

# TECHNOLOGY



## AWS Solution Architect

## Compute and Related Features





# A Day in the Life of a Cloud Architect

Mr. Well works for an IT company as a cloud architect. The company has several cloud-based applications. Recently, the company has been facing routing issues. The company is facing a huge loss due to this.

Now, the company wants Mr. Well to certify that these types of issues should not occur in the future.

In this lesson, he will get a brief idea about how to create routing requests, the development of AMI images, and various type of load balancers.



# Learning Objectives

By the end of this lesson, you will be able to:

- 🕒 Create an AMI image
- 🕒 Launch and connect to a Windows instance
- 🕒 Deploy types of Load Balancer
- 🕒 Create a routing request in ALB



## Introduction to Amazon EC2

# Elastic Compute Cloud (EC2)

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides scalable computing capacity in the Amazon Web Services (AWS) cloud.



# Benefits of EC2

Elastic Web-scale Computing

Flexible Cloud Hosting Services

AWS Integration

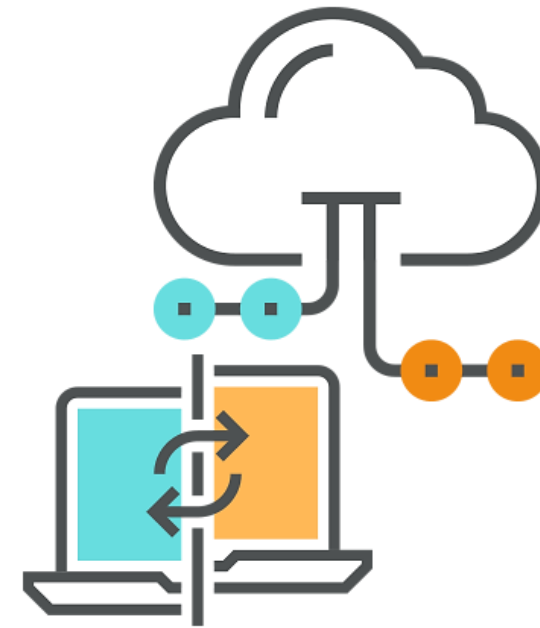
Reliability and Security

Low Cost

Completely Controlled



- EC2 increases or decreases the storage capacity in minutes.
- It launches thousands of server instances simultaneously.



# Benefits of EC2

Elastic Web-scale Computing

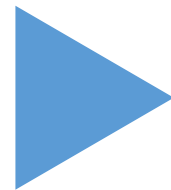
Flexible Cloud Hosting Services

AWS Integration

Reliability and Security

Low Cost

Completely Controlled



- EC2 launches numerous operating systems, instance types, and software in minutes.
- It allows the users to choose the memory, CPU, instance storage, and boot partition size that is best for their applications and OS.





# Benefits of EC2

Elastic Web-scale Computing

Flexible Cloud Hosting Services

AWS Integration

Reliability and Security

Low Cost

Completely Controlled

EC2 is integrated with other AWS products such as Amazon S3, Amazon RDS, and Amazon SQS, to provide a complete IT architecture solution.



# Benefits of EC2

Elastic Web-scale Computing

Flexible Cloud Hosting Services

AWS Integration

Reliability and Security

Low Cost

Completely Controlled

- Amazon EC2 provides a highly reliable environment in which replacement instances can be quickly and consistently deployed.
- The users can easily create secure and robust networks to run their Amazon EC2 instances using Virtual Private Cloud (VPC).



# Benefits of EC2

Elastic Web-scale Computing

Flexible Cloud Hosting Services

AWS Integration

Reliability and Security

Low Cost

Completely Controlled

- AWS charges the users by seconds, and they only pay for what they use.
- The rates are lower than the existing on-premise infrastructure.



# Benefits of EC2

Elastic Web-scale Computing

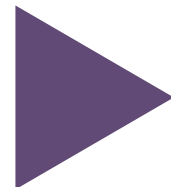
Flexible Cloud Hosting Services

AWS Integration

Reliability and Security

Low Cost

Completely Controlled




- The users have complete control of their instances. They have root access to all of their instances.
- The users can use web service APIs to stop their instance while keeping the data on their boot partition, and then resume it.





# Amazon EC2 Storage

Amazon EC2 offers flexible, cost-effective, and simple data storage options for instances. The storage options include the following:



Amazon EBS  
with EC2

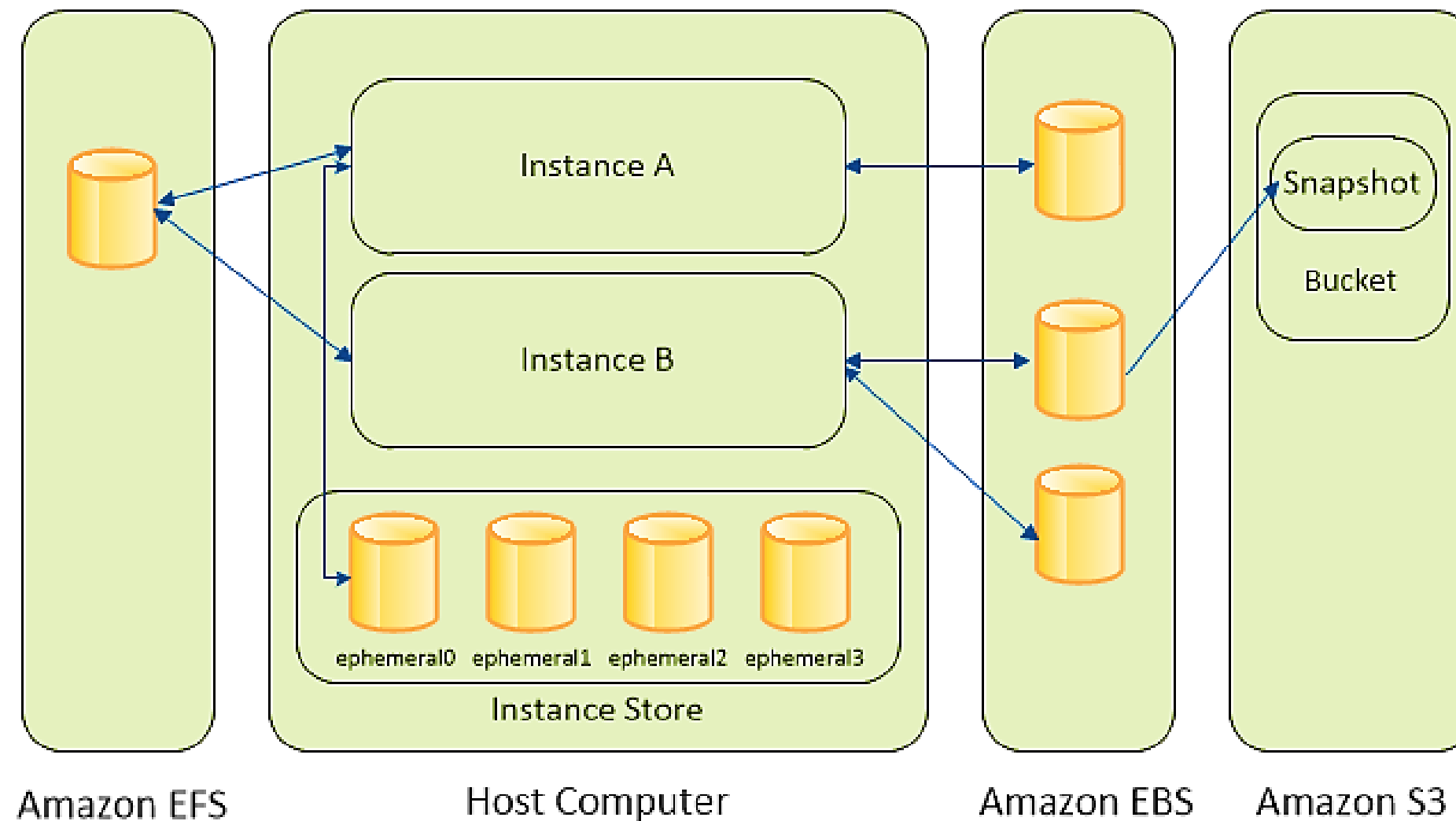
Amazon EC2  
instance store

Amazon EFS  
with EC2

Amazon S3  
with EC2

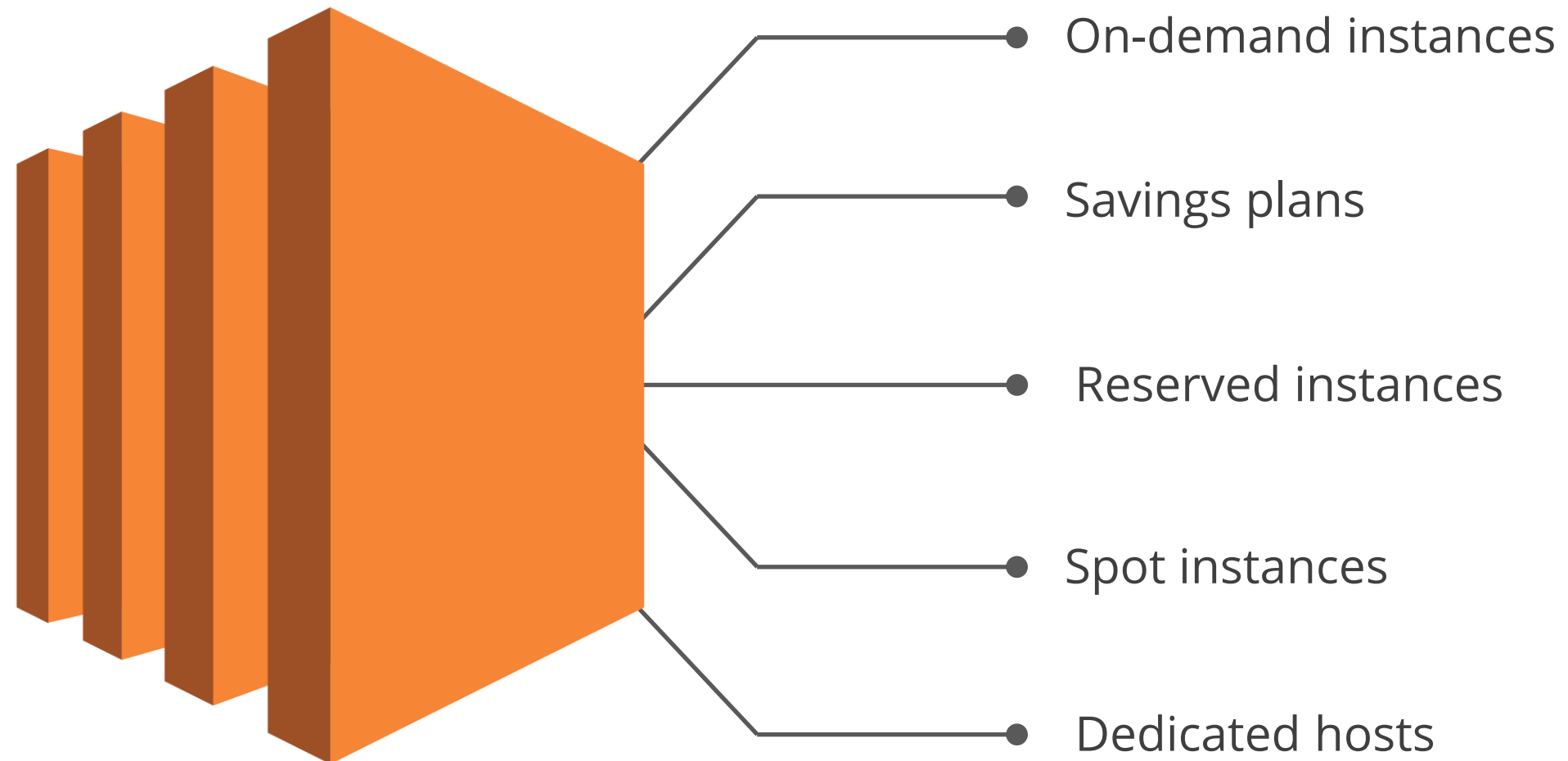
# Amazon EC2 Storage: Linux Instances

The following image depicts the relationship between the storage options and Linux instances:



# EC2 Instance Purchasing Options

Amazon EC2 provides the following purchasing options to enable users to optimize their costs based on their needs:



## Amazon Machine Images (AMI)



# Amazon Machine Images (AMI)

An Amazon Machine Image (AMI) is an AWS-supported and maintained image that contains the information required to launch an instance.



- The users must specify an AMI when they launch an instance.
- The users can launch multiple instances from a single AMI when they require multiple instances with the same configuration.
- The users can use different AMIs to launch instances when they require instances with different configurations.

# Amazon Machine Images (AMI)



AMI

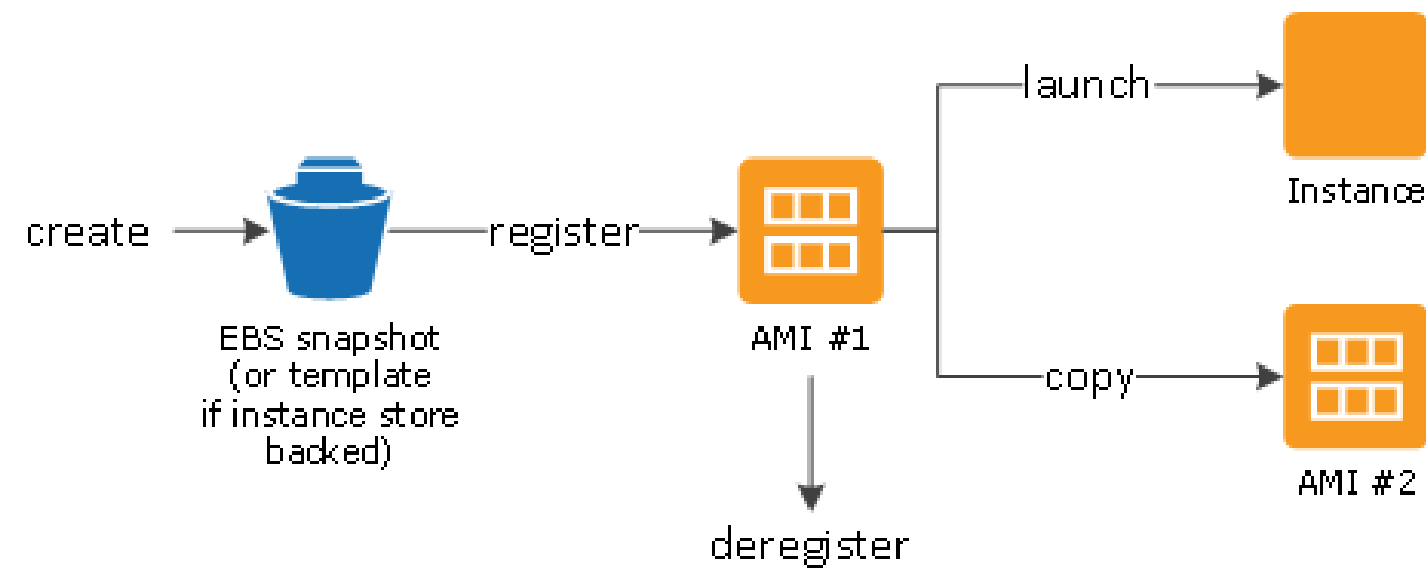


AMI is a virtual instance that includes:

- A template for the root volume for the instance
- Launch permissions to control AMI launch instances
- A block device mapping that specifies volumes to attach to the instance

# AMI Lifecycle

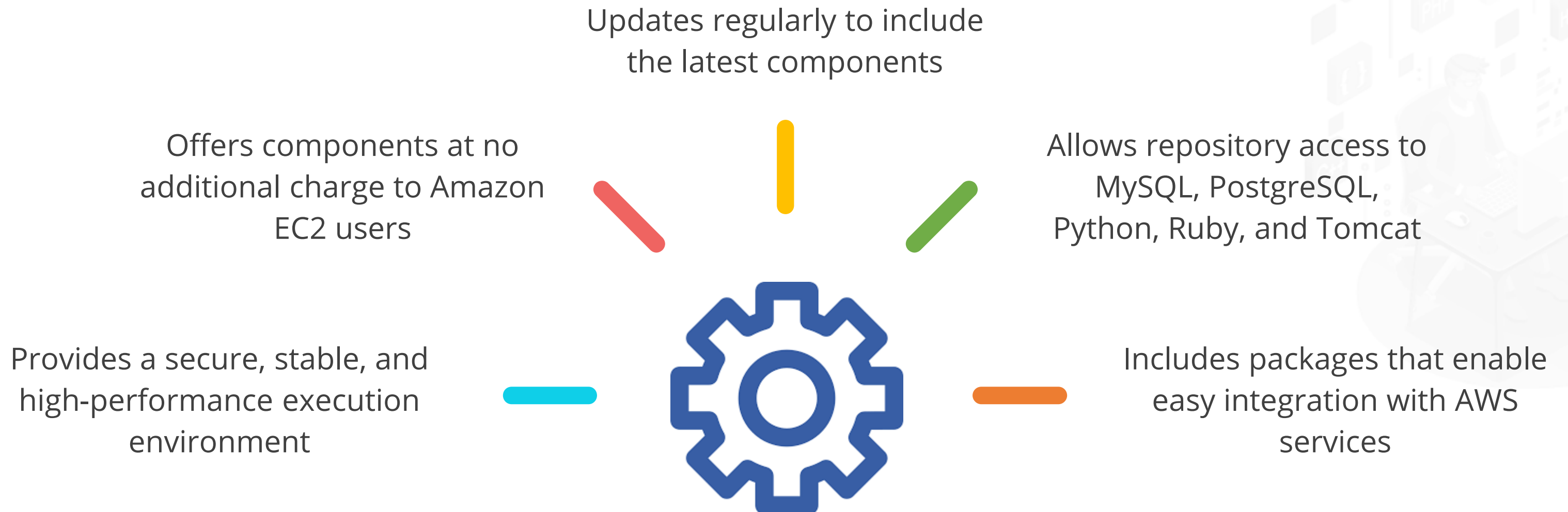
The following diagram represents the AMI lifecycle:



- After creating and registering an AMI, the users can use it to launch new instances.
- The users can copy an AMI within the same AWS Region or to different AWS Regions.
- When an AMI is no longer required, the users can deregister it.

# Amazon Linux AMI

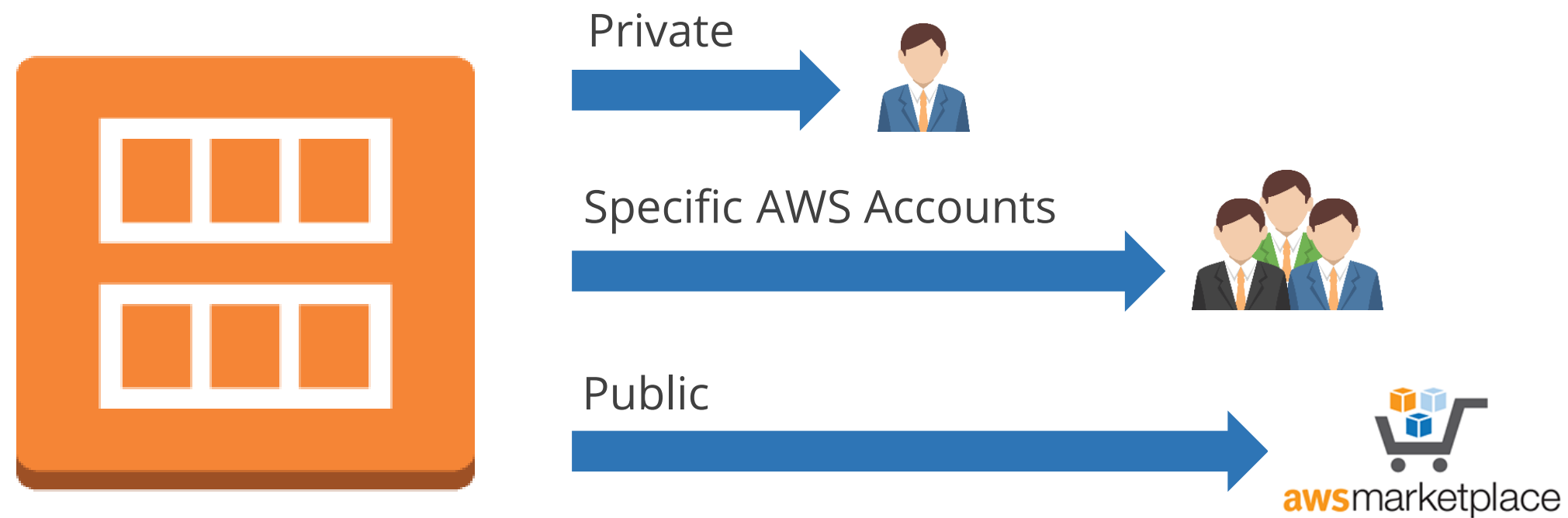
Amazon Linux AMI is a supported and maintained Linux image provided by AWS. The following are the features of the Amazon Linux AMI:





# AMI Distribution

An AMI can be kept private, shared with a specific list of AWS accounts, or made public.



# AMI Types

An AMI can be categorized into either EBS-backed AMI or instance store-backed AMI:

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	64 TiB	10 GiB
Root device volume	EBS volume	Instance store volume
Data persistence	By default, the root volume is deleted when the instance terminates. Data on any other EBS volumes persists after instance termination by default.	Data on any instance store volumes persists only during the life of the instance.

# Create an AMI image



**Duration: 10 Min.**

## Problem Statement:

You have been assigned a task to create an AMI image.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

Steps to be followed:

1. Open the Trusted Advisor console
2. Select EC2 and click on it
3. Select Launch instances to create an instance
4. Select Image and templates from Launch instances
5. Select Create Image from Image and templates
6. Click on the Create Image option after filling the details

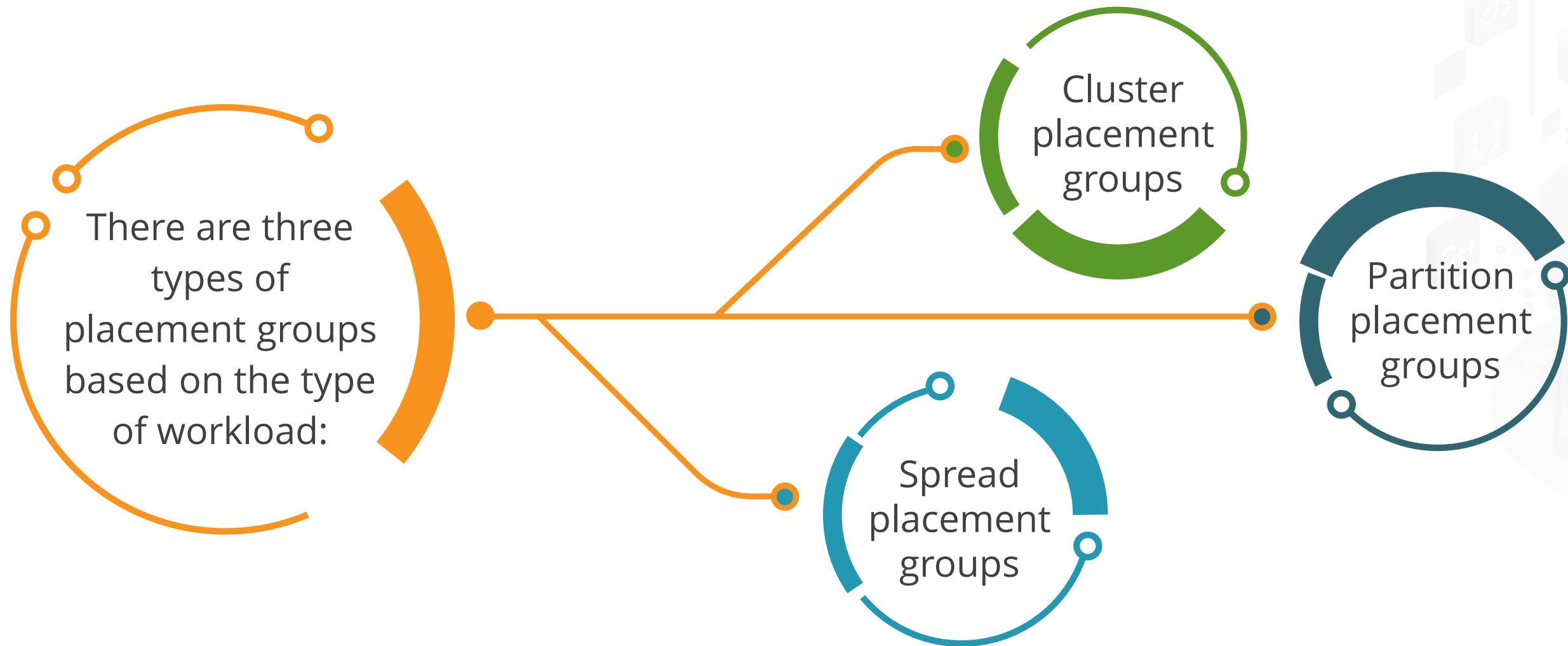




## Placement Groups

# Placement Groups

The placement groups help the users to deploy a group of interdependent instances to handle their workload more effectively.



# Placement Groups

The general rules and limitations of placement groups are as follows:



- A maximum of 500 placement groups per account can be created in each Region.
- The name of a placement group must be unique within the AWS account for the Region.
- The placement groups cannot be merged.

# Placement Groups

The general rules and limitations of placement groups are as follows:



- An instance can only be launched in one placement group at a time.
- The users cannot launch Dedicated Hosts in placement groups.
- A zonal Reserved Instance cannot be used to explicitly reserve capacity in a placement group.

# Cluster Placement Groups

A cluster placement group is a logical grouping of instances inside a single Availability Zone. It can span peered VPCs in the same Region.



Instances deployed in the same cluster placement group share a high-bisection bandwidth network segment and have a higher TCP or IP traffic throughput limit per flow.





# Cluster Placement Groups

The rules and limitations of cluster placement groups are as follows:



- A cluster placement group cannot be deployed across multiple Availability Zones.
- The maximum network throughput speed of traffic between two instances is limited by the slower of the two instances.
- The cluster placement group instances can use up to 10 Gbps for single-flow traffic.

# Cluster Placement Groups

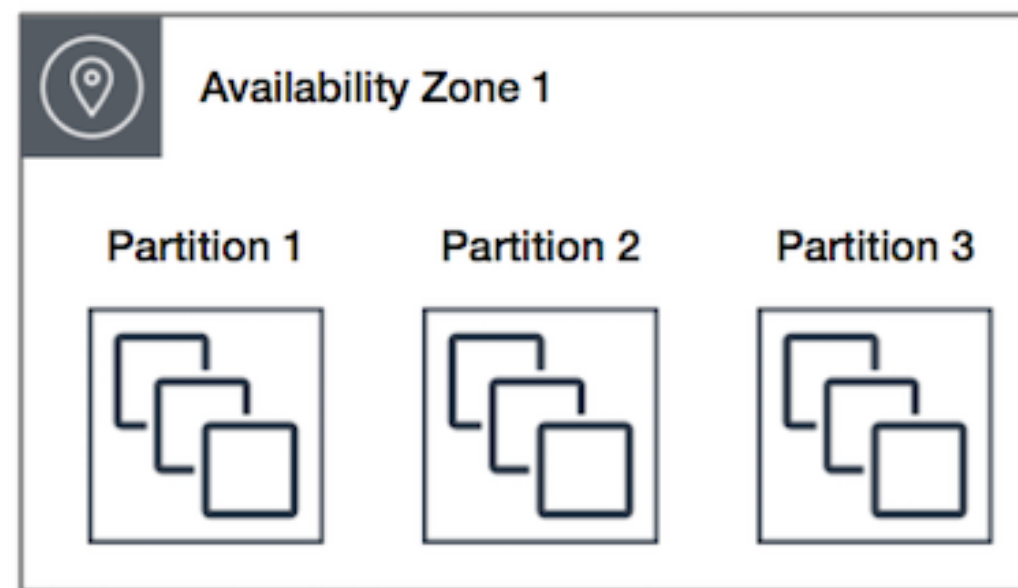
The rules and limitations of cluster placement groups are as follows:



- The users can launch multiple instance types into a cluster placement group.
- The network traffic to the internet and on-premises resources via an AWS Direct Connect connection is limited to 5 Gbps.
- All available instance aggregate bandwidth can be used for traffic to and from Amazon S3 buckets across the same Region.

# Partition Placement Groups

A partition placement group distribute the instances across logical partitions so that groups of instances in one partition do not share the same underlying hardware as groups of instances in other partitions.



It can be used by large distributed and replicated workloads, such as Hadoop, Cassandra, and Kafka.

# Partition Placement Groups

The rules and limitations of partition placement groups are as follows:



- A partition placement group can accommodate a maximum of seven partitions per Availability Zone.
- Amazon EC2 offers no guarantee that instances are evenly allocated across all partitions.
- A partition placement group containing Dedicated Instances can only have two partitions.
- The Capacity Reservations cannot be used to reserve capacity in a partition placement group.

# Spread Placement Groups

A spread placement group distributes a small group of instances across different underlying hardware to avoid correlated failures.



The placement groups can distribute instances among hosts or racks. The users can only use host level spread placement groups with AWS Outposts.

# Spread Placement Groups

The rules and limitations of spread placement groups are as follows:



- A rack spread placement group can support a maximum of seven instances per Availability Zone.
- The spread placement groups are not supported for Dedicated Instances.
- The host level spread placement groups are only accessible for AWS Outpost placement groups.
- The Capacity Reservations cannot be used to reserve capacity in a spread placement group.

# User data & Metadata – Installing apps when launching an instance



**Duration: 8 mins**

## **Problem Statement:**

You have been asked to install apps when launching an instance using user data and metadata

ASSISTED PRACTICE



# Assisted Practice: Guidelines

---

Steps to be followed:

1. Open the **EC2 Instance** dashboard
2. Click on **Launch Instance** to Create an Instance and this screen appears.
3. Fill in the details in the **Review and Launch** page.
4. In the User Data box, write this code and click Review and Launch.

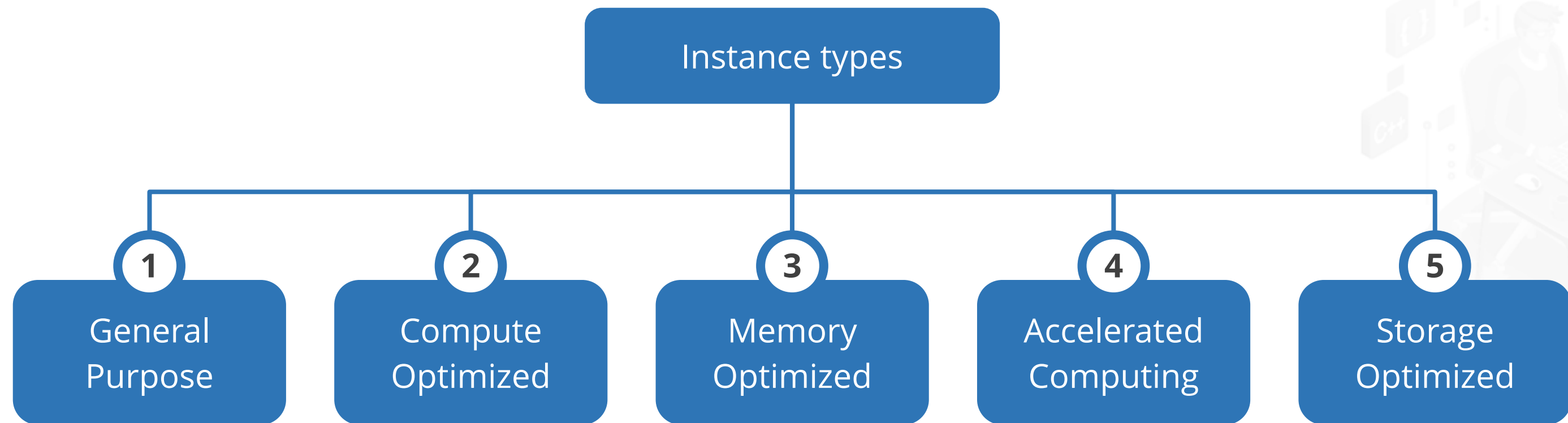


# TECHNOLOGY

## Instances

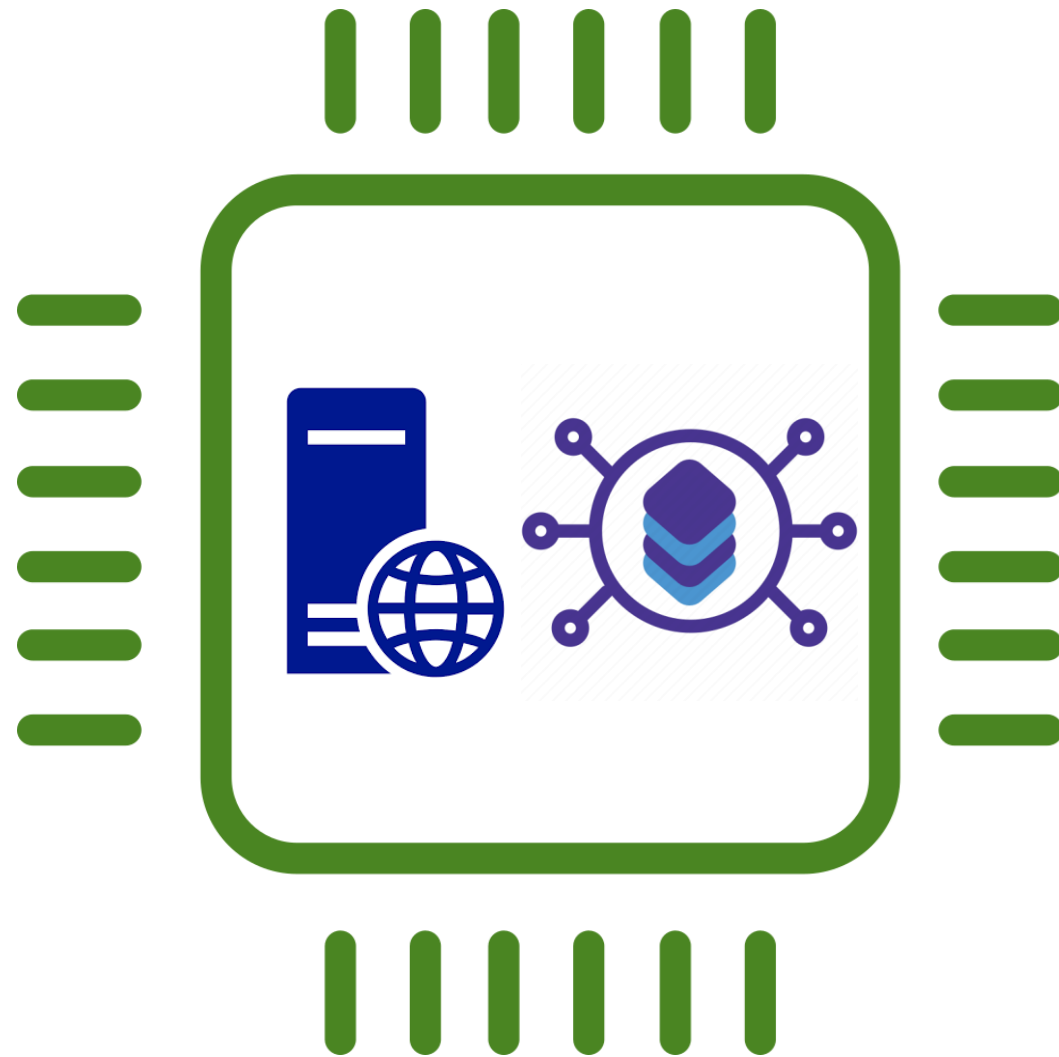
# Instances

An instance is a virtual server in the AWS cloud. Amazon EC2 offers various instance types, allowing the users to select the CPU, memory, storage, and networking resources required to run their applications.



# General Purpose Instances

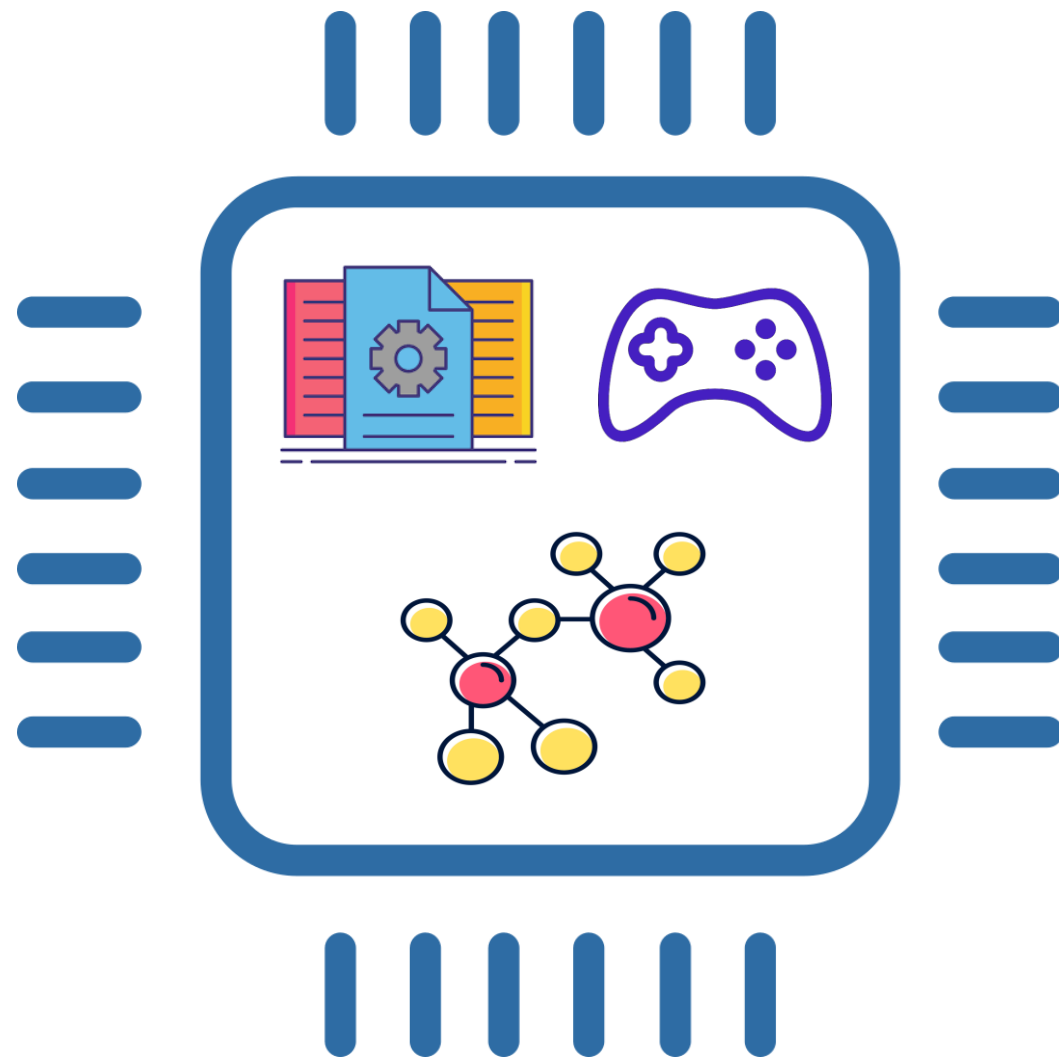
General purpose instances offer a good range of computing, memory, and networking resources and can be used for a wide range of workloads.



- These instances are useful for applications like web servers and code repositories that require such resources in equal parts.
- The examples are Mac, T4g, T3, T3a, T2, M6g, M6i, M6a, M5, M5a, M5n, M5zn, M4, and A1 instances.

# Compute Optimized Instances

Compute optimized instances are suitable for compute-intensive applications that benefit from high-performance processors.



- These instances are useful for batch processing workloads, dedicated gaming servers and ad server engines, scientific modeling, and so on.
- The examples are C7g, C6g, C6gn, C6i, C6a, Hpc6a, C5, C5a, C5n, and C4 instances.

# Memory Optimized Instances

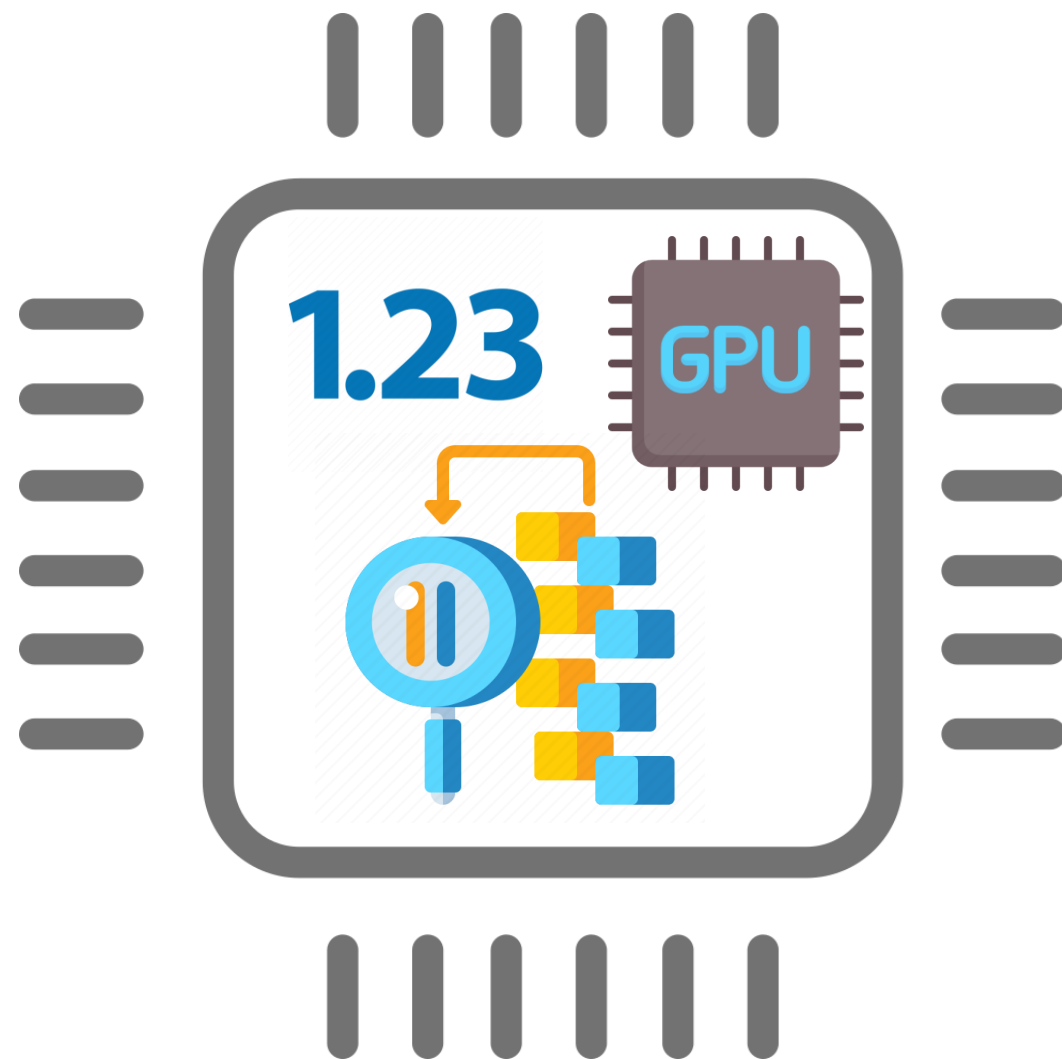
Memory optimized instances are intended to provide rapid performance for workloads that process large data sets in memory.



- These instances are useful for databases, electronic design automation (EDA) workloads, real-time analytics, and real-time caching servers.
- The examples are R6g, R6i, R5, R5a, R5b, R5n, R4, X2gd, X2idn, X2iezn, X1e, X1, high memory, and z1d instances.

# Accelerated Computing Instances

Accelerated computing instances use hardware accelerators, or co-processors, to execute tasks more effectively than software running on CPUs.



- These instances are useful for performing tasks such as floating-point number calculations, graphics processing, data pattern matching, and so on.
- The examples are P4, P3, P2, DL1, Trn1, Inf1, G5, G5g, G4dn, G4ad, G3, F1, and VT1 instances.



# Storage Optimized Instances

Storage optimized instances are designed to provide applications with tens of thousands of low-latency, random I/O operations per second (IOPS).



- These instances are useful for workloads that demand high, sequential read and write access to very large data sets on local storage.
- The examples are Im4gn, Is4gen, I4i, I3, I3en, D2, D3, D3en, and H1 instances.

## EC2 Instance Metadata

# EC2 Instance Metadata



- Instance metadata is the data about an instance that users can use to configure or manage it. It's divided into sections like hostname, events, and security groups.
- Instance metadata allows users to access user data that was specified when the instance was launched.

## Note

- Metadata is the information about the EC2 instance.
- User data is the launch script of the EC2 instance.

# EC2 Instance Metadata

The users can access instance metadata from a running instance using one of the following methods:

**01**

**Instance Metadata Service Version 1 (IMDSv1):** A request or response method

**02**

**Instance Metadata Service Version 2 (IMDSv2):** A session-oriented method

# EC2 Instance Metadata

To view all the categories of instance metadata from within a running instance, use the following URL:

IMDSv1	IMDSv2
<pre>curl http://169.254.169.254/latest/meta-data/</pre>	<pre>TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X- aws-ec2-metadata-token-ttl-seconds: 21600" ` \ &amp;&amp; curl -H "X-aws-ec2-metadata-token: \$TOKEN" -v http://169.254.169.254/latest/meta-data/</pre>

# Instance Metadata Categories

The different categories of instance metadata are:

ami-id	instance-type	public-hostname
ami-launch-index	local-hostname	public-ipv4
ami-manifest-path	local-ipv4	public-keys/
block-device mapping/	mac	reservation-id
hostname	metrics/	security-groups
iam/	network/	services/
instance-action	placement/	spot/
instance-id	profile	tags/



# Instance Metadata Service Version 2

Instance Metadata Service Version 2 (IMDSv2) uses session-oriented requests.

The logo for IMDSv2 consists of an orange rounded rectangle with the text "IMDSv2" in white, and an orange cylinder with a white outline positioned in front of the bottom right corner of the rectangle.

**IMDSv2**

With session-oriented requests:

1. Users generate a session token that specifies the session duration, which can range from one second to six hours.
2. Users can make further requests using the same session token during the specified duration.
3. Users need to create a new session token to use for subsequent queries once the time limit has passed.



# IAM Roles



An IAM role is included with an instance profile on Amazon EC2. The IAM console automatically creates an instance profile when users create an IAM role, giving it the same name as the role it belongs to.

## **IAM Roles - Access Other AWS Services From EC2**

# Connection Between IAM and EC2

The connection between IAM Roles and EC2 instances:

aws

Services

Search for services, features, blogs, docs, and more

[Alt+S]

Global

odl\_user\_705451 @ 0883-2994-0

IAM > Roles > Create role

Step 1

Select trusted entity

Step 2

Add permissions

Step 3

Name, review, and create

Add permissions

Permissions policies (760)

Choose one or more policies to attach to your new role.

Filter policies by property or policy name and press enter

26 matches

< 1 2 >

⚙

"Ec2" X

Clear filters

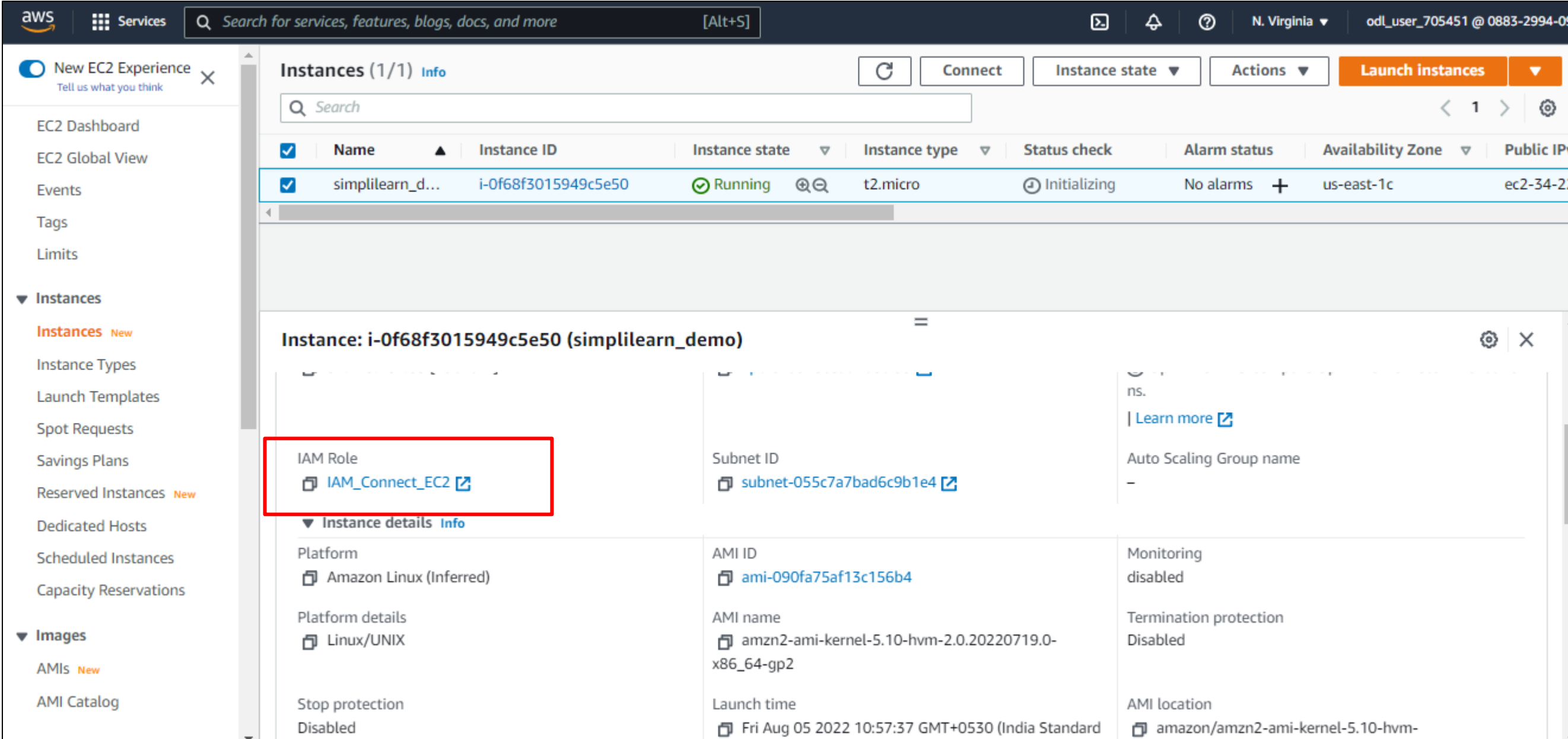
<input type="checkbox"/>	Policy name	Type	Description
<input type="checkbox"/>	<input type="checkbox"/> AmazonEC2FullAccess	AWS m...	Provides full access to Amazon EC2 via the ...
<input type="checkbox"/>	<input type="checkbox"/> AmazonEC2RoleforSSM	AWS m...	This policy will soon be deprecated. Please ...
<input type="checkbox"/>	<input type="checkbox"/> AmazonEC2RoleforAWSCodeDeploy	AWS m...	Provides EC2 access to S3 bucket to downl...
<input type="checkbox"/>	<input type="checkbox"/> AmazonEC2ContainerRegistryFullAccess	AWS m...	Provides administrative access to Amazon ...
<input type="checkbox"/>	<input type="checkbox"/> AmazonEC2ContainerRegistryReadOnly	AWS m...	Provides read-only access to Amazon EC2 ...
<input type="checkbox"/>	<input type="checkbox"/> AmazonElasticMapReduceforEC2Role	AWS m...	Default policy for the Amazon Elastic MapR...
<input type="checkbox"/>	<input type="checkbox"/> AmazonEC2ReadOnlyAccess	AWS m...	Provides read only access to Amazon EC2 ...

©Simplilearn. All rights reserved.

simplilearn

# Connection Between IAM and EC2

The connection between IAM Roles and EC2 instances:



The screenshot displays the AWS Management Console interface for the 'Instances' page. The left sidebar shows navigation options like 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Tags', 'Limits', and 'Instances'. The main content area shows a list of instances, with one instance 'simplilearn\_d...' selected. Below the list, the 'Instance details' section is expanded, showing various attributes. The 'IAM Role' attribute is highlighted with a red box, indicating the connection between the instance and its IAM role.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
simplilearn_d...	i-0f68f3015949c5e50	Running	t2.micro	Initializing	No alarms	us-east-1c	ec2-34-2...

Instance: i-0f68f3015949c5e50 (simplilearn_demo)		
IAM Role	Subnet ID	Auto Scaling Group name
<a href="#">IAM_Connect_EC2</a>	<a href="#">subnet-055c7a7bad6c9b1e4</a>	-
▼ Instance details Info		
Platform	AMI ID	Monitoring
<a href="#">Amazon Linux (Inferred)</a>	<a href="#">ami-090fa75af13c156b4</a>	disabled
Platform details	AMI name	Termination protection
<a href="#">Linux/UNIX</a>	<a href="#">amzn2-ami-kernel-5.10-hvm-2.0.20220719.0-x86_64-gp2</a>	Disabled
Stop protection	Launch time	AMI location
Disabled	<a href="#">Fri Aug 05 2022 10:57:37 GMT+0530 (India Standard</a>	<a href="#">amazon/amzn2-ami-kernel-5.10-hvm-</a>

# EC2 IAM Roles to Access S3



**Duration: 8 mins**

## **Problem Statement:**

You have been asked to access S3 using the EC2 instance.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

Steps to be followed:

1. Create a role.
2. Attach the IAM instance profile to the EC2 instance.
3. Validate access through the S3 bucket





## How to Manage Spot Interruptions with Demos?



# Spot Instance Interruptions

Spot Instances can be launched on spare EC2 capacity at substantial savings in exchange for returning them when Amazon EC2 requires the capacity back.



# Reasons for Interruption

1

Capacity

2

Price

3

Constraints

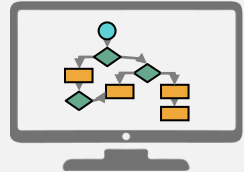


# Hibernate Interrupted Spot Instances

When an Amazon EC2 Spot Instance is hibernated, the following happens:



When the instance gets a signal from Amazon EC2, the agent instructs the operating system to enter a state of hibernation.



The root volume retains the instance memory (RAM).



Other than Elastic IP addresses, instance storage volumes and public IP addresses are not maintained.

# Launch and Connect to a Windows Instance



**Duration: 8 mins**

## **Problem Statement:**

You have been asked to launch and connect to a windows instance.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

Steps to be followed:

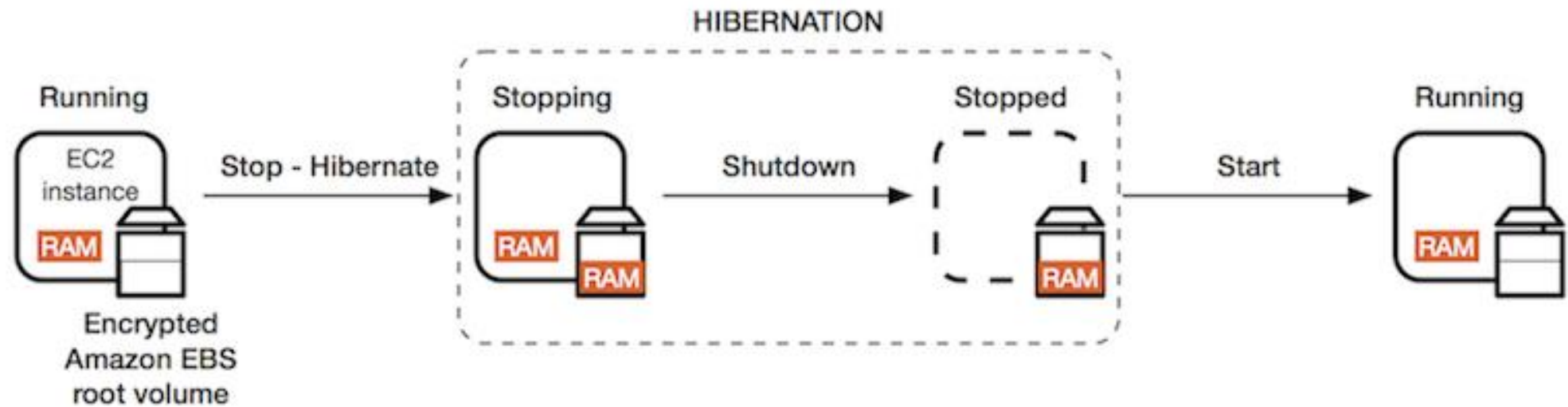
1. Launch an EC2 instance
2. Access the EC2 instance via RDP



## Hibernate EC2 Instance

# Hibernation: Overview

The following diagram shows an overview of the hibernation process:





# Hibernation: Overview

The changes that occur while hibernating a running instance:



- Amazon EC2 signals the operating system to perform hibernation.
- The instance moves to the stopped state.
- EBS volumes remain attached to it.
- The instance store volumes remain attached to the instance, but the data is lost.

# Hibernation: Overview

Changes that occur while hibernating a running instance:



- It gets migrated to a new underlying host computer when it starts again.
- It boots up, and the operating system reads in the contents of the RAM from the EBS root volume once it starts.
- It retains its private IPv4 addresses and any IPv6 addresses.
- It gets assigned to a new IPV4 address by Amazon EC2 once it starts again.

# Hibernation: Overview

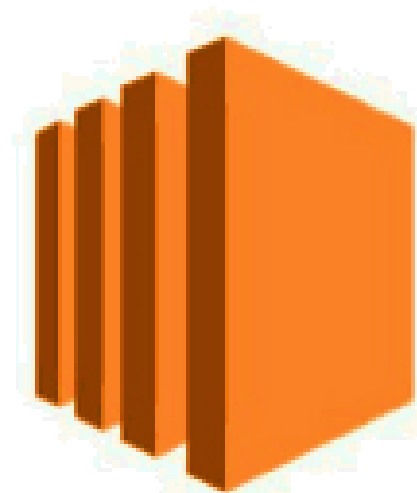
Changes that take place while a running instance is hibernated:



- While hibernating the instance with EC2-Classic, an Elastic IP address is disassociated from the instance.
- While hibernating a **ClassicLink** instance, it gets unlinked from the VPC to which it was linked.

# Hibernation for EC2 Instances

The user will launch EC2 instances by setting them up as desired, hibernating them, and then bringing them back to life when the user needs them.

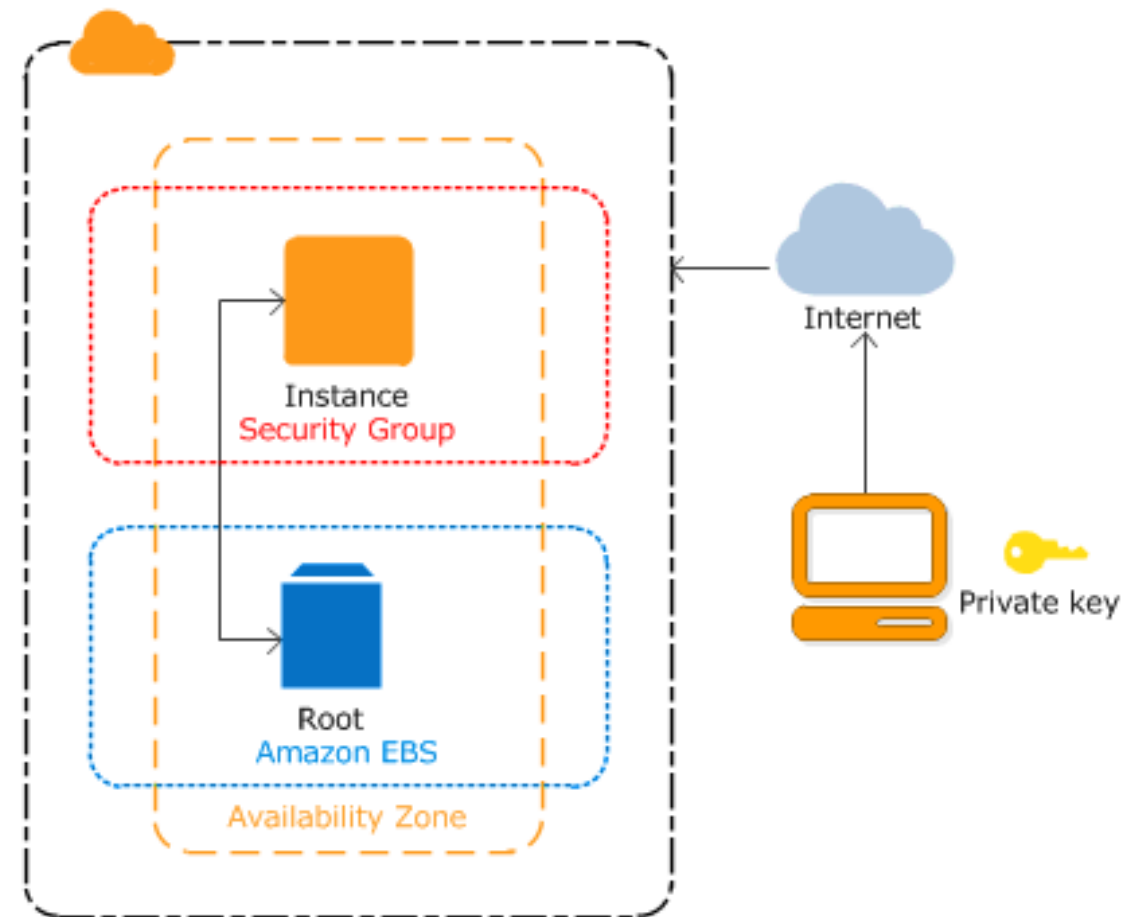


Amazon EC2

The hibernation process stores the in-memory state of the instance along with its private and elastic IP addresses, allowing it to pick up exactly where it left off.

# Hibernation For EC2 Instances

The feature is presently available, and users can use this on freshly launched M3, M4, M5, C3, C4, C5, R3, R4, and R5 instances running Amazon Linux 1.



It applies to on-demand instances and a few others which are running with reserved instance coverage.

# Hibernation For EC2 Instances

A hibernated instance writes the in-memory state to a file in the root EBS volume and then shuts itself down.



The AMI used to launch the instance must be encrypted.



# Hibernation For EC2 Instances

The encryption ensures proper protection for sensitive data when it is copied from memory to the EBS volume.



A user only pays for the EBS volumes and Elastic IP Addresses associated with the instance while it is in hibernation; there are no additional hourly fees.



# Important Points for Hibernating an Instance:

**Instance Type:** Users can enable and use hibernation on freshly launched instances.



**Root Volume Size:** The root volume must have free space equal to the amount of RAM on the instance for the hibernation to succeed.

# Important Points for Hibernating an Instance:

**Operating Systems:** The newest Amazon Linux 1 AMIs are configured for hibernation. Users need to create an encrypted AMI, using one AMI as a base.



**Modifications:** Users cannot modify the instance size or type while it is in hibernation but can modify the user data and the EBS Optimization setting.

# Important Points for Hibernating an Instance:

**Pricing:** While the instance is in hibernation, users only need to pay for the EBS storage, and any Elastic IP addresses attached to the instance.



**Performance:** The time to hibernate is dependent on the memory size of the instance, the amount of in-memory data to be saved, and the throughput of the root EBS volume.

# Limitations

The limitations regarding hibernating an instance:



- While hibernating an instance, the data on any instance store volumes is lost.
- Users can't hibernate an instance that has more than 150 GB of RAM.
- A user may not be able to connect to a new instance launched from a snapshot if the user creates the snapshot from an instance that is hibernating.

# Limitations

Below are the points regarding the limitations of hibernating an instance:



- Users can't change the instance type or size of an instance when hibernation is enabled.
- User can't hibernate an instance that is in an auto Scaling group or used by Amazon ECS.
- User can't hibernate an instance that is configured to boot in UEFI mode.

# Limitations

Below are the points regarding the limitations of hibernating an instance:



- If the user is using an instance that has been hibernating, it may not be able to resume after the user tries to start it.
- AWS does not support keeping an instance hibernated for more than 60 days.
- AWS constantly updates its platform, which can conflict with existing hibernated instances.



# Updates on AWS

AWS is working on support for Amazon Linux 2, Ubuntu, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016, along with the SQL Server variants of the Windows AMIs.

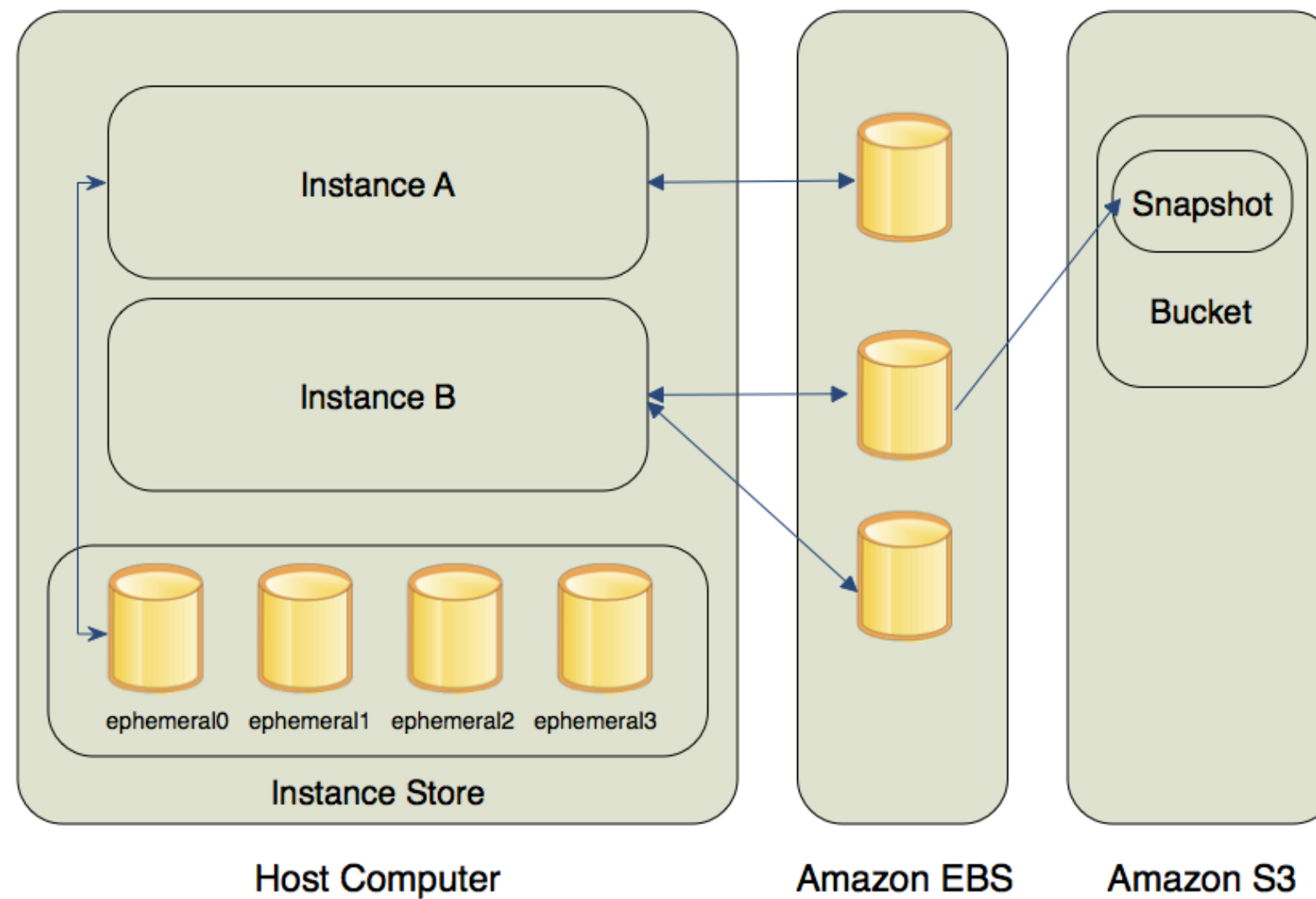




## Introduction to EBS

# EBS Optimized Instances

An Amazon EBS-optimized instance employs an optimized configuration stack and offers extra dedicated capacity for Amazon EBS I/O.



# Amazon EBS vs Instance Store

This table shows the different characteristics between the EBS-backed and Amazon Instance store-backed volumes.

Characteristics	Amazon EBS	Instance Store
Boot time	Usually less than 1 minute	Usually less than 5 minutes
Root device volume	Amazon EBS volume	Instance store volume
Upgrading	EBS-backed instances can be upgraded for instance type, Kernel, RAM disk, and user data	Instance store-backed instances cannot be upgraded
AMI creation	AMI can be easily created using a single command	AMI creation requires AMI tools and needs to be executed from within the running instance
Volume attachment	EBS volume can be attached as additional volumes when the instance is being launched and even when the instance is up and running	Instance store volume can be attached as additional volumes only when the instance is being launched and not when the instance is up and running

# Supported Instance

The supported instance types include the dedicated bandwidth to Amazon EBS, the typical maximum aggregate throughput that can be achieved on that connection with a streaming read workload.

Instance size	Maximum bandwidth (Mbps)	Maximum throughput (MB/s, 128 KiB I/O)	Maximum IOPS (16 KiB I/O)
c1.xlarge	1,000	125	8,000
c3.xlarge	500	62.5	4,000
g2.2xlarge	1,000	125	8,000
i2.xlarge	500	62.5	4,000
m1.large	500	62.5	4,000
r3.xlarge	500	62.5	4,000

# Get maximum performance

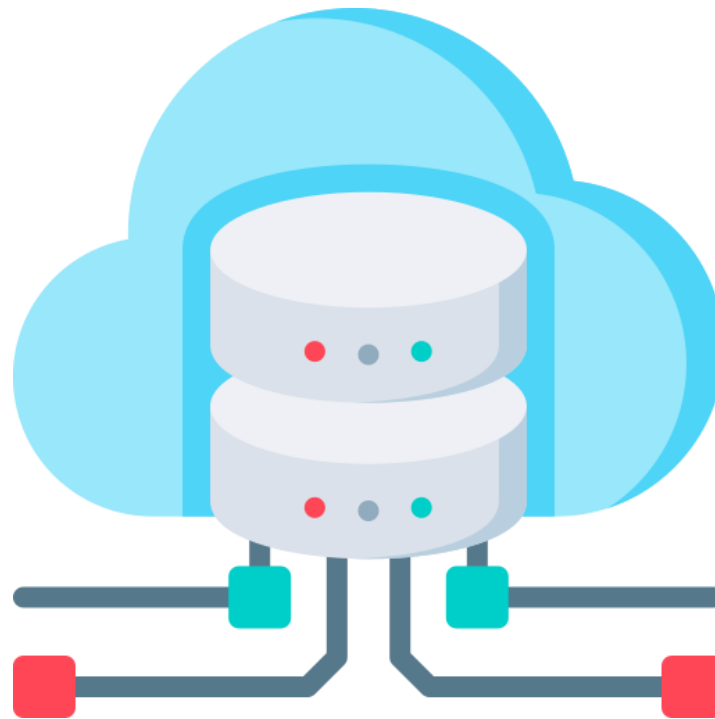


- The EBSIOBalance-percent and EBSByteBalance-percent metrics help to determine whether the instances are scaled efficiently or not.
- The cases with a low balance % on a regular basis are candidates for sizing up.
- The downsizing is appropriate in cases where the balance percentage never falls below 100%.

## Elastic File System

# Introduction to Storage

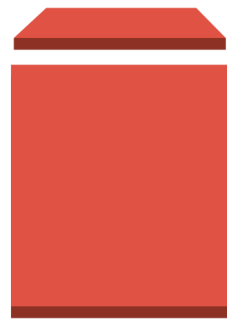
AWS offers scalable, reliable, and secure cloud services with faster data access.





# Storage units

AWS provides a variety of options, including object storage, file storage, block storage, and data migration. Some of the examples are as follows:



**EBS**



**EFS**

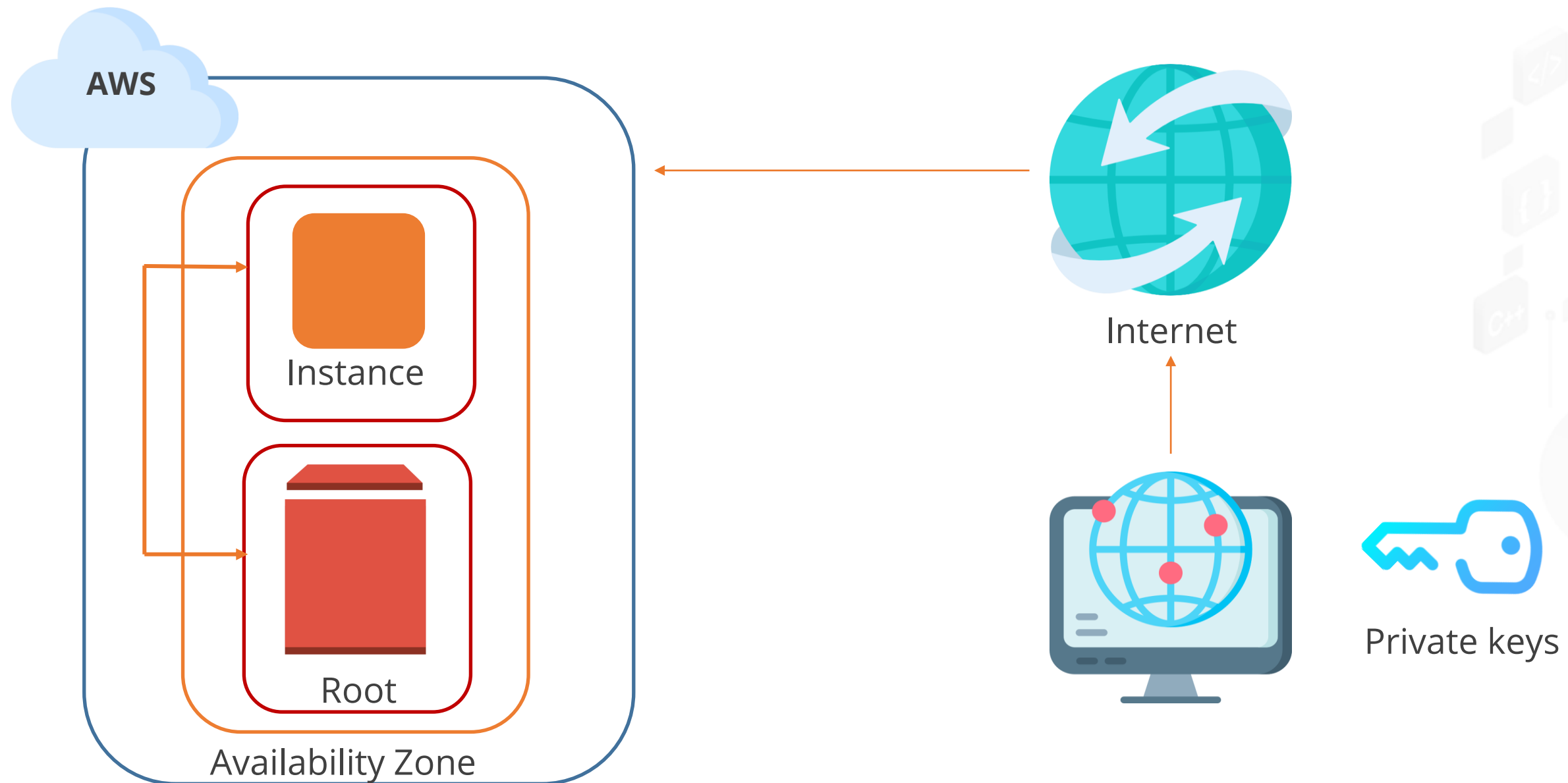


**S3**

Amazon  
**FSX**

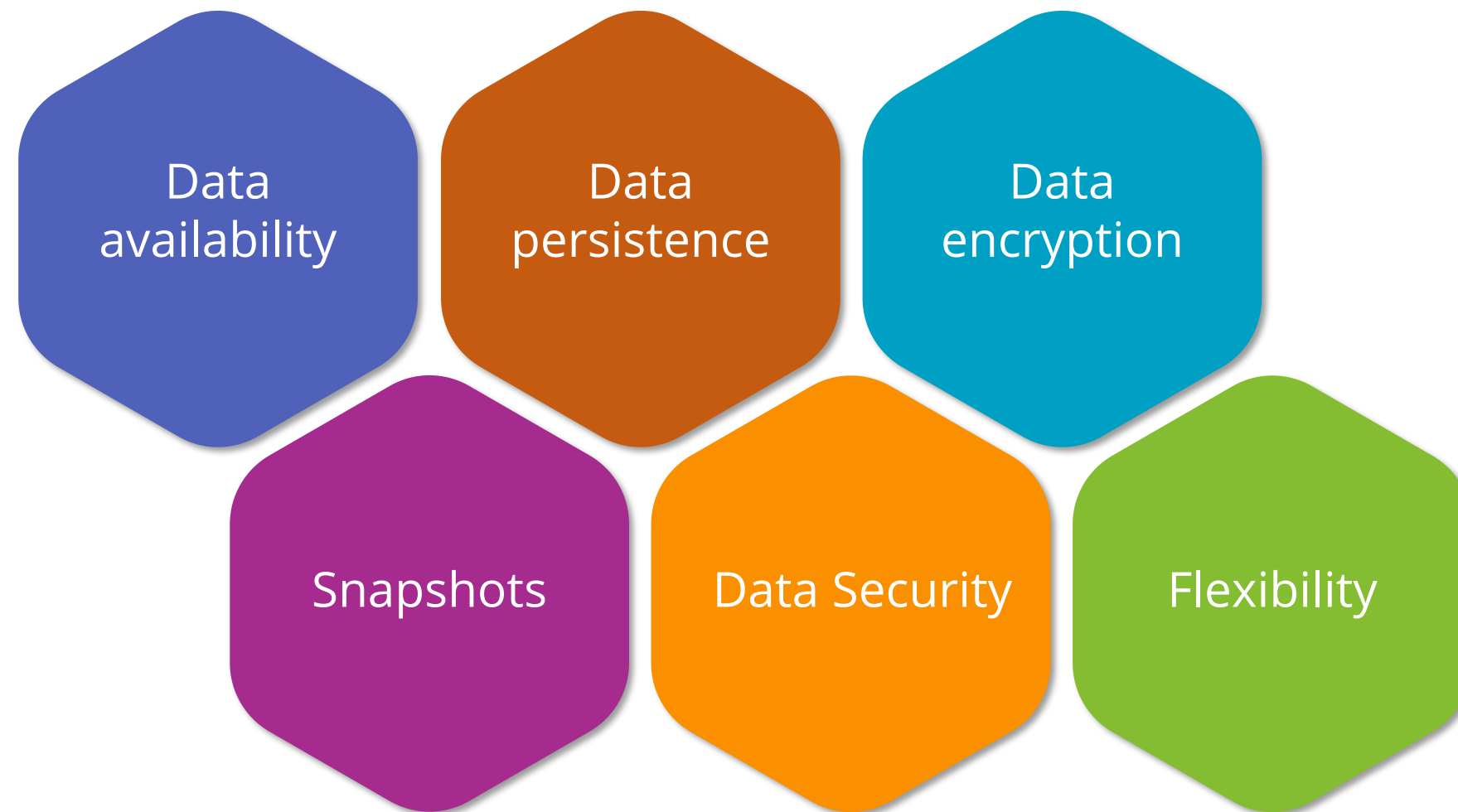
# EBS overview

Amazon Elastic Block Store (Amazon EBS) provides block-level storage volumes for use with EC2 instances.



# Benefits of EBS

The benefits of Amazon Elastic Block Store (Amazon EBS) are as follows:



# EBS Snapshot

Amazon EBS provides the ability to create snapshots (backups) of any EBS volume and write a copy of the data in the volume to Amazon S3, where it is stored redundantly in multiple Availability Zones.



- When performing incremental backups, back up only the blocks on the device that have changed after the most recent snapshot is saved.
- When a snapshot is deleted, only the data exclusive to that snapshot is removed.

# Storage Categories

The Storage categories can be classified into SSD-backed and HDD-backed:

	SSD-backed		HDD-backed	
Type	Provisioned IOPS SSD (io1)	General Purpose SSD (gp2)	Throughput Optimized HDD (st1)	Cold HDD (sc1)
Purpose	High 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 and test	Big data, data warehouses, log processing	Colder data requiring fewer scans per day
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

# Storage Categories

The Storage categories can be classified into SSD-backed and HDD-backed:

	SSD-backed		HDD-backed	
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

# Amazon Elastic File System

Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully manageable elastic NFS file system for use with AWS Cloud services and on-premises resources.





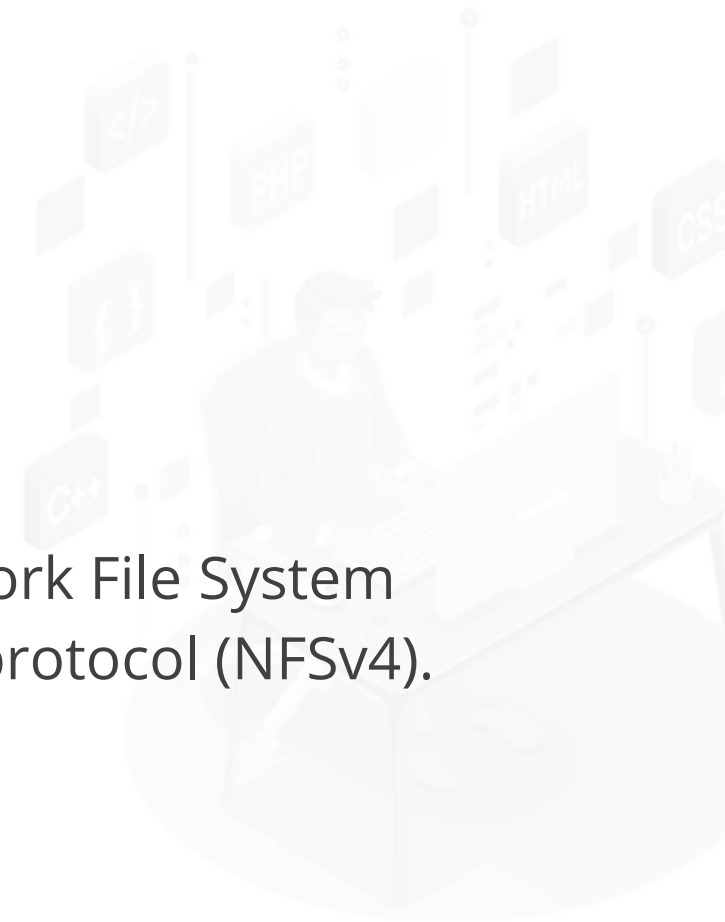
# Elastic File System Overview

The data in EFS is stored and accessed from all the availability zones in the AWS region.

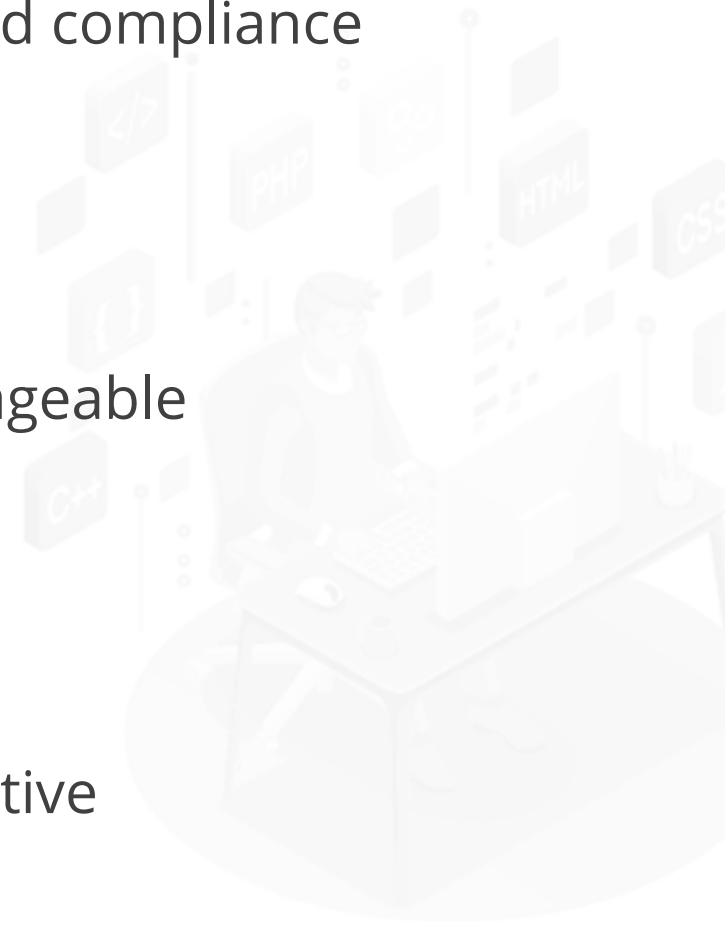
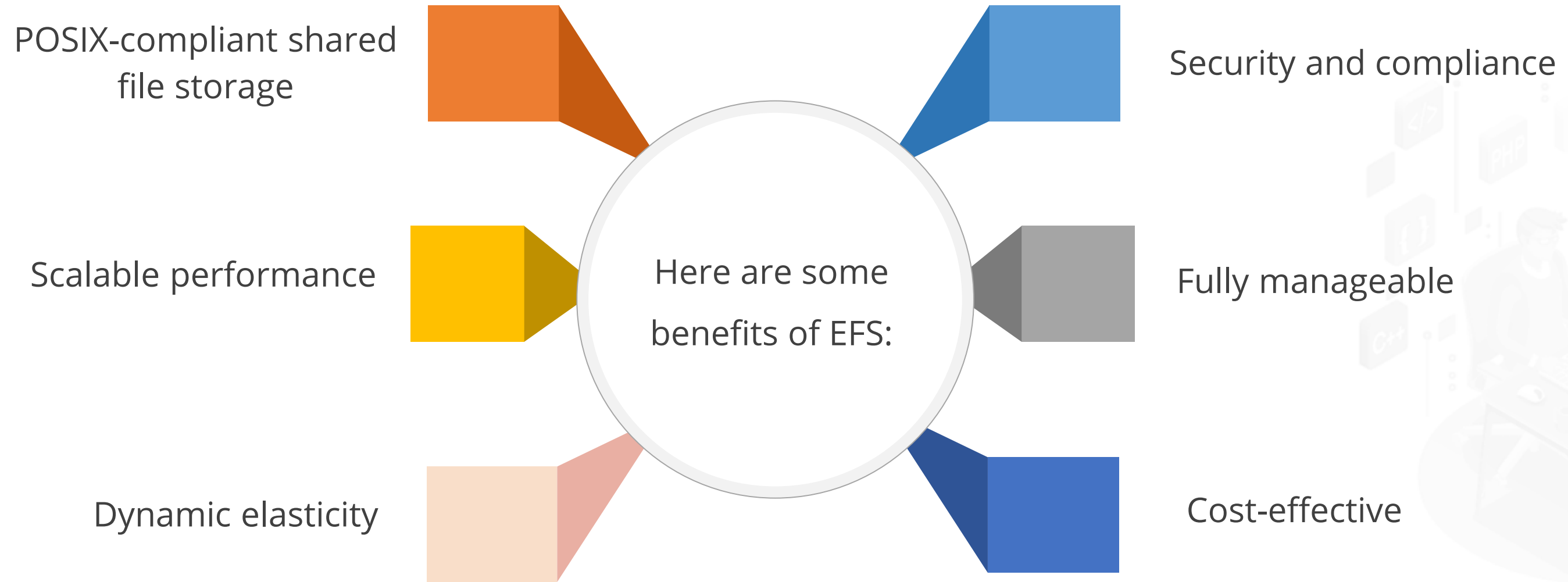
The users only pay for what they use with EFS storage.



EFS supports Network File System Version 4.0 and 4.1 protocol (NFSv4).



# Benefits of Elastic File System



# EFS vs. EBS

The difference between EFS and EBS are as follows:

Feature	EFS	EBS
Storage Size	No limitations	Maximum 16 TiB
Storage Type	Object storage	Block storage
Performance	Scalable	Hardly scalable
File Size Limitation	Maximum file size 47.9 TiB	No limitation
Data Throughput	Default throughput of 3 GB	SSD- and HDD-backed storage types
Data Access	Can be accessed concurrently	Limited to single EC2 instance
Availability Zone (AZ) Failure	Can survive one AZ failure	Cannot withstand AZ failure without snapshots

# TECHNOLOGY

## Amazon FSx

# Amazon FSx

The Amazon FSx family of services makes it easy to launch, run, and scale shared storage powered by popular commercial and open-source file systems. The Amazon FSx consists of two categories:



# FSx windows File server



- The FSx for the windows file system is a fully native managed windows file system and it is easily integrated with the whole AWS environment.
- Amazon FSx for Windows File Server provides fully managed, highly reliable, and scalable file storage that is accessible over the industry-standard Server Message Block (SMB) protocol.

# FSx windows File server

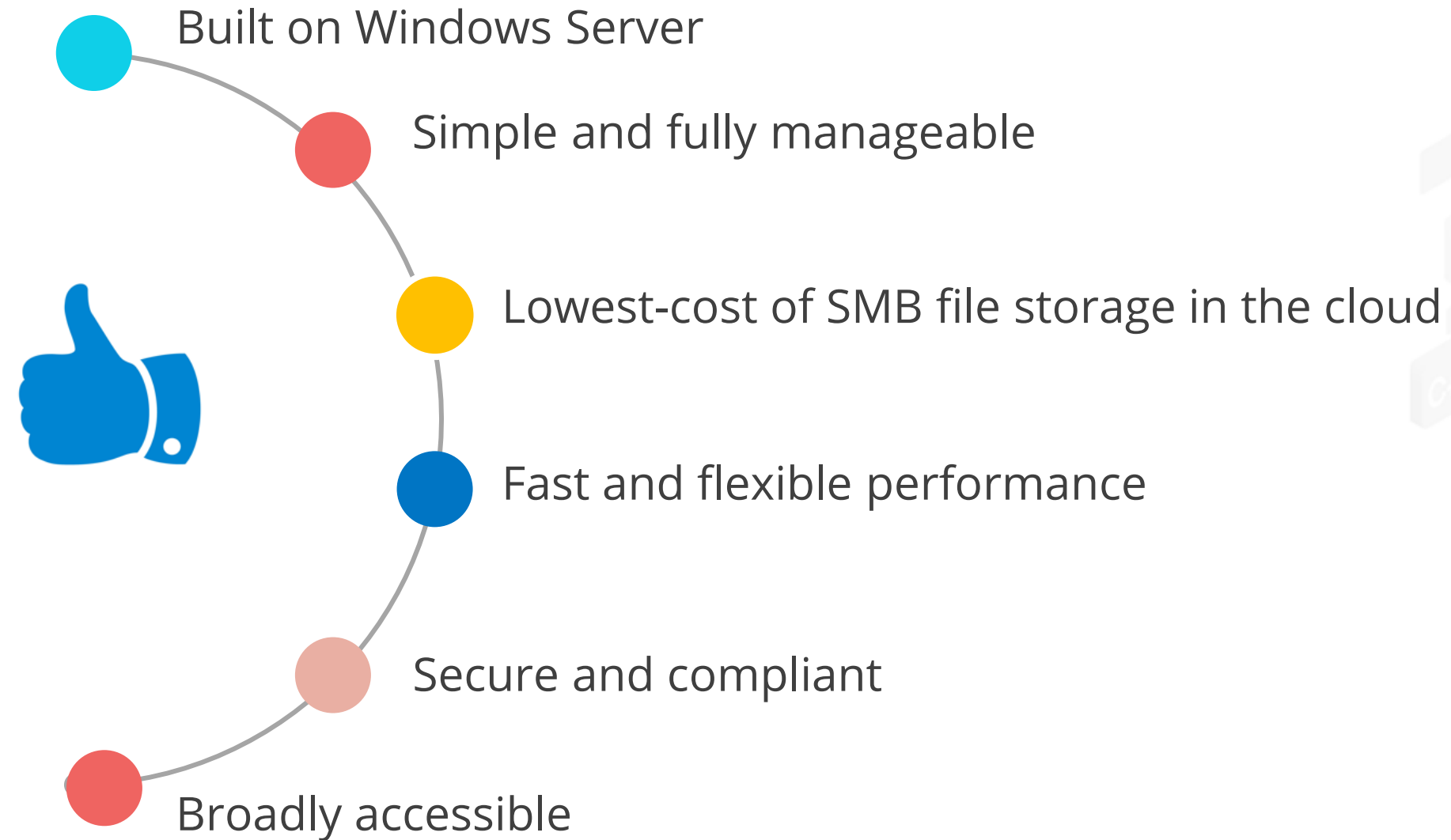


- It is built on Windows Server, delivering a wide range of administrative features such as user quotas, end-user file restore, and Microsoft Active Directory (AD) integration.
- It offers single-AZ and multi-AZ deployment options, fully managed backups, and encryption of data at rest and in transit.



# Amazon FSx for Windows

Here are a few benefits of FSx for windows:



# FSx Lustre



- FSx is integrated Natively and supports fast processing up to 100+ GB/s.
- The open-source Lustre file system is designed for applications that require fast storage that can keep up with your computing performance.
- Amazon FSx enables you to use Lustre file systems for any workload where storage speed matters.

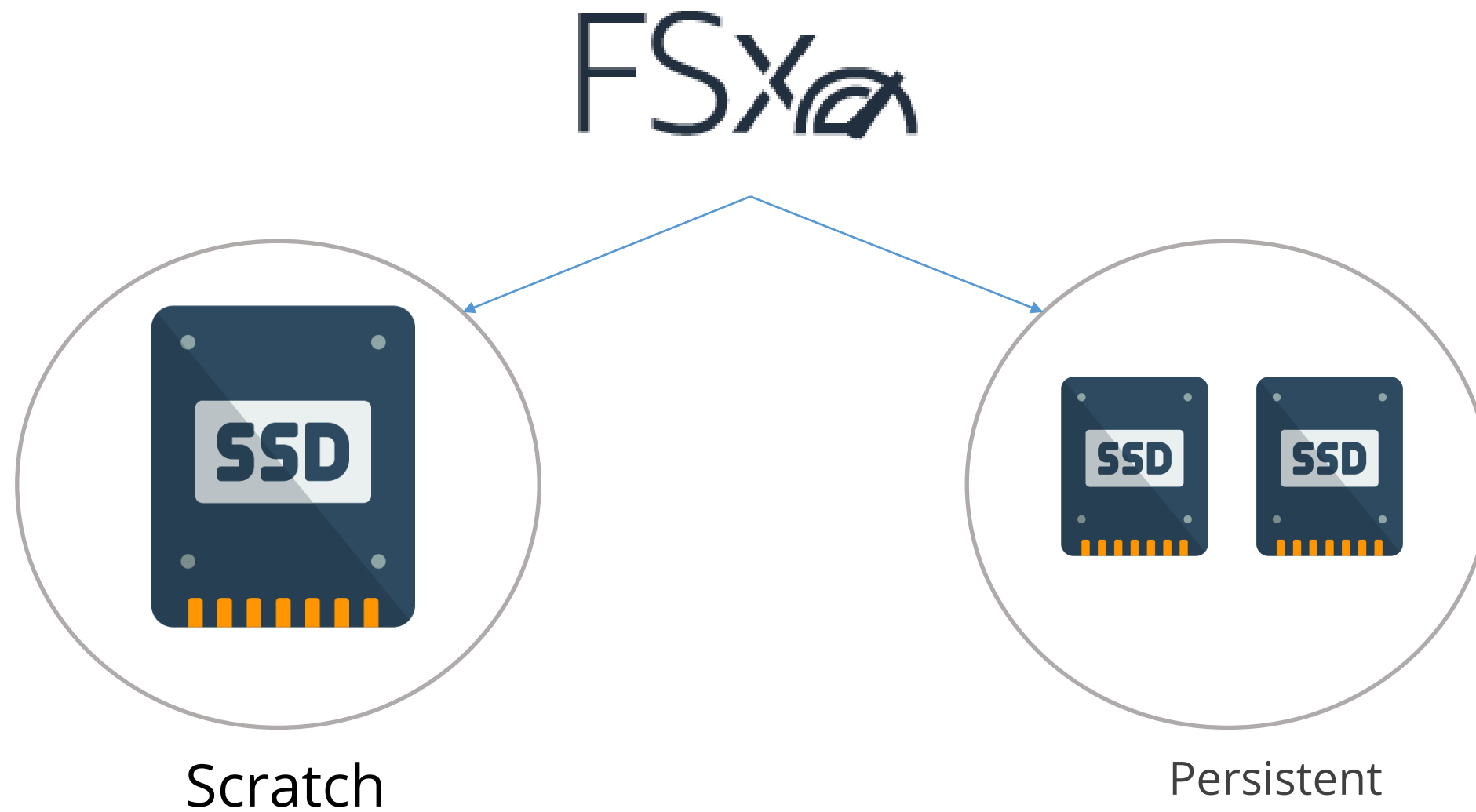
# FSx Lustre



- Amazon FSx for Lustre makes it easy and cost-effective to launch and run the world's most popular high-performance file system.
- Amazon FSx for Lustre integrates with Amazon S3, making it easy to use the Lustre file system to access data sets.

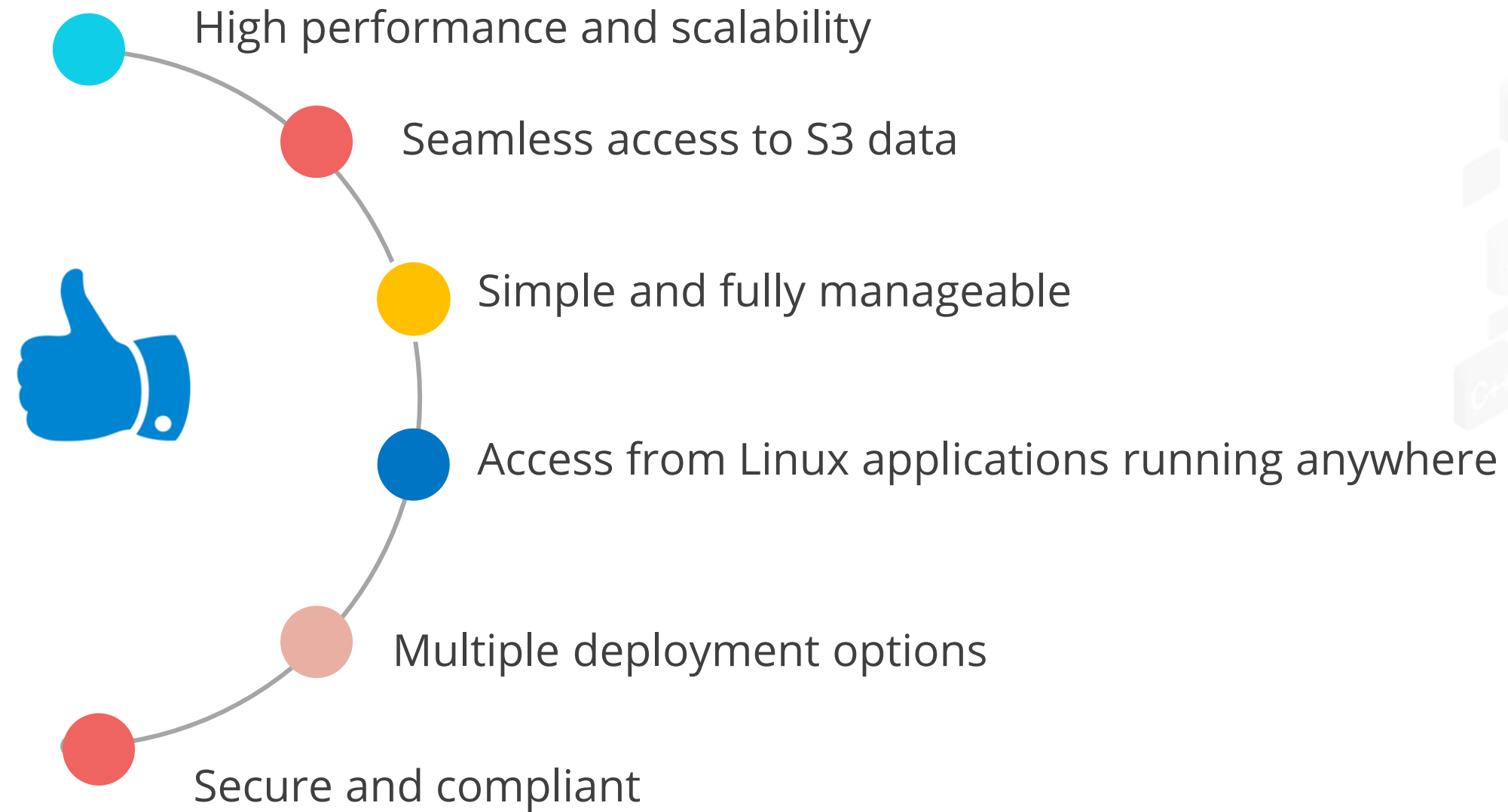
# FSx Lustre Deployment Options

FSx Lustre provides an option for deployment: scratch File system and persistent File system



# Amazon FSx for Lustre

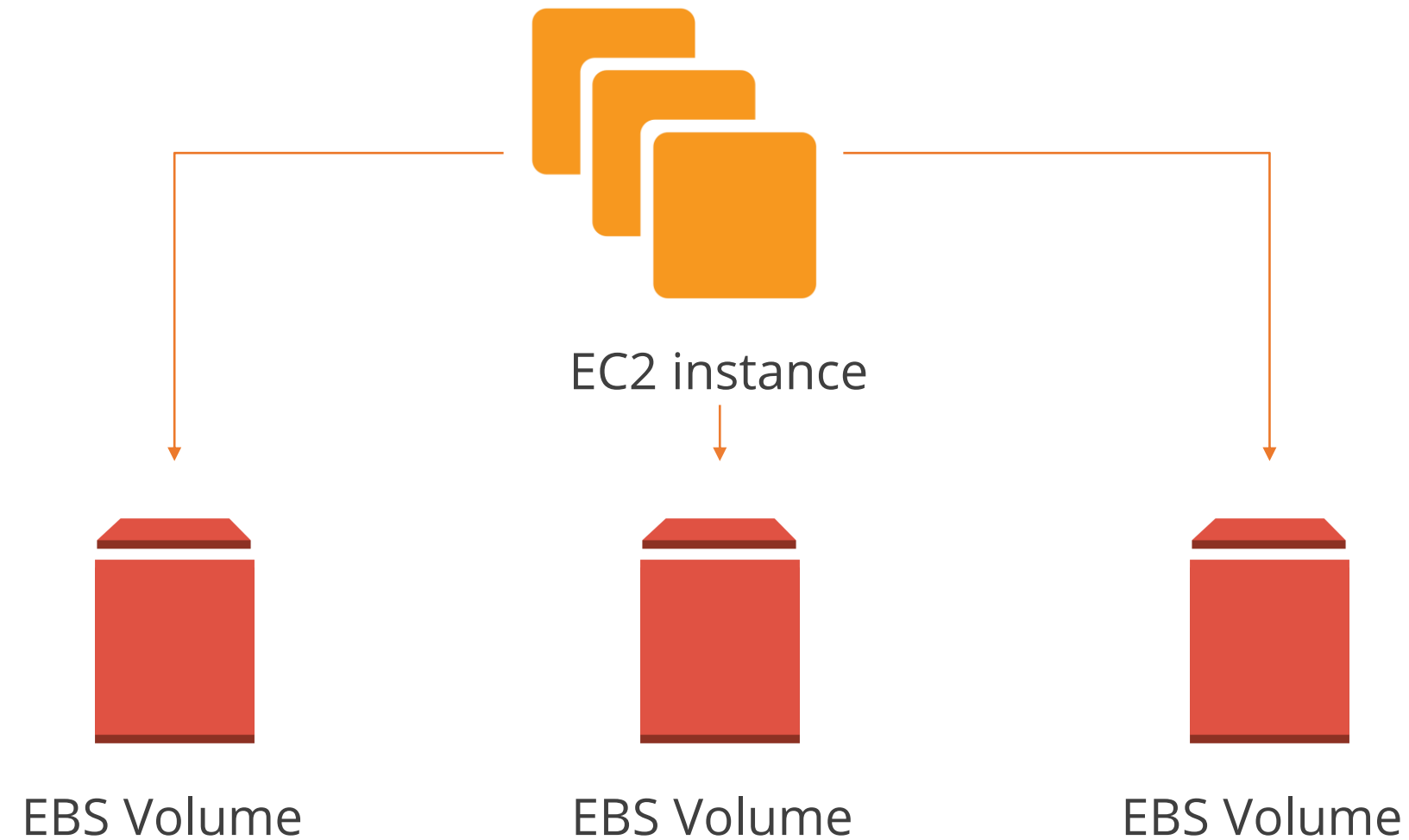
Here are a few benefits of FSx for Lustre:



## Amazon EBS volume

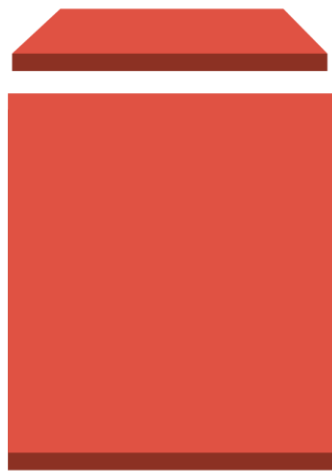
# Amazon EBS volume

An Amazon EBS volume is a long-lasting, block-level storage device that can be attached to instances. You can use a volume attached to an instance just like a physical hard disk drive.



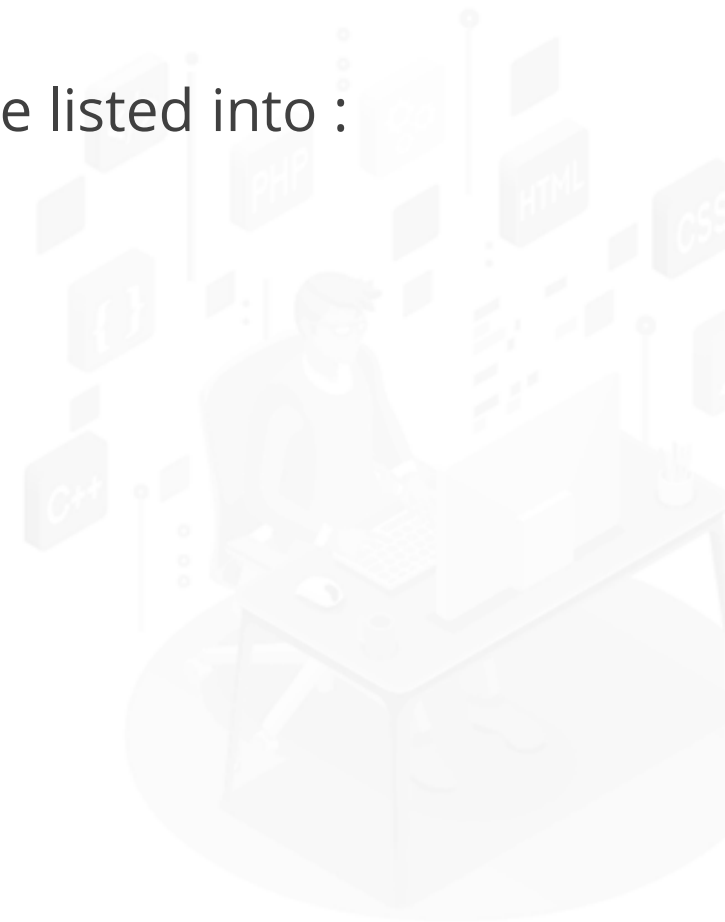


# EBS volume types



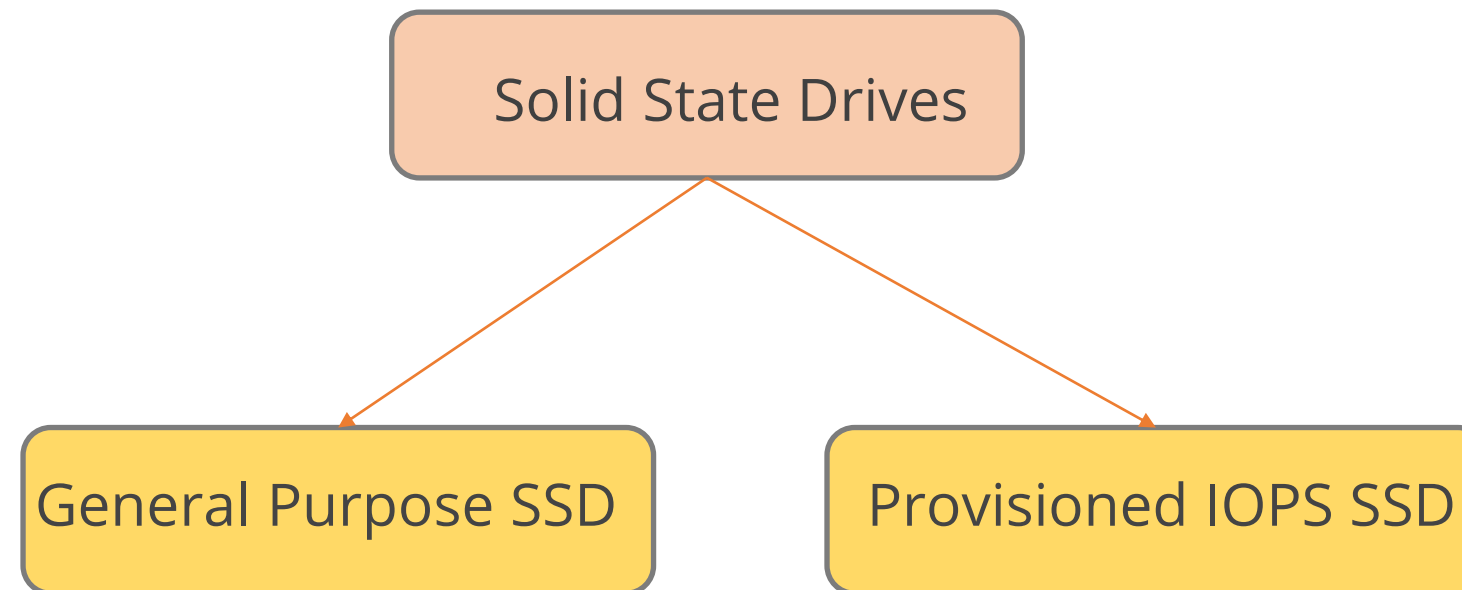
EBS Volume

- Amazon EBS volume types can be listed into :
- Solid state drives (SSD)
- Hard disk drives (HDD)
- Previous Generations
- Magnetic Volumes



# Solid state drives (SSD)

Optimized for transactional workloads with low I / O size and frequent read or write operations, where the primary performance attribute is IOPS.



# General Purpose SSD

The General-purpose SSD drive is used to balance the performance and price and this can be used for the most workload. There are two types of General-purpose SSD volumes, they are GP2 and GP3:

Volume type	gp3	gp2
Volume size	1 GiB – 16 TiB	
Use cases	<ul style="list-style-type: none"><li>•Lower latency apps</li><li>•Development and test environments</li></ul>	
Durability	99.8% – 99.9% durability	
Amazon EBS Multi-attach	Not supported	
Max IOPS per volume	16,000	
Max throughput per volume	1,000 MiB/s	250 MiB/s
Boot volume	Supported	

# Provisioned IOPS SSD

Provisioned IOPS SSD volumes are used for high-performance, low-latency, high-throughput, and for mission-critical workloads. Three types of Provisioned IOPS SSD volumes are io2 Block Express, io2, and io1. The io2 Block Express volume type is only supported with R5b instances.

Volume type	io2 Block Express		io2	io1
Volume size	4 GiB – 64 TiB		4 GiB – 16 TiB	
Use cases	<ul style="list-style-type: none"><li>• Sub-milliseconds latency</li><li>• More than 64,000 IOPS or 1,000 MiB of throughput</li></ul>		<ul style="list-style-type: none"><li>• Workloads that require sustained IOPS performance or more than 16,000 IOPS</li><li>• I/O intensive database workload</li></ul>	
Durability	99.999% durability		99.999% durability	99.8% – 99.9% durability
EBS Multi-attach	Supported			
Max IOPS per volume	256,000	64,000		
Max throughput per volume	4,000 MiB/s	1,000 MiB/s		
Boot volume	Supported			

# Hard Disk Drives (HDD)

For heavy streaming applications that demand higher throughput, hard disc drives are ideal. Throughput is the key performance factor for SSDs.



We will now talk about the various EBS volume types that fall under HDD-backed volumes.

# Throughput Optimized HDD

They are low-cost HDD volumes with defined performance. They are designed for sequential workloads like big data processing and log processing. The volume size differs from the range of 500GB-16TB.

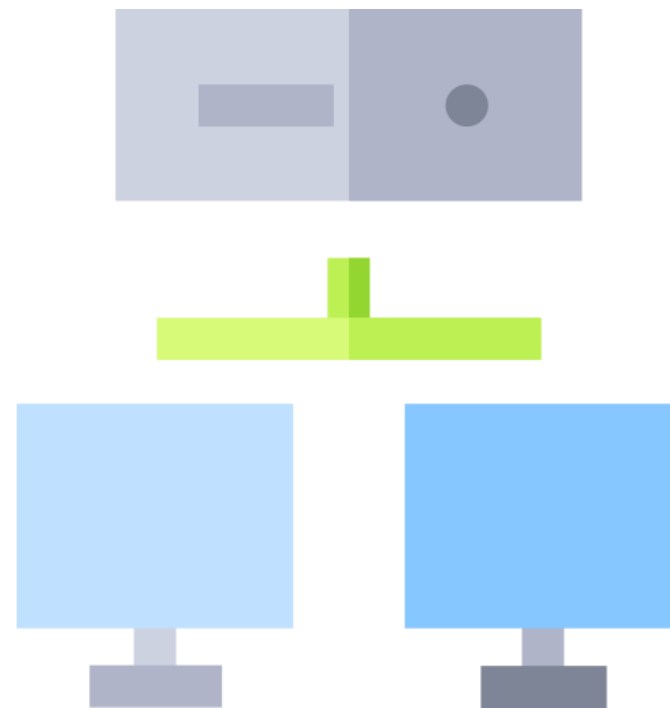
<b>Volume type</b>	st1
<b>Volume size</b>	125 GiB – 16 TiB
<b>Use cases</b>	<ul style="list-style-type: none"><li>•Big data</li><li>•Data warehouses</li><li>•Log processing</li></ul>
<b>Durability</b>	99.8% – 99.9% durability
<b>Amazon EBS Multi-attach</b>	Not supported
<b>Max IOPS per volume</b>	500
<b>Max throughput per volume</b>	500 MiB/s
<b>Boot volume</b>	Not supported

## Amazon Elastic Load Balancing



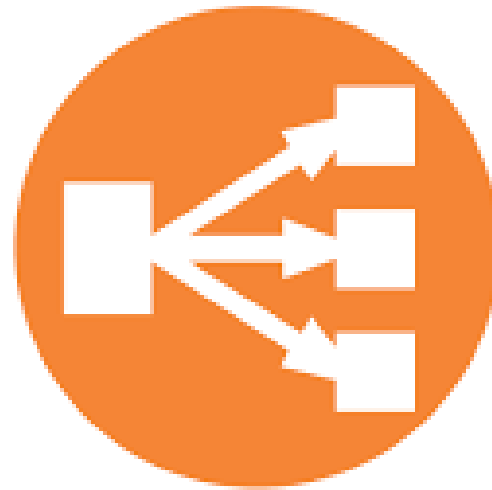
# What Is Load Balancing?

Load balancing refers to the distribution of network traffic across multiple servers or instances of virtual machines that host the application.



# Amazon Elastic Load Balancing

Amazon Elastic Load Balancing (ELB) is a load balancing service offered by AWS that distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, and Lambda functions.



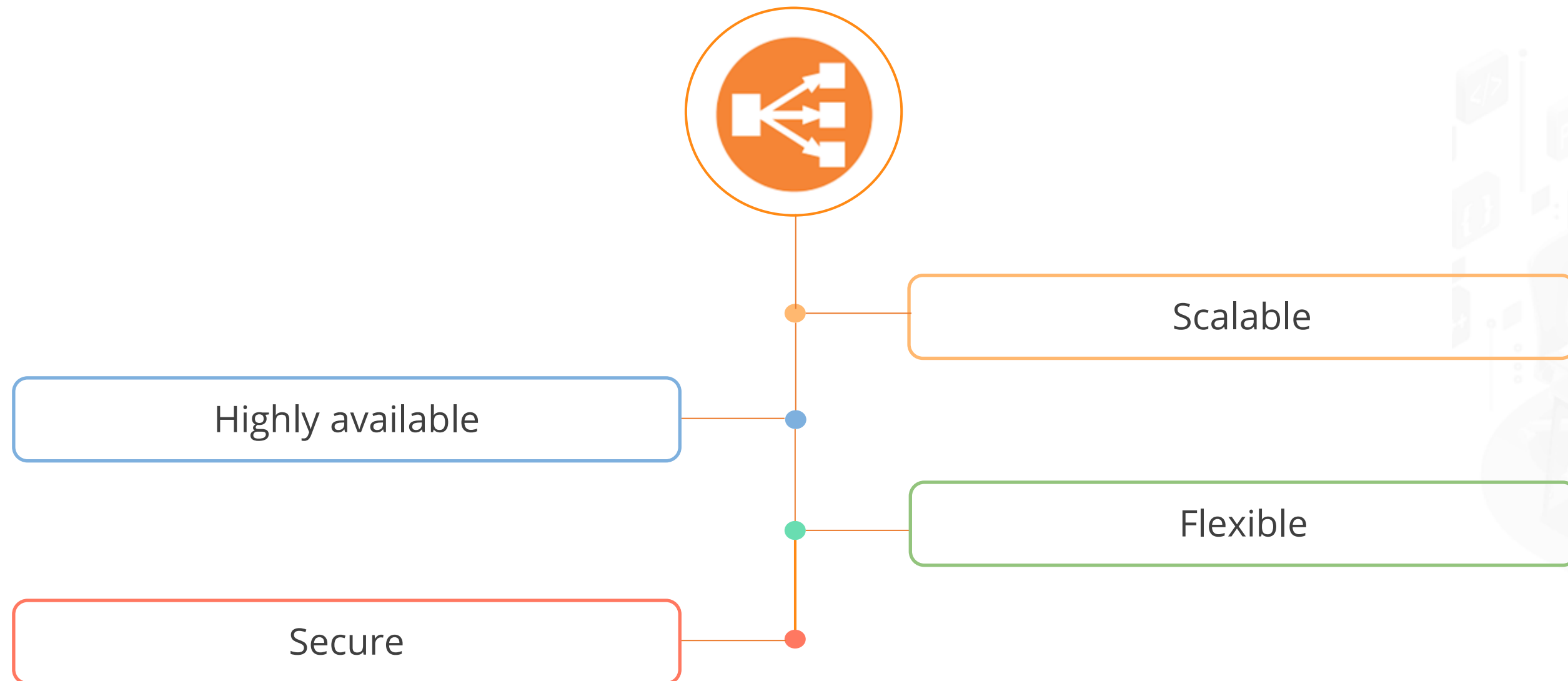
# Importance of Load Balancing

Load balancing is needed to:

01	Prevent traffic overload on any server
02	Improve application responsiveness
03	Increase availability of applications
04	Avoid single point of failure in servers

# Benefits of Amazon ELB

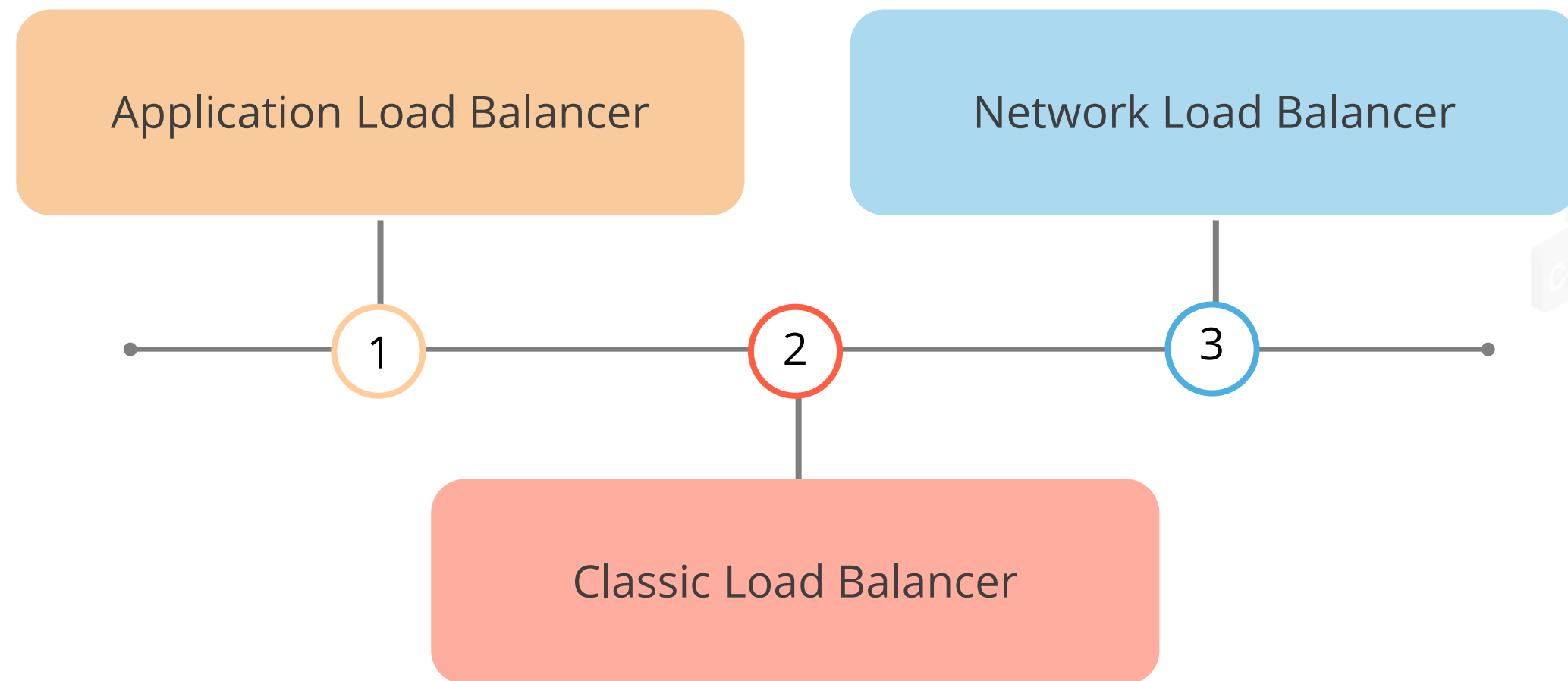
Amazon ELB offers the following benefits:



## Types of Amazon Load Balancers

# Types of Amazon Load Balancers

Amazon ELB offers the following types of load balancers:



# Application Load Balancer

Components of Application Load Balancer are:

**01**

An Application Load Balancer is used for load balancing of HTTP and HTTPS traffic.

**02**

It routes traffic to targets within Amazon Virtual Private Cloud (Amazon VPC) based on the content of the request.

**03**

It operates on layer 7.

# Network Load Balancer

Components of Network Load Balancer are:

**01**

A Network Load Balancer is used for load balancing of TCP, UDP, and TLS traffic.

**02**

It routes traffic to targets within Amazon Virtual Private Cloud (Amazon VPC) regardless of the content of the request.

**03**

It operates on layer 4.



# Classic Load Balancer

Components of Classic Load Balancer are:

**01**

A Classic Load Balancer provides basic load balancing across multiple Amazon EC2 instances.

**02**

It is best suited for applications built on the EC2-Classic network.

**03**

It operates on both layer 4 and layer 7.

# TECHNOLOGY

## AWS Lambda

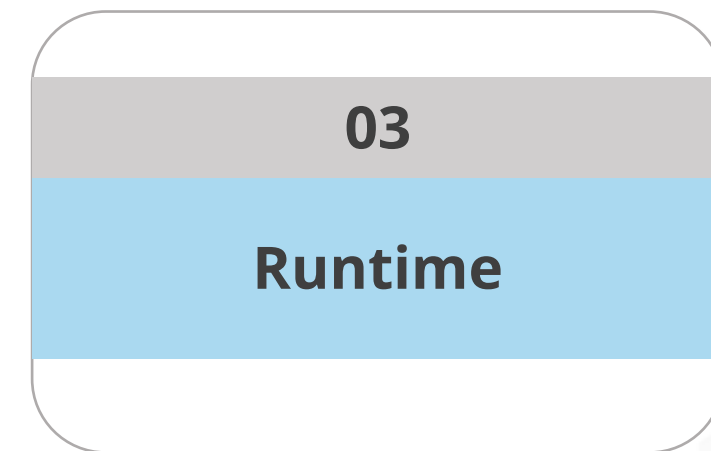
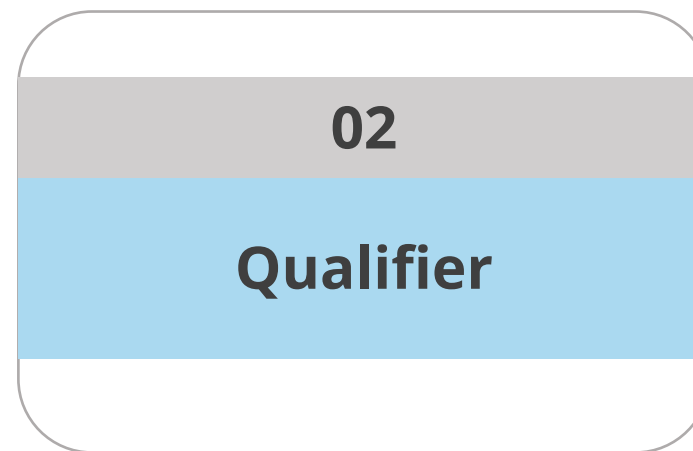
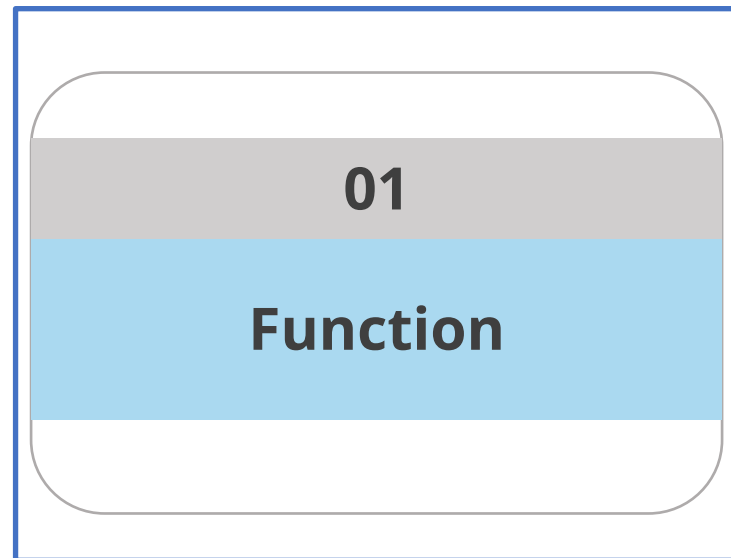
# What Is AWS Lambda?

AWS Lambda is a serverless compute service that allows users to run code without provisioning or managing servers. It executes the code only when needed and scales automatically, from a few requests per day to thousands per second.



# Terminologies in AWS Lambda

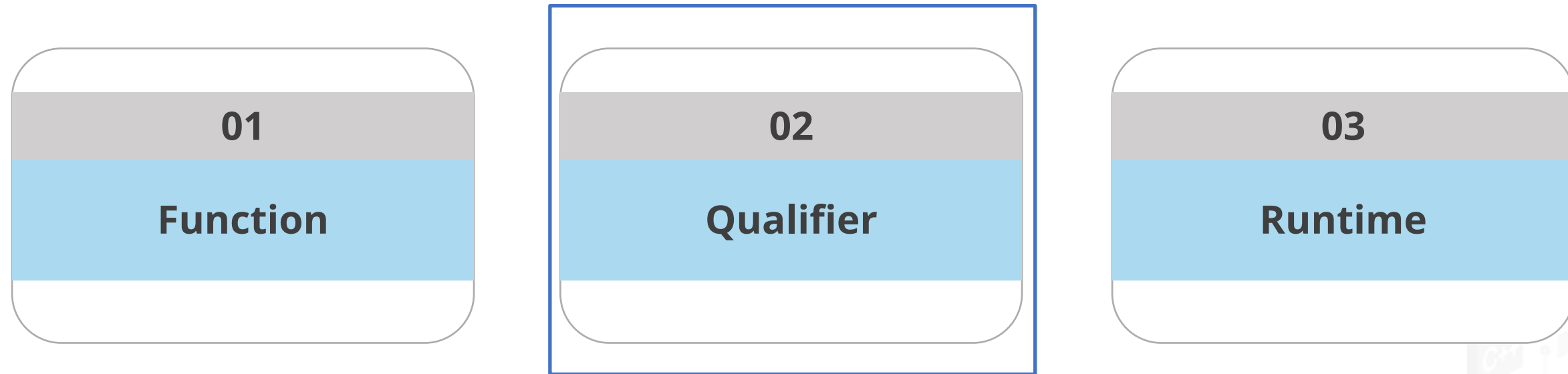
The terminologies used in context with AWS Lambda are:



A function is a resource that contains a code to process events and runtime to pass requests between Lambda and the function code.

# Terminologies in AWS Lambda

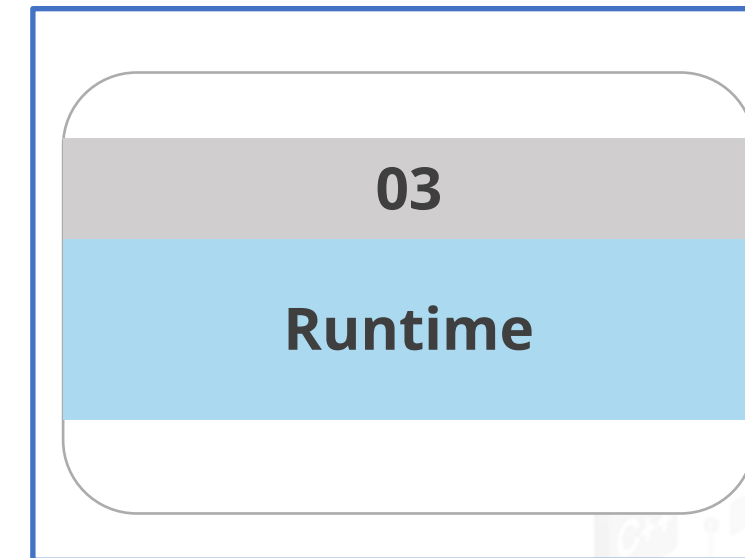
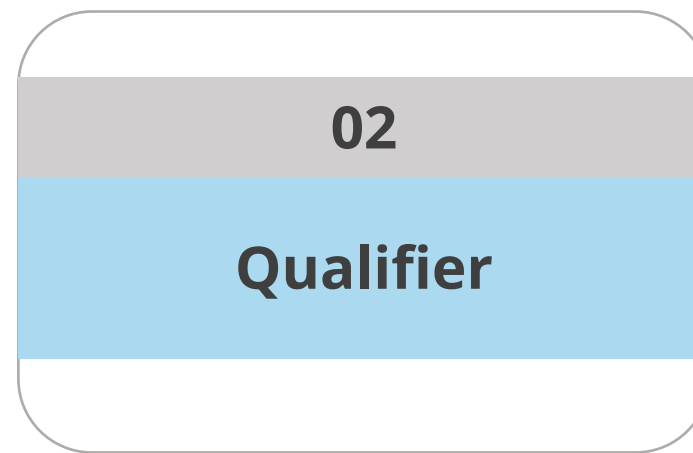
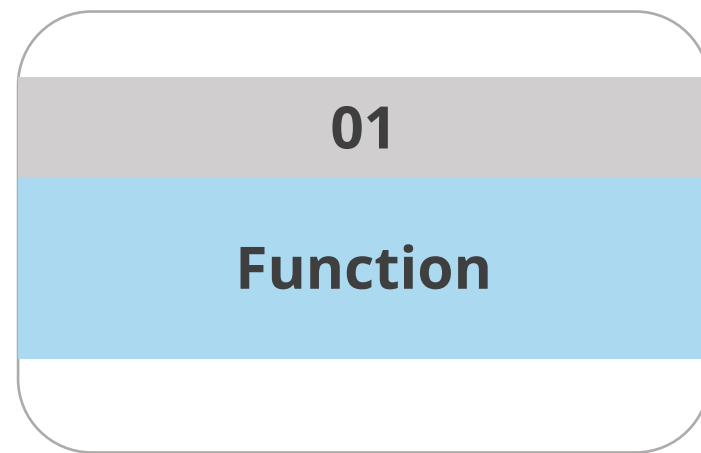
The terminologies used in context with AWS Lambda are:



A qualifier is used to specify a version or an alias for a Lambda function.

# Terminologies in AWS Lambda

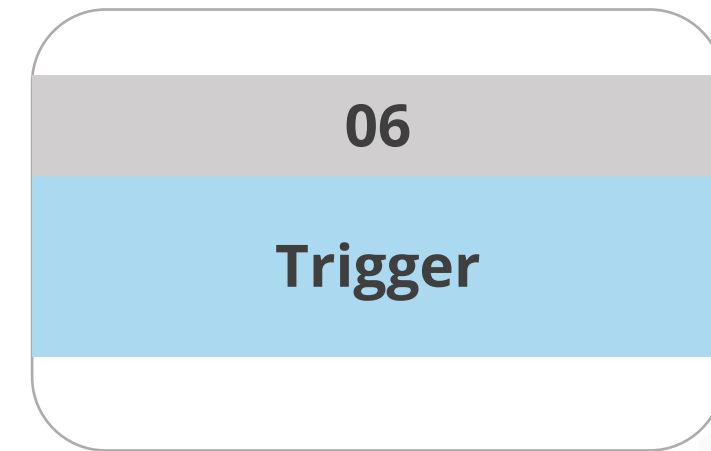
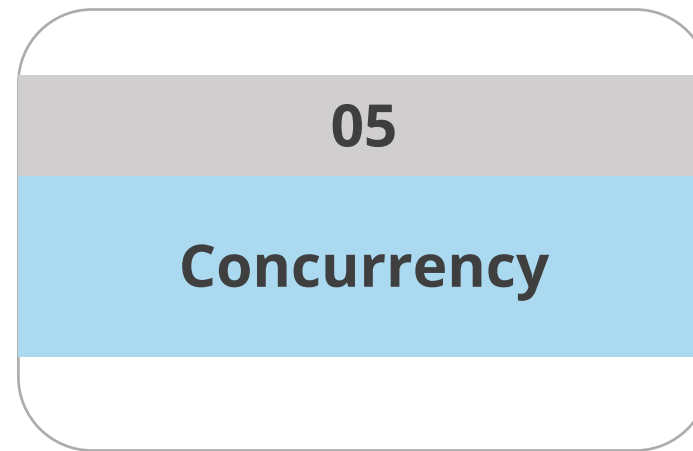
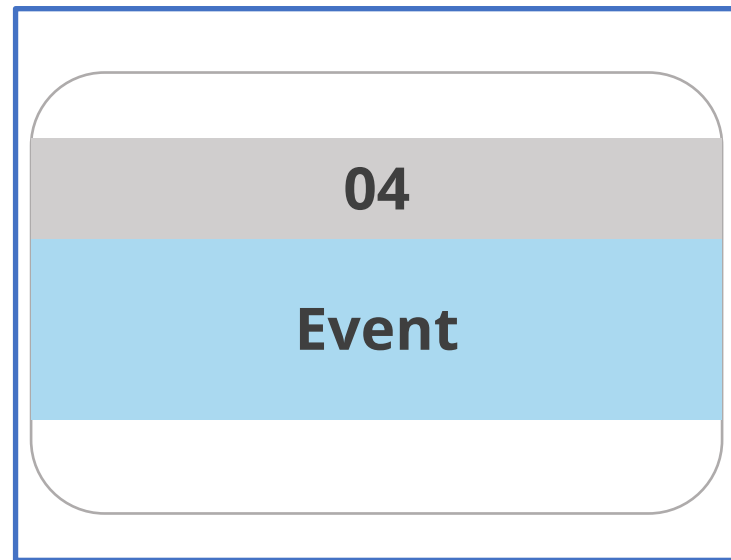
The terminologies used in context with AWS Lambda are:



Runtime terminology allows the function code written in different languages to run in the same base execution environment. Users can choose a runtime based on the programming language of their code.

# Terminologies in AWS Lambda

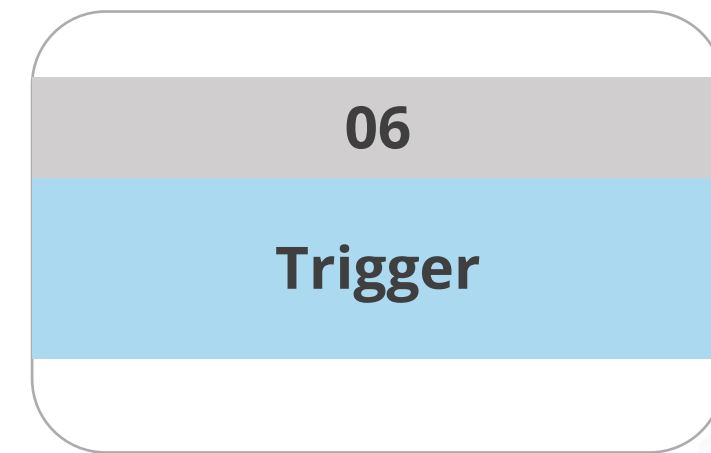
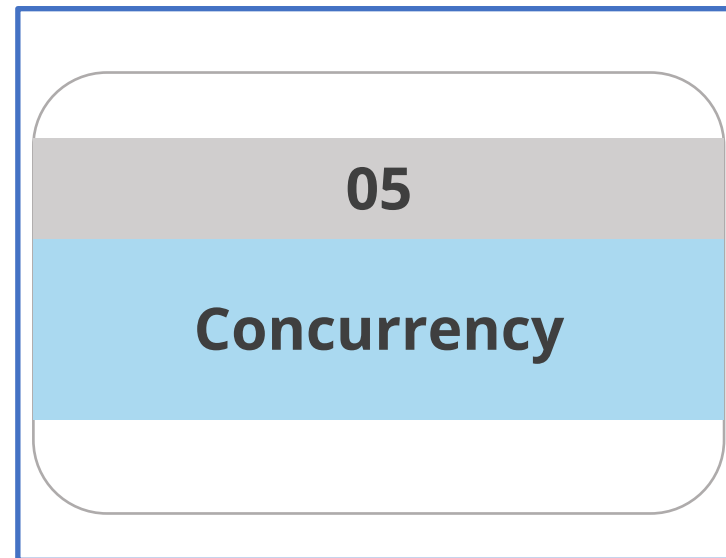
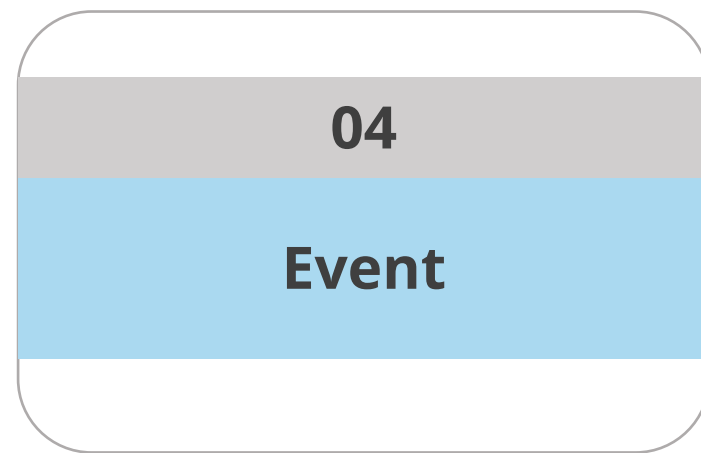
The terminologies used in context with AWS Lambda are:



An event is a JSON formatted document that contains data for a function to process. It is converted to an object and passed to the function code.

# Terminologies in AWS Lambda

The terminologies used in context with AWS Lambda are:

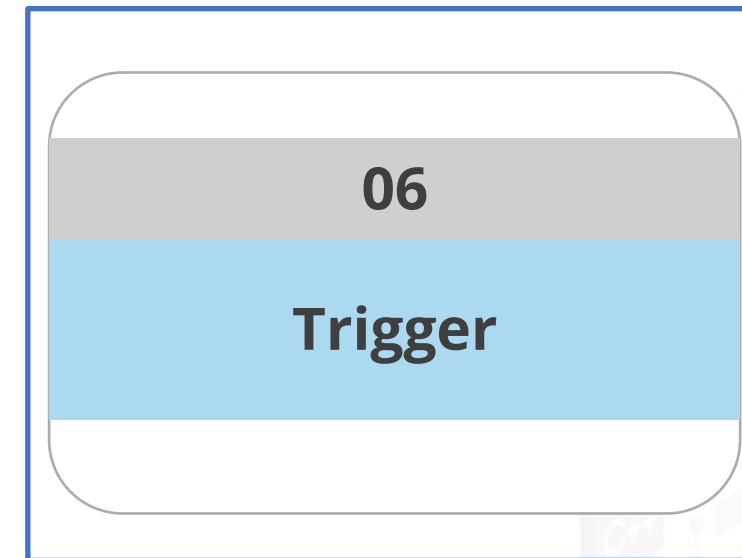
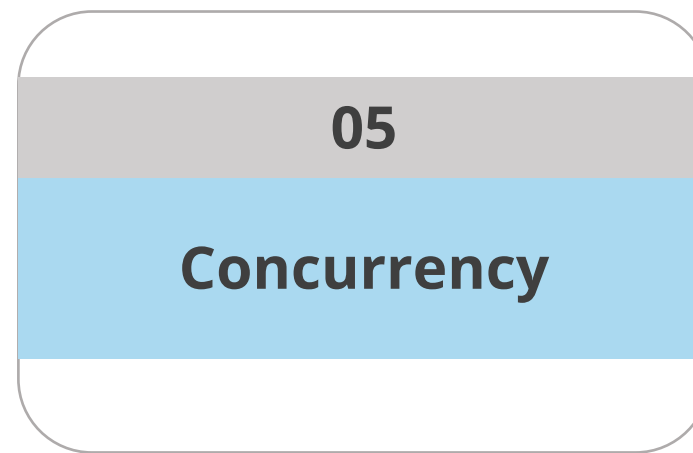
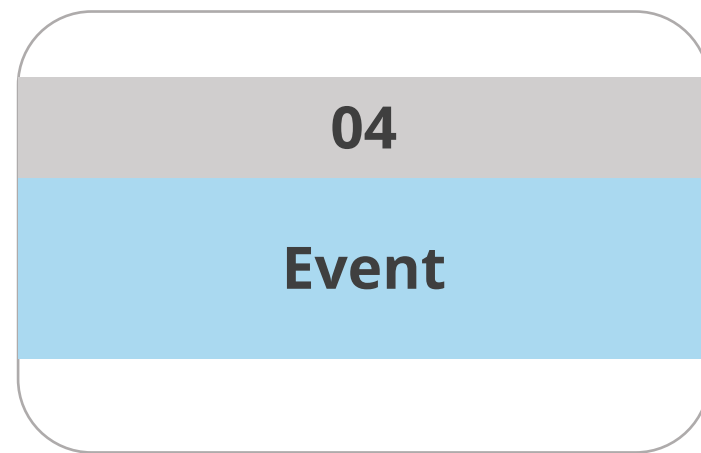


Concurrency is the number of requests that a function is serving at any given time. Users can configure their functions to limit their concurrency.



# Terminologies in AWS Lambda

The terminologies used in context with AWS Lambda are:

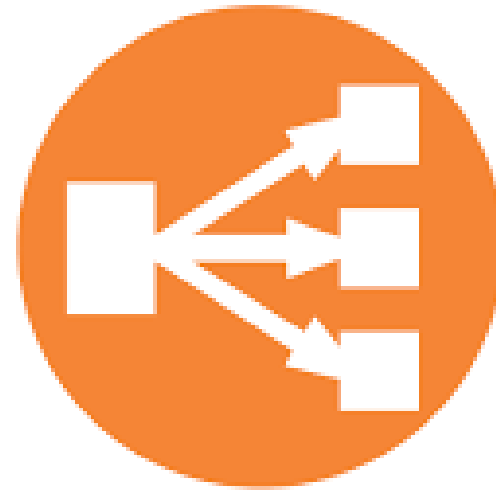


A trigger is a resource that invokes a Lambda function. It can be an AWS service, an application, or an event.

## Gateway Load Balancer

# Gateway Load Balancer

A Gateway Load Balancer operates at the third layer of the Open Systems Interconnection (OSI) model. It is used to deploy and manage a fleet of virtual appliances that support the GENEVE protocol.



## Note

Users can add or remove targets from the Load Balancer without disrupting the overall flow of requests.

# Gateway Load Balancer

Components of Gateway Load Balancer are:

**01**

It listens to all IP packets across all ports and forwards traffic to the target group that's specified in the listener rule using the GENEVE protocol on port 6081.

**02**

Elastic Load Balancing scales your load balancer as traffic to your application changes over time.

**03**

Elastic Load Balancing can scale to the vast majority of workloads automatically.

# Gateway Load Balancer: States

A Gateway Load Balancer can be in one of the following states:



01

## **Provisioning:**

The Gateway Load Balancer is being set up.

02

## **Active:**

The Gateway Load Balancer is fully set up and ready to route traffic.

03

## **Failed:**

The Gateway Load Balancer could not be set up.

# Launch WebServer Instances



**Duration: 2 mins**

## **Problem Statement:**

You need to launch webserver instances.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

Steps to be followed:

1. In the console navigation pane, choose VPC.
2. Click in Subnets and click on Create subnet.
3. Create an internet gateway
4. Select the instance, copy the Public IPv4 address



# Classic Load Balancer



**Duration: 10 mins**

## **Problem Statement:**

You have been asked to create and deploy a Classic Load Balancer.

ASSISTED PRACTICE



# Assisted Practice: Guidelines

---

Steps to be followed:

1. Create a Security Group
2. Launch two Instances with different Availability Zones
3. Create a Classic Load Balancer
4. Deploy the Classic Load Balancer to an EC2 Instance



## HTTP 503: Service Unavailable Error

# HTTP 503: Service Unavailable Error

The **HTTP 503: Service unavailable error** indicates that either the load balancer or the registered instances are causing the error.

The following are a few causes and solutions for the HTTP 503: Service unavailable error:



## **Cause 1:**

Insufficient capacity in the load balancer to handle the request

## **Solution 1:**

This should be a transient issue and should not last more than a few minutes. If it persists, go to the AWS Support Center for assistance

# HTTP 503: Service Unavailable Error

The following are a few causes and solutions for the HTTP 503: Service unavailable error:

## Cause 2:

There are no registered instances.

## Solution 2:

A user needs to register at least one instance in every Availability Zone, where the load balancer is configured to respond. Verify this by looking at the **HealthyHostCount** metrics in CloudWatch. If a user is unable to ensure that an instance is registered in each Availability Zone, enabling cross-zone load balancing is recommended.



# HTTP 503: Service Unavailable Error

The following are a few causes and solutions for the HTTP 503: Service unavailable error:



## Cause 3:

There are no healthy instances.

## Solution 3:

A user needs to ensure that they have healthy instances in every Availability Zone, where their load balancer is configured to respond. This can be verified by looking at the **HealthyHostCount** metric.

# HTTP 503: Service Unavailable Error

The following are a few causes and solutions for the HTTP 503: Service unavailable error:



## Cause 4:

The surge queue is full.

## Solution 4:

A user needs to ensure that their instances have sufficient capacity to handle the request rate. This can be verified by looking at the **SpillOverCount** metric.



# Application Load Balancer



**Duration: 10 Min.**

## Problem Statement:

You have been asked to create a load balancer using the AWS Management Console

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

Steps to be followed:

1. Configure a target group
2. Register targets
3. Configure a load balancer and a listener
4. Test the load balancer





# Routing Requests in ALB



**Duration: 8 Min.**

## **Problem Statement:**

You have been asked to create a Routing Requests in ALB.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

Steps to be followed:

1. Setting up the prerequisites for EC2
2. Creating a routing request in ALB



# Network Load Balancer



**Duration: 8 Min.**

## **Problem Statement:**

You have been asked to create a Network Load Balancer.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

Steps to be followed:

1. Configure your target group
2. Choose a load balancer type
3. Configure your load balancer and listener
4. Test the load balancer
5. Delete your load balancer



## EBS volume Encryption

# EBS Encryption

An AWS KMS key is a logical representation of an encryption key.



The KMS key also includes data about the key itself, such as ID, description, and creation date and EBS encrypts your volume with a data key using the industry-standard AES-256 algorithm.

# EBS Encryption

EBS encryption works when the snapshot is encrypted.



EBS encrypted

- Amazon EC2 sends a **GenerateDataKeyWithoutPlaintext** request to AWS KMS, specifying the KMS key to choose for volume encryption.
- If the volume is encrypted using the same KMS key as the snapshot, AWS KMS uses the same data key as the snapshot and encrypts it under that same KMS key

# EBS Encryption

EBS encryption works when the snapshot is encrypted



EBS encrypted

- When you attach the encrypted volume to an instance, Amazon EC2 sends a **CreateGrant** request to AWS KMS
- AWS KMS decrypts the encrypted data key and sends the decrypted data key to Amazon EC2.



# EBS Encryption

EBS encryption works when the snapshot is unencrypted



EBS unencrypted

- Amazon EC2 sends a **CreateGrant** request to AWS KMS.
- Amazon EC2 sends a **GenerateDataKeyWithoutPlaintext** request to AWS KMS, specifying the KMS key to choose for volume encryption.
- AWS KMS generates a new data key, encrypts it under the KMS key volume encryption, and sends the encrypted data key to Amazon EBS to be stored with the volume metadata.
- Amazon EC2 sends a Decrypt request to AWS KMS to get the encryption key to encrypt the volume data.

# EBS Encryption

EBS encryption works when the snapshot is unencrypted



EBS unencrypted

- When you attach the encrypted volume to an instance, Amazon EC2 sends a CreateGrant request to AWS KMS.
- When you attach the encrypted volume to an instance, Amazon EC2 sends a Decrypt request to AWS KMS, specifying the encrypted data key.
- AWS KMS decrypts the encrypted data key and sends the decrypted data key to Amazon EC2.
- Amazon EC2 uses the plaintext data key in hypervisor memory to encrypt disk I/O to the volume.

## Load Balancer Access Logs

# Load Balancer Access Logs

The different classifications of access logs for different Load Balancers are:

Application Load Balancer

Network Load Balancer

Classic Load Balancer

- Access logging is an optional feature of Elastic Load Balancing that is disabled by default.
- Each log contains information such as the time the request was received, the client's IP address, latencies, request paths, and server responses.
- These access logs can be used to analyze traffic patterns and troubleshoot issues.

# Load Balancer Access Logs

The different classifications of access logs for different Load Balancers are:

Application Load Balancer

Network Load Balancer

Classic Load Balancer

- Access logging is an optional feature of Elastic Load Balancing that is disabled by default.
- It captures the logs and stores them in the Amazon S3 bucket that the users specify as compressed files once the access logging is enabled.
- Access logging can be disabled at any time.

# Load Balancer Access Logs

The different classifications of access logs for different Load Balancers are:

Application Load Balancer

Network Load Balancer

Classic Load Balancer

- There is no additional charge for access logs.
- The users are charged storage costs for Amazon S3, but not charged for the bandwidth used by Elastic Load Balancing to send log files to Amazon S3.

# Access Log Files

Elastic Load Balancing publishes a log file for each Load Balancer node every 5 minutes.  
The file names of the access logs use the following format:

- bucket: The name of the S3 bucket
- prefix: The prefix (logical hierarchy) in the bucket
- aws-account-id: The AWS account ID of the owner
- region: The Region of a Load Balancer and S3 bucket
- yyyy/mm/dd: The date that the log was delivered



# Access Log Files

Elastic Load Balancing publishes a log file for each Load Balancer node every 5 minutes. The file names of the access logs use the following format:

- load-balancer-id: The resource ID of the Load Balancer
- end-time: The date and time that the logging interval ended
- ip-address: The IP address of the Load Balancer node that handled the request
- random-string: A system-generated random string

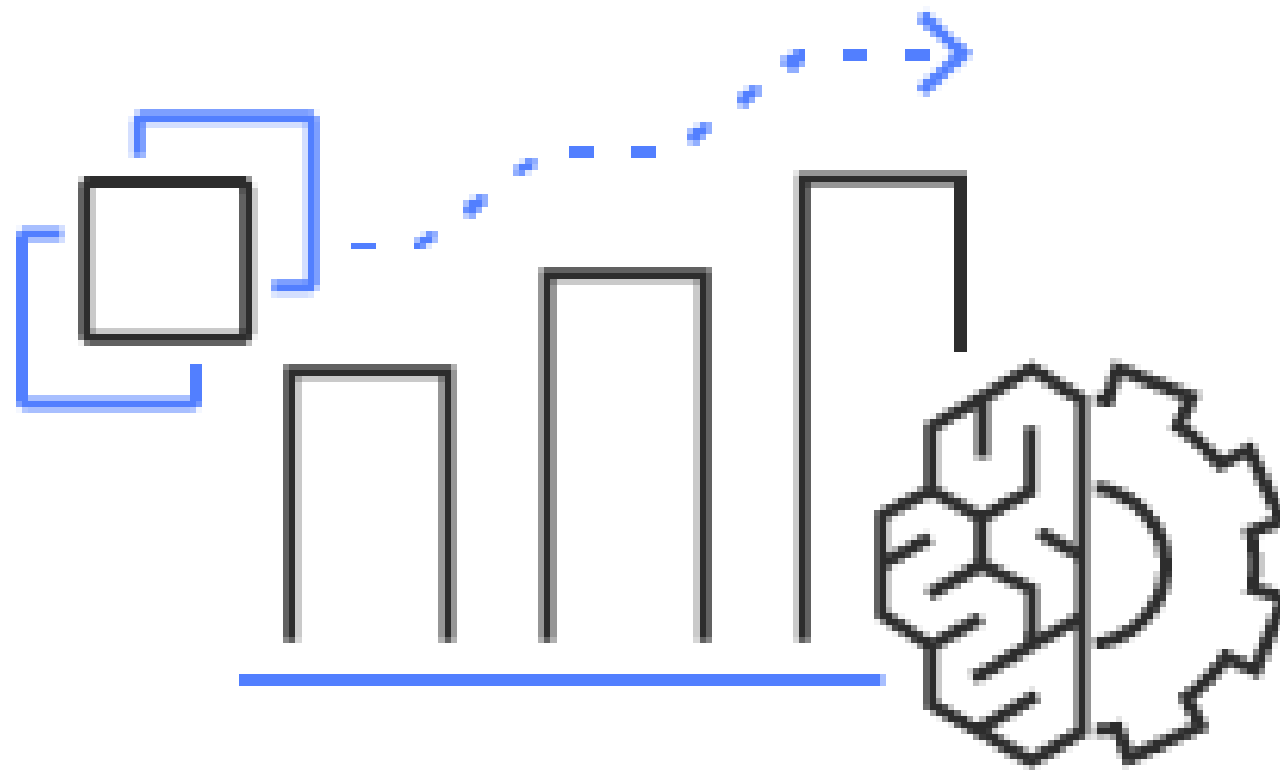




## Auto Scaling

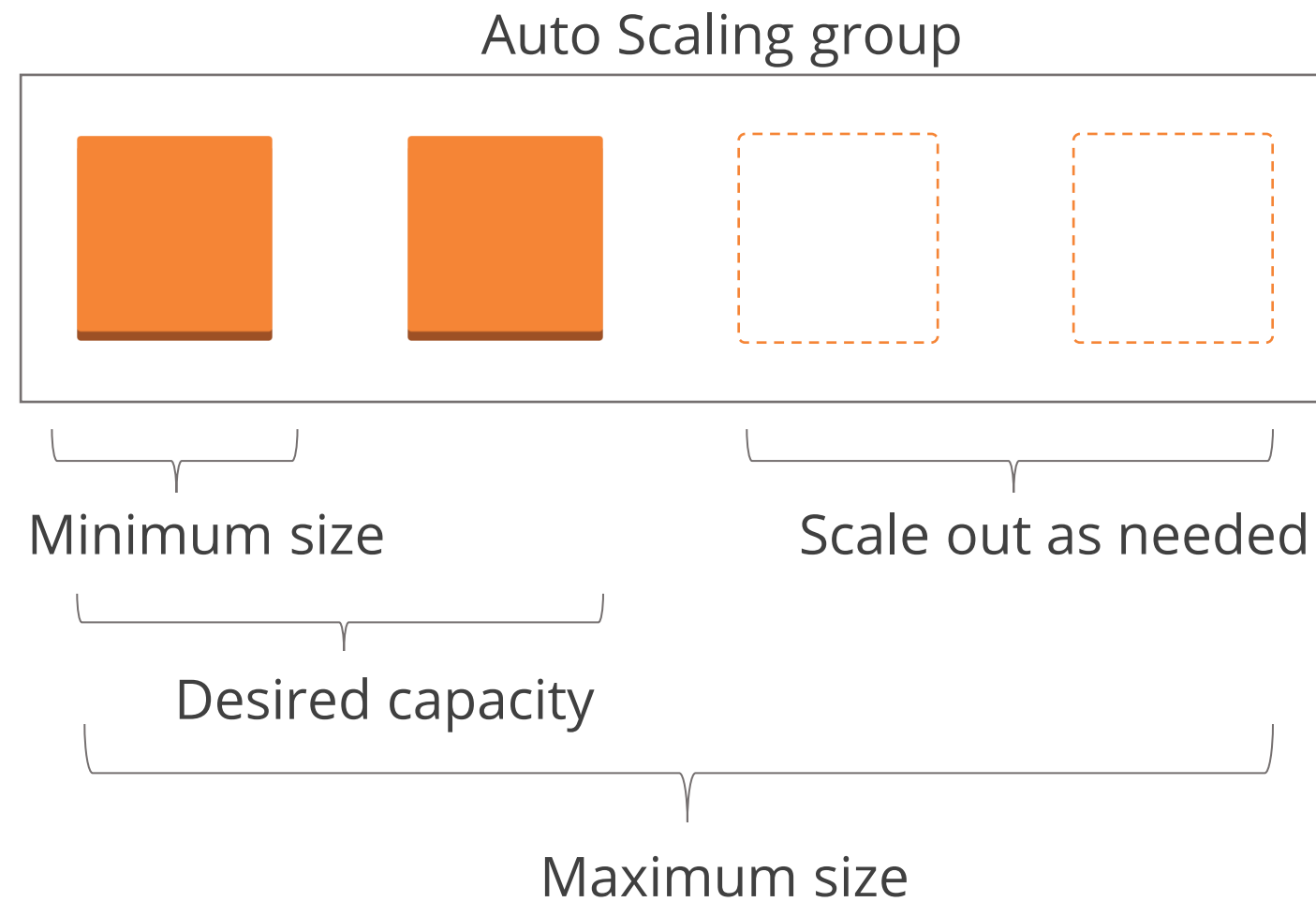
# Auto Scaling

Amazon EC2 Auto Scaling helps the users to maintain application availability. According to the conditions defined by the users, EC2 Auto Scaling allows them to automatically add or remove EC2 instances.



# Auto Scaling Groups

A collection of EC2 instances is called an Auto Scaling group. The users can specify the minimum number of instances in each group, and Auto Scaling ensures that the group never goes below the minimum size.



# Auto Scaling

The Difference between the load balancer and Auto Scaling

## Load Balancer

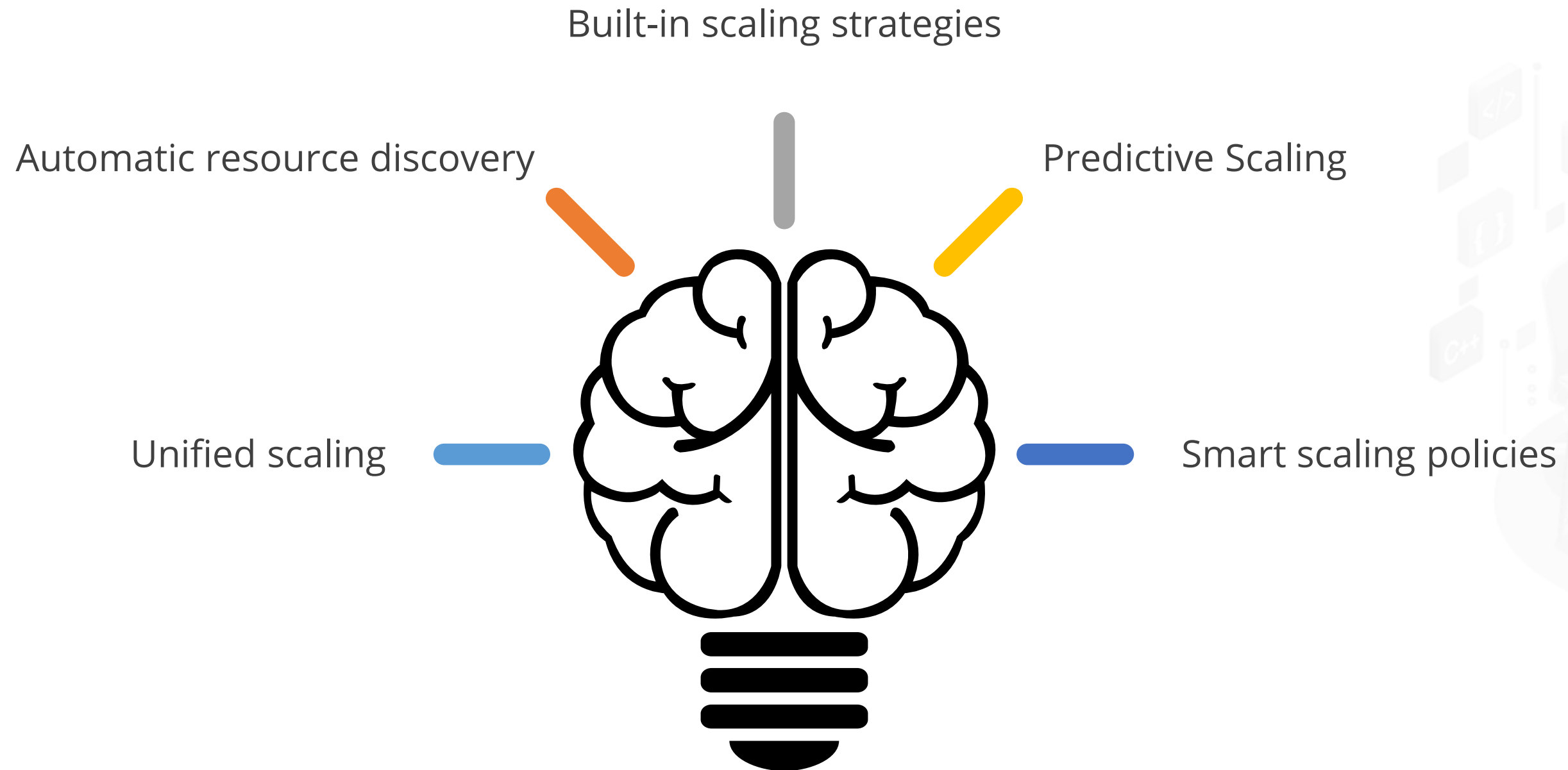
Elastic load balancer only monitors each instance's health, distributes traffic, and connects each request to the right target groups.

## Load Balancer

The user's ability to specify policies based on predefined criteria is enabled by auto-scaling, which enables multiple instances to work parallel.

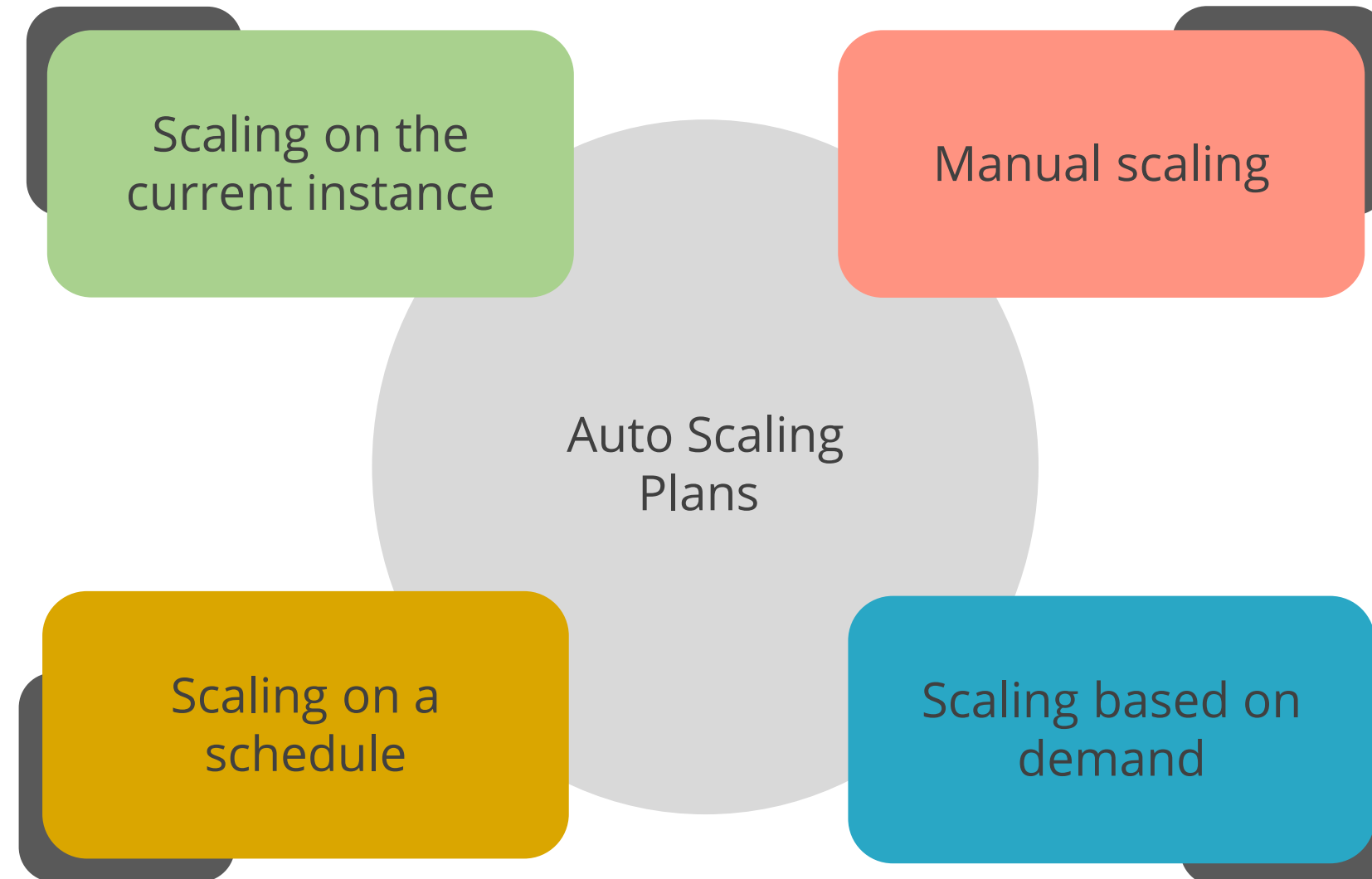
# Auto-Scaling Features

Some of the features of Auto scaling are:



# Auto Scaling Planning

Here are the four categories of auto scaling planning:

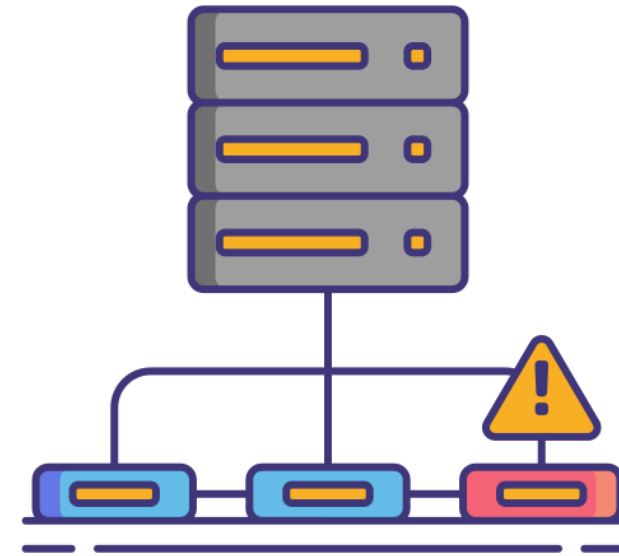


# Auto Scaling Benefits

Better fault tolerance

Increased application availability

Lower costs



Amazon EC2 Auto Scaling can determine the health of an instance. It can terminate the instance and replace it with a new one.

# Auto Scaling Benefits

Better fault tolerance

Increased application availability

Lower costs



Amazon EC2 Auto Scaling ensures that the application always has the right amount of computing and proactively provisions capacity with Predictive Scaling.



# Auto Scaling Benefits

Better fault tolerance

Increased application availability

Lower costs

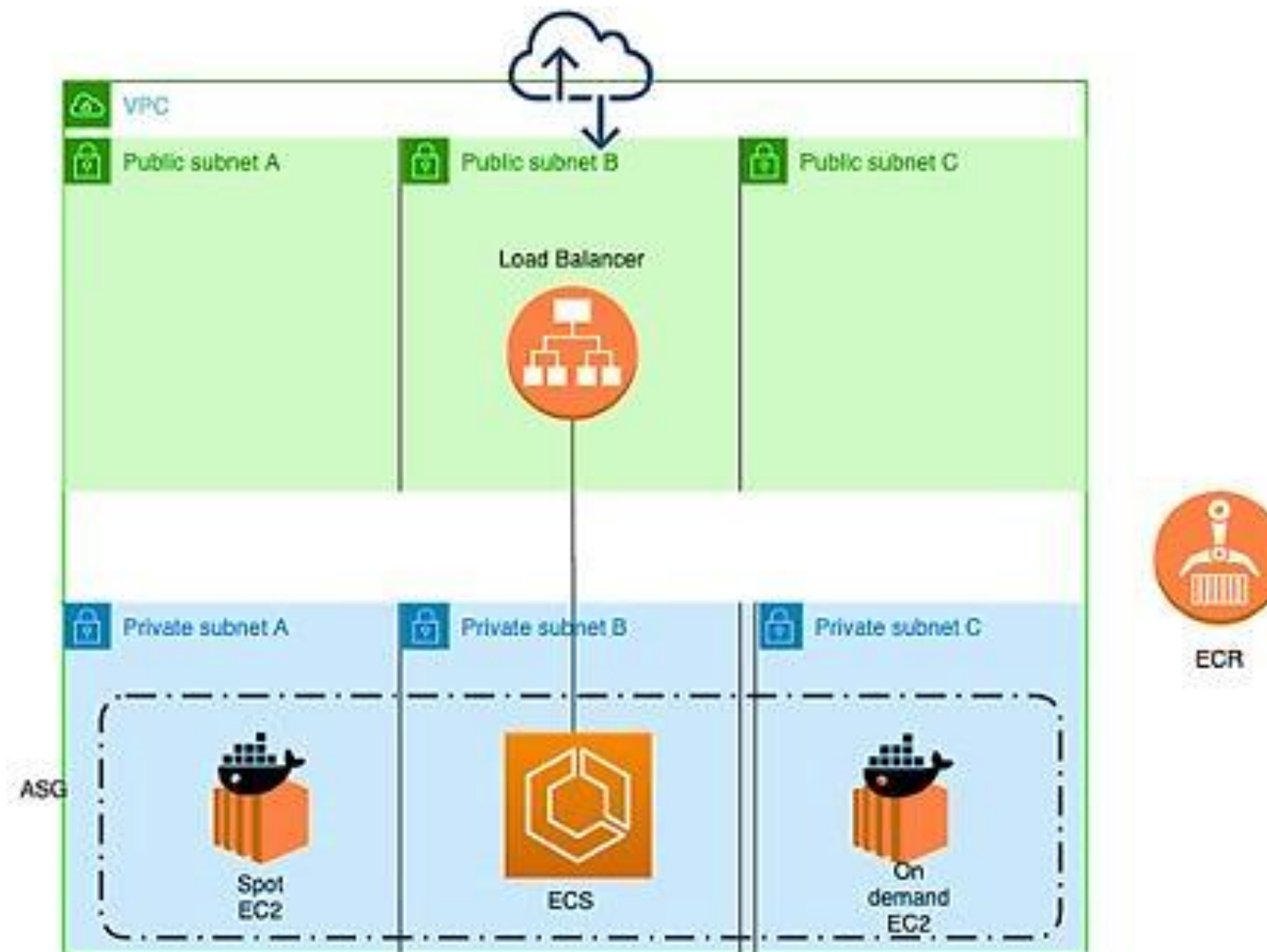


Amazon EC2 Auto Scaling adds instances only when needed and it can scale across purchase options to optimize performance and cost.

## Maintain Fleet with Auto Scaling

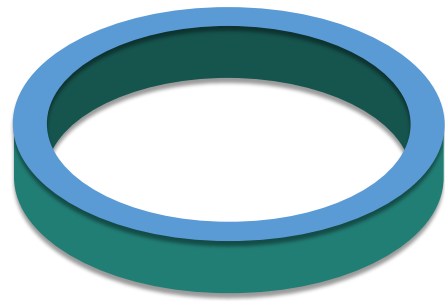
# Fleet Management with Auto Scaling

If the user's application runs on Amazon EC2 instances, the user has what is known as a fleet. The Users can automate the management of a user's fleet with Auto Scaling, and it is easy to set up.

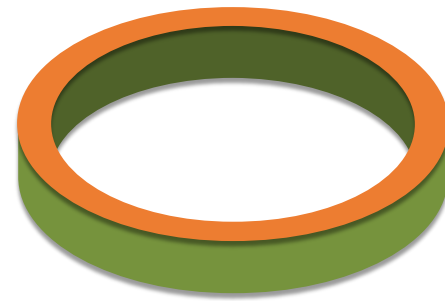


# Functions of Auto Scaling

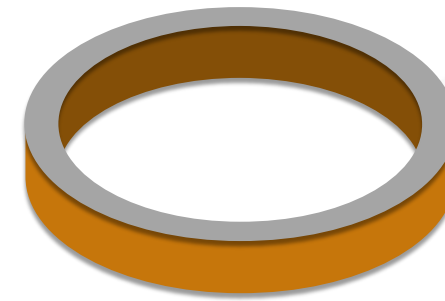
The three main functions that Auto Scaling performs to automate fleet management for EC2 instances are as follows:



Monitoring the health of running instances



Automatically replacing impaired instances

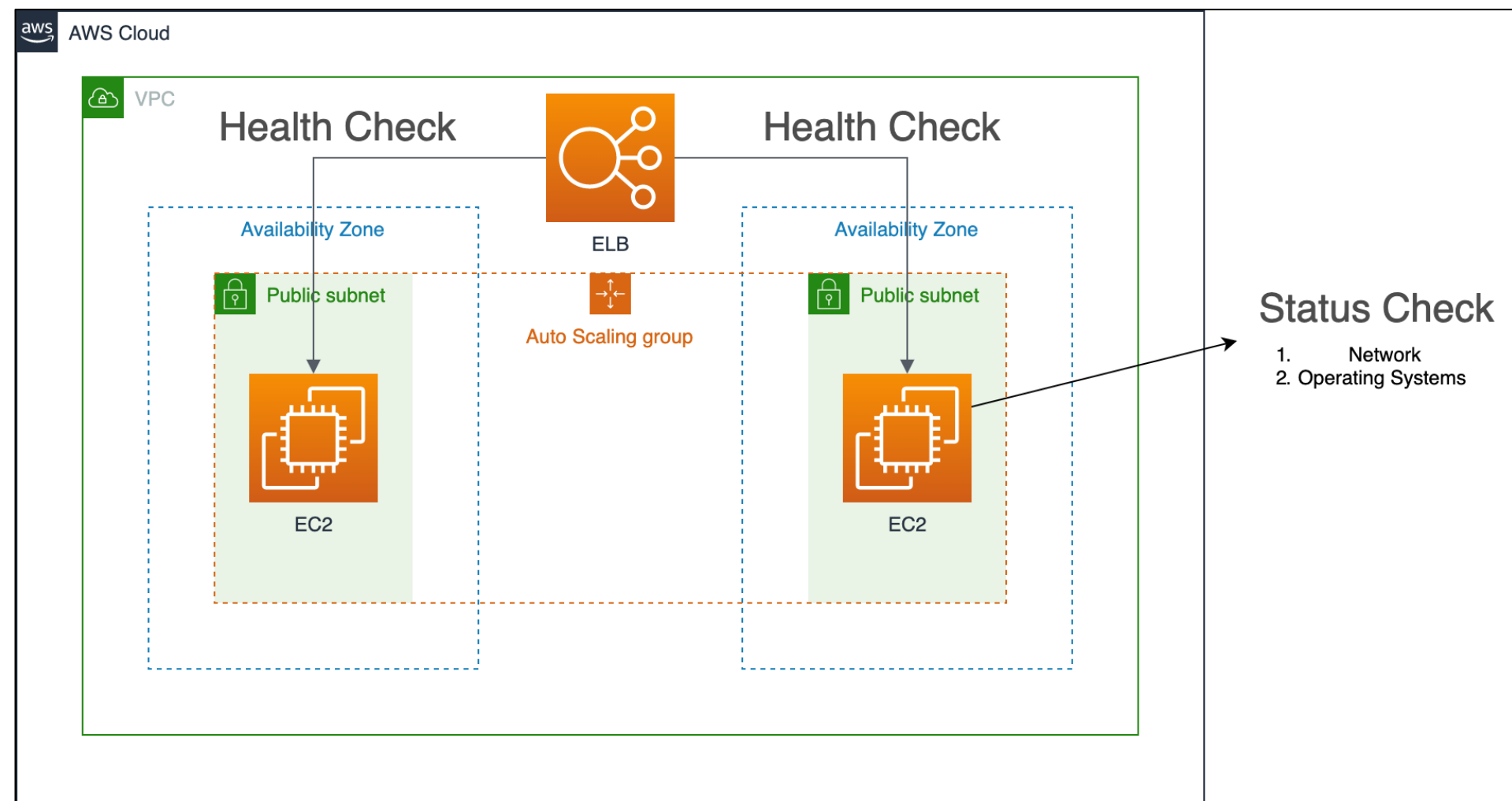


Balancing capacity across Availability Zones



# Monitoring the Health of Running Instances

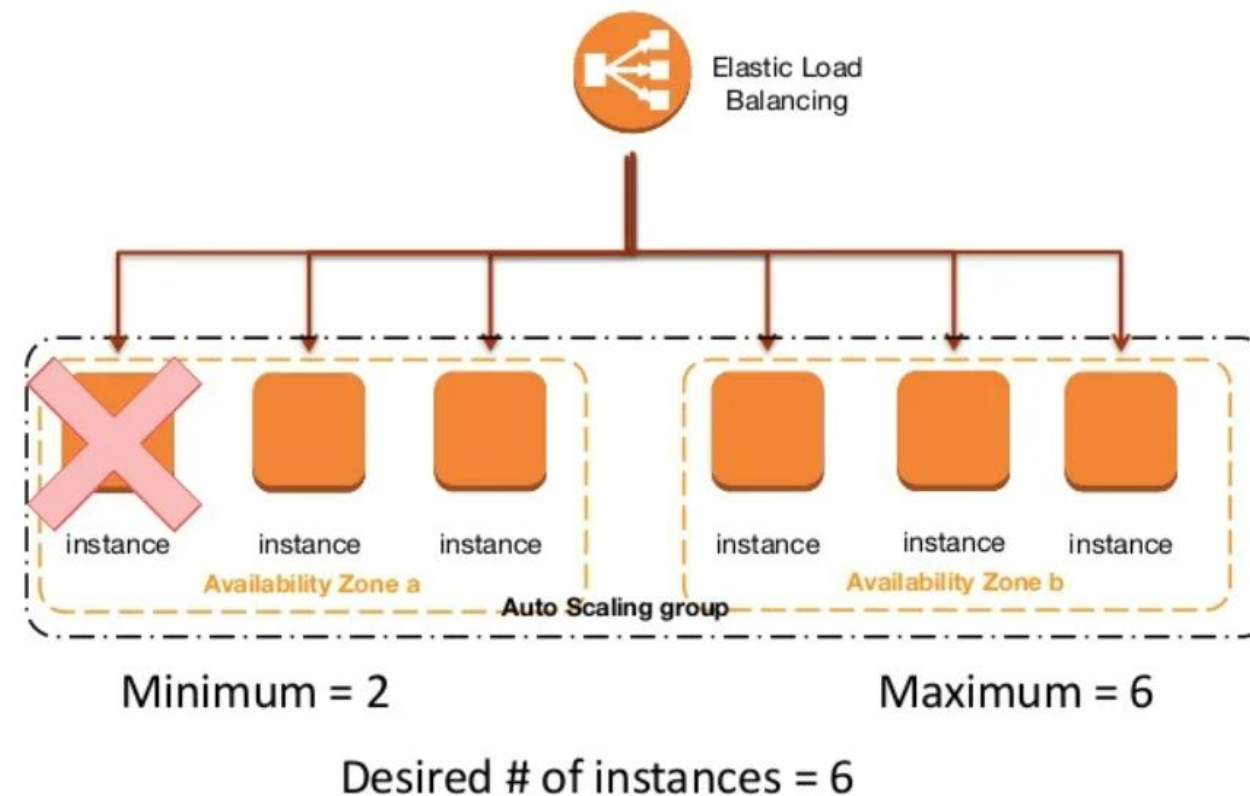
Auto Scaling monitors the health of all instances that are placed within an **Auto Scaling group**. Auto Scaling performs EC2 health checks at regular intervals, and if the instance is connected to an **Elastic Load Balancing** load balancer, it can also perform ELB health checks.



# Automatically Replacing Impaired Instances

When an impaired instance fails a health check, Auto Scaling automatically terminates it and replaces it with a new one.

## Unhealthy Instances Get Replaced...

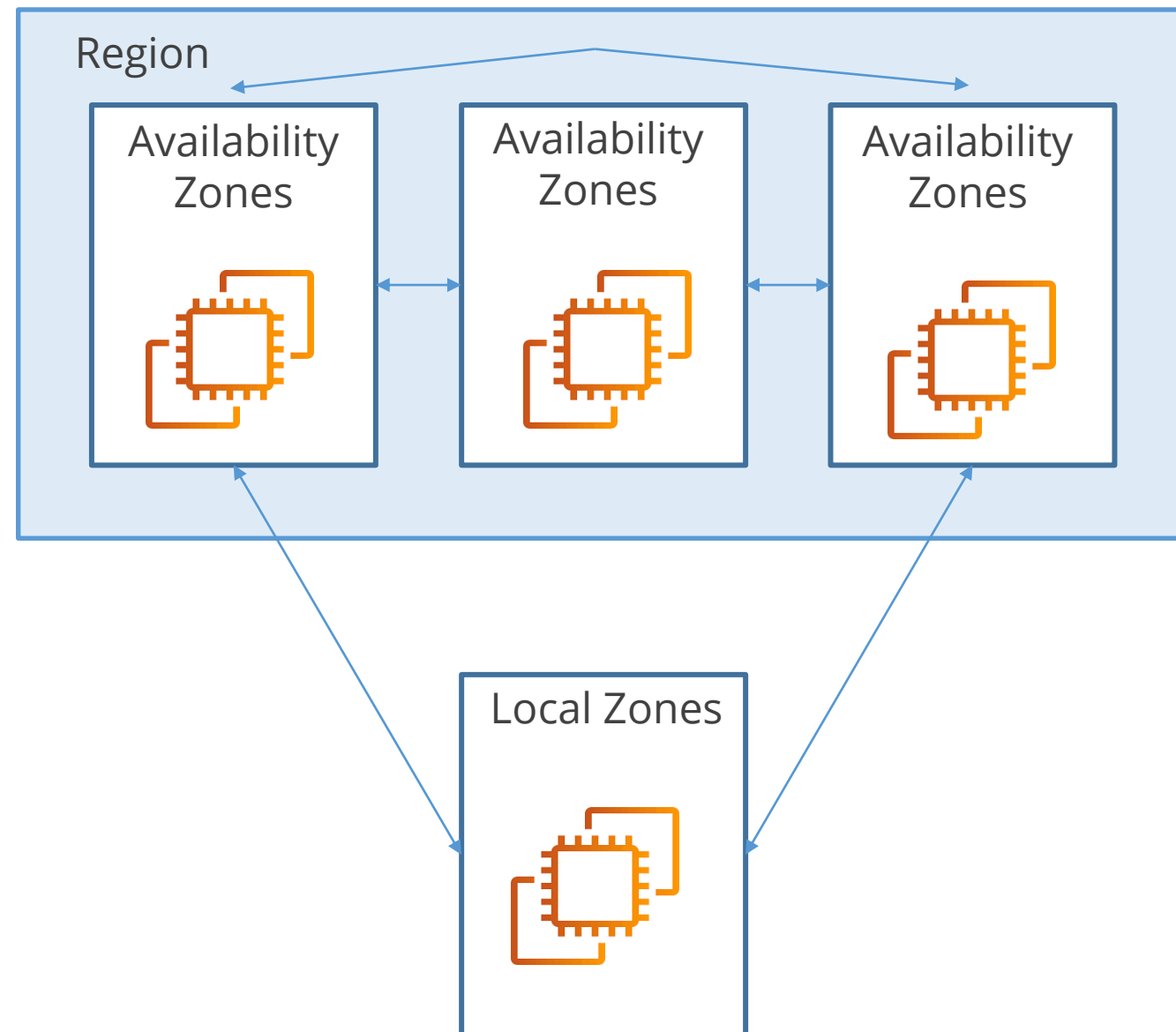


If the user is using an Elastic Load Balancing load balancer, Auto Scaling gracefully detaches the impaired instance from the load balancer before provisioning a new one and attaches it back to the load balancer.



# Balancing Capacity Across Availability Zones

Balancing resources across **Availability Zones** is the best practice for well-architected applications, as this greatly increases aggregate system availability.



Auto Scaling automatically balances EC2 instances across zones when you configure multiple zones in the Auto Scaling group settings.

# Manual and Dynamic Scaling



**Duration: 13 mins**

## **Problem Statement:**

You have been asked to perform manual and dynamic scaling.

ASSISTED PRACTICE



# Assisted Practice: Guidelines

---

Steps to be followed:

1. Create Auto Scaling Group
2. Create an EC2 Auto Scaling Group with Load Balancer



# Auto Scaling with Launch Templates



**Duration: 13 mins**

## **Problem Statement:**

You need to perform auto-scaling with launch templates.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

Steps to be followed:

1. Create a launch template
2. Create a launch configuration
3. Create an Auto Scaling group



## Key Takeaways

- Amazon EC2 provides a highly reliable environment in which replacement instances can be quickly and consistently deployed.
- The users can launch multiple instances from a single AMI when they require multiple instances with the same configuration.
- Auto Scaling performs EC2 health checks at regular intervals, and if the instance is connected to an Elastic Load Balancing load balancer, it can also perform ELB health checks.
- AWS Lambda executes the code only when needed and scales automatically, from a few requests per day to thousands per second.



# Vertical Scaling of EBS Volume for a Linux VM

Duration: 30 mins



**Project agenda:** To do vertical scaling of EBS volume for a Linux VM

## Description:

Your organization is experiencing business growth where solution deployment is happening with limited resources. In this case, the vertical scalability feature of AWS can be used to create a cost-optimized architecture.

## Perform the following:

1. Create an EC2 Instance
2. Identify the EBS volume that is created
3. Create a snapshot and create a new volume
4. Detach the existing volume and attach the new volume to the EC2 Instance



# TECHNOLOGY

**Thank You**