AWS Architecting and SysOps

File Storage on AWS

June-July 2019

Contents

Introduction

AWS Storing Fundamentals

Storage

Amazon Simple Storage Service (S3)

Amazon S3 Glacier

AWS Storage Gateway

Amazon Elastic File System

AWS Snowball

AWS Snowball Edge

AWS Snowmobile

Amazon Elastic Block Store (EBS)

Cloud storage with AWS

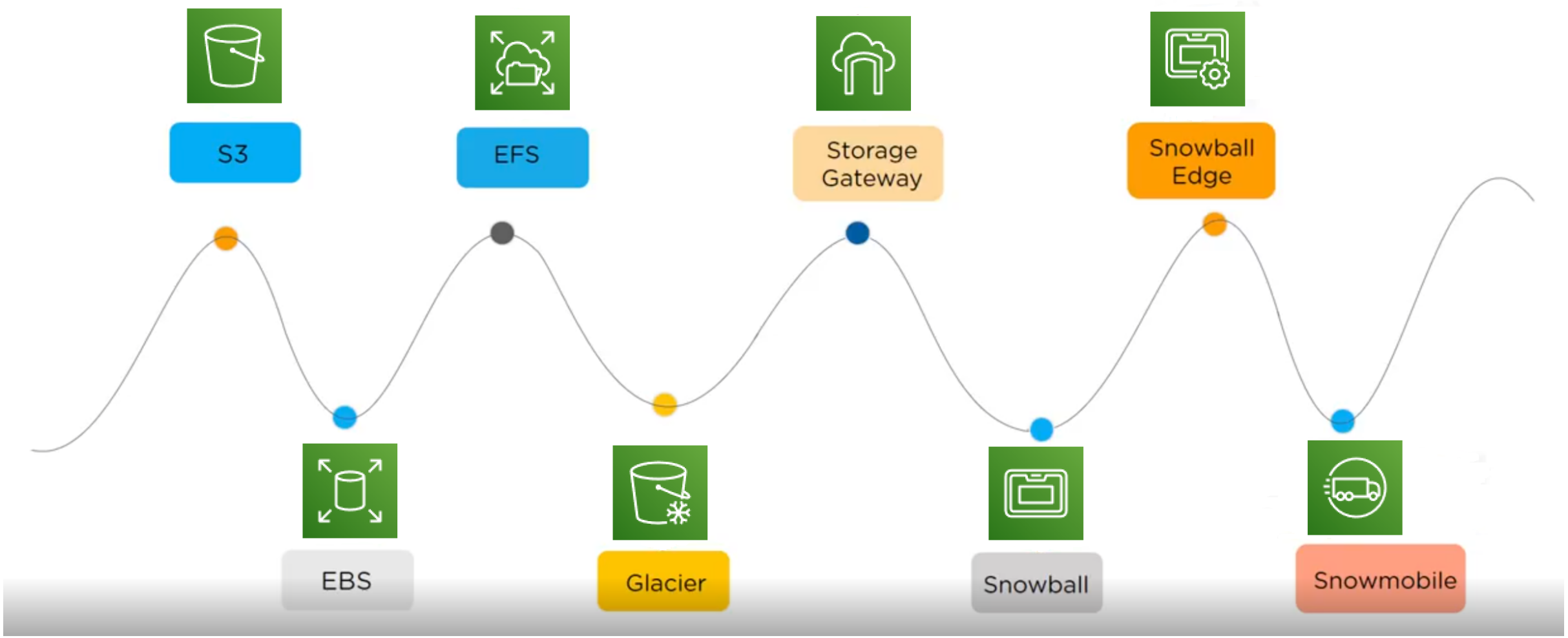
- Cloud storage is a critical component of cloud computing, holding the information used by applications

- It provides a web service where your data can be:
  - ✓Stored
  - ✓Accessed
  - ✓Easily backed up

- Cloud storage is typically more:
  - ✓Reliable
  - ✓Scalable
  - ✓Secure

  than traditional on-premises storage systems

Types of storage overview



Types of storage overview

- S3: cloud storage to make data accessible from any Internet location
  - Scalable storage in the cloud
  - User-generated content, active archive, serverless computing, big data storage..

- EBS: persistent local storage attached to EC2 instances
  - EC2 block storage volumes
  - For databases, data warehousing, enterprise applications, big data processing…

- EFS: shared storage to use with AWS cloud service and on-premises resources
  - Fully managed file system for EC2
  - Similar to EBS but shared between EC2 instances

Types of storage overview

- S3 Glacier: highly affordable long-term storage
  - The archiving solution for AWS cloud

- Storage gateway: a hybrid storage cloud augmenting your on-premises environment with Amazon cloud storage
  - Used to move data from local environment to the cloud, but still a local copy is necessary

- Snow family: massive data transfer services
  - Hardware is shipped to your local premises where you can copy data and shipped back to Amazon. Then Amazon will copy the data to the desired destination in AWS cloud

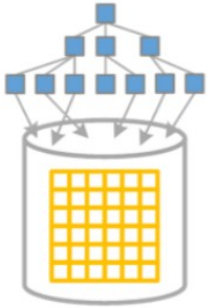
✓Snowball: data input and export hardware device

✓Snowball Edge: Snowball (with more capacity) + an embedded computer platform

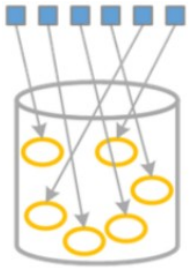
✓Snowmobile: a data center build in a truck that allows massive data transfers

Block, file, and object storage

- Block storage
  - Stores data organized as an array of unrelated blocks
  - Data is stored in blocks, and multiple blocks comprise a file
  - EBS provides persistent block storage volumes to use with EC2



- File storage
  - Stores data in a hierarchy of files and folders
  - Data stored on a file system contains attributes (permissions, file owner…) which are stored as metadata in the file system
  - EFS provides scalable file storage to use with EC2



- Object storage
  - Flat storage systems that stores data as objects
  - Objects consist of data, metadata, and a unique identifier
  - S3 provides object storage to store an almost limitless amount of data and scale easily

Amazon Elastic Block Store (EBS)

Amazon Elastic File System

Amazon Simple Storage Service (S3)

Amazon S3

Storing files in the Cloud

S3 (Simple Storage Service)

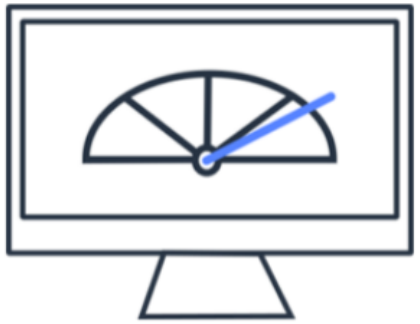
- S3 is an object storage service that stores and retrieves any amount of data from anywhere on the Internet
  - You can use S3 as an extension of your on-premises storage
- S3 offers industry-leading scalability, data availability, security and performance at low cost
  - You can access your objects any time via AWS Management Console, AWS CLI, & AWS SDKs
- The total volume of data and number of objects you can store are almost unlimited
- Customers of all sizes and industries can use it to store and protect any amount of data for a wide range of use cases
  - Websites, mobile applications, backup and restore, archive, enterprise applications, IoT devices, and big data analytics

S3 Benefits

Easy to use

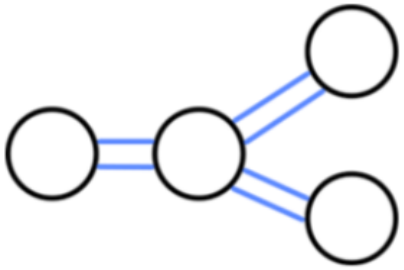
Durable

Available



High performance

Scalable



Integrated

S3 Benefits

- Easy to use: built with a focus on simplicity and robustness
- Designed for 99.999999999% durability across multiple Availability Zones
  - You don't need to pre-provision the size of S3 for your data
  - ❖Durability = ability to assure data is stored on the system as long as it is not changed by legitimate access. Data should not get corrupted or disappear because of a system malfunction
- Designed for at least 99.5% availability (depending on the product)
  - ❖Availability = ability to be operational and accessible if required. In AWS, the availability is increased by adding redundancy to it
- High performance
  - S3 supports 3500 requests/sec to add data or 5500 request/sec to retrieve data
- Scalable and pay-as-you-go
  - You don't need to pre-provision the size of S3 for your data, and you pay for what you use
- It integrates with other AWS services and third-party solutions

S3 terms

- Object
  - Base unit of storage in S3 that consist of object data and metadata
  - Metadata is a set of name-value pairs that describe the object
    - Date last modified, standard HTTP metadata, custom metadata…
  - Uniquely identified by a key (name) and version id
  - Max size: 5 TB

- Bucket
  - A container for storing objects
  - Identified by name and the selected region where the bucket has been created
  - The bucket name is unique across all existing bucket names in S3

- Access policy
  - Who can / can't access data stored in the bucket
  - What operations users can / can't perform on the bucket
  - By default, buckets are private

S3 Security and Encryption

- You can secure your S3 buckets with:
  - IAM bucket policies
  - S3 bucket policies
  - Access Control Lists (ACLs)
- S3 buckets can be configured to create access logs which logs all requests made to the S3 bucket
- Encryption available in S3

✓In transit SSL / TLS

✓At rest

  - AES-256, Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3)
  - AWS-KMS, Server-Side Encryption with AWS KMS-Managed Keys (SSE-KMS)
  - AWS-KMS, Server Side Encryption With Customer Provided Keys (SSE-C)

S3 Security

- IAM policies
  - IAM policies specify what actions are allowed / denied on what AWS resources
  - ➢ If you are more interested in “What can this user do in AWS?”
- S3 policies
  - S3 bucket policies specify what actions are allowed / denied for which principals on the bucket that the bucket policy is attached to
  - With S3 bucket policies you are able to set fine grain permissions for the bucket
  - ➢ If you are more interested in “Who can access this S3 bucket?”
- Be consistent with the method you choose!!!
  - ➢ You may choose S3 policies at some point, because you have reached your limit of IAM policies

- S3 ACLs
  - Legacy control access. A sub-resource attached to every S3 bucket and object
  - It defines which AWS accounts or groups are granted access and the type of access
  - When you create a bucket, S3 creates a default ACL that grants the resource owner full control

S3 Security

IAM policy

```
{
  "Version": "2012-10-17",
  "Statement":[{
    "Effect": "Allow",
    "Action": "s3:*",
    "Resource": ["arn:aws:s3:::my_bucket",
                 "arn:aws:s3:::my_bucket/*"]
    }
  ]
}
```

S3 policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": ["arn:aws:iam::111122223333:user/Alice",
                "arn:aws:iam::111122223333:root"]
      },
      "Action": "s3:*",
      "Resource": ["arn:aws:s3:::my_bucket",
                   "arn:aws:s3:::my_bucket/*"]
    }
  ]
}
```

Principal corresponds to the IAM entity that have the access specified in the S3 policy
(In IAM the principal is already the IAM entity)

S3 Security: Public access settings

- S3 block public access settings prevents the application of any settings that allow public access to data within S3 buckets

- There are 2 possibilities: management of public ACL and management of public bucket policies for the bucket, with each 2 possibilities each

✓Manage public ACLs for this bucket
  - Block new public ACLs and uploading public objects
  - Remove public access granted through public ACLs

✓Manage public bucket policies for this bucket
  - Block new public bucket policies
  - Block public and cross-account access if bucket has public policies

S3 Security: Public access settings

- Block new public ACLs and uploading public objects
  - Prevents setting or updating public bucket or object ACLs.
  - If you try to put public ACLs on a bucket or object, or upload objects with public ACLs, the request is rejected with an access denied error.
  - This setting doesn't affect your existing public ACLs.

- Remove public access granted through public ACLs
  - S3 does not evaluate public ACLs when authorizing requests for access to buckets or objects.
  - If objects with public ACLs are uploaded to your bucket, or a public ACL is added to a bucket, this setting denies public access granted through those ACLs

S3 Security: Public access settings

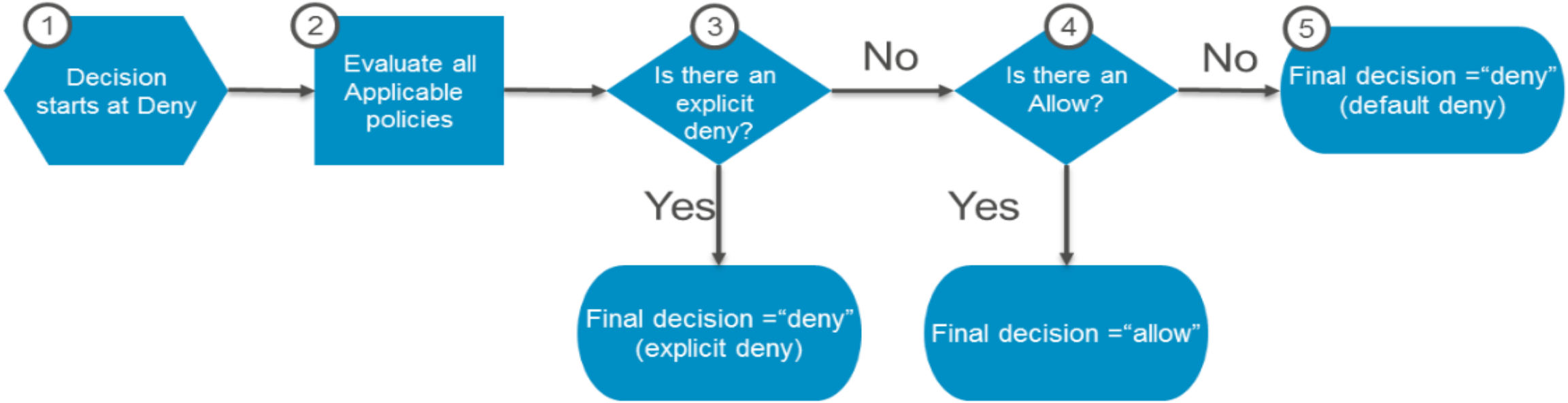
- Block new public bucket policies
  - Prevents adding a new public policy on a bucket and displays and access denied error.
  - This setting does not change the behavior of an existing bucket that has a public policy.
  - Use this setting to maker sure that no one can update a bucket policy to grant public access to the bucket

- Block public and cross-account access if bucket has public policies
  - If set, access to buckets with public policies is limited to the bucket owner and to AWS services.
  - Existing public access and cross-account access are denied.
  - Use this setting to protect buckets that have public policies while you work to remove the policies

S3 Security: authorization process

- How does authorization work with multiple access control mechanisms?
  - Whenever an AWS principal issues a request to S3, the authorization decision depends on the union of all the IAM policies, S3 bucket policies, and S3 ACLs that apply

- According to the least-privilege principle, decisions default to *deny* and an explicit deny always trumps an *allow*.



S3 Version Control

- Versioning enables you to keep multiple versions of an object in one bucket
  - Ex/ store my-image.jpg (version 111) and my-image.jpg (version 112) in a single bucket
- Versioning protects you from the consequences of unintended overwrites and deletions from users or applications
  - Enabling versioning's MFA delete setting, increases your protection
- You can simply use versioning to archive objects so you have access to previous versions
- Normal S3 pricing applies to each version of the object
- Buckets can be in one of these states:
  - ✓Un-versioned (default). When you create a bucket, this is the default state
  - ✓Versioning-enabled. Once activated, you cannot return to un-versioned state
  - ✓Versioning-suspended. When you don't want versioning for the bucket anymore

S3 Cross Region Replication (CRR)

- Enables automatic, asynchronous copying of object across buckets in different AWS regions
- Buckets configured for cross-regional replication can be owned by the same AWS account or by different accounts
- Cross-region replication is enabled with a bucket-level configuration
  - You add the replication configuration to your source bucket where you provide the destination bucket where you want Amazon S3 to replicate objects
- When to use it?
  - Minimize latency
  - Increase operational efficiency
  - Maintain object copies under different ownership

S3 Storage Classes

- S3 Standard – Frequent Access
  - Used for frequently accessed data. Ex: student's attendance
- S3 Intelligent–Tiering [Unknown or changing access]
  - Designed to optimize costs by automatically moving data to the most cost-effective access tier, without performance impact or operational overhead
- S3 Standard – Infrequent Access (S3 Standard-IA)
  - Used for less frequently accessed data, but requires rapid access when needed
  - Ex: student's record
- S3 One Zone – Infrequent Access (S3 One Zone-IA)
  - The same as S3 Standard-IA but for a single AZ. Ex: student's report card
- S3 Glacier
  - Used for long-term, low-cost data archive (the cost is almost 6 times less than S3 standard)
  - Ex: old student's records

S3 Storage Classes

	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive**
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	N/A	N/A
Availability SLA	99.9%	99%	99%	99%	N/A	N/A
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days
Retrieval fee	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	millseconds	milliseconds	milliseconds	select minutes or hours	select hours
Storage type	Object	Object	Object	Object	Object	Object
Lifecycle transitions	Yes	Yes	Yes	Yes	Yes	Yes

* S3 Intelligent-Tiering charges a small tiering fee and has a minimum eligible object size of 128KB for auto-tiering. Smaller objects may be stored but will always be charged at the Frequent Access tier rates

† Data stored will be lost in the event of AZ destruction

** Coming soon

S3 Lifecycle

- To manage your objects so that they are stored cost effectively throughout their lifecycle, you can configure their lifecycle
- A lifecycle configuration is a set of rules that define actions that S3 applies to a group of objects
  - You can configure S3 to move your data between various storage classes on a defined schedule
- Use cases:
  - Periodic logs in a bucket that your app needs for a period of time. Then, you can delete them
  - Documents frequently accessed for a limited period of time then they become infrequently accessed. You do not need real-time access to them, but your organization or regulations might require you to archive them for a specific period. After that, you can delete them
  - Data for archival purposes. For example, you might archive digital media, financial and healthcare records, raw genomics sequence data, long-term database backups, and data that must be retained for regulatory compliance.

S3 Automated Lifecycle Policy

- Transition
  - Define when objects transition to another storage class
  - Objects must be stored at least 30 days in the current class before you can transition them

- Expiration
  - Define when objects expire
  - S3 deletes expired objects on your behalf

S3 Transfer acceleration

- Enables fast, easy, and secure transfers of files over long distances between your client and an your S3 bucket
- It takes advantage of Amazon CloudFront's globally distributed edge locations.
  - ❖Edge location: a nearby site (generally deployed in major cities or highly populated areas) used to cache copies of your content for faster delivery
  - As the data arrives at an edge location, data is routed to Amazon S3 over an optimized network path.
- When using Transfer Acceleration, additional data transfer charges may apply
- You might want to use Transfer Acceleration on a bucket for various reasons, including the following:
  - You have customers that upload to a centralized bucket from all over the world
  - You transfer gigabytes to terabytes of data on a regular basis across continents
  - You are unable to utilize all of your available bandwidth over the Internet when uploading to Amazon S3

S3 usage charges

- Storage
  - You pay for storing objects in your S3 buckets
  - The rate you're charged depends on your objects' size, how long you stored the objects during the month, and the storage class
- Requests
  - You pay for requests, for example, GET or PUT requests, made against you S3 buckets
  - You also pay for requests associated with lifecycle actions
  - The rate depends on the kind of request you make and the storage class
- Management
  - You pay for the storage management features (inventory, analytics, object tagging, and so on) that are enabled on your account's buckets

S3 usage charges

- Data transfer. You pay all data transfers in and out of S3, except for the following
  - Data transferred in from the internet
  - Data transferred out to an EC2 instance, when it is in the same AWS region as the S3 bucket
  - Data transferred out to CloudFront

➢You also pay a fee for any data transferred during S3 Transfer Acceleration

- Intelligent-Tiering. You pay for:
  - Storing objects
  - Monthly monitoring and automation fee
  - Objects that are deleted, overwritten, or transitioned to a different storage class before 30 days
- This class has a minimum billable object size of 128 KB for auto-tiering. Smaller objects may be stored but will be charged at Frequent Access tier rates

S3 usage charges

- Infrequent Access (S3 Standard-IA / S3 One Zone-IA). You pay for:
  - Objects stored
  - Retrieving objects that are stored in Standard-IA and One Zone-IA
  - Objects that are deleted, overwritten, or transitions to a different storage class before 30 days

➢Both classes have a minimum billable object size of 128 KB. Smaller objects may be stored but will be charged for 128 KB of storage

- S3 Glacier. You pay for:
  - Objects stored
  - Retrieving objects that are stored in S3 Glacier storage
  - Deleting an object stored before the 90 day minimum storage commitment hast passed
- For each object archived in S3 Glacier, S3 requires 8 KB of store and maintain the user-defined name and metadata for the object. This is charged at S3 standard rate

Hosting a static website on S3

- You can host a static website on S3
  - In a static website, individual webpages include static content and client-side scripts
- S3 does not support server-side scripting
  - A dynamic website relies on server-side processing, including server-side scripts
- To host a static website, you configure an Amazon S3 for website hosting, and then upload your website content to the bucket
- This bucket must have public read access
  - It is intentional that everyone in the world will have access to this bucket
- The website is then available at the region-specific website endpoint of the bucket
  - It is intentional that everyone in the world will have access to this bucket

S3 Use Cases

Backup Storage

Media Hosting

Application Assets

Data Lake

Content Delivery

S3 Use Cases

- Backup storage
  - S3 buckets can provide storage for active data backup services (that need a good access time)
- Media hosting
  - You use S3 buckets to host video, photo, or music uploads and downloads
- Application assets
  - Storage for your application data. As your application need to read / write data, you can configure them to use Amazon S3.
- Data lake
  - Store all your organization data (structured or unstructured) using a S3 buckets as your centralized repository
- Content delivery
  - You can use S3 buckets to host content that customers can download


Amazon S3 Glacier
Cheap Back-Ups for long-term needings

S3 Glacier

- S3 Glacier is a secure, durable, and extremely low-cost cloud storage service for data archiving and long-term backup

- It is optimized for infrequently used data, or “cold data”, that it must be retained for future reference or compliance
  - It is not designed to store data that is accessed frequently
- It is designed to deliver 99.999999999% durability
  - Data is redundantly stored in multiple facilities and in multiple devices within each facility
  - It is automatically distributed across a minim of three physical AZ

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

S3 Glacier terms

- Archive
  - Base unit of storage. Any object such as a photo, video, document that you store in S3 Glacier
  - Each archive has its own unique id and its optional description

- Vault
  - A container for storing archives
  - Identified by the name and the selected region where the vault has been created

- Access policy
  - Who can / can't access data stored in the vault
  - What operations users can / can't perform on the vault
  - You can set a lock policy for the vault if necessary

Why do you need a vault lock?

- 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
- Locking a vault takes two steps:
  1. Initiate the lock by attaching a vault lock policy to your vault, which sets the lock to an in-progress state and returns a lock Id.
     - While in the in-progress state, you have 24 hours to validate your vault lock policy before the lock Id expires
  2. Use the lock Id to complete the lock process.
     - If the vault lock policy doesn't work as expected, you can abort the lock and restart from the beginning

S3 Glacier retrieval time options

Objects are not available in real time, there is a retrieval time associated

Expedited

1-5 minutes

$0.03 per GB

Standard

3-5 hours

$0.01 per GB

Bulk

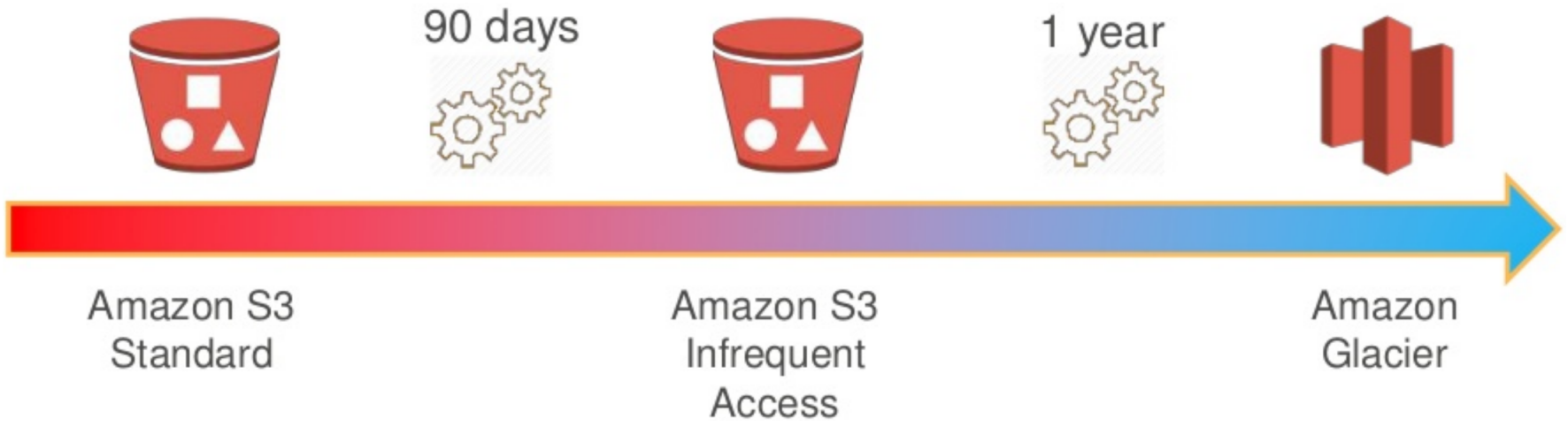
5-12 hours

$0.0025 per GB

S3 Glacier lifecycle

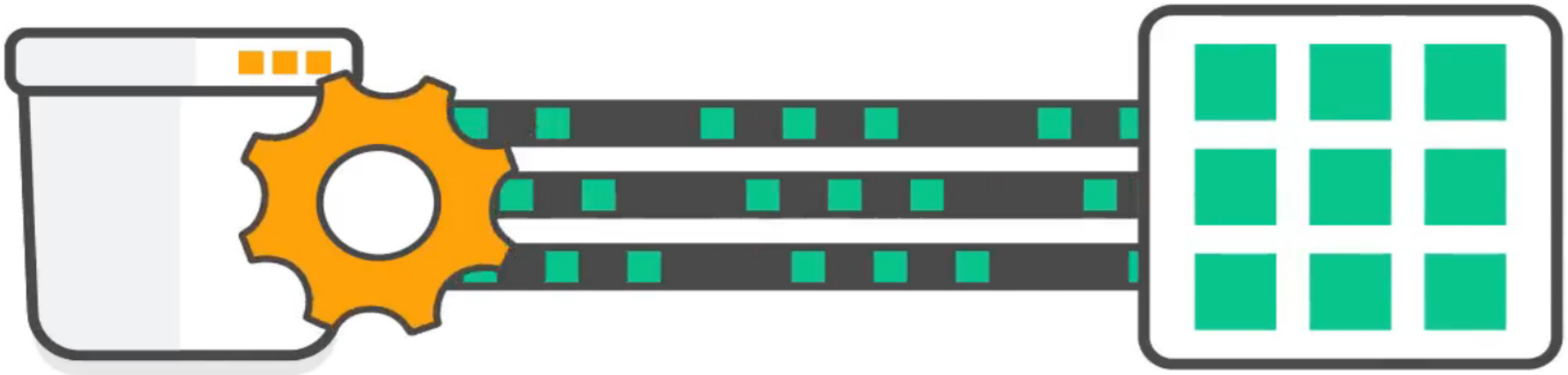
- You can automatically archive data from Amazon S3 into Glacier using lifecycle policies
- These policies will archive the data to Glacier based on whatever rules you have specified such as how long the data has been stored in S3, or specific data range when the data was stored

➢S3 objects will transition to S3 Glacier in the designed time, but these remain as S3 objects that you manage in S3



S3 Glacier Select

- S3 Glacier jobs can perform a select query on an archive, retrieve an archive, or get an inventory of a vault as asynchronous operations
- But S3 Glacier select function allows queries to run directly on data stored in S3 Glacier without having to retrieve the entire archive
- It changes the value of archive storage by allowing you to process and find only the bytes you need out of the archive to use for analytics
  - Now, your analytics application can retrieve only the relevant data for your query from the S3 Glacier archive



S3 Glacier usage charges

- Customers can store data for as little as $0.004 per gigabyte per month, a significant savings compared to on-premises solutions.
  - To keep costs low yet suitable for varying retrieval needs, S3 Glacier provides three options for access to archives, from a few minutes to several hours.
- S3 Glacier. You pay for:
  - Objects stored
  - Retrieving objects that are stored in S3 Glacier storage
  - Deleting an object stored before the 90 day minimum storage commitment hast passed
- For each object archived in S3 Glacier, S3 requires 8 KB of store and maintain the user-defined name and metadata for the object. This is charged at S3 standard rate
- S3 Glacier requires an additional 32 KB of data per object for S3 Glacier's index and metadata so you can identify and retrieve your data, this is charged in the S3 Glacier rate

AWS Storage Gateway

Transparent use of Cloud storage

Storage Gateway

- Storage Gateway is a hybrid storage service that enables your on-premises applications to seamlessly use AWS cloud storage
- You can use the service for backup and archiving, disaster recovery, cloud data processing, storage tiering (storage in different levels), and migration
- The service helps you reduce and simplify your datacenter and remote office storage infrastructure
- Your applications connect to the service through a virtual machine or hardware appliance using standard storage protocols, such as NFS, SMB and iSCSI
- The service includes:
  - a highly-optimized data transfer mechanism, with bandwidth management, automated network resilience, and efficient data transfer
  - a local cache for low-latency on-premises access to your most active data

Storage Gateway

- It connects storage services:
  - ✓ S3
  - ✓ S3 Glacier
  - ✓ EBS snapshots
- To provide storage for:
  - ✓ Files
  - ✓ Volumes
  - ✓ Tapes

within AWS

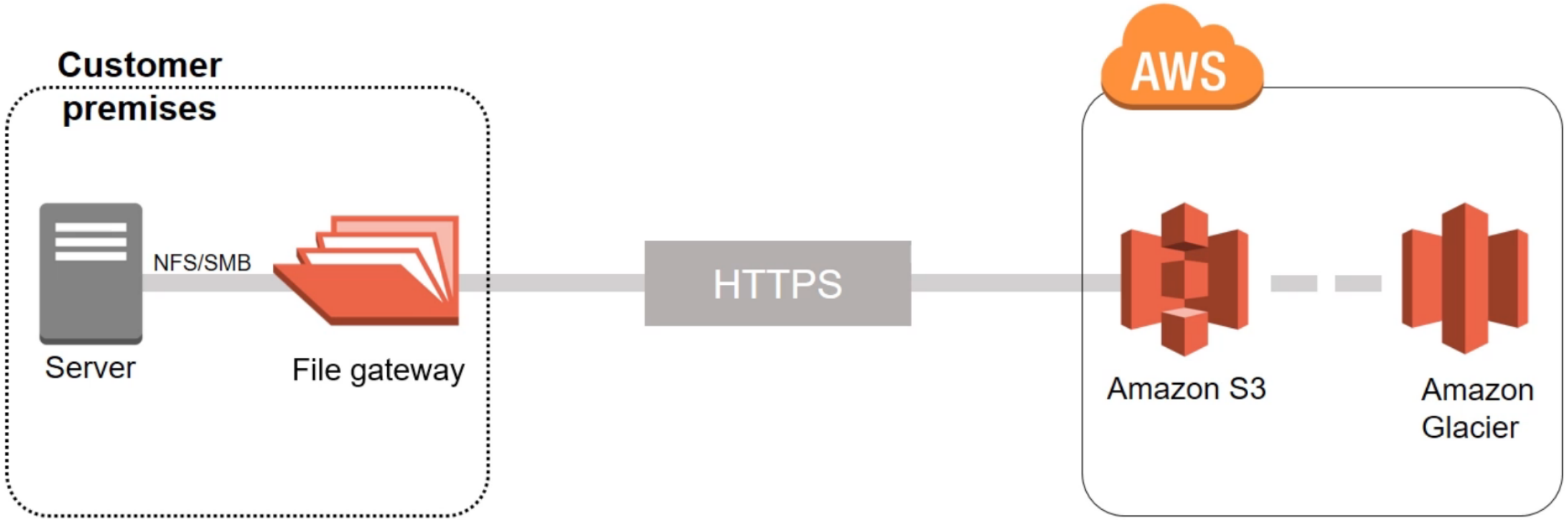
Storage Gateway interfaces

- File gateway interface
  - Store files as objects in Amazon S3, with a local cache for low-latency access to your most used data
- Volume gateway interface
  - Block storage in Amazon S3 with point-in-time backups as Amazon EBS snapshots
- Tape gateway interface
  - Back up your data to Amazon S3 and archive in Amazon Glacier using your existing tape-based processes

➢ These interfaces are integrated into a virtual machine (VM) that it is deployed into your data center and connects to AWS storage
  - You start by downloading a VM image for the gateway, then you active it from AWS console
  - After the VM activation, you configure your file / volume / tape share and associate it with you S3 bucket

File Gateway

- Allows to store / retrieve files from S3 via NFS / SMB and HTTPS
- As your on-premises files change, they are asynchronously updated as objects in your S3 bucket (with a one-to-one mapping between them)
  - ✓ Existing objects in the S3 bucket appear as files in the file system

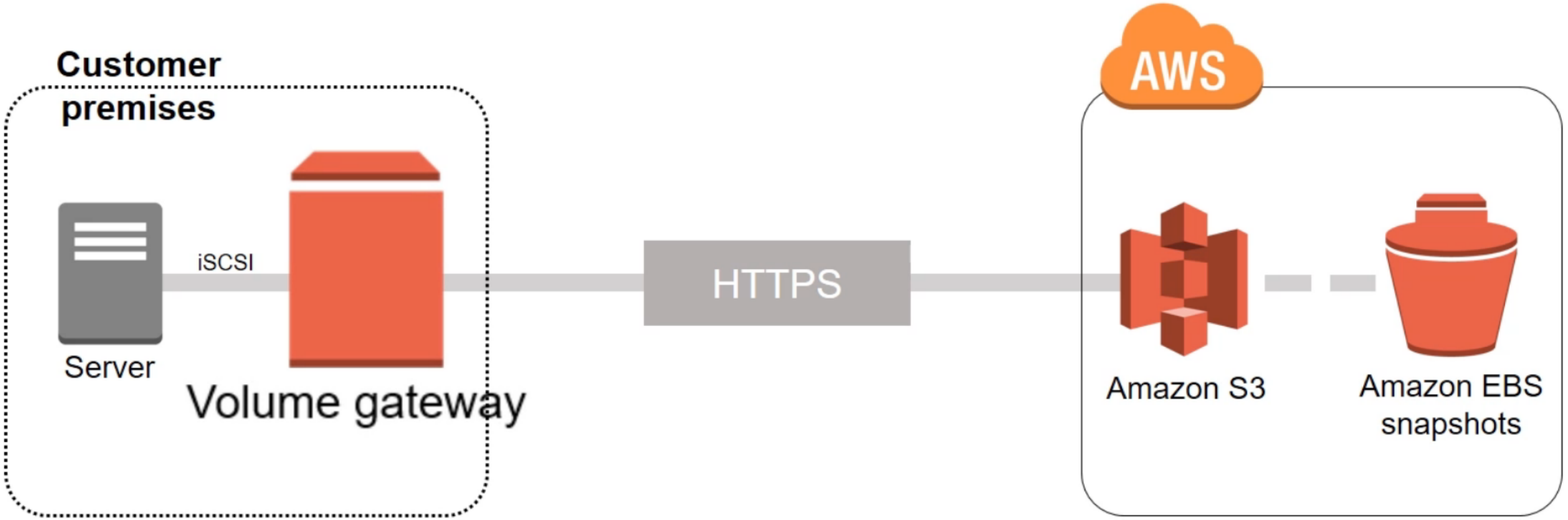


Storage Gateway: File gateway

- The file gateway interface allows you to store and retrieve files from S3 using industry standard file protocols
  - NFS (Network File System) and Server Message Block (SMB)
- Once your files are transferred to S3, they are stored as objects and they are accessed through a secured network system (HTTPS)
  - There, they can be used as S3 objects and things like versions, life cycle management and cross region replication apply directly to them
- There is a one-to-one mapping between files and objects, and the gateway asynchronously updates the objects in S3 as you change the files
  - Existing objects appear as files in the file system, and the key becomes the path
- The file gateway not only manages data transfer to / from AWS, but buffers applications from network congestion, optimizes and streams data in parallel, and manages bandwidth consumption

Volume Gateway

- Your applications work with block storage volumes using iSCSI protocol
- Data is backed up as point-in-time snapshots objects of your volumes
  - ✓ They stored in AWS cloud as EBS snapshots

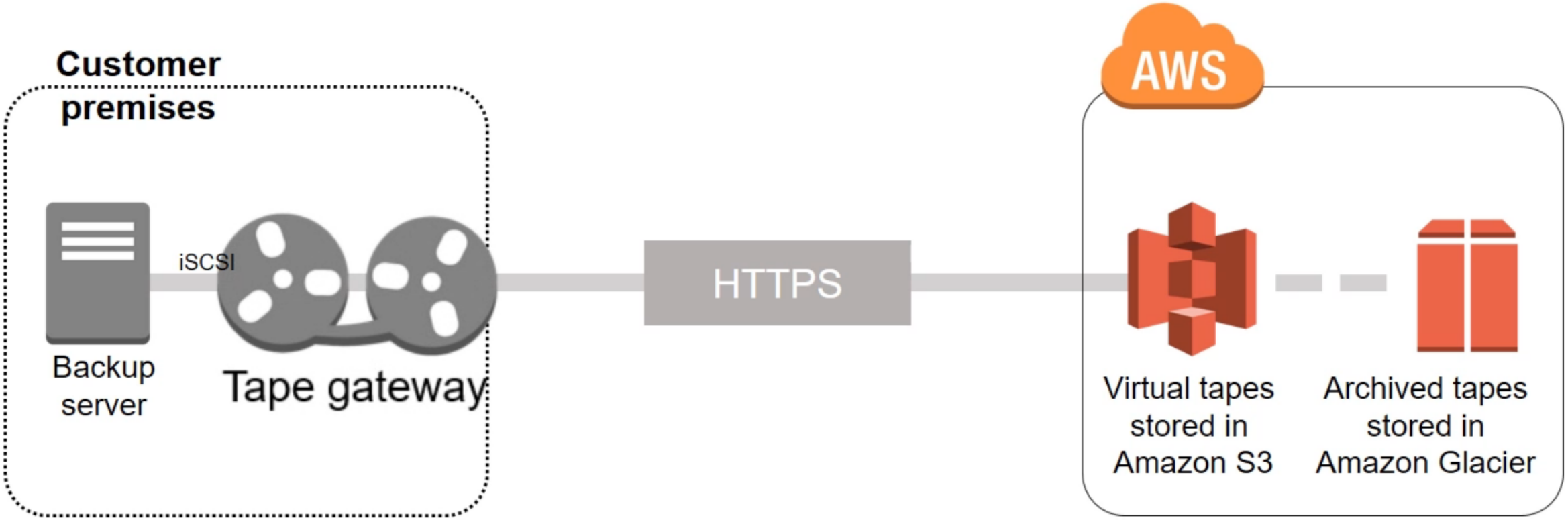


Storage Gateway: Volume gateway

- Volume interface presents your applications with disk volumes using industry protocols
  - iSCSI (Small Computer System Interface)
- Data on these volumes can be simultaneously backed up as point-in-time snapshots, and stored in the cloud as EBS snapshots
  - Snapshots are incremental backups that only capture changed blocks
- When connecting with this block interface you can run the gateway in 2 different modes: cached and stored
- In cached mode, your primary data is stored in S3 and you retain your frequently accessed data locally.
  - This result in substantially cost saving for primary storage because it minimizes the need to scale your storage on-premises retaining low latency access to your frequently accessed data.
- In stored mode, you store your entire data set locally while performing asynchronous backups of this data in S3.
  - This provides durable and inexpensive offsite backups that you can recover locally or from Amazon EC2

Tape Gateway

- Your backup application works with virtual tapes using iSCSI protocol
- Data is backed up as S3 objects and archived tapes are stored in Glacier
  - ✓ Glacier allows additional cost reduction



Storage Gateway: Tape gateway

- Tape interface presents the storage gateway to your existing backup application as a virtual tape library
- A virtual tap library consists of:
  - ✓ A virtual media changer - Analogous to a robot that moves tapes around in a physical tape library's storage slots and tape drives
  - ✓ Virtual tape drives - Analogous to a physical tape drive that can perform I/O and seek operations on a tape
- You can continue to use your existing backup applications while writing to an almost limitless collection of virtual tapes
- Each virtual tape is stored in S3 as an object
- When you no longer require access to data on virtual tapes, your backup application can archive it, from the virtual tape library, into Amazon Glacier.
  - This further reduces storage costs.

Storage Gateway key use cases

- Offsite Backup
  - Done via data snapshots with on-premises recovery
  - ✓When you need to restore a backup, just specify the S3 snapshot you want to restore, and the Storage Gateway will download the data to your local storage volume
  - Because it is a compatible SCSI interface, it works with your existing backup applications
  - ✓From your backup application, you can directly point to a Storage Gateway volume as it simply represents another disk drive
- Disaster recovery
  - Disaster recovery snapshots access in EC2 in a seamlessly way
  - If you keep your data as EBS snapshots, in the event your on-premises infrastructure goes down, you can easily re-construct operations in EC2
- Data mirroring
  - Run your applications in the cloud using your uploaded data without worrying about synchronization with your on-premises data

AWS Snow Family

Offline storage solutions for massive data

Snow Family

➢The Challenge

- Data is foundational for digital projects in the cloud, but moving large volumes of data is a serious challenge
- A dedicated 1Gbps network connection theoretically moves 1PB of data in 120 days
  - In 24h you can theoretically move 86400 GB
- Moving enterprise data centers to the cloud or migrating data-intensive analytics environments requires bulk data transport methods that are simple, affordable, secure and tamper-resistant

Quantities of bytes						
Common prefix				Binary prefix		
Name	Symbol	Decimal SI	Binary JEDEC	Name	Symbol	Binary IEC
kilobyte	KB/kB	10^3	2^{10}	kibibyte	KiB	2^{10}
megabyte	MB	10^6	2^{20}	mebibyte	MiB	2^{20}
gigabyte	GB	10^9	2^{30}	gibibyte	GiB	2^{30}
terabyte	TB	10^{12}	2^{40}	tebibyte	TiB	2^{40}
petabyte	PB	10^{15}	2^{50}	pebibyte	PiB	2^{50}
exabyte	EB	10^{18}	2^{60}	exbibyte	EiB	2^{60}
zettabyte	ZB	10^{21}	2^{70}	zebibyte	ZiB	2^{70}
yottabyte	YB	10^{24}	2^{80}	yobibyte	YiB	2^{80}

Snow Family

➢The solution: Snow family

- The Snow family offers a number of physical devices that help physically transport up to exabytes of data in and out of AWS

- You receive a device in your site, after requesting the service in the AWS console, where you fill it with data and return it to the AWS region where it came from. The end-to-end tracking is handled via Amazon SNS, text messages, or AWS console.

- These devices are designed to be secure and tamper-resistant while on site or in transit
  - Hardware and software is cryptographically signed and all data stored is automatically encrypted using 256-encryption keys owned and managed by you
  - Upon completion devices are erased using NIST media sanitation guidelines

Snowball	Snowball Edge	Snowmobile
Data transfer service built around a secure suitcase-sized device that moves data into and out AWS Cloud.	Data transfer service with on-board S3-compatible storage and compute support running AWS Lambda functions and Amazon EC2 instances.	Exabyte-scale data transfer service used to move extremely large amounts of data: up to 100PB of data (equivalent to 1250 Snowball devices) per Snowmobile.
Data filled in the device, is transferred into your S3 bucket. The total process takes a week.	Two options: compute optimized and storage optimized.	An almost 14 m long shipping container pulled by a semi-trailer truck that can connect to your Data Center with fiber cable.
Used to ship TB or PB of analytics data, scientific data, video and image libraries, migrations, backups…	Used in remote locations (military, maritime ops) or in environments with intermittent connectivity (manufacturing, transportation)	Used to migrate a complete Data Center
		

Feature comparison

	AWS Snowball	AWS Snowball Edge	AWS Snowmobile
Usage Scenario	Data Migration	Data Migration with Onboard pre-processing Options	Data Migration
Storage Capacity	50TB and 80TB	100TB	100PB
Onboard Computing Options	N/A	AWS Lambda Amazon EC2 AMIs	N/A
Encryption	Yes, 256-bit	Yes, 256-bit	Yes, 256-bit
Transfers via NFS	N/A	Yes	Yes
Transfers via HDFS	Yes	N/A	N/A
Transfers via S3 API	Yes	Yes	No
Clustering	N/A	Yes, up to 20 nodes	N/A
Rack-mountable	Shelf	Yes	N/A
HIPAA Compliant	Yes, eligible	Yes, eligible	No
Typical Job Lifetime	Days-Weeks	Data Migration: Days-Weeks Local Compute: Weeks-Months	Weeks-Months
Max Job Length	90 Days	Data Migration: 90 Days Local Compute: 120 Days	120-360 Days

Amazon EBS

Local storage for Amazon EC2

EBS (Elastic Block Storage)

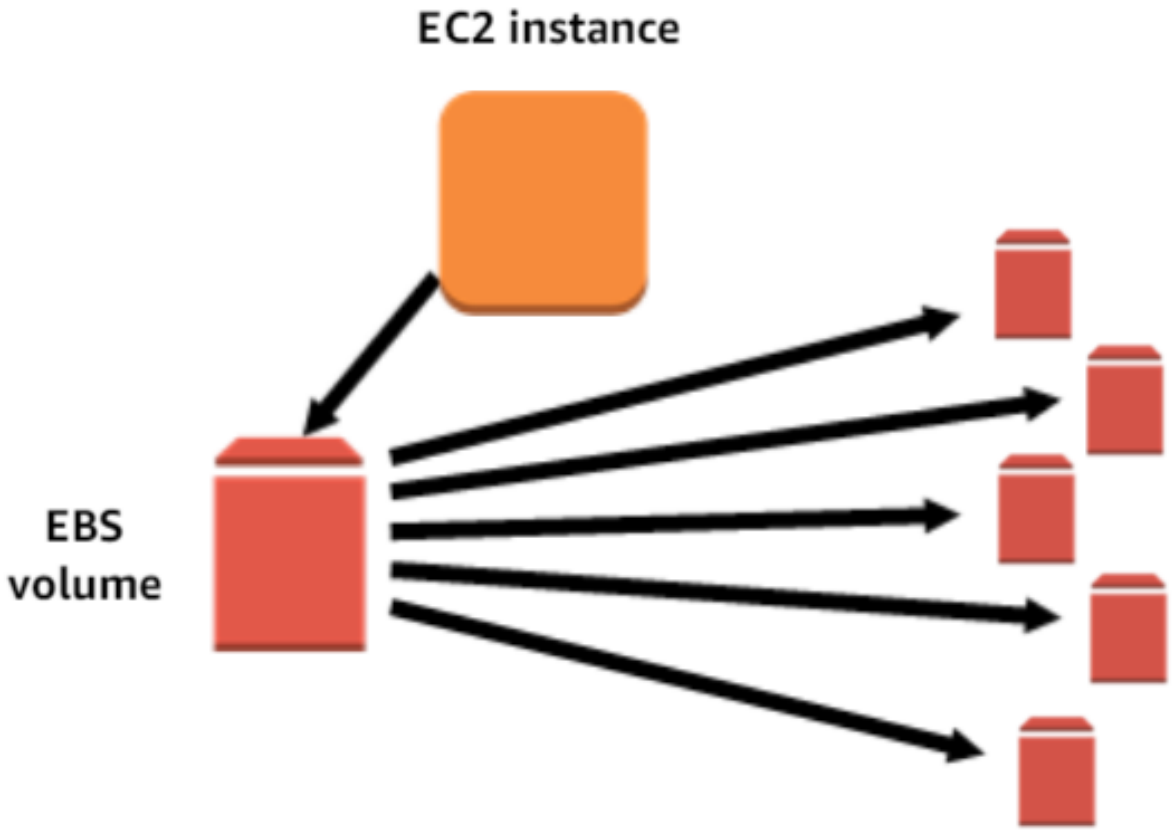
- EBS provides persistent block storage volumes for use with EC2 instances in the AWS Cloud
- Each Amazon EBS volume is automatically replicated within its Availability Zone to protect you from component failure, offering high availability and durability
- EBS volumes offer the consistent and low-latency performance needed to run your workloads
- With EBS, you can scale your usage up or down within minutes – all while paying a low price for only what you provision
- EBS is designed for application workloads that benefit from fine tuning for performance, cost and capacity

EBS Features

- Reliable, secure storage
  - EBS volumes provide redundancies within its AZ to protect against failures
  - Encryption and access control policies deliver a strong defense security strategy for your data
- Consistent, low-latency performance
  - EBS SSD volumes and EBS provisioned IOPS volumes deliver low-latency through SSD technology and consistent I/O performance scaled to the needs of your application
- Backup, restore, innovate
  - Protect your data by taking point-in-time snapshots of your Amazon EBS volumes
- Quickly scale up, easily scale down
- Geographic flexibility
  - EBS provides the ability to copy snapshots across AWS regions
- Optimized performance
  - EBS-optimized instance provides dedicated network capacity for EBS volumes

EBS Volume

- When an EBS is attached to an instance it looks like a disk
- But an EBS volume is not a single physical disk drive.
  - ➢EBS is a distributed system
  - ➢Each EBS volume is made up of multiple, physical devices
- By distributing volume across many devices, EBS service provides added performance and durability
  - ➢EBS volumes can be attached to any EC2 instance in the same AZ



EBS Volumes

- EBS volume can be attached to only one instance at a time
- Many volumes can attach to a single instance

✓Separate a boot volume from data volumes

- Recommended when data must be:

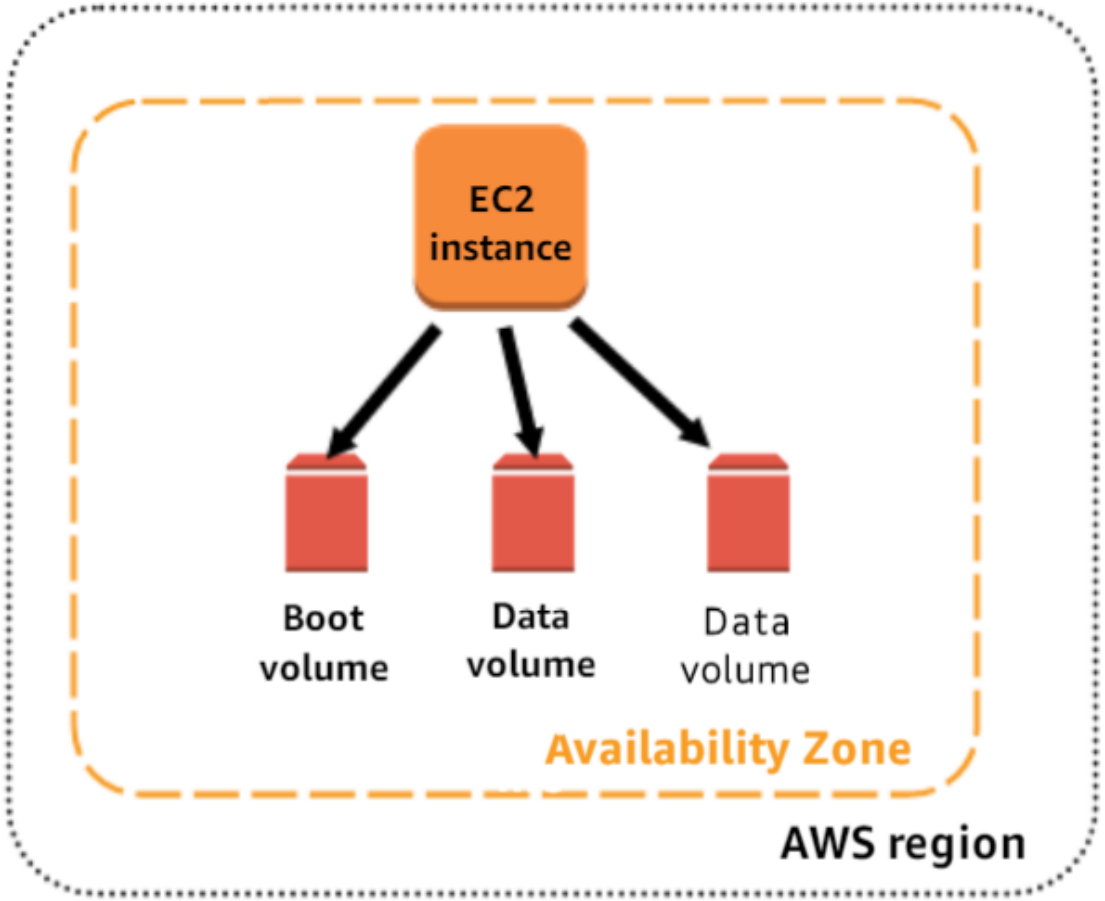
✓quickly accessible

✓requires long-term persistence

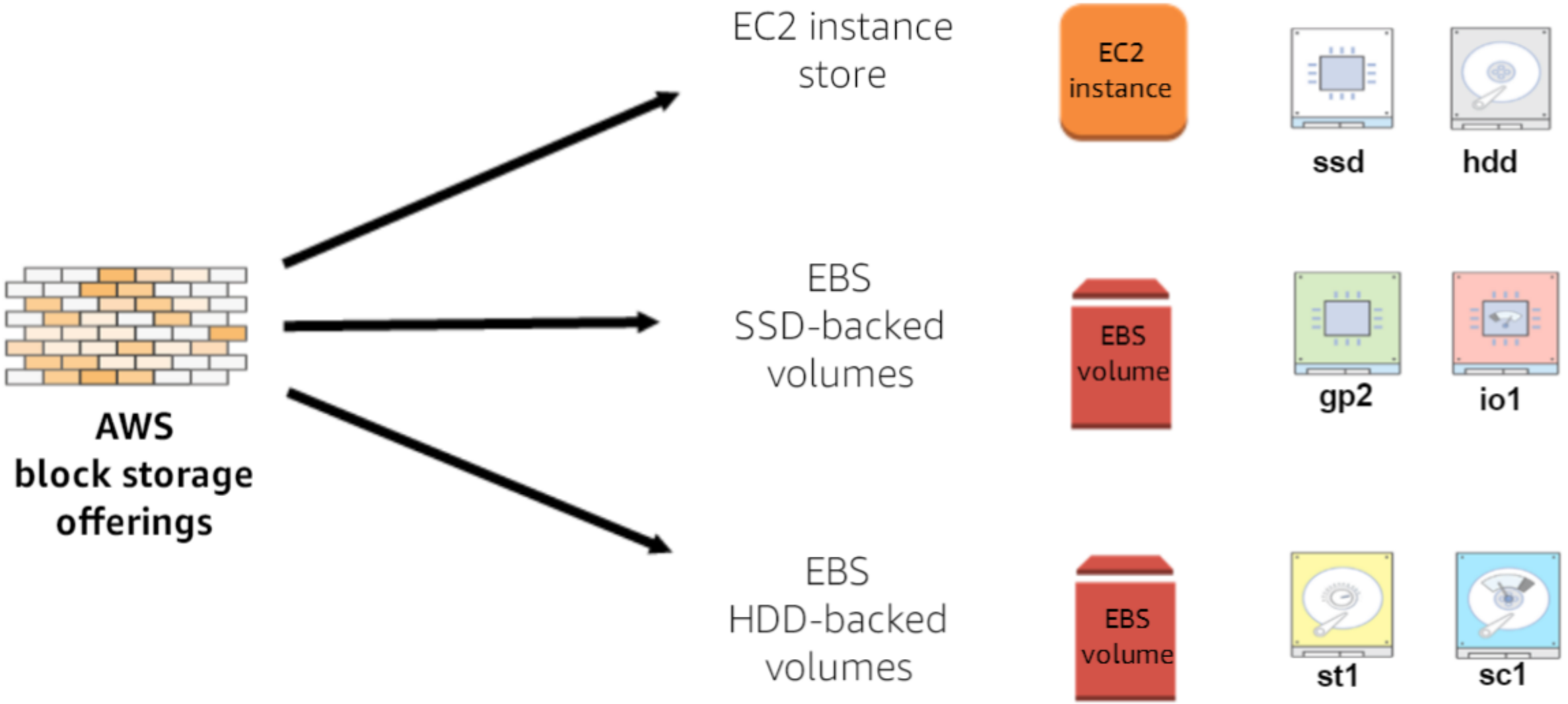
➢Data on EBS volume persists

✓You can detach and attach instances between EC2 instances in the same AZ

- Designed for high levels of volume availability and durability (99.999%)



Block storage offerings



EC2 instance store

- Provides ephemeral (non-persistent) block-storage for your instance
- This storage is located on disks that are physically attached to the host computer
- A good option when you need storage with very low latency
  - Only supported by C3, G2, HI1, I2, I3, M3, R3, and X1 instance families
- Ideal for temporary storage of information that changes frequently
  - Buffers
  - Caches
  - Scratch data and other temporary content
  - Replicated data across a fleet of instances
- Consists of one or more instance store volumes exposed as block devices
  - The size of an instance store and the number of devices available varies by instance type
- Although an instance store is dedicated to a particular instance, the disk subsystem is shared among instances and a host computer

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

EBS Volume Types

- General Purpose SSD (GP2)
  - Base performance of 3 IOPS/GiB, with the ability to burst to 3,000 IOPS for extended periods of time. GP2 volumes support up to 10,000 IOPS and 160 MB/s of throughput
- Provisioned IOPS SSD (IO1)
  - A specific level of I/O performance. Support up to 32,000 IOPS and 500 MB/s of throughput. For I/O intensive applications
- Throughput Optimized HDD (ST1)
  - Low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. With throughput of up to 500 MiB/s, it is a good fit for large, sequential workloads such as Amazon EMR, ETL, data warehouses, and log processing (not for boot volume)
- Cold HDD (SC1) volumes
  - Low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. With throughput of up to 250 MiB/s, sc1 is a good fit ideal for large, sequential, cold-data workloads (not for boot volume)

EBS Volume Types

	Solid-State Drives (SSD)		Hard Disk Drives (HDD)	
Volume Type	General Purpose SSD (gp2)*	Provisioned IOPS SSD (io1)	Throughput Optimized HDD (st1)	Cold HDD (sc1)
Description	General purpose SSD volume that balances price and performance for a wide variety of workloads	Highest-performance SSD volume for mission-critical low-latency or high-throughput workloads	Low-cost HDD volume designed for frequently accessed, throughput-intensive workloads	Lowest cost HDD volume designed for less frequently accessed workloads
Use Cases	• Recommended for most workloads • System boot volumes • Virtual desktops • Low-latency interactive apps • Development and test environments	• Critical business applications that require sustained IOPS performance, or more than 16,000 IOPS or 250 MiB/s of throughput per volume • Large database workloads, such as: ◦ MongoDB ◦ Cassandra ◦ Microsoft SQL Server ◦ MySQL ◦ PostgreSQL ◦ Oracle	• Streaming workloads requiring consistent, fast throughput at a low price • Big data • Data warehouses • Log processing • Cannot be a boot volume	• Throughput-oriented storage for large volumes of data that is infrequently accessed • Scenarios where the lowest storage cost is important • Cannot be a boot volume
API Name	gp2	io1	st1	sc1
Volume Size	1 GiB - 16 TiB	4 GiB - 16 TiB	500 GiB - 16 TiB	500 GiB - 16 TiB
Max. IOPS/Volume**	16,000***	64,000****	500	250
Max. Throughput/Volume	250 MiB/s***	1,000 MiB/s†	500 MiB/s	250 MiB/s
Max. IOPS/Instance††	80,000	80,000	80,000	80,000
Max. Throughput/Instance††	1,750 MiB/s	1,750 MiB/s	1,750 MiB/s	1,750 MiB/s
Dominant Performance Attribute	IOPS	IOPS	MiB/s	MiB/s

EBS Snapshots

- EBS provides the ability to create snapshots of any EBS volume and write a copy of the data to S3, where it is stored redundantly
  - ❖Snapshot = point-in-time incremental copy of a volume
- You can create AMI's from EBS-backed instances and Snapshots
- You can change EBS volumes sizes on the fly, including changing the size and storage type
- To move an EC2 volume from one AZ / region to another, take a snapshot or an image of it, then copy to a new AZ / region
- Encryption:
  - Snapshots of encrypted volumes are encrypted by default
  - Volumes restores from encrypted snapshots are encrypted by default.
- You can share snapshots (other AWS accounts or made public), but only if they are unencrypted

EBS Use Cases

- Relational database
  - EBS scales your performance needs for the most used relational database
- Enterprise applications
  - EBS meets the needs of any organization by providing reliable storage to run mission-critical applications
- Development and test
  - EBS provisions, duplicate, scale, or archive your development, test, and production environments
- NoSQL databases
  - EBS volumes provide the consistent and low-latency performance for NoSQL databases
- Business continuity
  - Minimize data loss and recovery times by regularly backing up your data and log files across different geographic regions. Copy EBS snapshots to deploy applications in new AWS regions

Amazon EFS

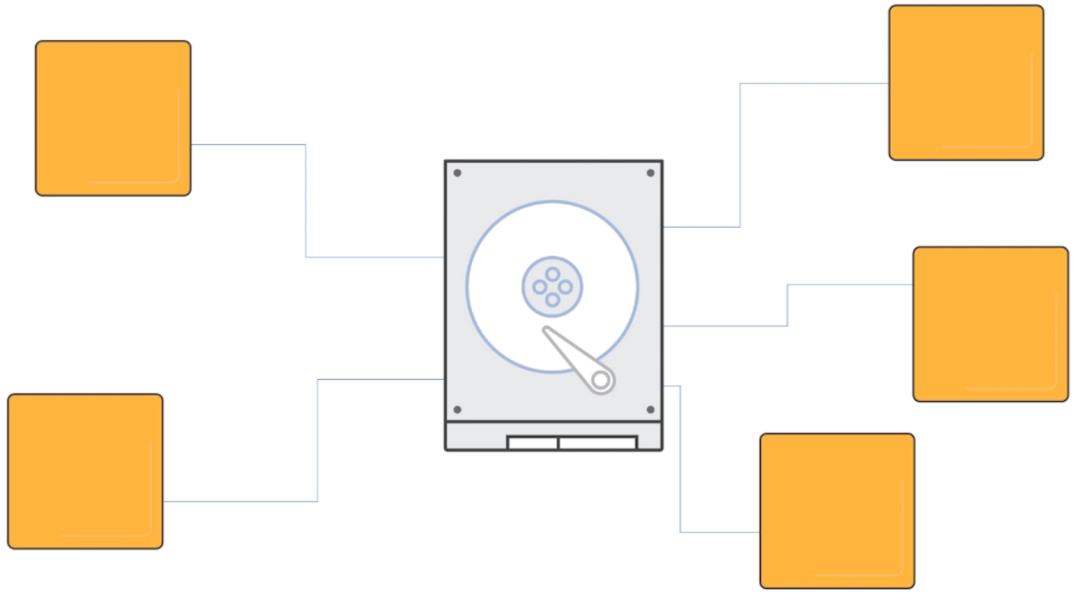
Distributed filesystem in AWS

EFS (Elastic File System)

- EFS provides a simple, secure, scalable, elastic file system for Linux-based workloads to use with AWS Cloud services and on-premises resources.
- A fully managed service that requires no changes to your existing applications and tools, providing access through a standard file system interface
- Provides high availability and durability in the same way as other AWS services
- ➢ Built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files, so your applications have the storage needed
- ➢ Designed to provide parallel shared access to thousands of Amazon EC2 instances
- In the same way as S3, EFS provides classes for standard and infrequent access. Then using lifecycle management, you can move your storage to a cost-optimized file system reducing costs by up to 85%

EFS

- Shareable file system access for EC2 Instances

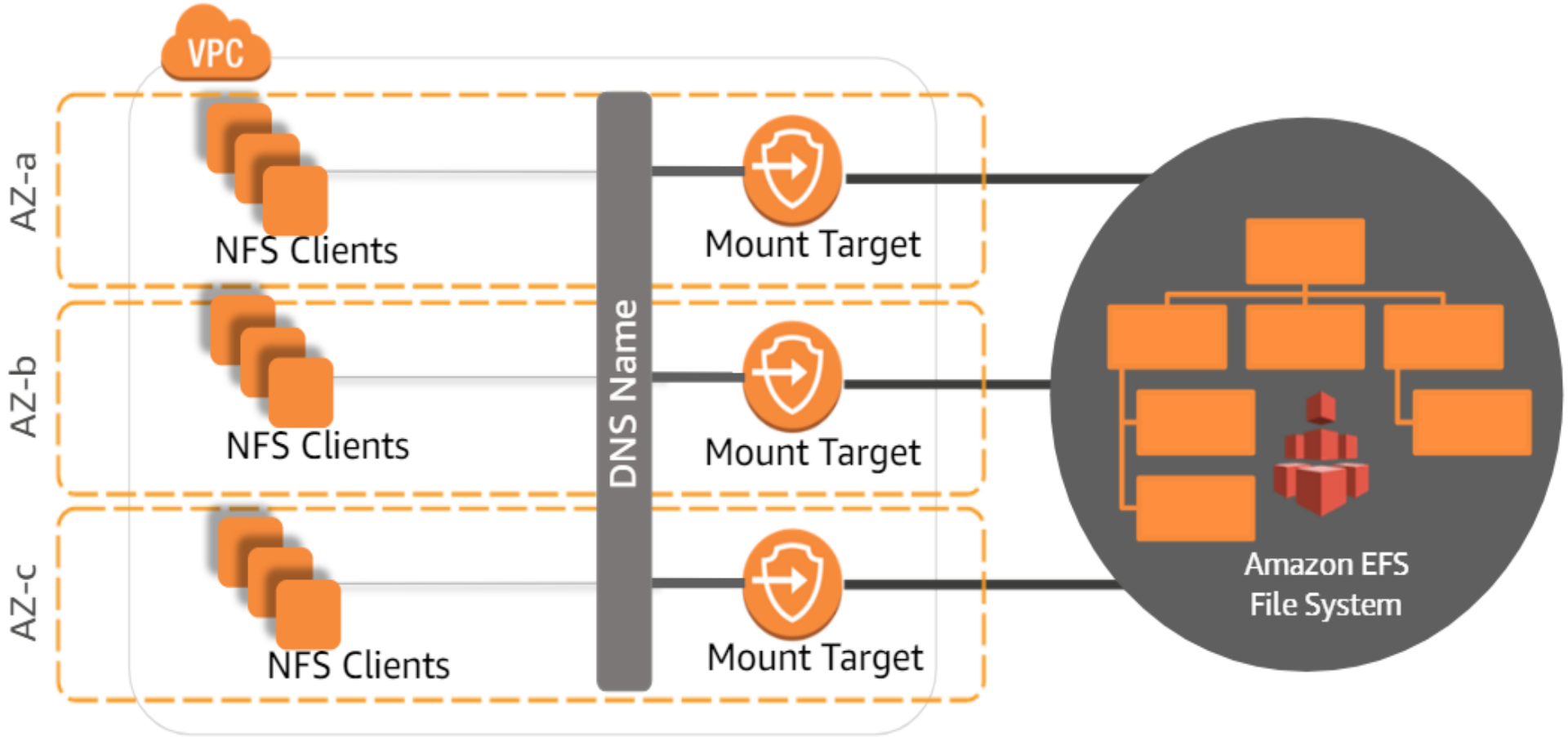


  - You can attach the same logical volume to multiple EC2 instances at a time
  - These EC2 instances can be span along multiple AZ

- When you request an EFS, the service gives you the mount point for your OS
  - Which may run on EC2 instances or on-premises hardware
- Manages file structure
  - You can deploy, share, and secure your file system share quickly (without worrying about infrastructure)
  - Stores data across multiple AZ
- Provides elastic capacity
  - Growing and shrinking automatically as you add or remove files
  - Billing only for capacity used

When to use EFS?

- This is how EFS is able to build a highly available and scalable NFS share



- Create mount targets in each AZ where your EC2 instances reside
- EFS provides a common DNS namespace in a NFS system that your EC2 instances can connect to
- EFS supports NFSv4 protocol
- Charging per GB

When to use EFS?

- Applications running on EC2
  - Provides not only shared, but permanent storage
- On-premises file systems with multi-host attachments
  - That would like to move to AWS file workloads
- High throughput application requirements
  - Multiple Gbps throughput requirements
- Applications that require high availability and durability

➢ It is important to understand your file workload requirements to determine whether EFS is a proper fit for your storage needs

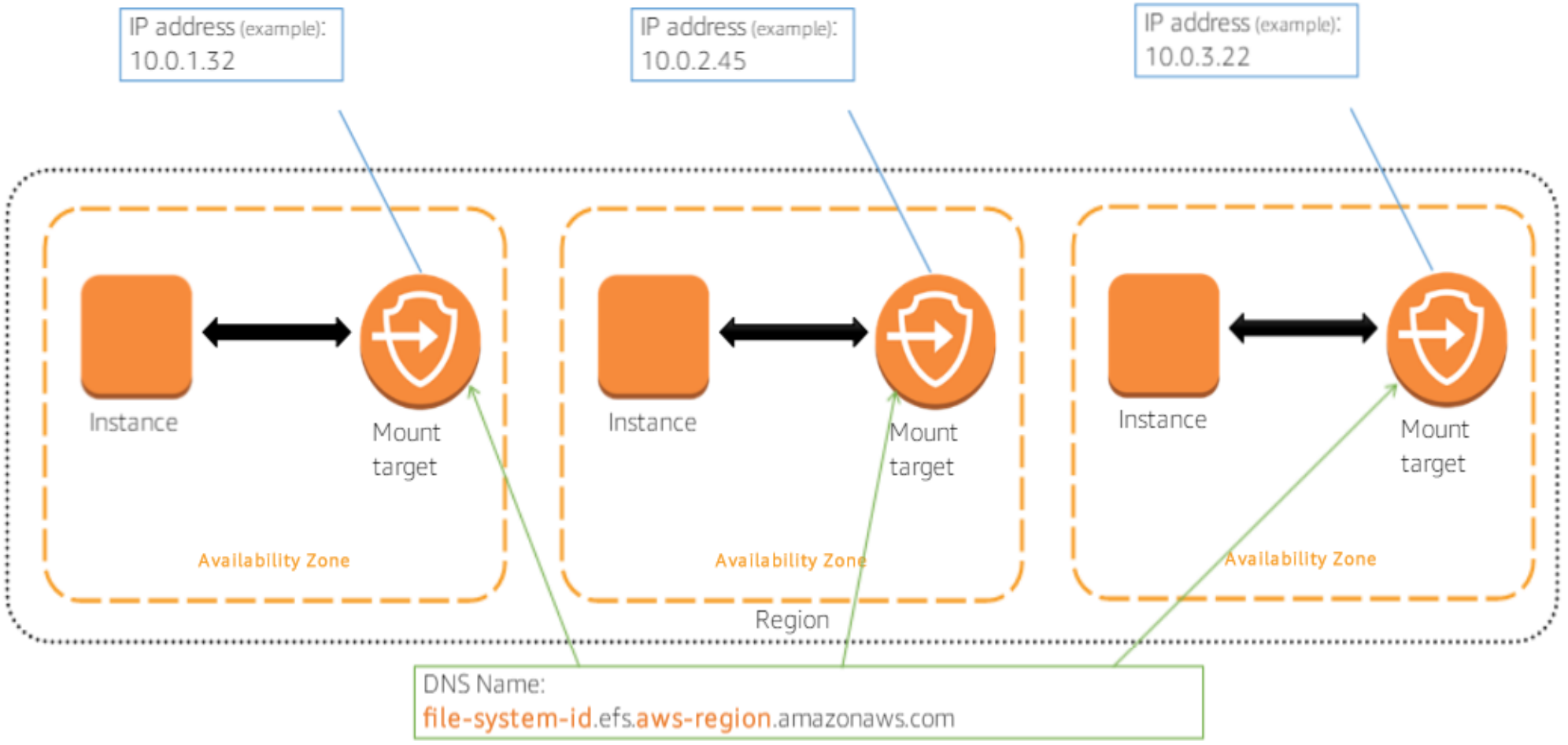
➢ And to choose among the different available modes

EFS options

- You can create up to 125 FS per AWS account
- You can choose between two modes: performance and throughput modes
- Performance modes:
  - General purpose: default and best suited for most workloads
  - Max I/O: recommended for workloads that must scale to higher I/O throughput and I/O operations.

  Ex/ Tens, hundreds, or thousand EC2 instances accessing the same file system.
- Throughput modes:
  - Bursting: default and it is recommended for most workloads
  - Provisioned: recommended for higher throughput to storage ratio

EFS mount target (mount point)



- The mount target has:
  - ✓ IP address
  - ✓ DNS name
  - ➢ DNS name contains the file system id + AWS region where the FS was created

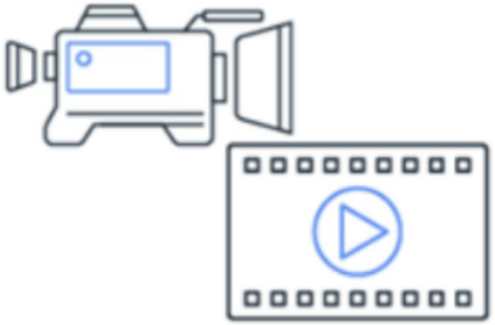
- ❖ A mount target is analogous to an NFS mount point

EFS Use Cases

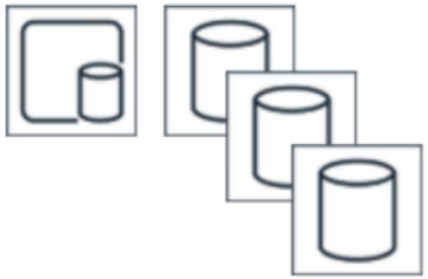
Big data analytics

Web serving and content management

Application testing and development



Media and entertainment



Database backups

Container storage

EFS Use Cases

- Big data analytics
  - Scale and performance file data sharing
- Web serving and content management
  - Durable and high throughput file system
- Application testing and development
  - Provides to development teams a common repository to be able to share code and other files
- Media and entertainment
  - Shared storage for your media (audio, video…)
- Database backups
  - Provides a file system that can be easily mounted from EC2 instances for data servers
- Container storage
  - Persistent shared access to a common file repository

Use EFS, S3 or EBS?

- EBS is a block level storage service to use with EC2
  - It can deliver performance for workloads that require the lowest-latency access to data from a single EC2 instance

Amazon Elastic Block Store (EBS)

- EFS is a file storage service to use with EC2
  - It provides a file system interface, file system access semantics (such as strong consistency and file locking), and concurrently-accessible storage for up to thousands of Amazon EC2 instances

Amazon Elastic File System

- S3 is an object storage service
  - It makes data available through an Internet API that can be accessed anywhere

Amazon Simple Storage Service (S3)

Storage references and documentation

- Amazon S3 documentation
  - https://docs.aws.amazon.com/s3/
- Amazon S3 Glacier documentation
  - https://docs.aws.amazon.com/glacier/
- AWS Storage Gateway documentation
  - https://docs.aws.amazon.com/storagegateway/
- AWS Snowball documentation
  - https://docs.aws.amazon.com/snowball/
- Amazon EFS documentation
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEFS.html
- Amazon EBS documentation
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html