

Elastic Compute Cloud

Elastic Compute Cloud(EC2)

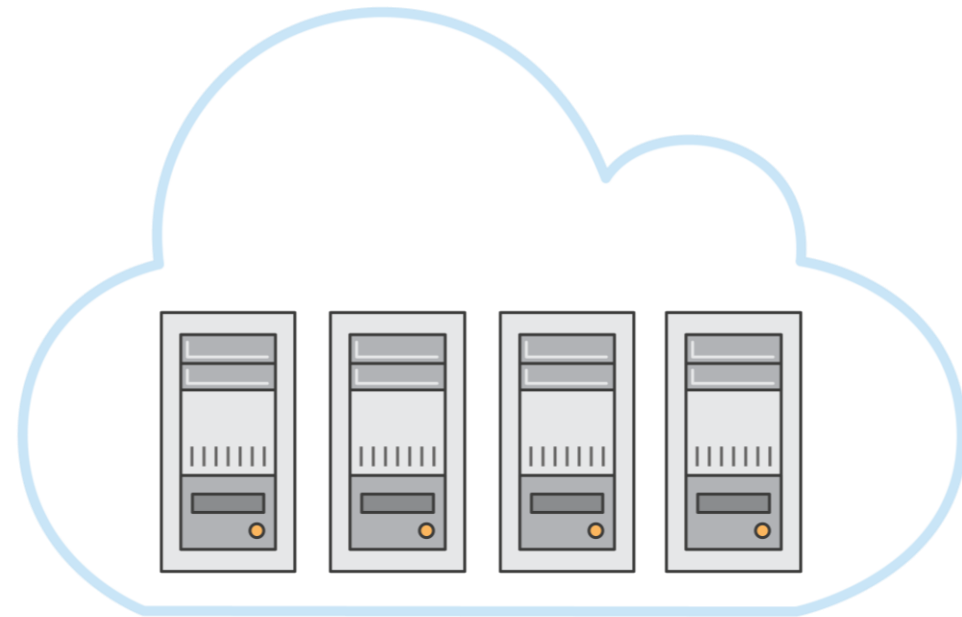
- Virtual computing environments , known as instances you can launch and manage with a few clicks of a mouse or a few lines of code.
- Resizable compute capacity
- Complete control of your computing resources
- Reduced time required to obtain and boot new server instances



EC2 Use Cases

Elastic Compute Cloud

- ✓ Application server
- ✓ Web server
- ✓ Database server
- ✓ Game server
- ✓ Mail server
- ✓ Media server
- ✓ Catalog server
- ✓ File server
- ✓ Computing server
- ✓ Proxy server



Amazon EC2 Facts

- [Scale capacity](#) as your computing requirements change
- Pay only for capacity that [you actually use](#)
- Choose [Linux](#) or [Windows](#)
- Deploy across [AWS Regions](#) and [Availability Zones](#) for reliability
- Use [tags](#) to help manage your Amazon EC2 resources



Launching an Amazon EC2 Instance via the Management Console

1. Determine the AWS Region in which you want to launch the Amazon EC2 instance.
2. Launch an Amazon EC2 instance from a pre-configured Amazon Machine Image (AMI).
3. Choose an instance type based on CPU, memory, storage, and network requirements.
4. Configure network, IP address, security groups, storage volume, tags, and key pair.



Amazon Machine Image Details

An AMI includes the following:

- Template for the **root volume** for the instance.
- **Launch permissions** that control which AWS accounts can use the AMI to launch instances.
- Block device mapping that specifies the **volumes to attach** to the instance when it is launched



AMI Selection

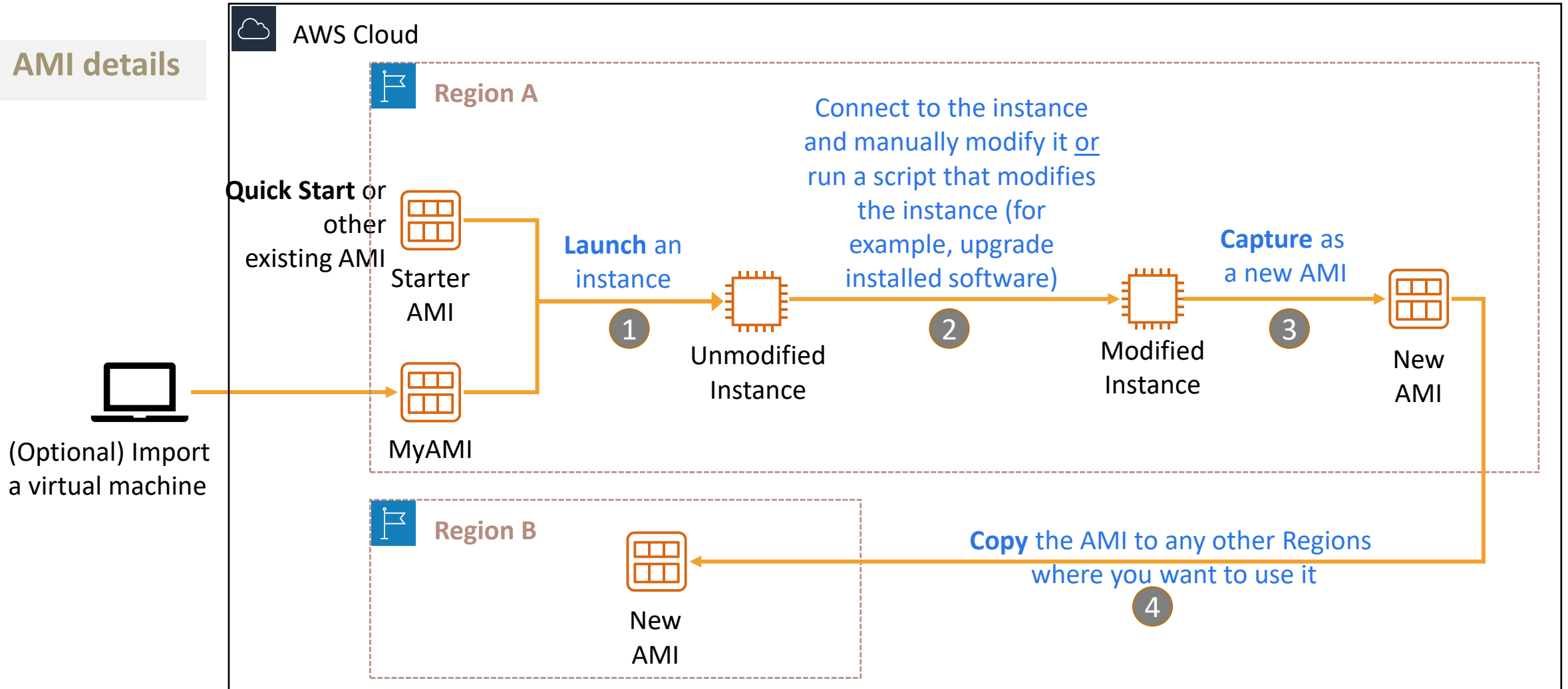
Select an AMI based on:

- ✓ Region
- ✓ Operating system
- ✓ Architecture (32-bit or 64-bit)
- ✓ Launch permissions
- ✓ Storage for the root device



Creating a new AMI

AMI details



Instance Type?

- AWS uses Intel® Xeon® processors to provide customers with high performance and value.
- EC2 instance types are optimized for Different use cases, Workload Requirements and come in Multiple Sizes.
- Consider the following when choosing your instances:
 - Core count
 - Memory size
 - Storage size and type
 - Network performance
 - CPU technologies



EC2 instance type naming and sizes

Instance type details

Instance type naming

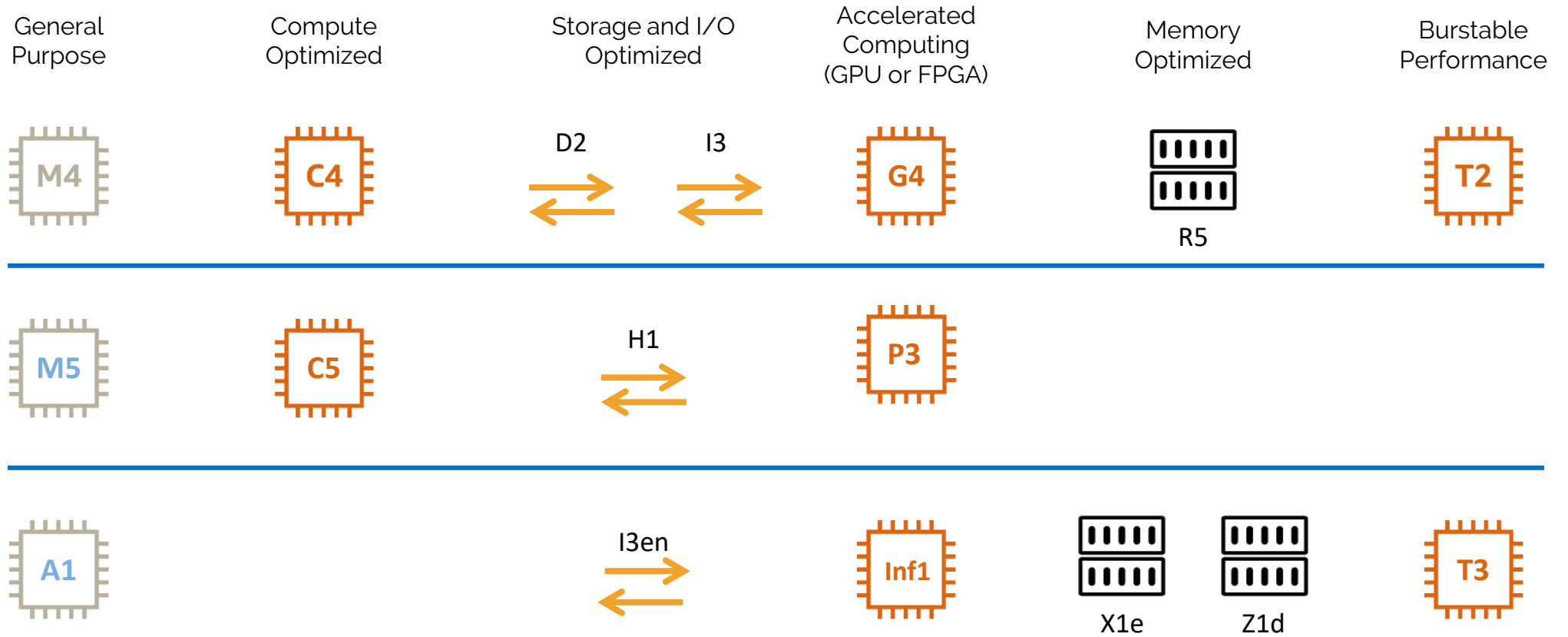
- Example: **t3.large**
 - **T** is the family name
 - **3** is the generation number
 - **Large** is the size

Example instance sizes

Instance Name	vCPU	Memory (GB)	Storage
t3.nano	2	0.5	EBS-Only
t3.micro	2	1	EBS-Only
t3.small	2	2	EBS-Only
t3.medium	2	4	EBS-Only
t3.large	2	8	EBS-Only
t3.xlarge	4	16	EBS-Only
t3.2xlarge	8	32	EBS-Only





Broad Set of Compute Instance Types



Security Groups

- A **security group** is a **set of firewall rules** that control traffic to the instance.
 - It exists *outside* of the instance's guest OS.
- Create **rules** that specify the **source** and which **ports** that network communications can use.
 - Specify the **port** number and the **protocol**, such as Transmission Control Protocol (TCP), User Datagram Protocol (UDP), or Internet Control Message Protocol (ICMP).
 - Specify the **source** (for example, an IP address or another security group) that is allowed to use the rule.

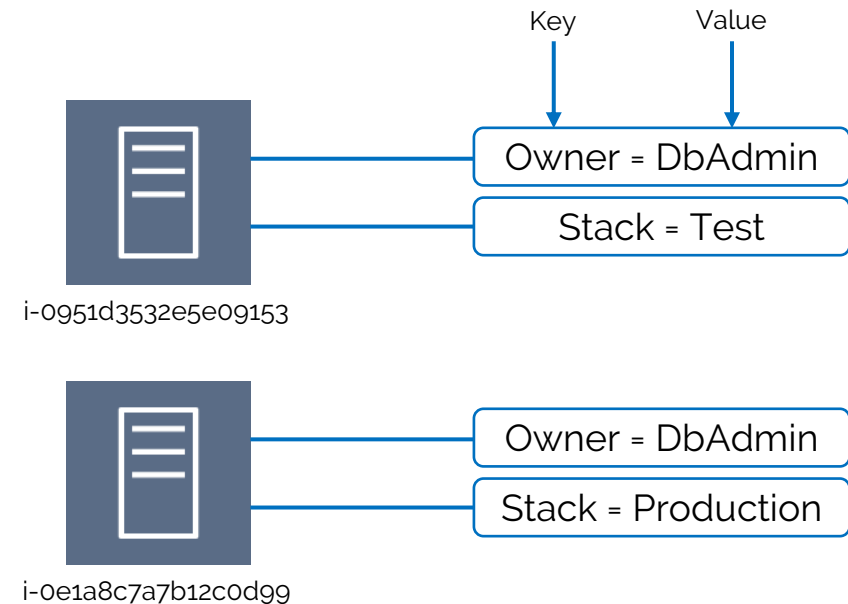
Example rule:

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ
SSH 	TCP	22	My IP  72.21.198.67/32



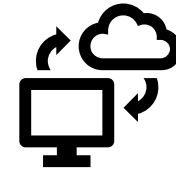
Tagging

- Tags - Identification of your Resources
- Consist of KEY and optional VALUE
- Categorize EC2 Instances based on the Tags

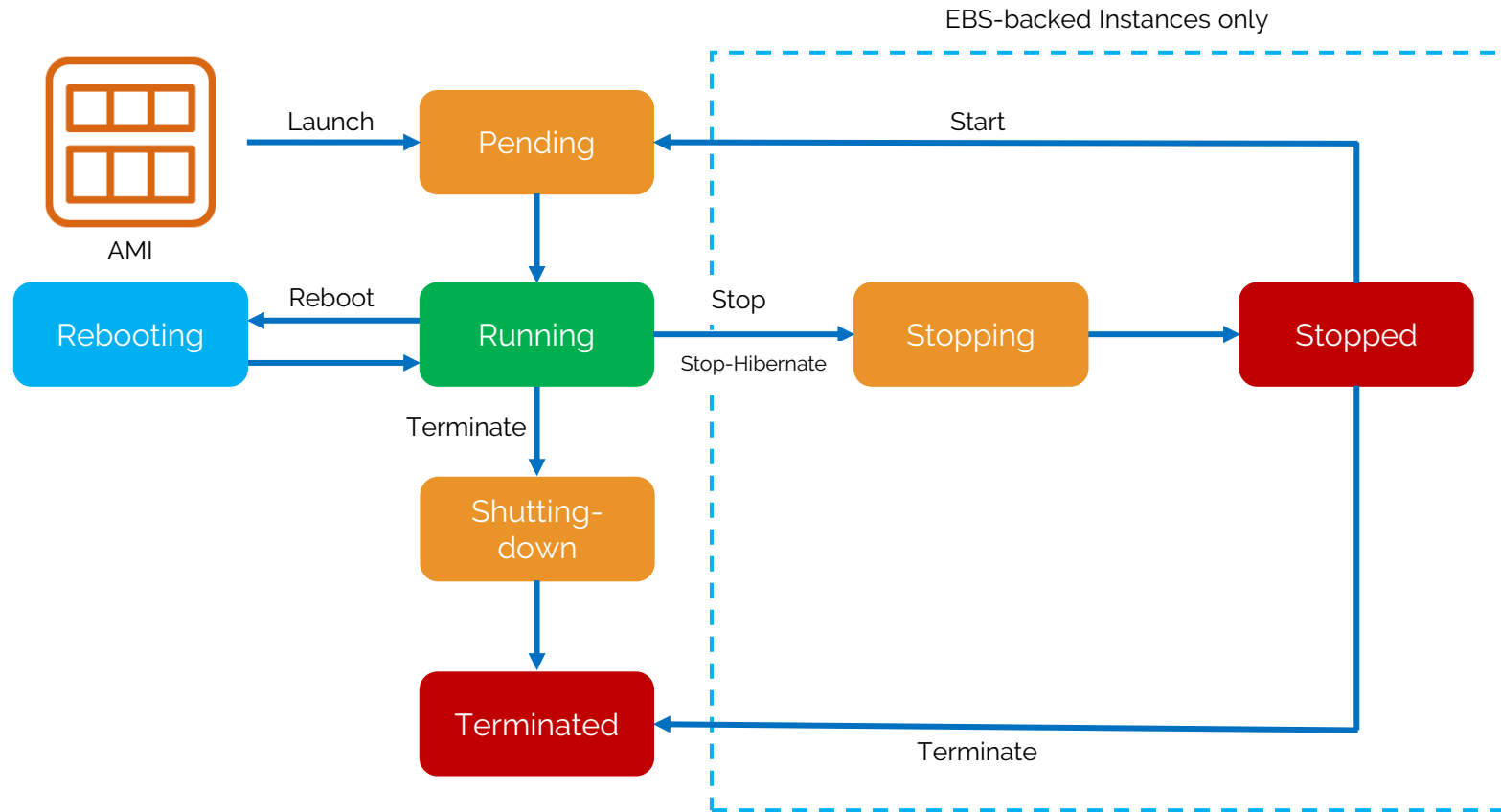


Key Pairs

- At instance launch, you specify an existing key pair *or* create a new key pair.
- A **key pair** consists of –
 - A **public key** that AWS stores.
 - A **private key** file that you store.
- It enables secure connections to the instance.
- For **Windows AMIs** –
 - Use the private key to obtain the administrator password that you need to log in to your instance.
- For **Linux AMIs** –
 - Use the private key to use SSH to securely connect to your instance.



Instance Lifecycle



Amazon EC2 Pricing & Models

Amazon EC2 Pricing

With EC2 you pay for-

- Number of Hours
- Instance Configuration
- Number of Instances
- Data Storage
- Purchasing Model

Purchasing Models



The diagram consists of four rounded square boxes arranged horizontally. The first three boxes (On-Demand, Spot Instances, and Reserved Instances) have a light blue outer border and an orange inner box. The fourth box (Dedicated Hosts) has a light orange outer border and an orange inner box. All inner boxes contain white text.

On-
Demand

Spot
Instances

Reserved
Instances

Dedicated
Hosts

On-Demand Instances

- Start and Stop at any point
- No long-term commitment or upfront payment
- Increase size as per requirement
- Ideal for spiky workloads

Reserved Instances

- Fixed capacity commitment
- Significant discount

Types of Reserved Instances:

Standard Reserved Instance	Convertible Reserved Instance
One-year to three-year term	One-year to three-year term
Modify settings within the same instance type	Exchange 1 or more Convertible Reserved Instances for another Convertible Reserved Instance with new attributes.
Can be sold in the Reserved Instance Marketplace	Cannot be sold in the Reserved Instance Marketplace

- Can pay in different terms: AURI, PURI and NURI

Spot Instances

- Bid for unused capacity
- Discount as much as 90 %
- 2-minute warning before instance shuts off
- Spot Blocks for predefined duration (1-6 hours)
- Spot Instance Hibernation

Dedicated Models

- Dedicated Tenancy
- Two Types based on Instance Placement:
 - Dedicated Instances-Pay for the instances, but no choice in hardware.
 - Dedicated Hosts- Pay for the entire physical server
- Useful when you have a requirement of not using shared hardware.

Hosts Vs Instances

	Dedicated Host	Dedicated Instance
Billing	Per-host billing	Per-instance billing
Visibility of sockets, cores, and host ID	Provides visibility of the number of sockets and physical cores	No visibility
Host and instance affinity	Allows you to consistently deploy your instances to the same physical server over time	Not supported
Targeted instance placement	Provides additional visibility and control over how instances are placed on a physical server	Not supported
Automatic instance recovery	Supported	Supported
Bring Your Own License (BYOL)	Supported	Not supported

Question #1

1. A Developer is creating an application that needs to locate the public IPv4 address of the Amazon EC2 instance on which it runs. How can the application locate this information?
 - A. Get the instance metadata by retrieving `http://169.254.169.254/latest/metadata/`.
 - B. Get the instance user data by retrieving `http://169.254.169.254/latest/userdata/`.
 - C. Get the application to run `IFCONFIG` to get the public IP address.
 - D. Get the application to run `IPCONFIG` to get the public IP address.

Question #1

1. A Developer is creating an application that needs to locate the public IPv4 address of the Amazon EC2 instance on which it runs. How can the application locate this information?
 - A. Get the instance metadata by retrieving <http://169.254.169.254/latest/metadata/>.
 - B. Get the instance user data by retrieving <http://169.254.169.254/latest/userdata/>.
 - C. Get the application to run IFCONFIG to get the public IP address.
 - D. Get the application to run IPCONFIG to get the public IP address.

Question #2

A company is implementing a data lake solution on Amazon S3. Its security policy mandates that the data stored in Amazon S3 should be encrypted at rest. Which options can achieve this? (Select TWO.)

Tick all that apply.

- A. Use S3 server-side encryption with an Amazon EC2 key pair.
- B. Use S3 server-side encryption with customer-provided keys (SSE-C).
- C. Use S3 bucket policies to restrict access to the data at rest.
- D. Use client-side encryption before ingesting the data to Amazon S3 using encryption keys.
- E. Use SSL to encrypt the data while in transit to Amazon S3.

Question #2

A company is implementing a data lake solution on Amazon S3. Its security policy mandates that the data stored in Amazon S3 should be encrypted at rest. Which options can achieve this? (Select TWO.)

Tick all that apply.

- A. Use S3 server-side encryption with an Amazon EC2 key pair.
- B. Use S3 server-side encryption with customer-provided keys (SSE-C).
- C. Use S3 bucket policies to restrict access to the data at rest.
- D. Use client-side encryption before ingesting the data to Amazon S3 using encryption keys.
- E. Use SSL to encrypt the data while in transit to Amazon S3.

Question #3

A photo-sharing website running on AWS allows users to generate thumbnail images of photos stored in Amazon S3. An Amazon DynamoDB table maintains the locations of photos, and thumbnails are easily re-created from the originals if they are accidentally deleted. How should the thumbnail images be stored to ensure the LOWEST cost?

- A. Amazon S3 Standard-Infrequent Access (S3 Standard-IA) with cross-region replication
- B. Amazon S3
- C. Amazon Glacier
- D. Amazon S3 with cross-region replication

Question #3

A photo-sharing website running on AWS allows users to generate thumbnail images of photos stored in Amazon S3. An Amazon DynamoDB table maintains the locations of photos, and thumbnails are easily re-created from the originals if they are accidentally deleted. How should the thumbnail images be stored to ensure the LOWEST cost?

- A. Amazon S3 Standard-Infrequent Access (S3 Standard-IA) with cross-region replication
- B. Amazon S3**
- C. Amazon Glacier
- D. Amazon S3 with cross-region replication