S3	Public access to S3 bucket is disabled by default.
S3 Lifecycles Policies	**Exam Tips** S3 Lifecycle policies are used to ensure you are using the most cost effective option to store your objects in S3. Lifecycle rules are based on object creation date. S3 can transition your objects to **Infrequently Accessed Storage** or to **Glacier** based on the rules you configure. You can also set an expiry date for objects you want S3 to delete after a certain time period has elapsed.
S3 Versioning	• S3 Versioning enables you to revert to older versions of S3 objects. • Multiple versions of an object are stored in the same bucket. • Versioning also protects you from accidental / malicious deletes. • With versioning enabled, a **DELETE** action doesn't delete the object version, but applies a delete marker instead. • To permanently delete, provide the object Version ID in the delete request.
MFA Delete	• It provides an additional layer of protection to S3 Versioning. • Use MFA Delete to protect against accidental or malicious deletions of your version-controlled S3 buckets. • Two things that MFA Delete enforces: 1. Need a valid code from your MFA device to enable permanent deletion of an S3 object. 2. Need a valid code from your MFA device to suspend or reactivate versioning on the S3 bucket.
S3 Encryption	• **Encryption In-Transit** 1. SSL/TLS (HTTPS) Encrypts the data over the network (B/w your PC and S3). • **Encryption At Rest** 1. Server Side Encryption a. SSE-S3 - Amazon S3 managed keys b. SSE-KMS - Amazon KMS managed keys c. SSE-C - Customer managed keys 2. Client Side Encryption Encrypt your file locally before uploading to S3. **Note**: - If you want to enforce the use of encryption for your files stored in S3, use an S3 Bucket Policy to deny all PUT requests that don't include the x-amz-server-side-encryption parameter in the request header.
EC2 Volume Types	**EBS vs Instance Store** • Root device volumes can either be EBS volumes or Instance Store volumes. • An instance store root device volume's maximum size is 10GB

<table>
<tr>
<td></td>
<td>

EBS root device volume can be up to 1 or 2TB depending on the OS.

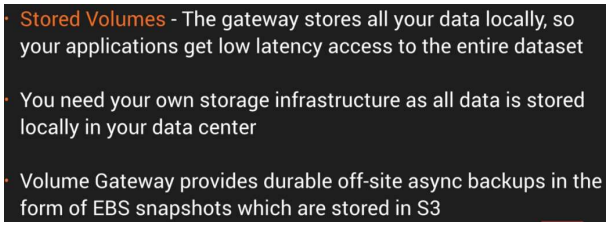
<table>
<tr><th>Characteristic</th><th>Amazon EBS-Backed</th><th>Amazon Instance Store-Backed</th></tr>
<tr><td>Boot time</td><td>Usually less than 1 minute</td><td>Usually less than 5 minutes</td></tr>
<tr><td>Size limit</td><td>1 TiB</td><td>10 GiB</td></tr>
<tr><td>Root device volume</td><td>Amazon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

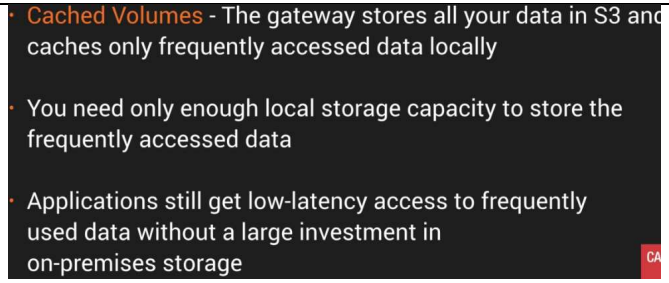
‘Delete on Termination’ is the default for all EBS root device volumes. You can set this to false however but only at instance creation time.
Additional volumes will persist automatically. You need to delete these manually when you delete an instance.
Instance Store is known as ephemeral storage, meaning that data will not persist after an instance is deleted. You cannot set this to false, data will always be deleted when that instance disappears.

</td>
</tr>
<tr>
<td>Volume & Snapshots</td>
<td>
・ Volumes exist on EBS:
　・ Virtual Hard Disk
・ Snapshots exist on S3.
・ Snapshots are point in time copies of Volumes.
・ Snapshots are incremental – this means that only the blocks that have changed since your last snapshot are moved to S3.
・ If this is your first snapshot, it may take some time to create.
</td>
</tr>
</table>

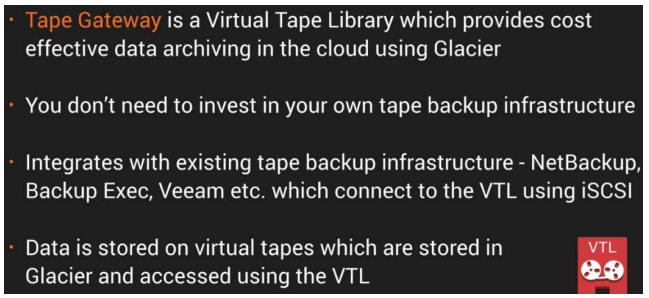
Snapshots of Root Device Volume	· To create a snapshot for Amazon EBS volumes that serve as root devices, you should stop the instance before taking the snapshot. · However you can take a snap while the instance is running. · You can create AMI's from both Images and Snapshots · You can change EBS volume sizes on the fly, including changing the size and storage type. · Volumes will ALWAYS be in the same availability zone as the EC2 instance. · To move an EC2 volume from one AZ/Region to another, take a snap or an image of it, then copy it to the new AZ/Region
Volumes & Snapshots	· Snapshots of encrypted volumes are encrypted automatically. · Volumes restored from encrypted snapshots are encrypted automatically. · You can share snapshots, but only if they are unencrypted. · These snapshots can be shared with other AWS accounts or made public.
Encryption & Downtime	**Enabling Encryption** For most AWS resources, encryption can only be enabled **at creation**. **EFS (Elastic File System) -** If you want to encrypt an EFS filesystem that already exists, you will need to create a new encrypted EFS and migrate your data. **RDS (Relational Database**) - If you want to encrypt an existing RDS, you will need to create a new encrypted database and migrate your data. **EBS Volumes -** Encryption must be selected **at creation time**. You cannot encrypt an unencrypted volume or unencrypt an encrypted volume. So you cannot change the encryption status of EBD volume. You can migrate data between encrypted and unencrypted volumes. **S3 Buckets -** You can enable encryption on your S3 Buckets **at any time**. **S3 Objects -** You can enable encryption individual S3 objects **at any time.** **Exam Tips:** Remember that for the majority of services, you will need to enable encryption at creation time. • EFS • RDS • EBS Volumes • To add an encryption later will involve migrating your data in some way, you may wish to stop your applications at this time. **S3 has greater flexibility, and you can enable for S3 Buckets or Objects at any time and without disrupting your applications.**
KMS & CloudHSM	• Both allow you to generate, store and manage cryptographic leys used to protect your data in AWS.

	• HSMs (Hardware Security Modules) are used to protect the confidentiality of your keys. It is a physical device. It is often used in financial payment systems.it can be used in Credit/Debit Card payment systems. • Both offer a high level of security. **KMS Vs CloudHSM** **KMS:-** • Shared harware, multi-tenant managed service. • Allows you to generate, store and manage your encryption keys. • Suitable for applications for which multi-tenancy is not an issue. • Free-tier eligible. • Encrypt data stored in AWS, including EBS Volume, S3, RDS, DynamoDB etc. **Cloud HSM:-** • Dedicated HSM instance, hardware is not shared with other tenants no Free-Tier. • Allows you to generate, store and manage your encryption keys. • HSM is under your exclusive control within your own VPC. • FIPS 140-2 Level 3 compliance (US Government stnadard for HSMs). • Use cases include: Database encryption, Digital Rights Management (DRM), Public Key Infrastructure (PKI), Authentication and Authorization, Document Signing. And Transaction processing. **Exam Tips:** • Both KMS and CloudHSM enable you to generate, store and manage your own encryption keys to encrypt data stored in AWS. • KMS is multi-tenancy and good for use cases which do not require dedicated hardware. • If your application has a requirement for dedicated hardware for managing keys, use CloudHSM.
AMIs	**Exam Tips:** • It provide a template for launching EC2 instances • You can create your own custom AMI from a customized EC2 insance. • AMIs are region-bound, so if you are attempting to launch an instances in a new region using a custom AMI, make sure you have copied your AMI to the new destination region. **Sharing AWS AMIs:** • AMI can be shared and copied between user accounts. • Generally AMI is stored within S3. • The owner of the source AMI must grant you read permission for the storage in order to enable yout to copy the AMI.

	• Remember the 2 restrictions: **1. Encrypted AMIs -** Copy the underlying snapshot, re-encrypt using your own key and create a new AMI from the snapshot. **2. AMI with an associated Billing Prodcts code -** You cannot directly copy an AMI with an associated **Billing Products** code (Applies to Windows, RedHat and Amis from AWS Marketplace)..
Snowball & Snowball Edge	**Snowball Vs Snowball Edge** **Snowball** is for Data transfer only **Snowball Edge** provides Edge Computing in addition to data transfer. If you have 100s of TB to upload or your data is taking a few days to upload – you need **Snowball** Use **Snowball Edge** if you need to process the data locally before returning the device to AWS.
Storage Gateway	**Storage Gateway** consists of an on-premises software appliance which connects with AWS cloud-based storage to give you a seamless and secure integration between your on premises IT environment and AWS. **Types of Storage Gateway** • **File Gateway - NFS / SMB** a. Files stored as objects in your S3 buckets b. Accessed using NFS or SMB mount point c. To your on-premises systems this appears like a file system mount backed by S3 d. All the benefits of S3 – bucket policies, S3 versioning, lifecycle management, replication etc. e. Low-cost alternative to on-premises storage. • **Volume Gateway (iSCSI)** Volume Gateway provides cloud backed storage which is accessed using iSCSI protocol. Two different Volume Gateway types available. 1. Stored Volumes - Store your all data locally and only backup to AWS  2. Cached Volumes – Use S3 as your primary storage and cache frequently accessed data in your Storage Gateway.



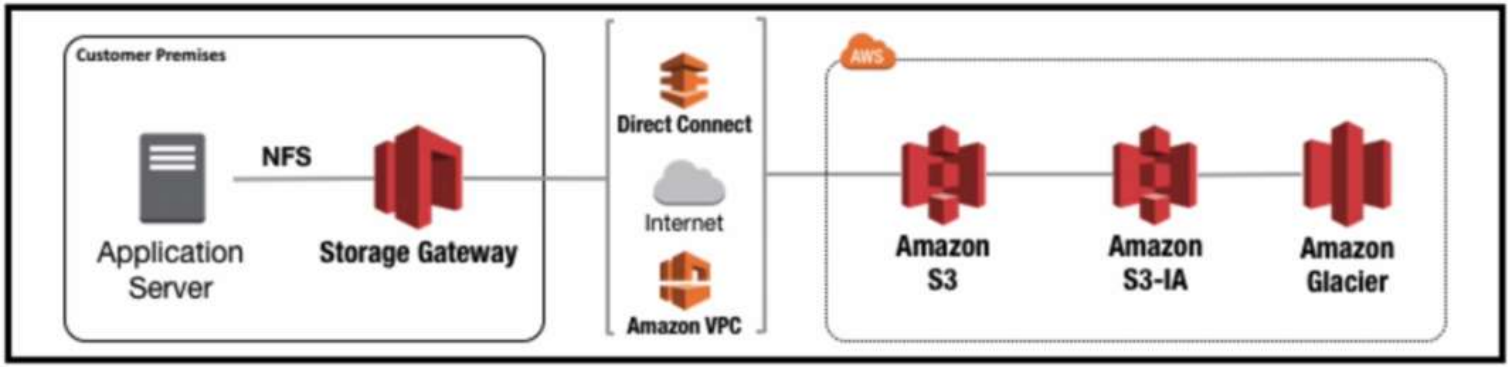
- **Tape Gateway (VTL)**



Exam Tips

- File Gateway - Flat files stored on S3, accessed using NFS or SMB
- Volume Gateway – 2 Types:
  1. Stored Volumes - Entire dataset **stored on-site**, backed-up to S3 as EBS Snapshots.
  2. Cached Volumes - Entire dataset stored in S3, only frequently accessed data **cached on-site**
- Tape Gateway - VTL
  1. Used for archiving your backups to Glacier
  2. Can be used with or without your own backup application.

Storage Gateway Diagram



Athena	**Exam Tips** • Athena is an interactive query service. • Allows you to query data located in S3 using standard SQL • Serverless
S3 Exam Tips	• Remember that S3 is **Object-based (**Object-based storage only for files**)**: i.e. allow you to upload files. • Not suitable to install an operating system or running a database on. • Files can be from 0 Bytes to 5 TB. • There is unlimited storage. • Files are stored in Buckets. • S3 is a universal namespaces. That is, names must be unique globally. • Read after Write consistency for PUTS of new Objects • Eventual Consistency for overwrite PUTS and DELETS (Can take some time to propagate) • S3 Storage Classes/Tiers: - S3 (durable, immediately available, frequently accessed) - S3 - IA (durable, immediately available, infrequently accessed) - S3 - One Zone IA: Same as IA. However, data is stored in a single Availability Zone only. - S3 - Reduced Redundancy Storage (data that is easily reproducible, such as thumbnails, etc.) - Glacier - Archived data, where you can wait 3 - 5 hours before accessing