS)



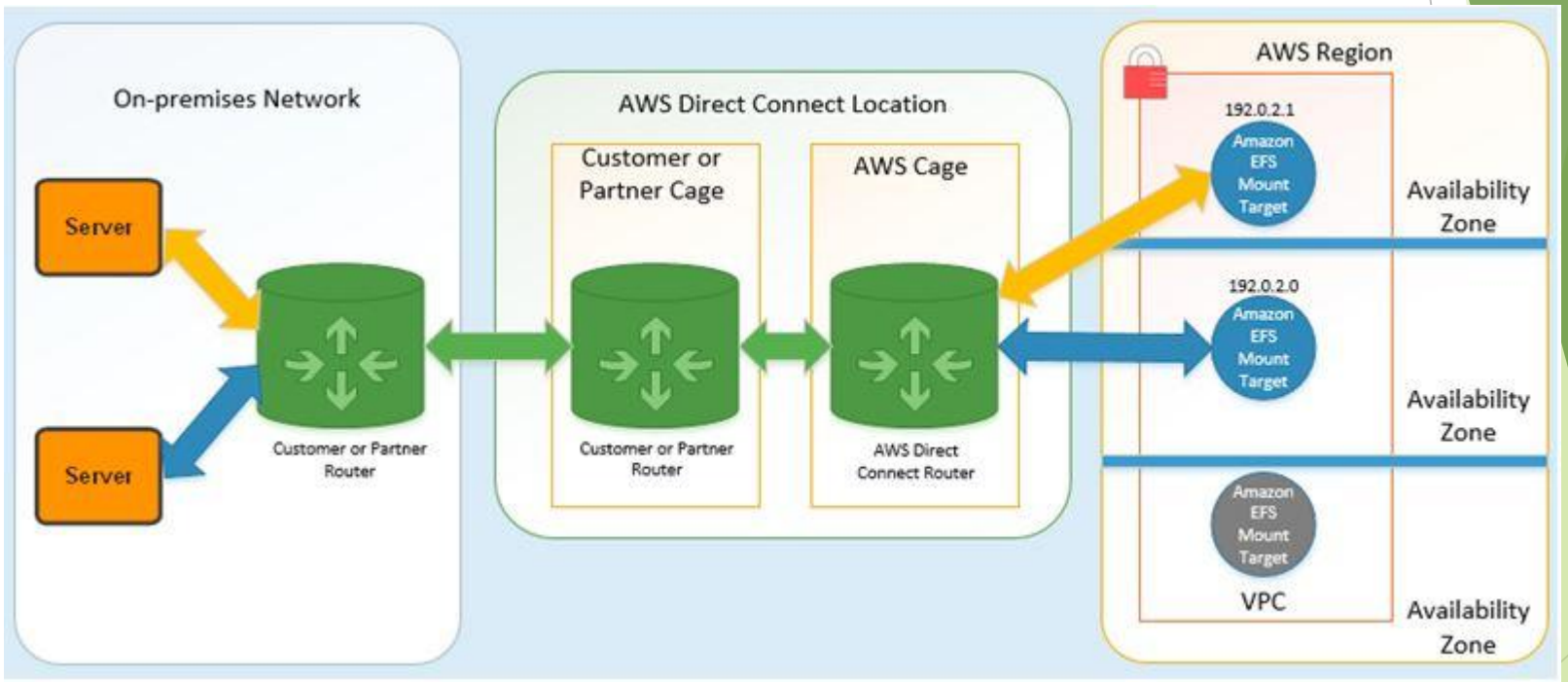
Elastic File System(EFS)



Elastic File System(EFS)



Elastic File System(EFS)



Elastic File System(EFS)

- Big Data Analytics, Media Processing Workload, Web Hosting, Home Directories(Sometimes Many Users Need To Access A Common Data Or File)
- It Is Used By SAS Applications
- It Can Be Used For Disaster Recovery

  By Using Inter-region VPC Peering,You Can Connect Ec2 Instances In One Region To Another Region
- Means, It Allow You To Copy Frequently Used Data Across Region

Cost

- No Additional Cost For This.You Will Be Billed For Direct Connect, VPC Peering And VPN Connection And Associated Data Transfer

Elastic File System(EFS)

Two Type Storage Class

1. EFS Standard
2. EFS-1A

EFS Standard

- Active Work Load Data Stored
- Pay For The Amount Of Data Stored

EFS-1A

- Infrequent Workloads
- For Each Access And Pay Less For Storage Compare To Standard
- You Can Set Life Cycle Policy Ie 7,14,30,60,90 Days Since Last Access (Move From Standard To EFS-1A)

Elastic File System(EFS)

Two Type Of Throughput Mode

1. Burst Throughput
2. Provisioned Throughput

Burst

- Bursting Depends Upon File Size
- Commonly 100MB/S Up To 1TB
- If The File Size Increase 10tb Then The Throughput Will Be 1000MB/S
- If 95% Inactive ,Then 5% Can Burst.So The Burst Credit Is 5.
- If The File System Running Out Of Credit Then You Can Change To Provisioned Mode

Elastic File System(EFS)

Provisioned

- You Can Specify That The Throughput You Need In MB/S
- Price Is High

Two Type Of Performance Mode

1. General Purpose
2. Max I/O Performance Mode

General Purpose

- Used For Latency Sensitive Workloads Like Content Management System,Home Directories Web Servers

Max I/O Performance

- Higher Level Of Aggregate Throughput & Operations Per Second

Elastic File System(EFS)

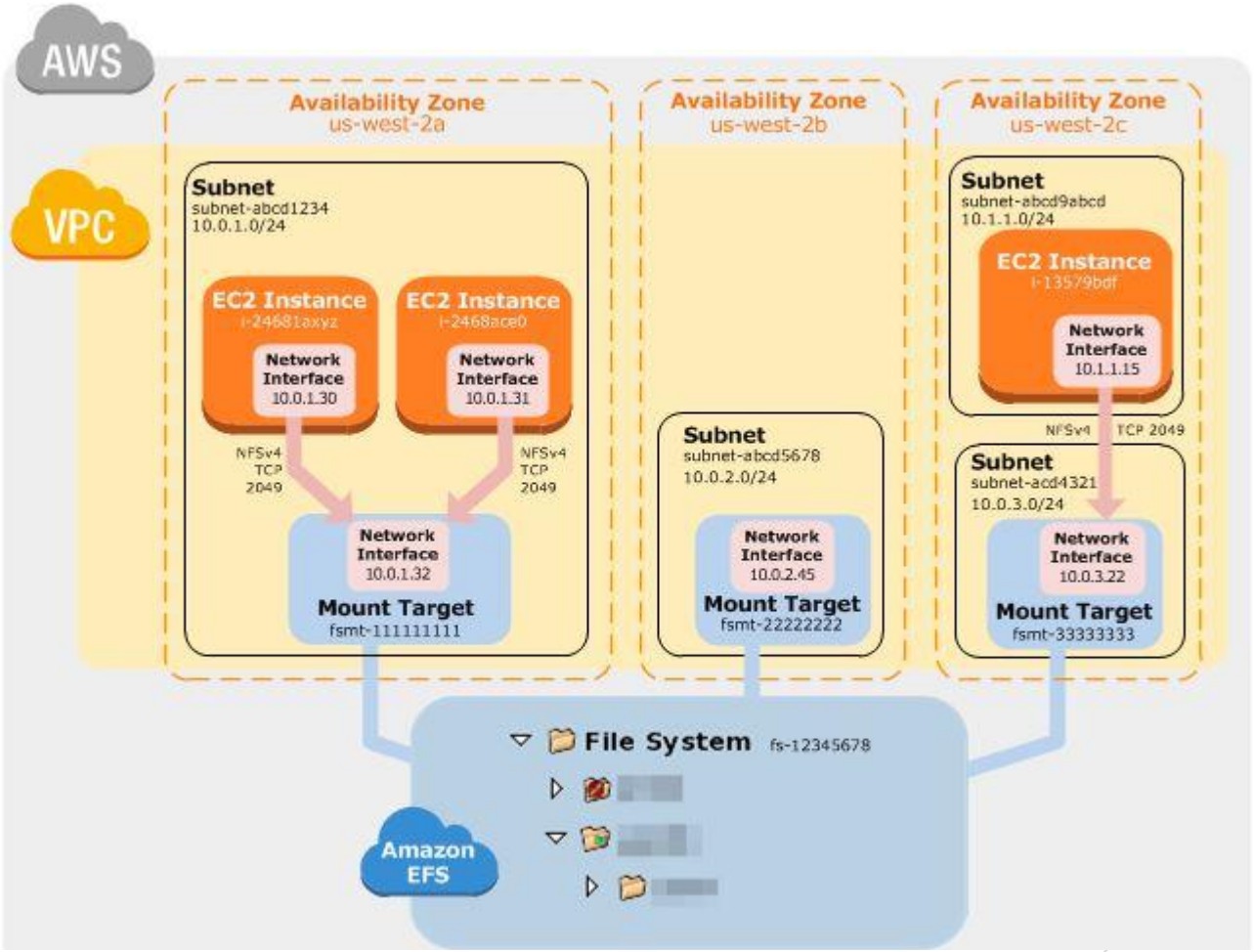
- Slightly Higher Level Of Latencies For File Operations
- Used For Paralyzed Applications And Workload Like Big Data Analytics,Media Processing Etc
- It Is Optimized For Tens,Hundreds,Thousands Of Ec2 Instances Are Accessing The File System

EFS

- Some Ami's Using NFS Client To Mount The EFS File System
- Read After Write Consistency

Elastic File System(EFS)

Creating a Mount Target



Elastic File System(EFS)

Creating a Mount Target

- Choose A Subnet To Create A Mount Target
- You Can Create A Mount Target In Each Availability Zone
- If ,Multiple Subnet In An Availability Zone,No Need To Create A Mount Target In The Same Subnet(Where Your Ec2 Instance Launched),It Can Be Any Subnet In The AZ
- Select Automatic IP Address For The Mount Target.Amazon Will Select One Available Ip Address From The Subnet Pool
- Ec2 System Use The DNS To Point EFS By Checking It In The DNS server

EFS Lab

Create EFS FIle System

https://docs.aws.amazon.com/efs/latest/ug/gs-step-two-create-efs-resources.html

EC2 Security Group Port To Be Opened

http and ssh

Go And Edit The Security Group You Created, go to Inbound Rule And Add NFS And Select Source As EC2 Security Group

EFS Lab

Commands To Enter On First Ec2

yum update -y

yum install httpd -y

service httpd start

chkconfig httpd on

yum install -y amazon-efs-utils

echo “<html><body><h1>this is my website</h1></body></html>” > index.html

cat index.html

- Spin Up Second Ec2 Install All Packages And Mount To Created Efs Volume

Simple Storage Services(S3)

- It Is An Object Storage
- Maintain Three Copies in a region
- No Limit For Storage
- Create A Bucket To Store Object
- You Can Upload Any Number Of Object To The Bucket
- Object Size Is 0-5tb
- Choose A Region Geographically Close To You To Create A Bucket
- Bucket Name Is Globally Unique

Simple Storage Services(S3)

- By Using Url You Can Access A Bucket
- You Can Create A Bucket By Using Console Or By Using AWS Sdk
- Using Rest Api You Can Create And Access Bucket [PUT,POST, DEL, GET Etc]
- Access Bucket Using Virtual Hosted Style And Path Style
- By Default ,100 Bucket You Can Create For Each Account
- You Can Increase Bucket Limit To 1000 By Submitting A Service Request

Simple Storage Services(S3)

- It is Highly Scalable,Durable And Secure
- Durability 99.99999999(11--9's)

  [If You Store 100 Billion Objects In S3,You Will Lose One One Object)
- Availability Is 99.99 Of The Time
- Used For Web Servers,Backup And Recovery,Mobile Applications,Enterprise Applications,Disaster Recovery,Big Data,Videos And Images
- S3 Archive -it Is Not Actively Used For Long Time And Stored For Years

S3 Storage Class

S3 Storage Class

1. S3-Standard
2. S3-Intelligent Tier
3. S3-Standard-1A
4. S3-One Zone-1A
5. S3-Glacier
6. S3-Glacier Deep Archive

Simple Storage Services(S3)

S3 Standard Class

- General Purpose Storage For Frequently Accessed Data
- Minimum Duration Is Not Applicable

S3 Intelligent Tier

- Automatically Move To Cost Effective Tier For Cost Optimization
- It Uses Two Access Tier
    1. Frequent Access Tier
    2. Infrequent Access Tier

Simple Storage Services(S3)

1. Frequent Access Tier -it Is Designed For Frequent Access
1. Infrequent Access Tier -designed For Infrequent Access It Is Lower Cost Tier

- Amazon S3 Monitor The Access Pattern Of The Object.
- If The Object Is Not Accessed 30 Consecutive Days,It Will Be Moved To Infrequent Access Tier.
- If The Data Being Accessed From Infrequent Trier,It Will Move Back To Frequent Tier
- There Is No Fees For Retrieving The Data

Simple Storage Services(S3)

- But Charged Small Amount Monthly For Monitoring And Automation Fee Per Object
- You Can Also Store Data From Intelligent To Glacier

S3 -Standard-1A(Infrequent Access)

- Minimum 30 Days Retention Period
- Data Accessed Less Frequently
- Requires Rapid Access When Needed

Simple Storage Services(S3)

S3 -One Zone-1A

- Minimum Retention Period 30 Days
- Less Frequently Accessed
- Other Storage Class Uses Minimum Of 3 Availability Zone.But This Stored Data In One AZ
- 20% Less Than Standard-1A

Simple Storage Services(S3)

S3 -Glacier

- Not Used Data For Long Time And Can Be Stored For Years
- Retrieval Time Few Minutes To Hours
- Minimum 90 Days Retention Period
- Low Cost

S3-Deep Archive

- Lowest Cost Among All Storage Class
- Data Accessed One Or Twice A Year
- Stored For 7-10 Years Or Longer
- Retrieval Time:With In 12 Hours

Simple Storage Services(S3)

Object Attributes In S3

- Key [Name Of The Object]
- Value [Data]
- Version Id [For Versioning]-Version1, Version2 Etc
- Meta Data

Security

- Access List For Bucket
  - It Defines Which AWS Accounts Or Groups Are Granted Access To The Bucket And The Type Of Access

    [READ, WRITE, READ_ACP, WRITE_ACP,FULL_CONTROL]acp=acl

Simple Storage Services(S3)

- Bucket Policy For Bucket
  - Granting Permissions To Multiple Accounts With Added Conditions
  - Granting Read-only Permission To An Anonymous User
  - Restricting Access To Specific IP Addresses
  - Adding A Bucket Policy To Require MFA
  - Granting Cross-account Permissions To Upload Objects
  - Ensuring The Bucket Owner Has Full Control Granting Permissions For Amazon S3 Inventory And AWS S3 Analytics
  - Example Bucket Policies For Vpc Endpoints For Amazon S3

S3-Data Consistency

Strong Read After Write Consistency

- Amazon S3 provides strong read-after-write consistency for PUTs and DELETEs of objects in your Amazon S3 bucket in all AWS Regions

S3 Object Level consistency

- A process writes a new object to Amazon S3 and immediately lists keys within its bucket. The new object will appear in the list.
- A process replaces an existing object and immediately tries to read it. Amazon S3 will return the new data.

S3-Data Consistency

- A process deletes an existing object and immediately tries to read it. Amazon S3 will not return any data as the object has been deleted.
- A process deletes an existing object and immediately lists keys within its bucket. The object will not appear in the listing.

Links

- https://aws.amazon.com/s3/consistency/
- https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html#ConsistencyModel
- https://aws.amazon.com/blogs/aws/amazon-s3-update-strong-read-after-write-consistency/

S3-Bucket Consistency

S3 Eventual Consistency

- Bucket configurations have an eventual consistency model
- If you delete a bucket and immediately list all buckets, the deleted bucket might still appear in the list.
- If you enable versioning on a bucket for the first time, it might take a short amount of time for the change to be fully propagated.
- We recommend that you wait for 15 minutes after enabling versioning before issuing write operations (PUT or DELETE) on objects in the bucket.

S3-Versioning

- Stores All Versions Of Object Uploaded
- Once Enabled, Cannot Be Disabled
- It Acts Like A Backup Tool
- MFA Delete-prevent Accidental Deletion Of Any Versioned S3 Object
- Can Apply Life Cycle Policy

S3-Lifecycle Policy

- used to store object cost effectively throughout their lifecycle
  - https://docs.aws.amazon.com/AmazonS3/latest/dev/lifecycle-transition-general-considerations.html
- So It Automate Moving your Object Between Different Storage Tier
  - https://docs.aws.amazon.com/AmazonS3/latest/dev/lifecycle-transition-general-considerations.html
- Can Be Used With Versioning
- Can Be Used With Current And Previous Version

Cross Region Replication

- You Can Configure S3 For Automatically Replicate Across Different AWS Region By Using Amazon Cross Region Replication

Cross Region Replication

When to Use CRR

- Meet Compliance Requirements – Although S3 Stores Your Data Across Multiple Availability Zones By Default, Compliance Requirements Might Dictate That You Store Data At Even Greater Distances. Cross-region Replication Allows You To Replicate Data Between Distant Aws Regions To Satisfy These Requirements.
- Minimize Latency – If Your Customers Are In Two Geographic Locations, You Can Minimize Latency In Accessing Objects By Maintaining Object Copies In Aws Regions That Are Geographically Closer To Your Users

Cross Region Replication

When to Use CRR

- Increase Operational Efficiency — If You Have Compute Clusters In Two Different Aws Regions That Analyze The Same Set Of Objects, You Might Choose To Maintain Object Copies In Those Regions

Same Region Replication

- Bucket Replication Within The Same Region

Same Region Replication

When to Use SRR

- Aggregate Logs Into A Single Bucket — If You Store Logs In Multiple Buckets Or Across Multiple Accounts, You Can Easily Replicate Logs Into A Single, In-region Bucket. This Allows For Simpler Processing Of Logs In A Single Location
- Configure Live Replication Between Production And Test Accounts — If You Or Your Customers Have Production And Test Accounts That Use The Same Data, You Can Replicate Objects Between Those Multiple Accounts, While Maintaining Object Metadata, By Implementing SRR Rules

Same Region Replication

- Abide By Data Sovereignty Laws — You Might Be Required To Store Multiple Copies Of Your Data In Separate Aws Accounts Within A Certain Region. Same-region Replication Can Help You Automatically Replicate Critical Data When Compliance Regulations Don't Allow The Data To Leave Your Country

S3-Replication

1. When Configured,New Object Uploaded To S3 Automatically Replicated To Destination Bucket
2. Allowed Only Versioning Enabled Bucket
3. Replicate In Your Account Or Bucket In Another AWS Account
4. By default Delete Marker Will Not Be Replicated,but can configure it to replicate https://aws.amazon.com/blogs/storage/managing-delete-marker-replication-in-amazon-s3/
5. Existing Files Will Not Be Replicated
6. Deleting Individual File Will Not Be Replicated for non-versioned bucket

S3 LAB

Create A Bucket

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/create-bucket-overview.html

Uploading Object To Bucket

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/upload-objects.html

Accessing Bucket

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/access-bucket-intro.html

Deleting A Bucket

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/empty-bucket.html
- https://docs.aws.amazon.com/AmazonS3/latest/userguide/delete-bucket.html

S3 LAB

S3 Versioning and LifeCycle Policy

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/how-to-set-lifecycle-configuration-intro.html

Configuring S3 Replication

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/replication-walkthrough1.html

Quick Lab To Create,Access And Delete A Bucket

- https://docs.aws.amazon.com/quickstarts/latest/s3backup/step-1-create-bucket.html

S3-Object Lock

- You Can Only Enable Object Lock For New Buckets. If You Want To Turn On Object Lock For An Existing Bucket, Contact AWS Support
- Object Lock Works Only In Versioned Buckets
- When You Create A Bucket With Object Lock Enabled, Versioning Also To Be Enabled
- Once You Create A Bucket With Object Lock Enabled, You Can't Disable Object Lock Or Suspend Versioning For The Bucket.
- Working In Two Modes -Governance Mode & Compliance Mode

S3-Object Lock

- S3 Not Supporting Object Locking For Concurrent Updates. If Two PUT Requests Are Simultaneously Made To The Same Key, The Request With The Latest Timestamp Wins
- If This Is An Issue, You Will Need To Build An Object-locking Mechanism Into Your Application
- S3 Object Lock Feature Is Different From Actual Object Lock
  - https://aws.amazon.com/blogs/storage/protecting-data-with-amazon-s3-object-lock/
- With S3 Object Lock, You Can Store Objects Using A Write-once-read-many (WORM) Model And Prevent An Object From Being Deleted Or Overwritten of existing content For A Fixed Amount Of Time Or Indefinitely

S3-Governance Mode

- Users Cannot Overwrite Or Delete An Object Version Or Alter Its Lock Settings Unless They Have Special Permissions
- With Governance Mode,You Protect Object Against Being Deleted By Most Users
- But You Can Still Grant Some Users Permission To Alter The Retention Settings Or Delete The Object If Necessary

Compliance Mode

- A Protected Object Version Cannot Be Overwritten Or Deleted By Any User Including The Root User For The Duration Of Retention Period
- Its Retention Period Cannot Be Changed

Retention Period

- It Protect An Object Version For A Fixed Amount Of Time
- When You Place A Retention Period On An Object Version, Amazon S3 Stores A Timestamp In The Object Version's Metadata To Indicate When The Retention Period Expires
- After The Retention Period Expires,The Object Version Can Be Overwritten Or Deleted Unless You Also Placed A Legal Hold On The Object Version

Legal Hold

- S3 Object Lock Also Enables You To Place A Legal Hold On An Object Version
- Like A Retention Period,A Legal Hold Prevents An Object Version From Being Overwritten Or Deleted
- But A Legal Hold Does Not Have A Retention Period And Remains In Effect Until Removed
- Legal Hold Can Be Freely Placed And Removed By Any User Who Has The S3:PutObjectLegalHold Permission

S3-Object Lock Replication

In April 2019, S3 Object Lock added support for Cross-Region Replication (CRR). This means that in addition to locking objects, you can now configure your S3 buckets to enable automatic, asynchronous copying of locked objects, and associated metadata, to an S3 bucket in a different AWS Region. Often, if your data is important enough to use a retention date, it is important enough to replicate it to another AWS region.

Glacier Vault Lock

- S3 Glacier Vault Lock allows you to easily deploy and enforce compliance controls for individual S3 Glacier vaults with a vault lock policy.
- You can specify controls such as “write once read many” (WORM) in a vault lock policy and lock the policy from future edits.
- Once locked, the policy can no longer be changed

S3 Object Lock Lab

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/object-lock-console.html

S3 Object Lock Document Reference

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/object-lock.html

S3-CORS(Cross-Origin Resource Sharing

- It Defines The Client Web Applications Loaded In One Domain Can Interact With Resources In A Different Domain By Using Cors Support
- You Can Build Web Applications With Amazon S3 And Allow Cross-origin, Access To Your Amazon S3 Resources

1. You Are Hosting A Website In An Amazon S3 Bucket Named Website1
2. Now You Want To Use Javascript File On The Web Pages That Are Stored In Another Bucket Named Website2
3. To Accept Request For Accessing(Authenticated Put And Get Request)File From Bucket1,Need To Enable Cors Configuration In Bucket 2

LAB

- S3-CLI
- AWS CLI Documentation
  - https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-welcome.html

- Installing latest version of the AWS CLI
  - https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html

S3 CLI commnads

- https://docs.aws.amazon.com/cli/latest/userguide/cli-services-s3-commands.html
- https://docs.aws.amazon.com/cli/latest/reference/s3/
- EX:

  aws s3 cp file1.txt s3://BucketName

Advanced S3 Api CLI Commands

- https://docs.aws.amazon.com/cli/latest/userguide/cli-services-s3-commands.html
- https://docs.aws.amazon.com/cli/latest/reference/s3api/
- EX:

  aws s3api delete-objects --bucket BucketName --delete file://hi.json

File hi.json Contents Saved current Directory

```
   {
  "Objects": [
    {
      "Key": "file1.txt.txt",
      "VersionId": "KnEtWljGHRL4muxIwgclJp525MzTayEF"
    }
  ],
  "Quiet": false
}
```

S3-Encryption

Two types

- In-Transit
- At-Rest

In-Transit

- It uses SSL/TLS Encryption

At-Rest(Server Side)

- Server-side Encryption Is The Encryption Of Data At Its Destination By The Application Or Service That Receives It
- Amazon S3 Encrypts Your Data At The Object Level As It Writes It To Disks In Its Data Centers And Decrypts It For You When You Access It

AWS-KMS

- AWS Key Management Service (KMS) Is To Create And Manage Cryptographic Keys And Control Their Use Across A Wide Range Of AWS Services And In Your Applications.
- AWS KMS Is A Secure And Resilient Service That Uses Hardware Security Modules (That Have Been Validated Under FIPS 140-2), Are being used to Protect Your Keys.
- AWS KMS Is Integrated With AWS Cloudtrail To Provide You With Logs Of All Key Usage To Help Meet Your Regulatory And Compliance Needs
- https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html
- https://docs.aws.amazon.com/crypto/latest/userguide/concepts-algorithms.html

S3-Server Side Encryption(SSE)

Four Methods Of Encrypting Object In S3

- SSE-S3
- SSE-KMS
- SSE-C
- Client Side Encryption

S3-Server Side Encryption(SSE)

Concepts

- https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html
-

- S3 Managed Key - SSE-S3
  - Amazon Will Manage The Key
  - Encryption And Decryption By Amazon
  - Amazon S3 Encrypts An Object Before Saving It To Disk And Decrypts It When You Download The Object
  - Amazon S3 Encrypts Each Object With A Unique Key
  - Amazon S3 will manage the data Key and master encryption keys
  - As An Additional Safeguard, It Encrypts The Key Itself With A Master Key That It Rotates Regularly
  - It Uses 256-bit Advanced Encryption Standard (AES-256), To Encrypt Your Data

SSE-S3 Lab

SSE-S3 using the S3 console

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/specifying-s3-encryption.html

Developing with Amazon S3 using the AWS SDKs, and explorers

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingAWSSDK.html

Documentation

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingEncryption.html

S3-At-Rest(SSE-KMS)

- Amazon S3 supports only symmetric encryption KMS keys, and not asymmetric KMS keys
- Public Key of asymmetric key pair, which you can export for use outside of AWS
- generate data keys that you can use outside of AWS KMS.

Above Document link

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingKMSEncryption.html

S3-At-Rest(SSE-KMS)

- SSE-KMS
  - AWS Key Management Service (KMS)Can Be Used To Manage The Encryption Key
  - AWS KMS Is Integrated With AWS Cloudtrail,A Service That Provides A Record Of Actions Performed By A User, Role, Or An AWS Service In AWS KMS
  - Object Is Encrypted In Server Side
  - SSE-KMS Requires That AWS Manage The Data Key But You or AWS Manage The Customer Master Key
  - There Are Additional Charges For Using AWS KMS CMKs

S3-At-Rest(SSE-KMS)

Note

- AWS KMS is replacing the term customer master key (CMK) with AWS KMS key and KMS key. The concept has not changed

Link For Above Documentation

- https://docs.aws.amazon.com/kms/latest/cryptographic-details/kms-keys.html

Bucket Key

- using AWS KMS (SSE-KMS), you can configure your bucket to use S3 Bucket Keys for SSE-KMS.
- Using a bucket-level key for SSE-KMS can reduce your AWS KMS request costs by up to 99 percent by decreasing the request traffic from Amazon S3 to AWS KMS.
- to use S3 Bucket Keys for SSE-KMS on new objects, AWS KMS generates a bucket-level key that is used to create unique data keys for objects in the bucket.

Bucket Key

- This bucket key is used for a time-limited period within Amazon S3, further reducing the need for Amazon S3 to make requests to AWS KMS to complete encryption operations

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingKMSEncryption.html
- https://docs.aws.amazon.com/AmazonS3/latest/userguide/bucket-key.html
- https://docs.aws.amazon.com/AmazonS3/latest/userguide/configuring-bucket-key.html

S3-At-Rest(SSE-KMS)

Customer Master Key(CMK)

- KMS CMK Cannot Accept More Than 4 KB (4096 Bytes) Of Data For Encryption
- CMK Can Use To Encrypt Small Amounts Of Data, Such As A Password Or RSA Key
- They Are Not Designed To Encrypt Application Data
- Three Types-customer Managed CMK,AWS Managed CMK AWS Owned CMK
- Two Types Of CMK Key-symmetric And Asymmetric

Customer Master Key(CMK)

Symmetric CMK

- Represents AES 256-bit Key
- Same Key Is Used For Encryption And Decryption

Asymmetric CMK

- Asymmetric CMK Represents An RSA Key Pair(2048,3072,4096) That Is Used For Encryption And Decryption Or Signing And Verification
- An Elliptic Curve (ECC) Key Pair That Is Used For Signing And Verification
- AWS KMS Will Not Leave Symmetric CMKs And The Private Keys Of Asymmetric CMKs Unencrypted

S3-At-Rest(SSE-KMS)

AWS Owned CMK

- AWS Owned CMKs Are A Collection Of CMKs That An AWS Service Owns And Manages It
- It Is Used In Multiple AWS Accounts.
- Although AWS Owned CMKs Are Not In Your AWS Account, An AWS Service Can Use Its AWS Owned CMKs To Protect The Resources In Your Account.
- You Are Not Charged A Monthly Fee Or Usage Fee For AWS Owned CMKs

S3-At-Rest(SSE-KMS)

- SSE-KMS

Two Types

- Customer Managed CMKs(Customer Master Key)
- AWS Managed CMKs (Customer Master Key)

Customer Managed CMK

- You Create, Own, And Manage The Key in your account. You Have Full Control Over These CMKs
- The Master Keys Are Stored In AWS KMS Is Called Customer Master Key(CMK)
- The CMK Includes Metadata(The Key ID, Creation Date, Description) And Key State

Customer Managed CMK

- The CMK Also Contains The Key Material Used To Encrypt And Decrypt Data
- By Default, AWS KMS Creates The Key Material For A CMK. You Cannot Extract, Export, View, Or Manage This Key Material. Also, You Cannot Delete This Key Material.Instead,You Must Delete The CMK
- You Can Import Your Own Key Material Into A CMK
- Each Customer Master Key (CMK) That You Create In AWS Key Management Service (KMS) Costs $1/Month Until You Delete It
- https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_job-functions.html

AWS Managed CMK

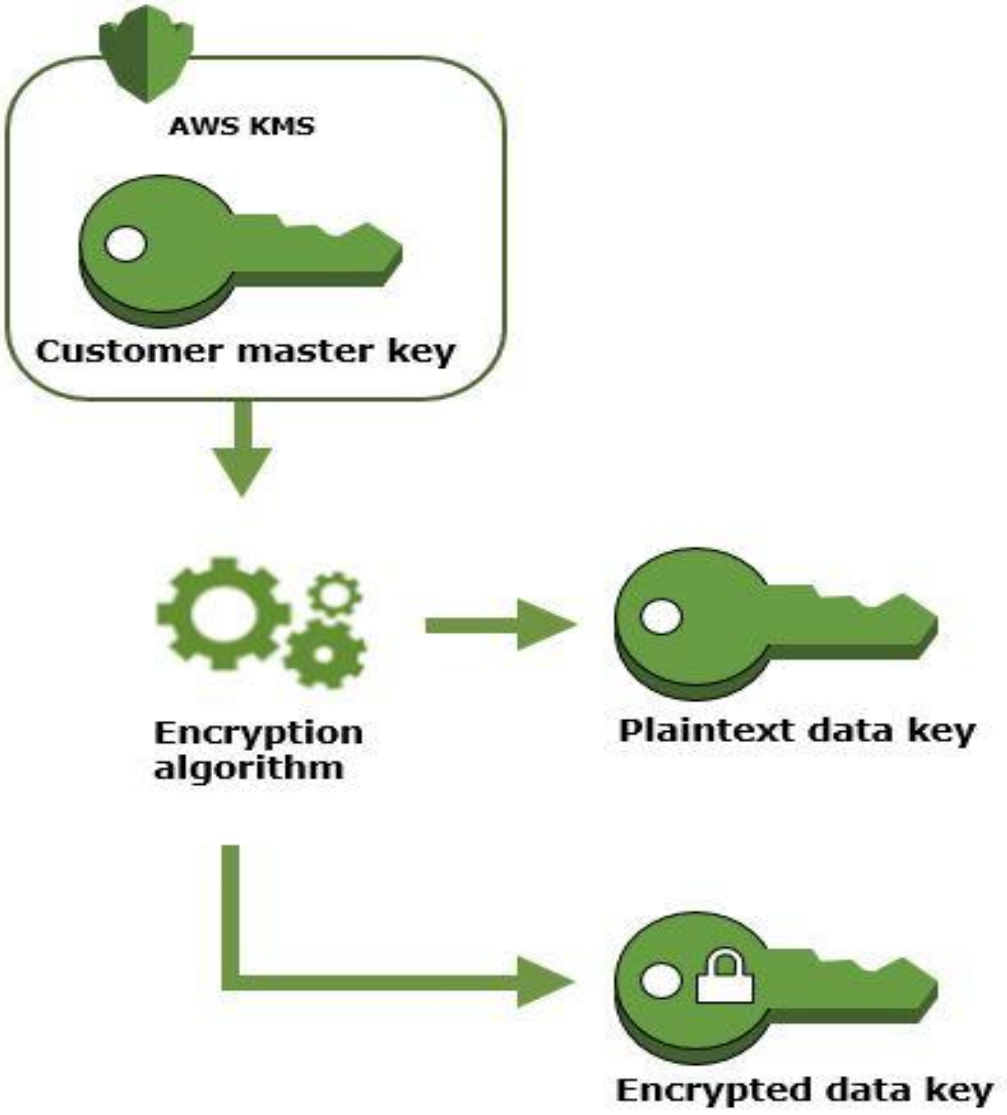
- AWS Managed CMKs Are CMKs In Your Account That Are Created, Managed, And Used On Your Behalf By An AWS Service That Is Integrated With AWS KMS. Some AWS Services Support Only An AWS Managed CMK
- You Do Not Pay A Monthly Fee For AWS Managed CMKs.They Can Be Subject To Fees For Use In Excess Of The Free Tier

S3-At-Rest(SSE-KMS)

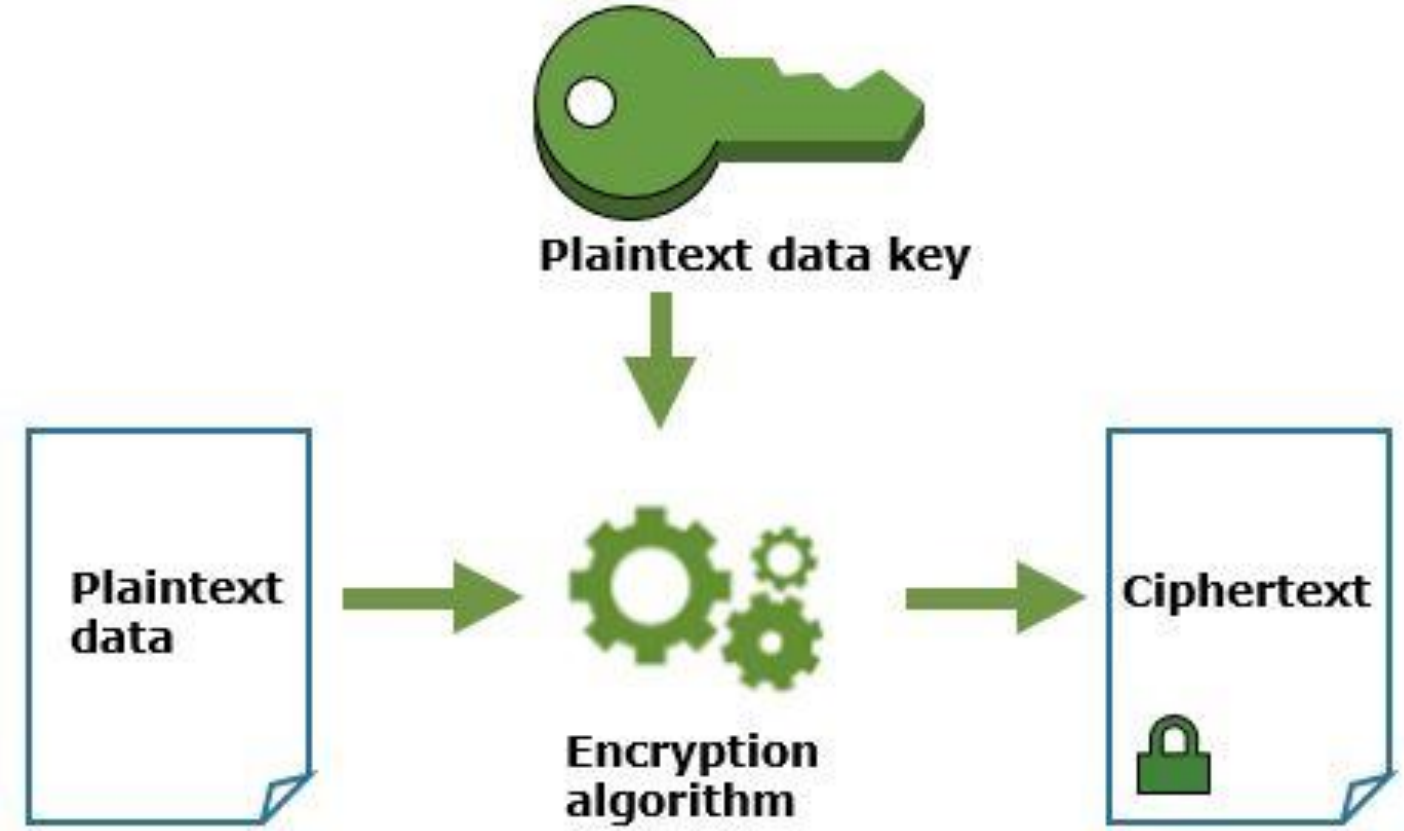
- SSE-KMS Data Encryption Method

1. Amazon S3 Requests A Plaintext Data Key And A Copy Of The Data Key Encrypted Under The Specified CMK
2. AWS KMS Creates A Data Key, Encrypts It By Using The Master Key, And Sends Both The Plaintext Data Key And The Encrypted Data Key To Amazon S3
3. Amazon S3 Encrypts The Data Using The Data Key And Removes The Plaintext Key From Memory As Soon As Possible After Use
4. Amazon S3 Stores The Encrypted Data Key As Metadata With The Encrypted Data

Create Plain Text Data Key And Encrypted Data Key



Encrypt Data With A Plain Text Data Key

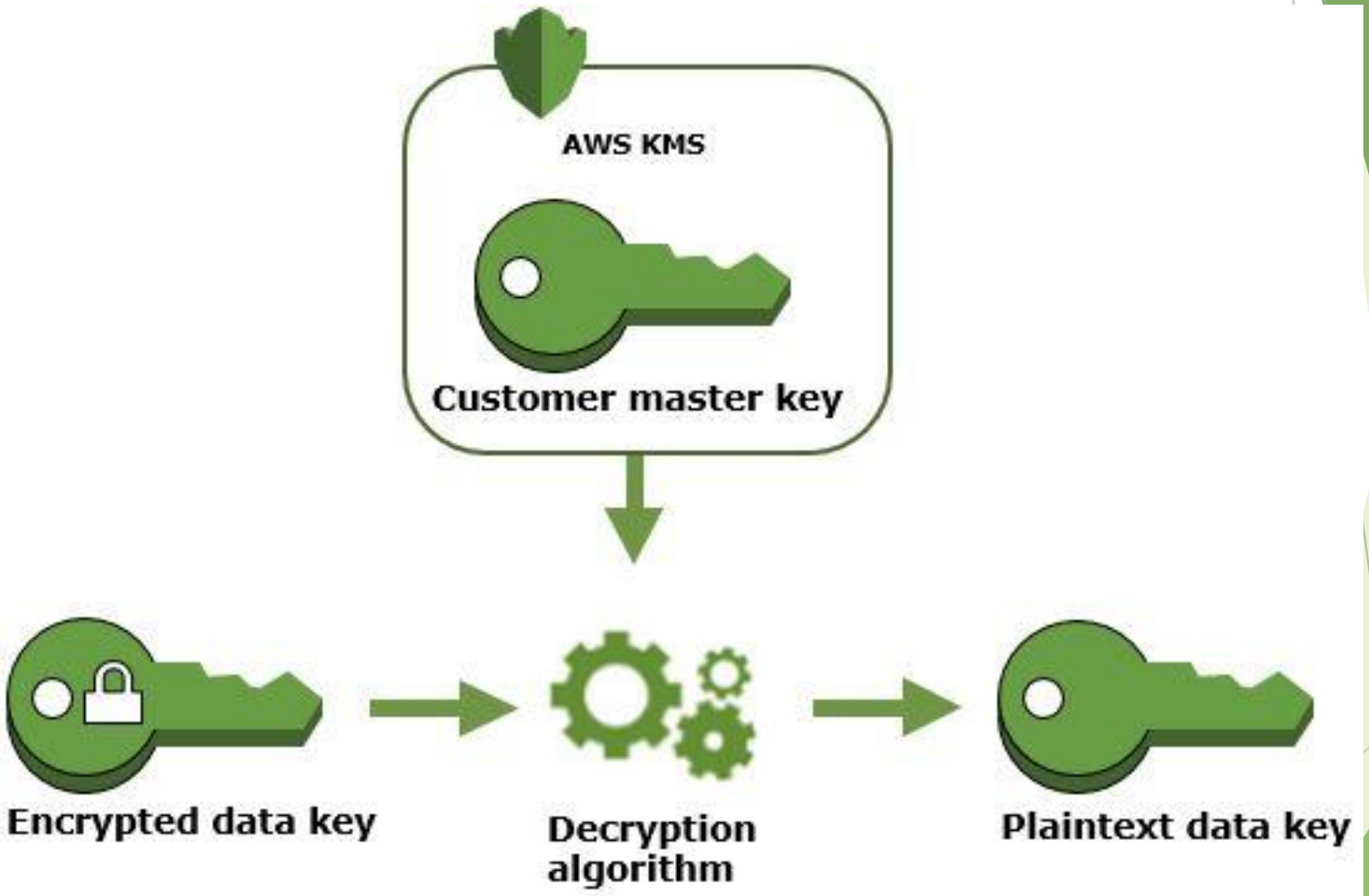


S3-At-Rest(SSE-KMS)

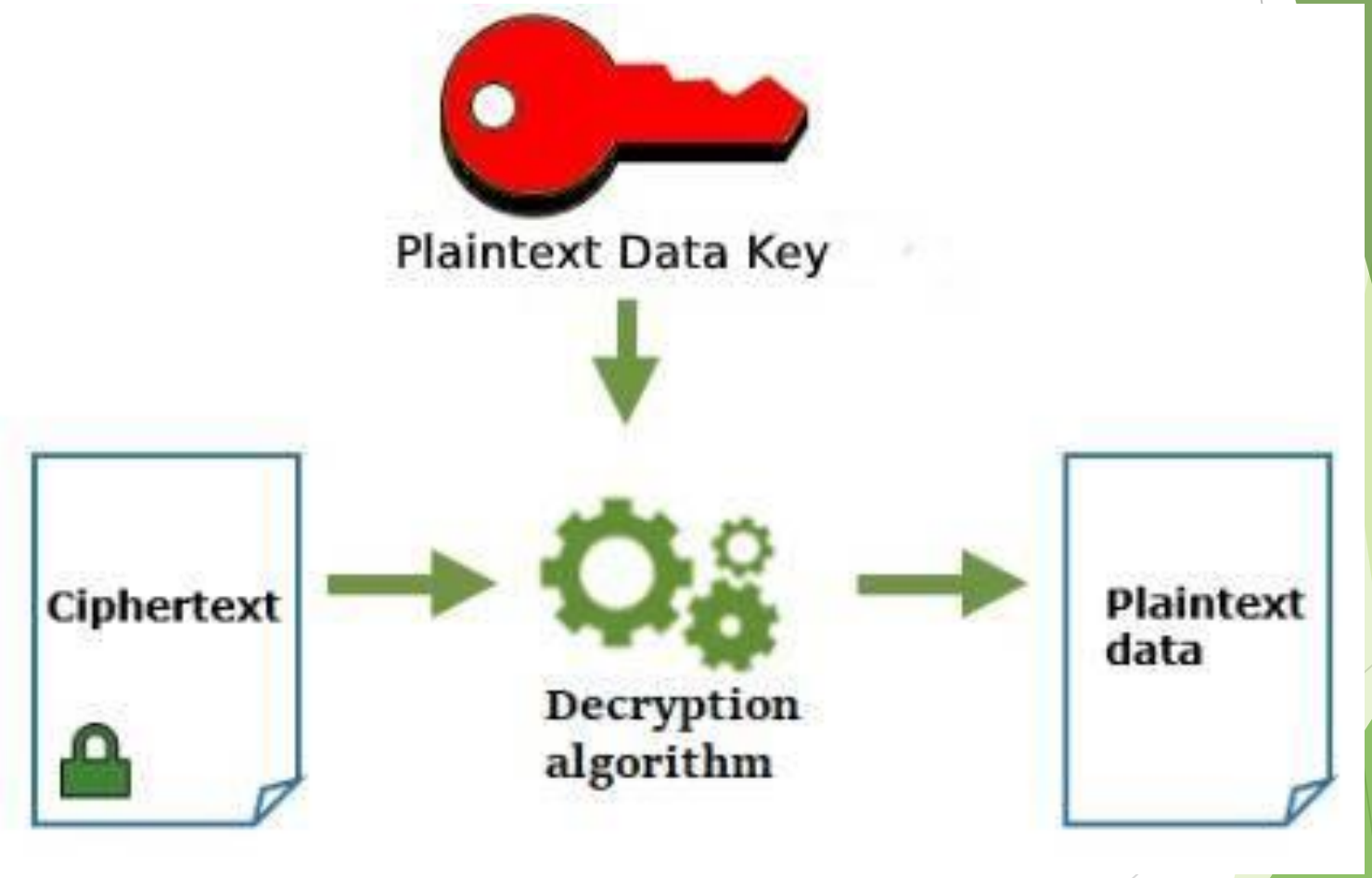
- Amazon S3 And AWS KMS Perform The Following Actions When You Request That Your Data Be Decrypted

1. Amazon S3 Sends The Encrypted Data Key To AWS KMS
2. AWS KMS decrypts The Key By Using The Appropriate Master Key And Sends The Plaintext Key Back To Amazon S3
3. Amazon S3 Decrypts The Ciphertext And Removes The Plaintext Data Key From Memory As Soon As Possible
4. For https access use aws signature version 4 otherwise use aws sdk or aws cli
5. https://docs.aws.amazon.com/general/latest/gr/signature-version-4.html

Decrypt Encrypted Data Key using CMK



Decrypt Data Using Plaintext Data Key



SSE-KMS Lab

SSpecifying server-side encryption with AWS KMS (SSE-KMS)

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/specifying-kms-encryption.html

S3-At-Rest(SSE-C)

- SSE-C Allows You To Set Your Own Encryption Keys
- Amazon S3 Manages The Encryption As It Writes To Disks And Decryption When You Access Your Object
- But User Has To Manage The Encryption Keys

Steps

1. When You Upload An Object, Amazon S3 Uses The Encryption Key And Apply AES-256 Encryption To Your Data And Removes The Encryption Key From Memory

S3-At-Rest(SSE-C) Customer-Provided Key

2. Amazon S3 Does Not Store The Encryption Key You Provide. Instead, It Stores A Randomly Salted(Unpredictable Value) HMAC Value Of The Encryption Key To Validate Future Requests
3. When You Retrieve An Object, You Must Provide The Same Encryption Key
4. Amazon S3 First Verifies That The Encryption Key You Provided Matches With The HMAC Value And Then Decrypts The Object Before Returning The Object Data To You

LAB

- SSE-C(Customer-Provided-Key)
- SSE-C Documentation
  - https://docs.aws.amazon.com/kms/latest/developerguide/services-s3.html

Lab

- Download Open SSL
- https://code.google.com/archive/p/openssl-for-windows/downloads
- Openssl-0.9.8k_X64.zip second file
- Go to user variable
- Set PATH
- %PATH%;c:\openssl\filename\bin
- Enter Below command to generate Customer-Provided-Key
- openssl enc -aes-128-cbc -k secret -P
- echo "copy created key from cli" >c:\s3test\key.txt

Upload and Encrypt File1.txt Using Below Commands

- aws s3 cp c:\test\test\file1.txt s3://tvm200 --sse-c --sse-c-key keyvalue

DownLoad and Decrypt File1/txt

- aws s3 cp s3://tvm25/testfile.txt c:\s3text --sse-c --sse-c-key keyvalue

S3-Client Side Encryption

- Client-side Encryption Is The Act Of Encrypting Data Before Sending It To Amazon S3
- Symmetric and Asymmetric data Data Keys Are Being Used

S3-Client Side Encryption

Data Keys

- AWS KMS Provides Symmetric Data Keys And Asymmetric Data Key Pairs That Are Used For Client-side Cryptography Outside Of AWS KMS https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-keys
- You Cannot Use An Asymmetric CMK To Generate Data Keys
- The Symmetric Data Key And The Private Key In An Asymmetric Data Key Pair Are Protected By A Symmetric CMK In AWS KMS
- Can Be Used 128-bit Or 256-bit Data Keys

S3-Client Side Encryption

Symmetric Data Key

- A Symmetric Encryption Data Key (AES)That You Can Use To Encrypt Data Outside Of AWS KMS. This Key Is Protected By A Symmetric CMK In AWS KMS

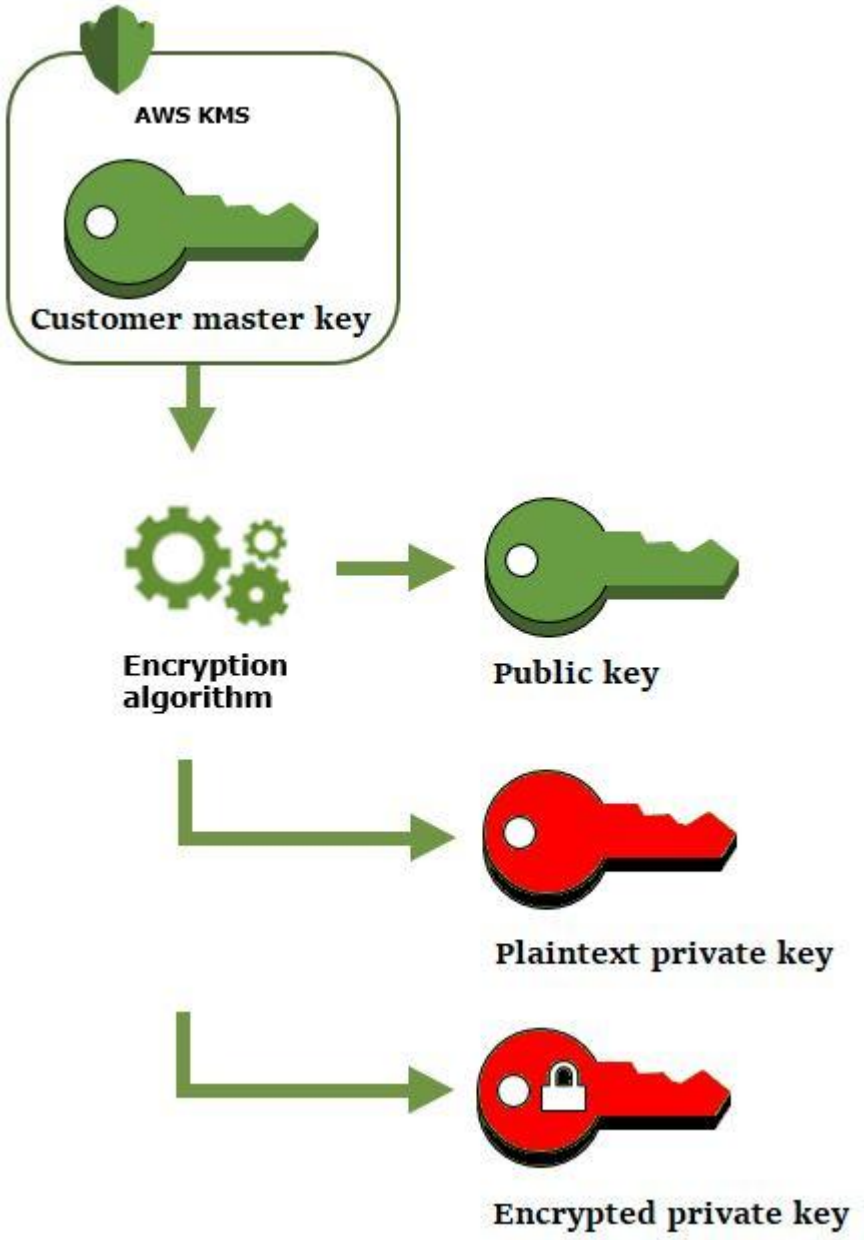
Asymmetric Data Key Pair

- An RSA Or Elliptic Curve (ECC) Key Pair That Consists Of A Public Key And A Private Key.
- You Can Use Your Data Key Pair Outside Of AWS KMS To Encrypt And Decrypt Data, Or Sign Messages And Verify Signatures.
- The Private Key Is Protected By A Symmetric CMK In AWS KMS

https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-key-pairs
https://docs.aws.amazon.com/kms/latest/developerguide/symmetric-asymmetric.html

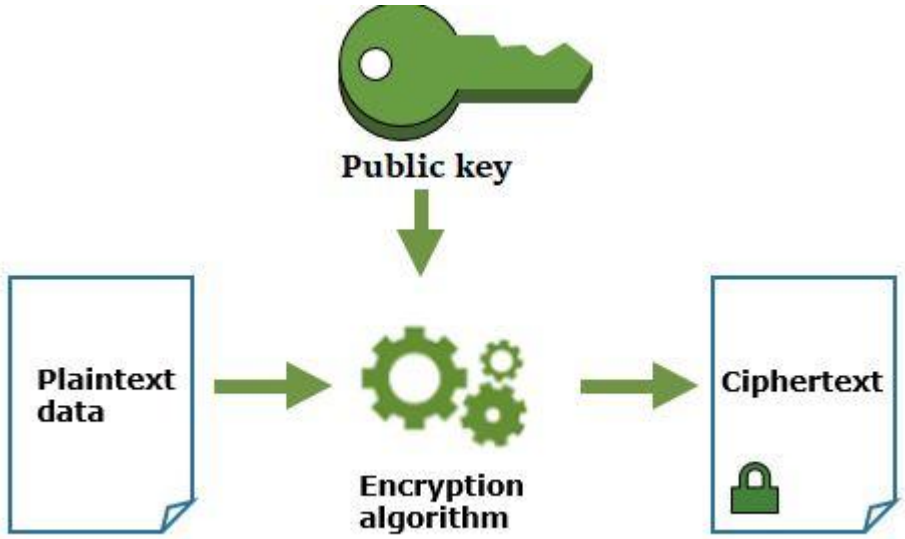
Client side encryption using Asymmetric Data Key Pair

Create a data key pair



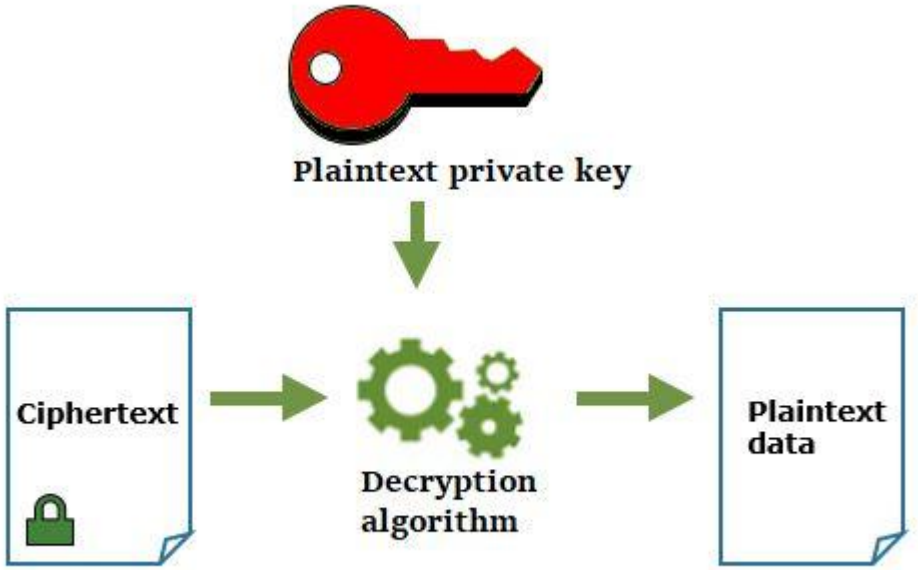
Client side encryption using Asymmetric Data Key Pair

Encrypt data with a data key pair



Client side Decryption using Asymmetric Data Key Pair

Decrypt data with a data key pair



S3-Client Side Encryption

Two Options

Use A Customer Master Key (CMK) Stored In AWS Key Management Service (AWS KMS)

1. Using a CMK stored in AWS KMS

   (Using Amazon S3 Encryption Client) https://aws.amazon.com/blogs/security/how-to-encrypt-and-decrypt-your-data-with-the-aws-encryption-cli/

   1. https://docs.aws.amazon.com/general/latest/gr/aws_sdk_cryptography.html
   2. https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingClientSideEncryption.html
   3. https://docs.amazonaws.cn/en_us/kms/latest/developerguide/programming-encryption.html

   Asymetric CMK

   1. https://docs.amazonaws.cn/en_us/kms/latest/developerguide/programming-encryption.html

   signing Request

   1. https://docs.aws.amazon.com/AmazonS3/latest/API/sig-v4-authenticating-requests.html

S3-Client Side Encryption

- Option 1: Using A CMK Stored In AWS KMS

When Uploading An Object

1. —using The Customer Master Key Id, The Client First Sends A Request To AWS KMS For A CMK .
2. CMK Generate A Plaintext Data Key(Symmetric) And Cypher Blob(A Copy Of The Data Key Encrypted) Upon Request
3. AWS KMS Returns Two Versions Of A Generated Data Key
4. A Plaintext Version Of The Data Key That The Client Uses To Encrypt The Object Data
5. A Cipher Blob Of The Same Data Key That The Client Uploads To Amazon S3 As Object Metadata

S3-Client Side Encryption

When downloading an object

1. The Client Downloads The Encrypted Object From Amazon S3 Along With The Cipher Blob Version Of The Data Key Stored As Object Metadata
2. The Client Then Sends The Cipher Blob To AWS KMS To Get The Plaintext Version Of The Data Key So That It Can Decrypt The Object Data

Option 2: Using A Master Key Stored Within Your Application

- When Uploading An Object—you Provide A Client-side Master Key To The Amazon S3 Encryption Client. The Client Uses The Master Key Only To Encrypt The Data Key That It Generates Randomly. The Process Works Like This:

- The Amazon S3 Encryption Client Generates A One-time-use Symmetric Key (Also Known As A Data Key) Locally.
  - https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingClientSideEncryption.html

S3-At-Rest(Client Side Encryption)

- It Uses The Data Key To Encrypt The Data Of A Single Amazon S3 Object. The Client Generates A Separate Data Key For Each Object
- The Client Encrypts The Data Encryption Key Using The Master Key That You Provide.
- The Client Uploads The Encrypted Data To Amazon S3 And Saves The Encrypted Data Key And Its Description As Object Metadata In Amazon S3
- The Client Uses The Description To Determine Which Client-side Master Key To Use For Decryption.

S3-At-Rest(Client Side Encryption)

- When Downloading An Object—the Client Downloads The Encrypted Object From Amazon S3.
- Using The Description From The Object's Metadata, The Client Determines Which Master Key To Use To Decrypt The Data Key.
- The Client Uses That Master Key To Decrypt The Data Key And Then Uses The Data Key To Decrypt The Object.
- The Client-side Master Key That You Provide Can Be Either A Symmetric Key Or A Public/Private Key Pair. The Following Examples Show How To Use Both Types Of Keys.

S3-At-Rest(Client Side Encryption)

The Following Example Shows How To Do Encryption Using Symmetric CMK Key

- Generate A 256-bit AES Key
- Save And Load The AES Key To And From The File System
- Use The AES Key To Encrypt Datakey On The Client Side Before Sending It To Amazon S3
- Use The AES Key To Decrypt Datakey Received From Amazon S3
- Verify That The Decrypted Datakey Is The Same As The Original Datakey

S3-At-Rest(Client Side Encryption)

The Following Example Shows How To Do Encryption Using Asymmetric CMK Key

- Generate A 1024-bit RSA Key Pair.
- Save And Load The RSA Keys To And From The File System.
- Use The RSA public Key To Encrypt Datakey On The Client Side Before Sending It To Amazon S3.
- Use The RSA private Key To Decrypt Datakey Received From Amazon S3.
- Verify That The Decrypted Datakey Is The Same As The Original Data

- https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingClientSideEncryption.html
- https://aws.amazon.com/articles/client-side-data-encryption-with-the-aws-sdk-for-java-and-amazon-s3/

LAB

- Client Side Encryption Using KMS KEY(AWS Encryption Client CLI)

client Side encryption using KMS KEY Lab

- Install AWS Encryption CLI
- https://docs.aws.amazon.com/encryption-sdk/latest/developer-guide/crypto-cli-install.html

1.After installing Encryption CLI

2.check powershell version

3.$host or $psversiontable

4.power shell version="Get-Host | Select-Object Version".

5.if it is 5.1 or Above then run the following commands

- Parameters Used

key =KMS Key ID

file1.txt =file to be encrypted

encrypted.file1.txt=File1.txt will be saved after encryption

```
aws-encryption-cli --encrypt `
    --master-keys key=b0ab7775-f135-4154-96ff-7525ec9e44aa `
    --encryption-context purpose=test* `
    --metadata-output "./Metadata" `
    --input "file1.txt" `
    --output "./encrypted.file1.txt"
```

client Side Decryption using KMS KEY

- Move the files encrypted.file1.txt and metadata in the directory test to bucket tvm200

```
aws s3 mv c:\test s3://tvm200 --recursive --exclude file1.txt
```

- For Decrypting uploaded file, use below commands

```
aws s3 mv s3://tvm200 c:\test --recursive
aws-encryption-cli --decrypt `
    --encryption-context purpose=test* `
    --metadata-output "./Metadata" `
    --input "./encrypted.file1.txt" `
    --output "./decrypted.file1.txt"
```

Asymmetric and Symmetric CMK differences

Key Pairs

- one key usage for encryption and decryption
- public and private key pair for encrypt/decrypt or signing/verification

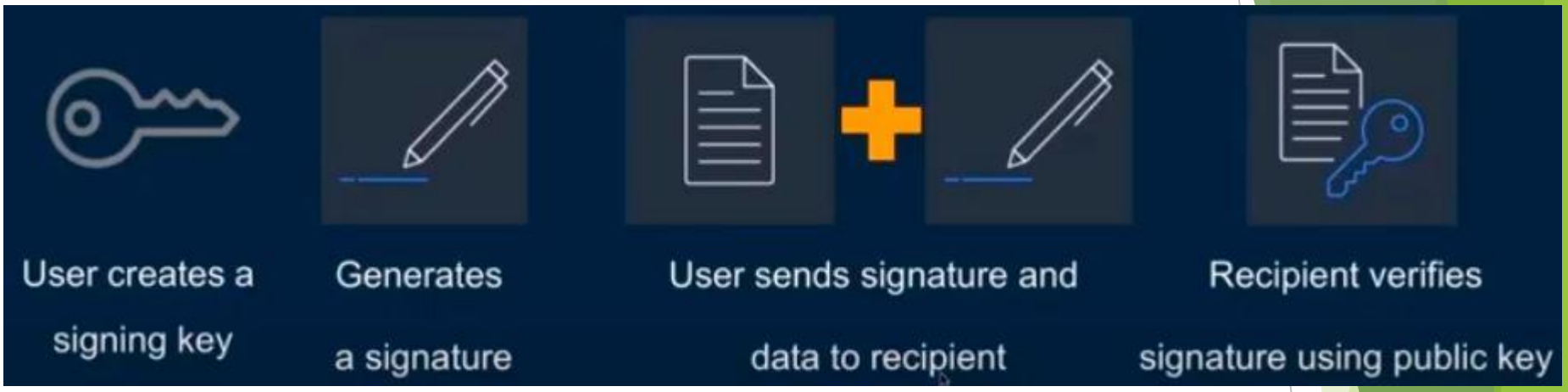
operation inside and outside aws kms

- symmetric key operations happen within the service
- encryption and verification operation can happen outside the service

interoperability

- ciphertext produced by symmetric keys has kms specific metadata(don't have to send key id)
- asymmetric keys allows you interoperability with other systems
- ciphertext and signature produced does not have the metadata
- the plain text/cipher text being sent to kms for encryption/decryption or signing along with the cmk key id

Digital Signature Workflow Using Asymmetric cmk



Asymmetric CMK Encryption Workflow

S3-Server Access Logging

- used to track requests for access to your bucket
- There is no extra charge for enabling server access logging on an Amazon S3 bucket, and you are not charged when the logs are PUT to your bucket
- usual charges for storage.
- You can delete these log files at any time.
- Subsequent reads and other requests to these log files are charged normally, as for any other object, including data transfer charges

Use Case

https://docs.aws.amazon.com/AmazonS3/latest/dev/ServerLogs.html

- access log information can be useful in security and access audits.
- It can also help you learn about your customer base and understand your Amazon S3 bill

S3-Transfer Acceleration

- Not Uploading Files Or Data Directly To S3.Using A Distinct Url,You Can Upload Directly To Edge Location And Then Transfer To S3 Using An Optimized Network Path
- It Enables Fast And Secure Data Transfer Over Long Distance Between Your Client And S3 Bucket

When To Use

- Many Customers Using A Centralized Bucket From All Over The World And Transfer GB's Or TB's Of Data On A Regular Basis,Across Countries
- Customers Not Able To Use The Full Bandwidth Of Internet For Uploading Files To S3

S3-Transfer Acceleration

- You Can Use A Transfer Acceleration Tool To Check The Bandwidth
- If The Bandwidth Is Same As The Regular S3 Transfer,They Will Not Charge.
- Minimum 20 Minutes Will Take To Realize The Performance Benefit

S3-Transfer Acceleration Lab

Transfer Acceleration Speed Comparison Tool

- https://s3-accelerate-speedtest.s3-accelerate.amazonaws.com/en/accelerate-speed-comparsion.html

Lab (Console & AWS CLI & AWS SDK's)

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/transfer-acceleration-examples.html

AWS CLI

- `$ aws s3 cp file.txt s3://bucketname/keyname --region region --endpoint-url https://s3-accelerate.amazonaws.com`

OR

- `$ aws s3 cp file.txt s3://bucketname/keyname --region region --endpoint-url https://BucketName.s3-accelerate.amazonaws.com`

Document

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/transfer-acceleration.html

S3-Pricing

- Storage Price
- Request (PUT, COPY, POST, LIST,GET, SELECT)Price
- Management Price(If You Require Cloudwatch)
- Data Transfer Pricing For In And Out Of S3
- Transfer Acceleration (Optional) In And Out
- Replication Pricing
- S3 Encryption Pricing(Using KMS)

S3-Pricing

-

Free Usage

- Transfers Between S3 Buckets Or From Amazon S3 To Any Service(S) Within The Same AWS Region Are Free
- Data transferred in from the internet.
- Data transferred out to Amazon CloudFront (CloudFront).

S3 Access

- Using Console You Can Perform Almost All Bucket Operations Without Writing Any Code
- If You Access Programmatically ,You Have To Write Rest Api Codes

S3 CLIENT APPLICATION

https://s3browser.com

https://docs.aws.amazon.com/AWSJavaSDK/latest/javadoc/com/amazonaws/services/s3/AmazonS3Client.html

Uploading to Amazon S3 directly from a web or mobile application

- https://aws.amazon.com/blogs/compute/uploading-to-amazon-s3-directly-from-a-web-or-mobile-application/

Uploading objects using pre signed URLs

- https://docs.aws.amazon.com/AmazonS3/latest/userguide/PresignedUrlUploadObject.html

Signing AWS requests with Signature Version 4

- https://docs.aws.amazon.com/general/latest/gr/sigv4_signing.html

Amazon S3 Browser-Based Uploads

- https://s3.amazonaws.com/doc/s3-example-code/post/post_sample.html