Introducing Cloud, EC2, S3 Bucket, and EBS Volume

Course(s):

1. Learn ELK stack 6.0

2. Cloud basics and web hosting

- Explain ELK stack

  - Analyzing log data
  - Visualising data on Kibana

- Launch EC2 instance and demonstrate cloudfront

  - Launch EC2 instance with both windows and linux
  - Demonstrate cloudfront by pulling content from S3

A Day in the Life of a MEAN Full Stack Developer

Joe has performed remarkably in the last sprint. Based on his expertise, the company has asked Joe to host the Kubernetes application on AWS cloud.

In this sprint, he has to deploy the Spring Boot application on AWS EC2 instance. He needs to then assign the required privileges and scale up the volume size of the instance.

In this lesson, we will learn how to solve this real-world scenario and help Joe effectively complete his task.

Learning Objectives

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

- Explain cloud computing

- Launch and connect to EC2 instance

- Explain placement groups and launch instances in a placement group

- Create an EBS volume and perform operations on it

- Create and delete an S3 bucket
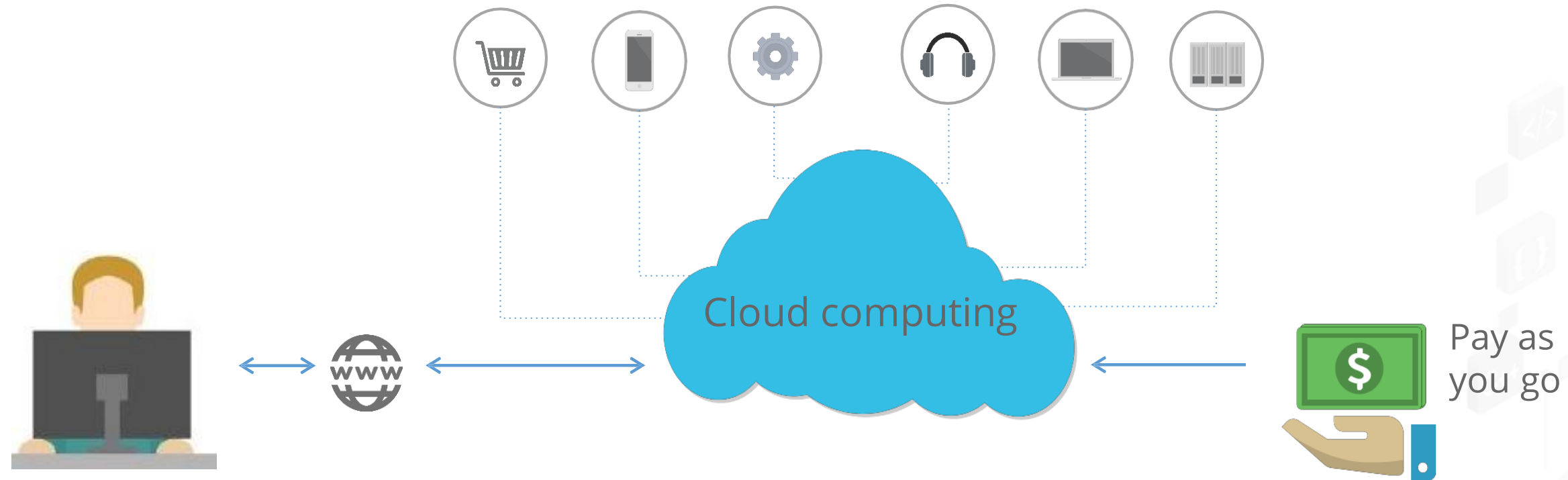
- Create an IAM user, role, and group

Cloud Introduction

Cloud Computing

Cloud computing refers to on-demand provisioning of IT resources and applications through the Internet.

Cloud computing

Pay as you go

Cloud computing facilitates:

- Quick access to cost-efficient and flexible IT resources
- Accessing servers, databases, storage media, and a variety of application services on the World Wide Web
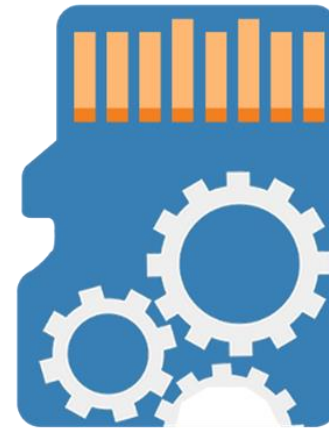
Cloud Computing

Cloud computing supports the following:

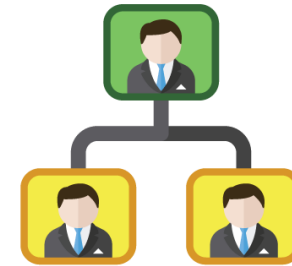|			
Servers	Databases	Storage media	Services on World Wide Web

Cloud Computing

Cloud computing providers such as Amazon Web Services possess and maintain the hardware required for different services.

With Cloud computing, you don't need:

Hardware investments

With Cloud computing, you only need to:

Determine type and size of resources

Empower the latest business project or idea

Identify files to be stored as backup on the Internet

Features of AWS

1 Cloud-based infrastructure helps business growth and scaling

2 Hybrid capabilities help you run hybrid applications across your existing on-premises and cloud services

3 Pay-as-you-go pricing allows you to pay only as long as you use the resources

4 You can build sophisticated applications with increased flexibility, scalability, and reliability

5 Greater security is provided through data encryption capabilities and exceptional hardware

6 AWS has a global presence in 190 countries through 18 geographic regions

simpl¦learn

Forms of Cloud Computing

There are three distinct forms of cloud computing.

Customer 1   Customer 2   Customer 3

Public Cloud

Machine 1   Machine 2   Machine 3

Private Cloud

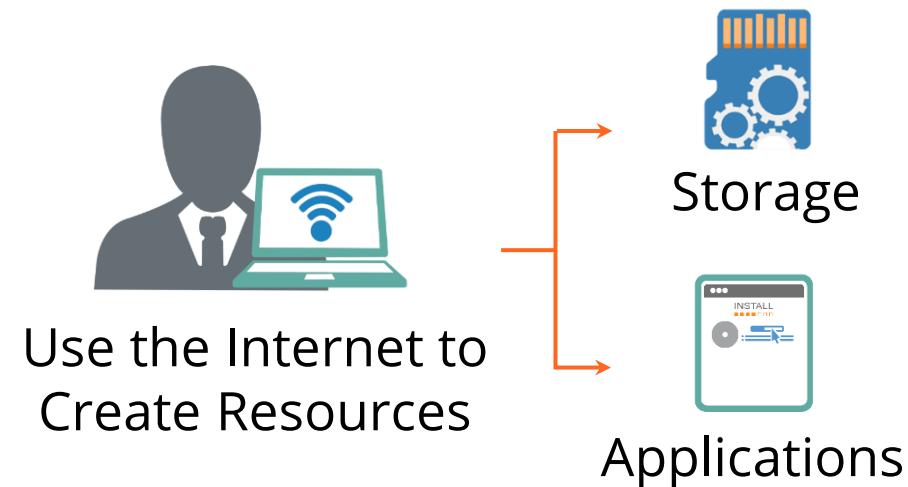Hybrid Cloud
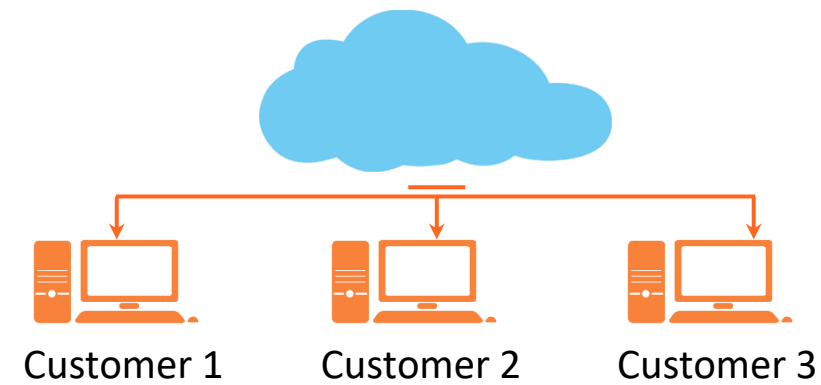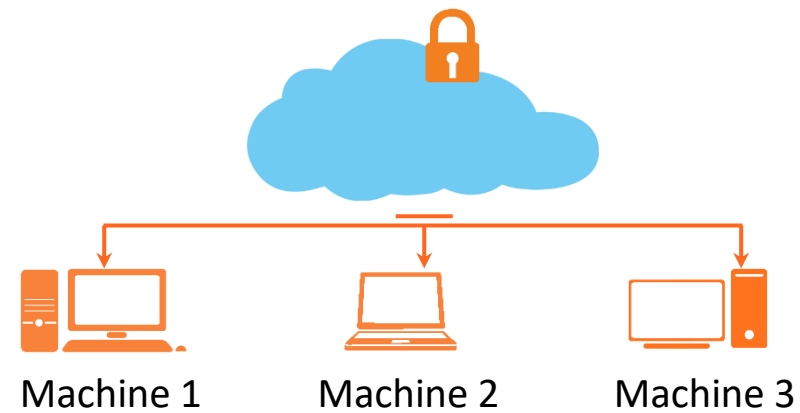
Public Cloud   Connect   Private Cloud
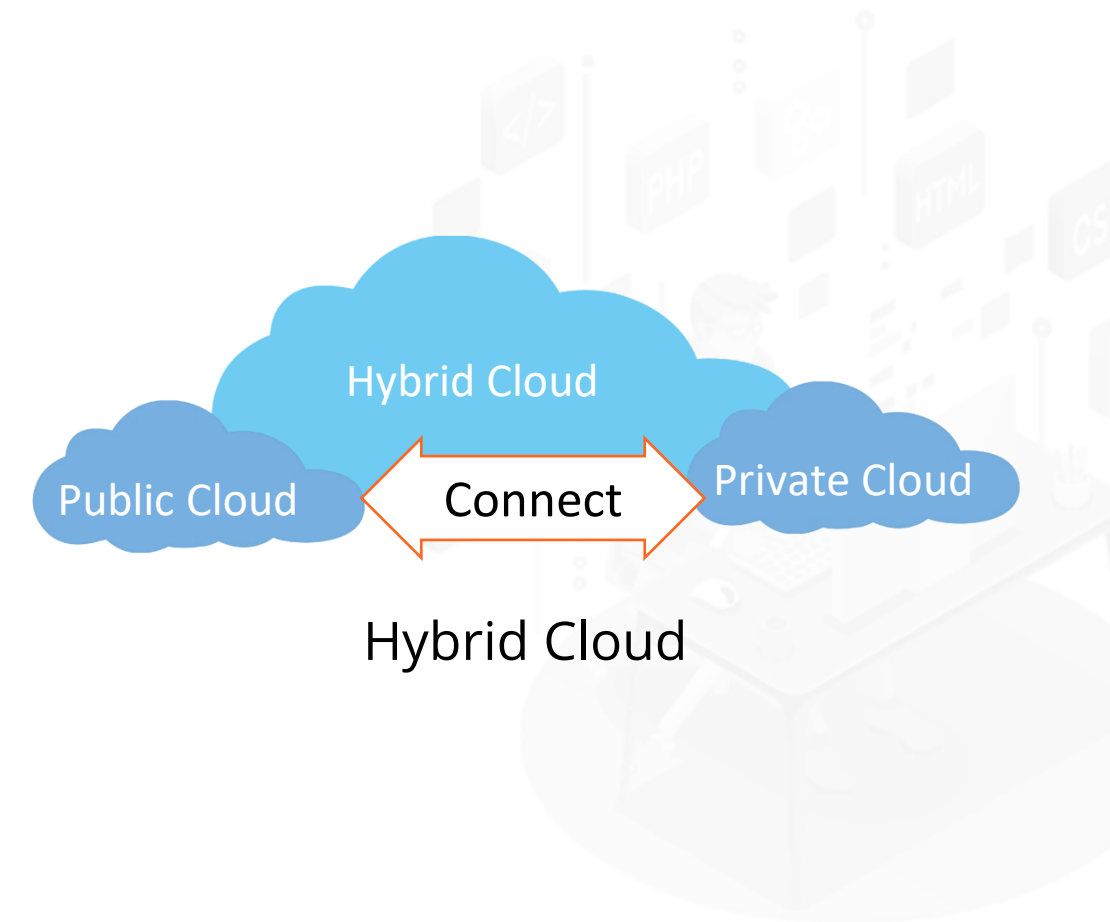
Hybrid Cloud

You need to choose the type of cloud you require depending on the type of data you need.

Public Cloud

Public cloud service providers use the Internet to create resources, such as storage and applications available to the public.

Use the Internet to Create Resources

Storage

Applications

A public cloud is ideal for:

INSTALL
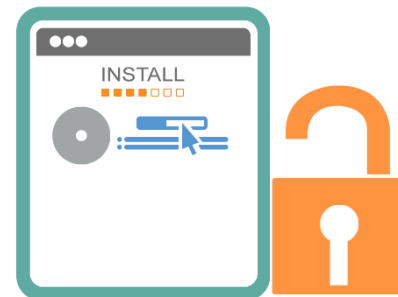
Sharing Non-Secured Applications

1010 1000 0111
0100 1010
0111 0100
1000 0100
1010 1010
0111 0100 0111
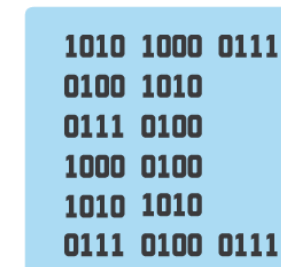
Testing Application Code
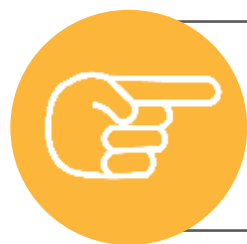
Examples of public cloud: Windows Azure Services Platform, Amazon Elastic Compute Cloud or EC2, Sun Cloud, and IBM's Blue Cloud

Private Cloud

A private cloud has an architecture similar to a data center, and a company owns it to ensure provisioning, monitoring, automation, and scalability.

Infrastructure   Services   Private Network
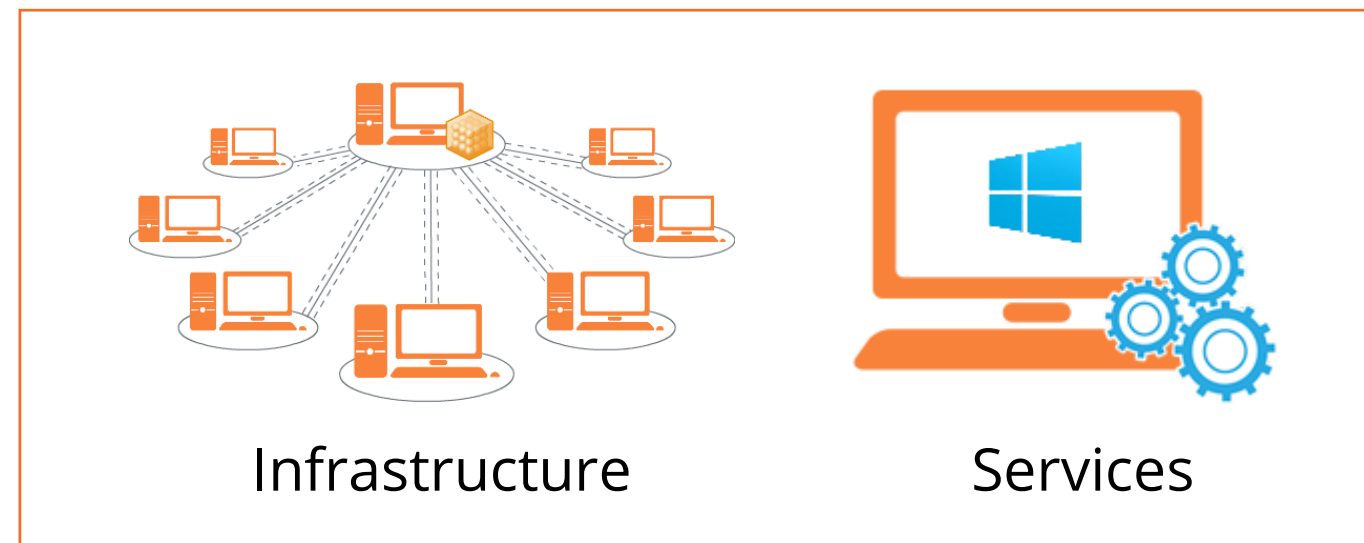
Features of Private Cloud

- Provides the highest level of security and control
- Offers modest economies of scale and is expensive
- Is used in large enterprises or projects that demand the best control and security

Hybrid Cloud

Hybrid cloud is a combination of public and private cloud services offered by multiple providers.

Through hybrid cloud, a company does the following:

Machine 1   Machine 2   Machine 3

Retains
control

Customer 1   Customer 2   Customer 3

Migrates data
to the public cloud

Tracks different
applications in each
cloud
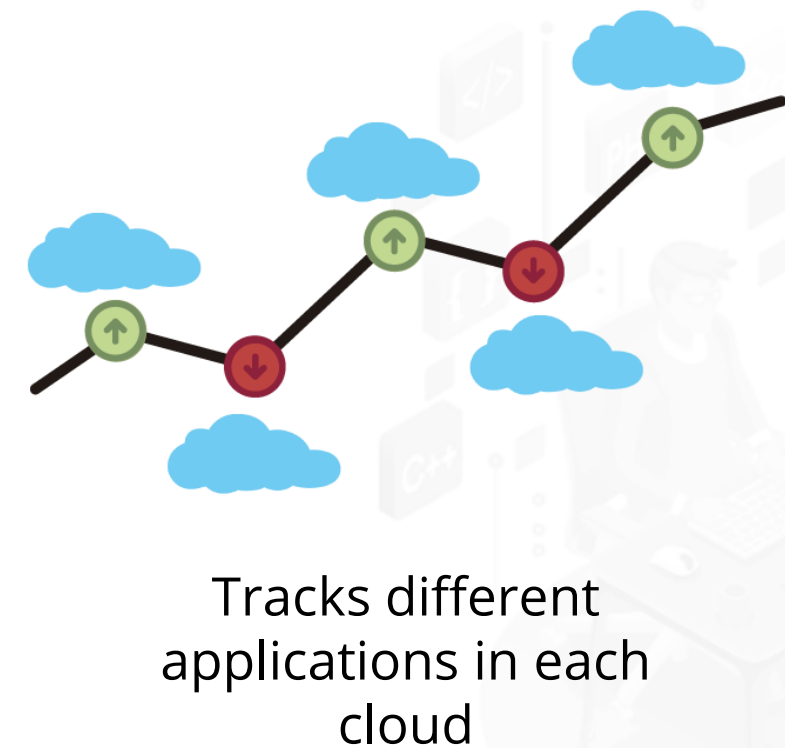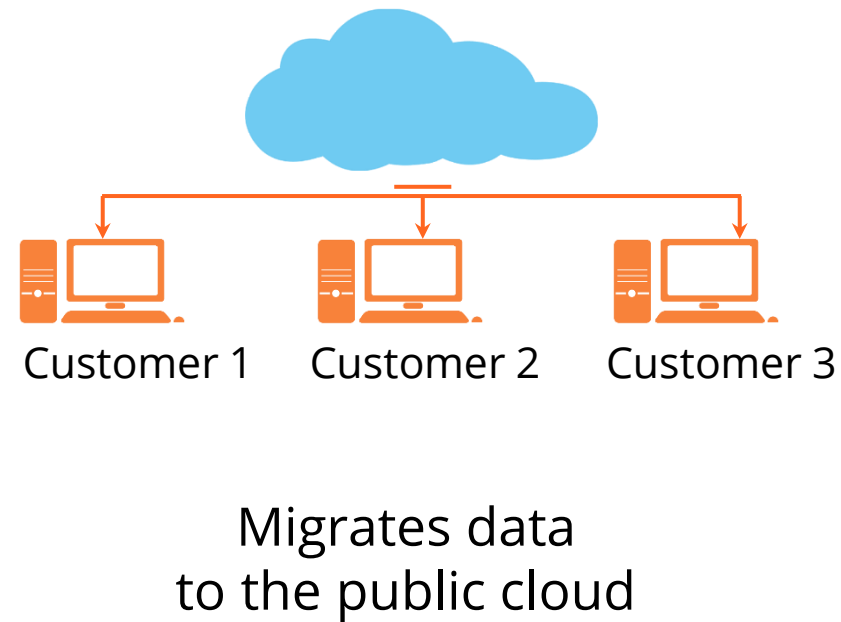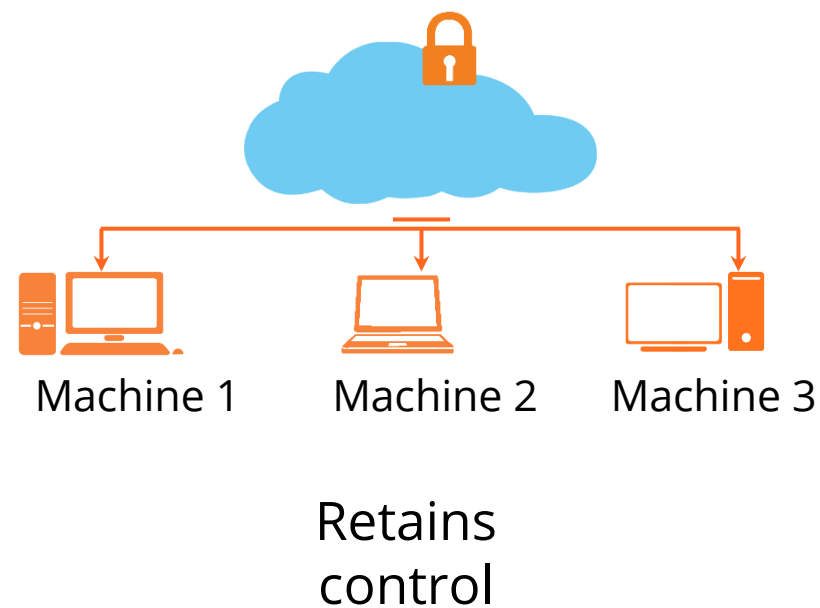
Hybrid cloud caters to different market verticals. While a public cloud allows you to interact with clients, a private cloud helps in securing their data.

Models of Cloud Computing

The three models of cloud computing are Infrastructure as a Service, Platform as a Service, and Software as a Service.

Each of these models comes with different levels of management, control, and flexibility.

Models of Cloud computing

Infrastructure as a Service

Platform as a Service

Software as a Service
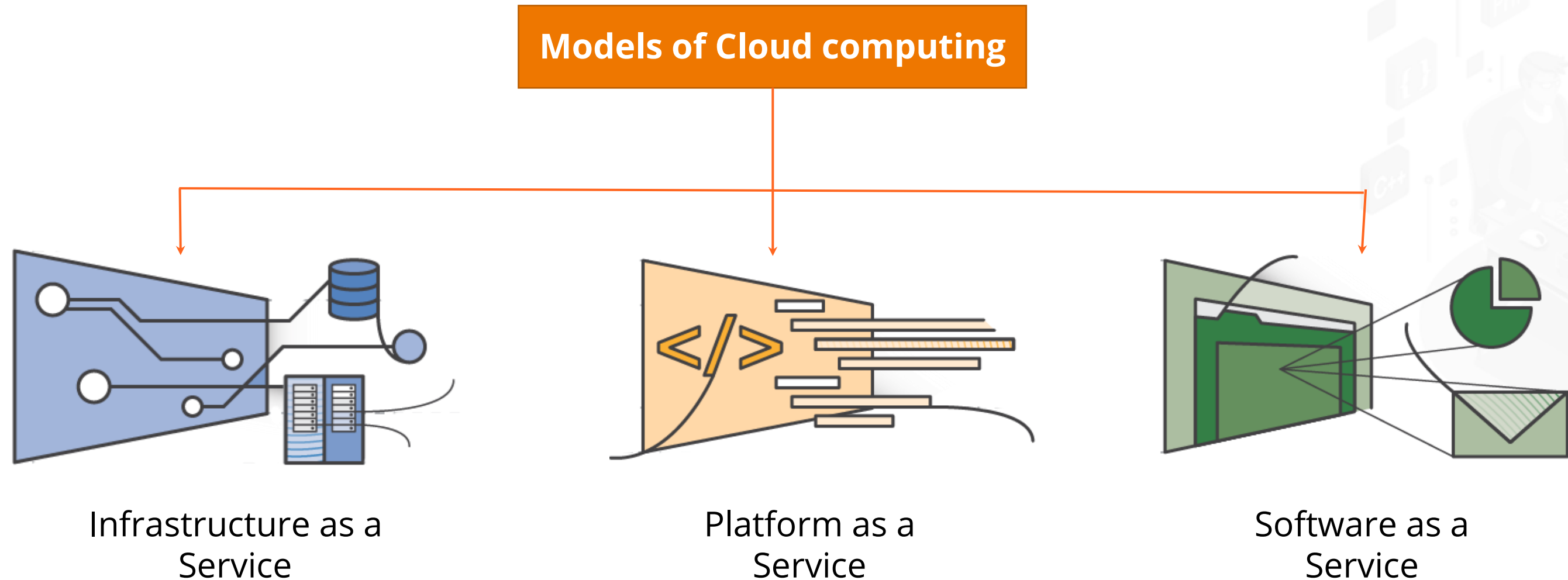
EC2 Introduction

Overview of EC2

- Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud

- Amazon EC2 has the following features:

  ➜ Virtual computing environments, known as *instances*

  ➜ Preconfigured templates for your instances, known as *Amazon Machine Images (AMIs)*

  ➜ Various configurations of CPU, memory, storage, and networking capacity for your instances, known as *instance types*

Overview of EC2

➜ Secure login information for your instances using *key pairs*

➜ Storage volume for temporary data that's deleted when you stop or terminate your instance, known as *instance store volumes*

➜ Persistent storage volumes for your data using Amazon Elastic Block Store (Amazon EBS), known as *Amazon EBS volumes*

➜ Multiple physical locations for your resources, such as instances and Amazon EBS volumes, known as *Regions* and *Availability Zones*

Overview of EC2

➔ A firewall that enables you to specify the protocols, ports, and source IP ranges that can reach your instances using *security groups*

➔ Static IPv4 addresses for dynamic cloud computing, known as *Elastic IP addresses*

➔ Metadata, known as *tags*, that you can create and assign to your Amazon EC2 resources

➔ Virtual networks you can create that are logically isolated from the rest of the AWS cloud, and that you can optionally connect to your own network, known as *virtual private clouds* (VPCs)

Benefits of EC2

Elastic	Flexible	Reliable	Inexpensive	Controlled	Secure

Instance Types of EC2

Instance Types

When you launch an instance, the instance type determines the hardware of the host computer used for your instance. Instances can be of the following types:

- Burstable performance instances (T3 and T2)

- General purpose instances (M5, M5a, M5d, T2, and T3)

- Compute optimized instances (C4, C5, Cn, and Cd)

- Memory optimized instances (R4, R5, R5a, and R5d)

- Storage optimized instances (D2, H1, and I3)

- Windows accelerated computing instances

- T1 Micro Instances

Instance Purchasing Options

Amazon EC2 provides the following purchasing options to enable you to optimize your costs based on your needs:

On-Demand Instances

Reserved Instances

Scheduled Instances

Dedicated Hosts

Dedicated Instances

Capacity Reservations

EC2 Pricing

Pricing

- Amazon EC2 is free to use

- There are three ways to pay for Amazon EC2 instances:
    - On-Demand
    - Reserved Instances
    - Spot Instances

- You can also pay for Dedicated Hosts which provide you with EC2 instance capacity on physical servers dedicated for your use

Duration: 15 min.

Problem Statement:

You are given a project to launch and connect to EC2 Linux instance.

Assisted Practice: Guidelines to Launch and Connect to EC2 Linux Instance

1. Login to your AWS lab.

2. Launch an EC2 Linux instance.

3. Select the AMI.

4. Attach the policy groups.

5. Connect to the instance.

Change the Volume Size of the Instance

Problem Statement:

You are given a project to change the volume size of an EC2 instance.

Assisted Practice: Guidelines to Change the Volume Size of the Instance

1. Login to your AWS lab.

3. Create the new volume which you need to add to an instance.

4. Validate the Availability zone of the EC2 instance with which you want to add volume.

5. Provide type, size, availability zone, and snapshot name on the **create volume** tab.

6. Select newly created volume and click on the action to attach the volume to the instance.

7. Select the instance of the same availability zone.

8. Verify the instance by adding one more device to the block device.

Placement Groups

Placement Groups

- When a new EC2 instance is launched, the EC2 service attempts to place it in such a way that all instances are spread across underlying hardware.

- *Placement groups* are used to influence the placement of a group of *interdependent* instances to meet the needs of your workload.

- There is no charge for creating a placement group.

Placement Groups

- You can create a placement group using one of the following placement strategies:

- **Cluster:** It packs instances close together inside an Availability Zone. This strategy enables workloads to achieve the low-latency network performance necessary for tightly-coupled node-to-node communication.

- **Partition:** It spreads your instances across logical partitions such that groups of instances in one partition do not share the underlying hardware with groups of instances in different partitions.

- **Spread:** It strictly places a small group of instances across distinct underlying hardware to reduce correlated failures.

Rules and Limitations

You need to take care of the following rules while using placement groups:

- The name you specify for a placement group must be unique within your AWS account for the region.

- You cannot merge placement groups.

- An instance can be launched in one placement group at a time; it cannot span multiple placement groups.

- On-demand capacity reservation and zonal reserved instances provide a capacity reservation for EC2 instances in a specific Availability Zone. The capacity reservation can be used by instances in a placement group. However, it is not possible to explicitly reserve capacity for a placement group.

- Instances with a tenancy of host cannot be launched in placement groups.

Launch Instances in a Placement Group

Problem Statement:

You are given a project to launch instances in a placement group.

Assisted Practice: Guidelines to Launch Instances in a Placement Group

1. Login to your AWS lab.

2. Create a placement group from the EC2 console.

3. Provide name and strategy as required.

4. Launch the new EC2 instance.

5. Select the placement group in which you want to launch your instance.

6. Verify the placement group of instance.

Security Groups

Security Groups

- A *security group* acts as a virtual firewall that controls the traffic for one or more instances.

- When you launch an instance, you can specify one or more security groups; otherwise, we use the default security group.

- You can add rules to each security group that allows traffic to or from its associated instances. You can modify the rules for a security group at any time; new rules are automatically applied to all instances associated with the security group.

- When you decide whether to allow traffic to reach an instance, you evaluate all the rules applied to security groups associated with the instance.

Security Group Rules

- By default, security groups allow all outbound traffic.

- Security group rules are always permissive; you can't create rules that deny access.

- Security groups are stateful. If you send a request from your instance, the response traffic for that request is allowed to flow in regardless of inbound security group rules.

- When you associate multiple security groups with an instance, the rules from each security group are effectively aggregated to create one set of rules.

Default and Custom Security Groups

Default Security Group:

- Your AWS account has a *default security group* for the default Virtual Private Cloud (VPC) in each region.
- It allows inbound traffic from all instances which are associated with it.
- It allows outbound traffic from all instances.

Custom Security Group:

- You can create your own security group and specify it when you launch your instances. You must provide a name and a description.
- It allows no inbound traffic.
- It allows all outbound traffic.
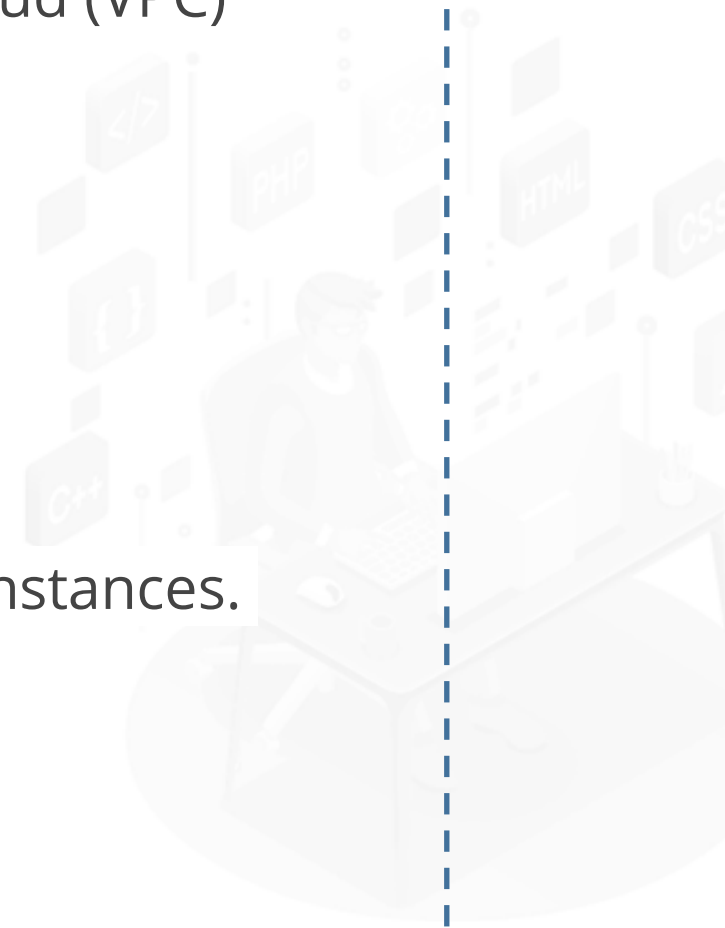
FULL STACK

Amazon Virtual Private Cloud (VPC)

simplilearn

VPC

- Amazon Virtual Private Cloud (Amazon VPC) enables you to launch AWS resources into a virtual network that you've defined.

- This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

- You can specify an IP address range for the VPC, add subnets, associate security groups, and configure route tables.

Amazon Elastic Block Store (EBS)

EBS

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

- EBS volumes behave like raw, unformatted block devices. You can mount these volumes as devices on your instances.

- EBS volumes are highly available and reliable storage volumes that can be attached to any running instance that is in the same Availability Zone.

EC2 instance store   **EBS**
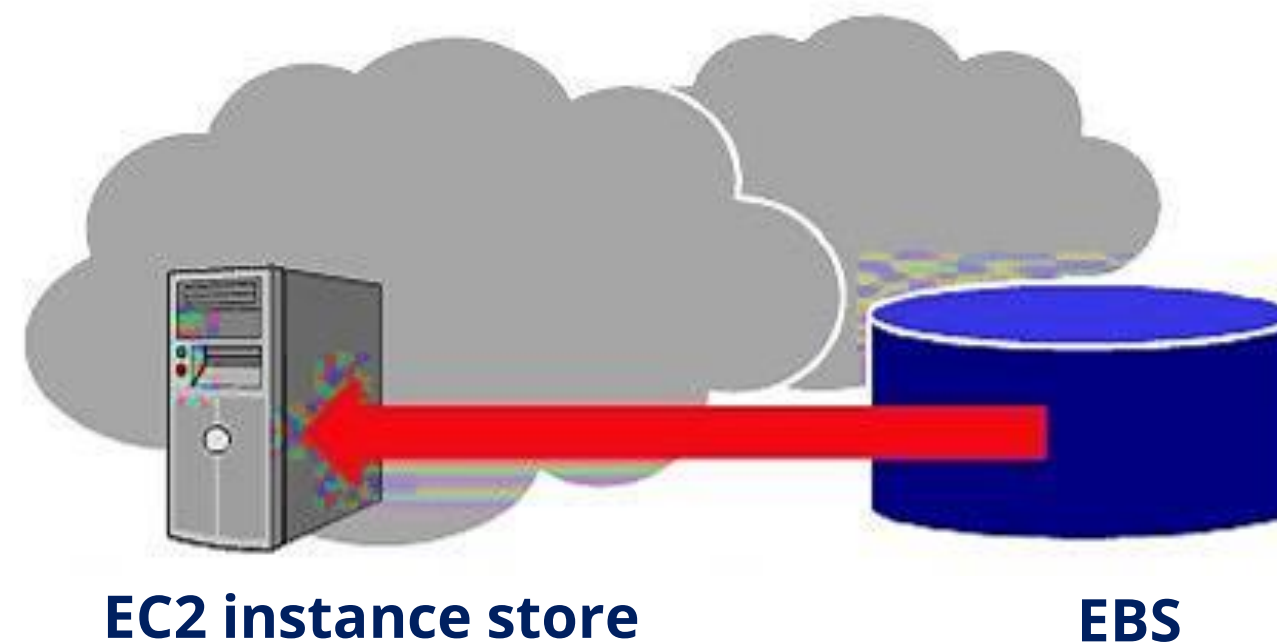
Features of EBS

- EBS volumes are created in a specific Availability Zone and can be attached to any instance in that same zone. To make a volume available outside the zone, you can create a snapshot and restore it to a new volume anywhere in that region.

- Amazon EBS provides the following volume types: General Purpose SSD (gp2), Provisioned IOPS SSD (io1), Throughput Optimized HDD (st1), and Cold HDD (sc1).

- You can create your EBS volumes as encrypted volumes, in order to meet a wide range of data-at-rest encryption requirements for regulated or audited data and applications.

Features of EBS

- You can create point-in-time snapshots of EBS volumes. Snapshots protect data for long-term durability, and they can be used as the starting point for new EBS volumes.

- Performance metrics, such as bandwidth, throughput, latency, and average queue length are available through the AWS Management Console.

- The metrics, provided by Amazon CloudWatch, allow you to monitor the performance of your volumes to make sure that you are providing enough performance for your applications without paying for resources you don't need.

Duration: 15 min.

Problem Statement:

You are given a project to create an EBS volume.

ASSISTED PRACTICE

Assisted Practice: Guidelines to Create an EBS Volume

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Select the **volume** tab in the EC2 dashboard.

4. Create a new volume by clicking on **Create.**

5. Select the volume type according to the project requirement.

6. Create a new volume by specifying the size and defining the zone.

7. Verify the new volume created on the volume console and its availability.

Attaching an Amazon EBS Volume to an Instance

Attaching an EBS Volume to an Instance

You can attach an available EBS volume to one of your instances. The instance has to be in the same Availability Zone as the volume.

While attaching an EBS volume, you need to take care of the following prerequisites:

- You need to determine the number of volumes you can attach to your instance

- An encrypted volume can only be attached to an instance that supports Amazon EBS encryption

Attaching an EBS Volume to an Instance

If a volume has an AWS Marketplace product code:

- It can only be attached to a stopped instance

- You must be subscribed to the AWS Marketplace code that is on the volume

- The configuration of the instance must support that specific AWS Marketplace code

- AWS Marketplace product code are copied from the volume to the instance

Format and Mount an EBS Volume

Problem Statement:

You are given a project to format and mount an EBS volume on Linux.

Assisted Practice: Guidelines to Mount an EBS Volume on Linux

1. Login to your AWS lab.

2. Attach an existing EBS volume.

3. Format an EBS volume on Linux.

4. Mount an EBS volume on Linux.

5. Unmount the EBS volume.

Detaching an EBS Volume

Problem Statement:

You are given a project to detach an EBS volume from an instance.

Assisted Practice: Guidelines to Detach an EBS Volume from an Instance

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Select the EBS volume you want to detach.

4. Detach the volume.

Duration: 10 min.

Problem Statement:

You are given a project to delete an EBS volume.

ASSISTED PRACTICE

Assisted Practice: Guidelines to Delete an EBS Volume

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Select the EBS volume you want to delete.

4. Delete the volume.

EBS Snapshots

EBS Snapshots

EBS snapshots are incremental backups; every snapshot only copies the blocks in the volume that change after the last snapshot. The only changed blocks are copied (in compressed form) to the S3 in subsequent snapshots.
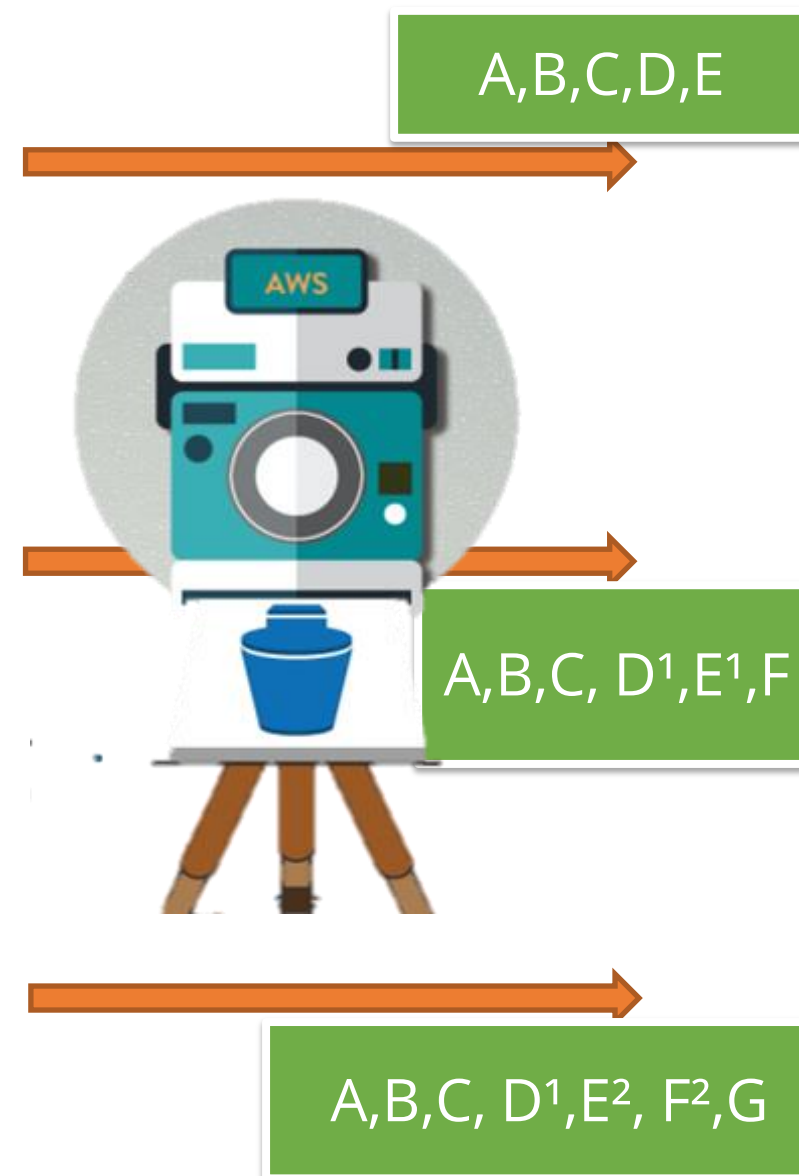
EBS Volume 1

A		B	C
	D		
		E	

EBS Volume 2

A		B	C
	D^1		
		E^1	F

EBS Volume 3

A		B	C
	D^1		
		E^2	F^2
			G

A,B,C,D,E

A,B,C, $D^1,E^1,$F

A,B,C, $D^1,E^2,$ $F^2,$G

EBS Snapshot 1

A		B	C
	D		
		E	

EBS Snapshot 2

A		B	C
	D^1		
		E^1	F

EBS Snapshot 3

A		B	C
	D^1		
		E^2	F^2
			G

EBS Snapshots

If you have a volume with 10 GB of data but only 2 GB of data has changed since your last snapshot, only that modified data is written to Amazon S3 during the snapshot process.

Example:

Step 1: When you take a snapshot of an EBS volume for the first time, it is a full snapshot. It only copies the blocks in the EBS volume that contains data. During the first snapshot, all blocks containing data (A, B, C, D, and E) are moved asynchronously to S3.

Step 2: Meanwhile, blocks D and E are changed and F is newly added from snapshot 1. When you take snapshot 2, only blocks D1, E1, and F are moved to S3.

Step 3: When you take snapshot 3, blocks E and F are changed and G is newly added. Only blocks E2, F1, and G are moved to S3.

Step 4: Since snapshot 3 contains the latest data, you can go ahead and delete older snapshots. Data of blocks D, E, F, and E1 is no more relevant. The blocks are released and not charged by AWS.

Creating Amazon EBS Snapshots

Creating EBS Snapshots

- A snapshot of an EBS volume can be created and used as a baseline for new volumes or for data backup.

- Snapshots are incremental when periodic snapshots of a volume are made.

- Snapshots occur asynchronously.

- An in-progress snapshot is not affected by ongoing reads and writes to the EBS volume.

Creating EBS Snapshots

- Snapshots only capture data that has been written to the Amazon EBS volume when the snapshot command is issued.

- Your snapshot will be complete if you can pause any file writes to the volume long enough to take a snapshot. However, if you can't pause all file writes to the volume, you should unmount the volume from within the instance, issue the snapshot command, and then remount the volume.

Create an EBS Snapshot

Problem Statement:

You are given a project to create a snapshot.

Assisted Practice: Guidelines to Create an EBS Snapshot

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Create an EBS snapshot.

Duration: 10 min.

Problem Statement:

You are given a project to view a snapshot.

Assisted Practice: Guidelines to View an EBS Snapshot

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Go to your Amazon EC2 console.

4. Choose **Snapshots** in the navigation pane.

5. To reduce the list, choose an option from the **Filter** list.

Initialize a Volume restored from a snapshot on Linux

Problem Statement:

You are given a project to initialize a volume restored from a snapshot.

ASSISTED PRACTICE

Assisted Practice: Guidelines to Initialize a Volume Restored from a Snapshot

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Create an EBS volume from a snapshot.

4. Attach the volume to an instance.
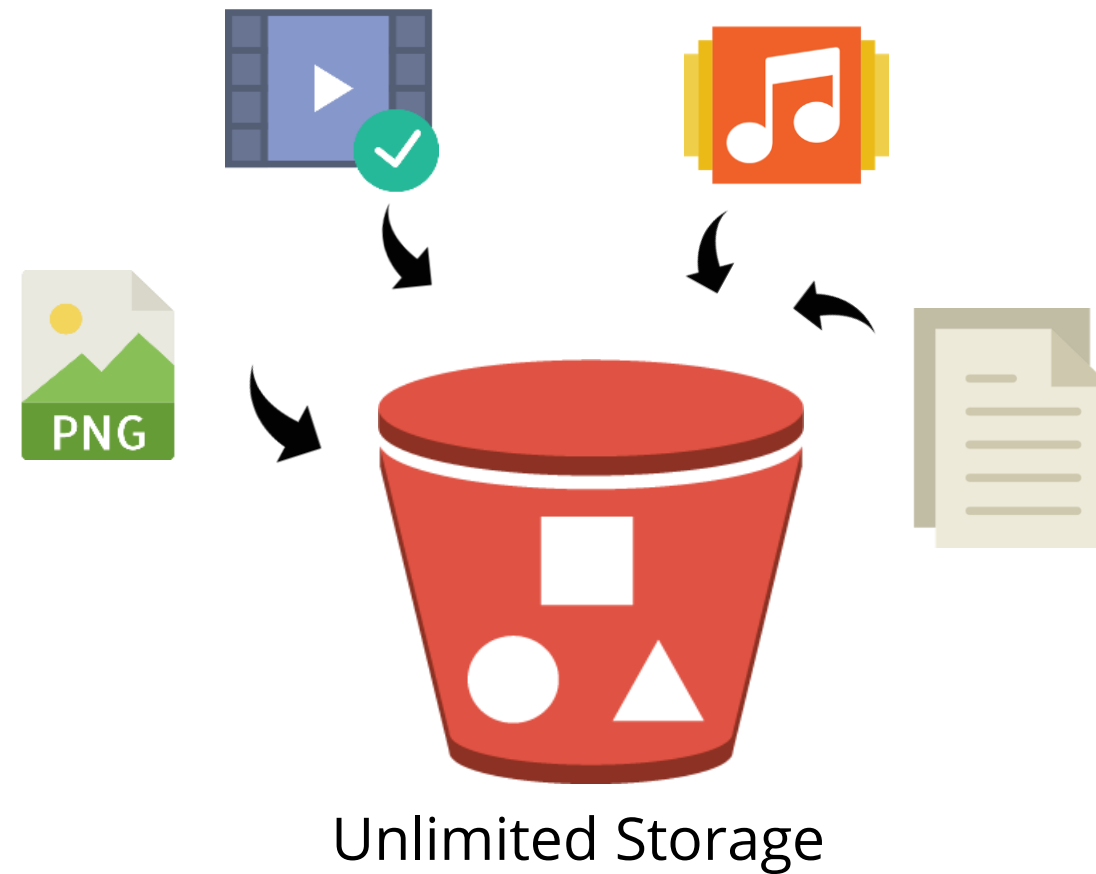
5. Initialize the volume.

Amazon Simple Storage Service (S3) Introduction

Simple Storage Service (S3)

Amazon S3 provides object storage and is built for storing and recovering data from anywhere over the Internet.

Unlimited Storage

Purpose of S3

Factors that make a repository expensive and time-consuming are:

- Need to purchase hardware and software components

- Need to hire a team of experts for maintenance

- Lack of scalability based on your requirements

- Requirement for data security

Benefits of S3

- Durability

- Low cost

- Scalability

- Availability

- Security

- Flexibility

- Simple data transfer

Amazon S3 Bucket

S3 Bucket

- You need to create an S3 bucket in one of the AWS regions in order to upload your data.

- When data is added to the bucket, Amazon S3 creates a unique version ID and allocates it to the object.

- The name of an S3 bucket cannot be used by another AWS account in any AWS region until the bucket is deleted.

- You should choose a location that is close to you in order to optimize latency, minimize costs, and address regulatory requirements.

Create a Bucket

Problem Statement:

You are given a project to create an S3 bucket.

Assisted Practice: Guidelines to Create an S3 Bucket

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Select S3 from the storage of the AWS service panel.

4. Verify the button to create a bucket.

5. Provide a bucket name and region.

6. Enter key and value for identification and tracking of bucket.

7. Block all the public access.

8. Review all parameters and create the bucket.

9. Verify the newly created bucket on the panel of S3.

Bucket Restrictions and Limitations

Bucket: Restrictions and Limitations

- You cannot transfer bucket ownership.

- You should not delete a bucket if you want to use the same name.

- You cannot change the region after you have created the bucket.

- You cannot create a bucket within another bucket.

- You can create 100 buckets in your AWS account by default.

Rules for Bucket Naming

Follow the listed bucket-naming conventions to avoid errors:

- Bucket names must be between 3 and 63 characters long.

- Bucket names must be a series of one or more labels.

- AWS recommends separating labels with a single period (.).

- Bucket names can contain lowercase letters, numbers, and hyphens.

- Each label must start and end with a lowercase letter or a number.

Deleting a Bucket

Deleting a Bucket

- You can delete a bucket that may or may not contain objects. All objects and their versions are permanently deleted when a bucket is deleted.

- Another AWS user can use the name after you delete a bucket.

- You need to clean up the Route 53 hosted zone settings if you have created and configured an Amazon Route 53 hosted zone.

- You need to stop the delivery of Elastic Load Balancing (ELB) logs before deleting a bucket.

Duration: 5 min.

Problem Statement:

You are given a project to delete an S3 bucket.

Assisted Practice: Guidelines to Delete an S3 Bucket

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Select S3 bucket you want to delete.

4. Delete the bucket.

Empty an S3 Bucket

Problem Statement:

You are given a project to empty an S3 bucket.

Assisted Practice: Guidelines to Empty an S3 Bucket

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Select S3 bucket you want to empty.

4. Empty the bucket.

S3 Objects

S3 Object

- Amazon S3 contains multiple objects with keys and values. An object consists of the following:
  -> **Key**: It is the name assigned to an object
  -> **Version ID**: It is a string generated by Amazon S3 when an object is added to a bucket
  -> **Value**: It is the content that you store in an object
  -> **Metadata**: It is a set of name-value pairs with which you can store information regarding the object
  -> **Subresource**: It is a mechanism used by Amazon S3 to store object-specific additional information
  -> **Access Control Information**: Amazon S3 supports both resource-based access control and user-based access control

Object Key and Metadata

S3 Object Key

- You specify a key name when you create an object.

- It uniquely identifies an object in a bucket.

- The name for a key is a sequence of Unicode characters . Its UTF-8 encoding is maximum 1024 bytes.

- You cannot download an object using the Amazon S3 console if an object key name consists of a single period (.) or two periods (..).

S3 Object Metadata

- There are two kinds of object metadata: system metadata and user-defined metadata.

- Amazon S3 maintains a set of system data for every object in a bucket.

- There are two kinds of system metadata. They can be metadata such as object creation date and other system metadata such as the storage class configured for the object.

- Metadata can be assigned to an object while uploading it. This information can be added as a name-value pair.

Storage Classes

Storage Classes

- There is a storage class associated with every object in Amazon S3.

- A storage class is chosen depending on the user case scenario and performance access requirements.

- Storage classes offer high durability.

S3 Storage Tiers

Amazon S3 is highly available, as it synchronizes the data across multiple Availability Zones. Data stored in S3 is spread across multiple devices and facilities to prevent data loss. S3 offers six different storage tiers.

STANDARD

STANDARD_IA

INTELLIGENT_TIERING

ONEZONE_IA

GLACIER

DEEP_ARCHIVE

STANDARD Storage

- It is the default storage class.
- It is designed for frequently accessed data.
- S3 Standard Storage provides 99.99% availability and 99.999999999% or "11 nines" durability of data.

S3 Storage Tiers

STANDARD

STANDARD_IA

INTELLIGENT_TIERING

ONEZONE_IA

GLACIER

DEEP_ARCHIVE

STANDARD_IA Storage

- It is designed for long-lived and infrequently accessed data.
- It provides 99.9% availability and 99.999999999% or "11 nines" durability of data.
- Amazon S3 stores the object data redundantly across multiple geographically separated Availability Zones (similar to the STANDARD storage class).
- STANDARD_IA objects are resilient to the loss of an Availability Zone.
- Minimum billable object size is 128 KB.

S3 Storage Tiers

STANDARD

STANDARD_IA

INTELLIGENT_TIERING

ONEZONE_IA

GLACIER

DEEP_ARCHIVE

INTELLIGENT_TIERING Storage

- It is designed for long-lived data with changing or unknown access patterns.
- It provides 99.9% availability and 99.999999999% or "11 nines" durability of data.
- It is designed to optimize storage costs by automatically moving data to the most cost-effective storage access tier, without performance impact or operational overhead.
- It stores objects in two access tiers: one tier that is optimized for frequent access and another lower-cost tier that is optimized for infrequently accessed data.

S3 Storage Tiers

STANDARD

STANDARD_IA

INTELLIGENT_TIERING

ONEZONE_IA

GLACIER

DEEP_ARCHIVE

ONEZONE_IA Storage
- It is designed for long-lived, infrequently accessed, and noncritical data.
- It provides 99.5% availability and 99.999999999% or "11 nines" durability of data.
- Minimum billable object size is 128 KB.
- Amazon S3 stores the object data in only one Availability Zone, which makes it less expensive than STANDARD_IA. However, the data is not resilient to the physical loss of the Availability Zone resulting from disasters, such as earthquakes and floods.
- The ONEZONE_IA storage class is as durable as STANDARD_IA, but it is less available and less resilient.

S3 Storage Tiers

STANDARD

STANDARD_IA

INTELLIGENT_TIERING

ONEZONE_IA

GLACIER

DEEP_ARCHIVE

GLACIER Storage
- It is a secure, durable, and an extremely low-cost storage service.
- Retrieval of data can take up to 3-5 hours.
- It is ideal for long-term archive and offsite backup solutions.
- It is used for archives where portions of the data might need to be retrieved in minutes.

simpl¦learn

S3 Storage Tiers

STANDARD

STANDARD_IA

INTELLIGENT_TIERING

ONEZONE_IA

GLACIER

DEEP_ARCHIVE

DEEP_ARCHIVE Storage
- It is designed for archiving rarely accessed data with a default retrieval time of 12 hours.
- If you have deleted, overwritten, or transitioned an object to a different storage class before the 180-day minimum, you are charged for 180 days.
- DEEP_ARCHIVE is the lowest-cost storage option in AWS. Storage costs for DEEP_ARCHIVE are less expensive than the GLACIER storage class. You can reduce DEEP_ARCHIVE retrieval costs by using bulk retrieval, which returns data within 48 hours.

simplilearn

Set the Storage Class of an object

Problem Statement:

You are given a project to set the storage class of an object.

Assisted Practice: Guidelines to Set the Storage Class of an Object

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Select the storage class of an object.

Operations on Objects

Operations on Objects

- You can store, delete, and retrieve objects in Amazon.

- You can retrieve a version of an object if versioning is enabled.

- You can upload objects up to 5 GB in size in one operation. You need to use multipart upload API if the size is greater than 5 GB.

- You can copy objects up to 5 GB in size in one operation. You need to use multipart upload API if the size is greater than 5 GB.

- You can retrieve a subresource associated with your object and update it if required.
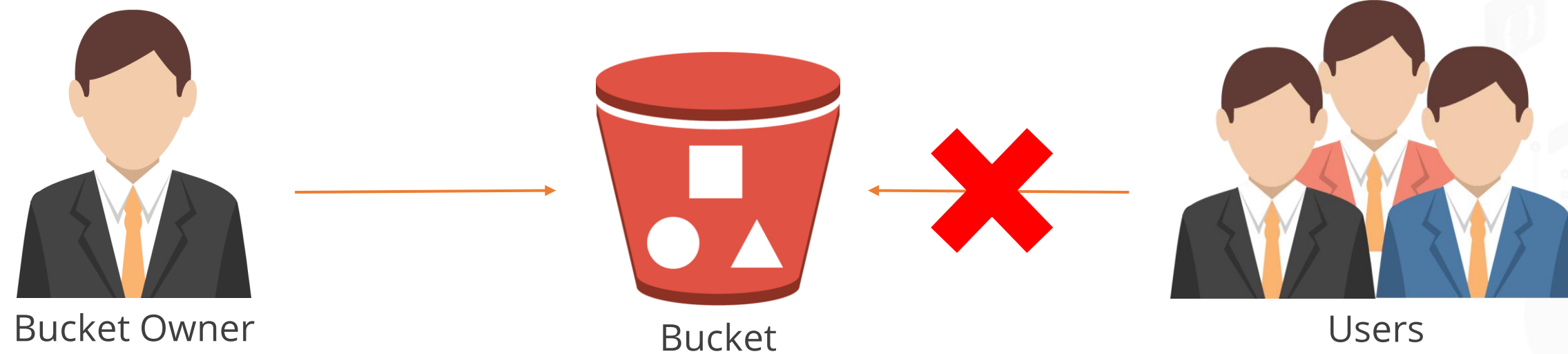
S3 Security

S3 Security

All data stored in Amazon S3 is secure by default as only bucket and object owners have access to the Amazon S3 resources they create.

Bucket Owner

Bucket

Users

S3 Security

- Security is a shared responsibility model between AWS and the customer which can be described as *security of the cloud* and *security in the cloud*.

- **Security of the cloud**: AWS protects the infrastructure that runs all services in AWS cloud. AWS has compliance programs that assign third-party auditors to regularly test and verify the security.

- **Security in the cloud**: Users are responsible for data sensitivity, organizational requirements, and applicable laws and regulations.

Data Protection

- The storage infrastructure provided by S3 is durable and is designed for mission-critical and primary data storage.

- Data is stored across multiple facilities by S3 PUT and PUT object copy operations to ensure data durability.

- S3 also detects and repairs any lost redundancy.

- S3 offers the following features:
  - It is backed with *Amazon S3 Service Level Agreement*
  - It is designed to provide 99.999999999% durability and 99.99% availability of objects over a given year
  - It is designed to sustain the concurrent loss of data in two facilities

Identity and Access Management

- Amazon resources, such as buckets, objects, and related subresources are accessible only by the resource owner.

- The owner can provide access permissions to others by writing an access policy.

- Access policy options can be resource-based policies and user policies.

- Resource-based policies are the policies attached to resources like bucket policies.

- Policies attached to users in your account are user policies.

Logging and Monitoring

AWS provides the following monitoring tools to monitor your S3 resources and prevent potential threats:

- Amazon CloudWatch Alarms

- AWS CloudTrail Logs

- Amazon S3 Access Logs

- AWS Trusted Advisor

Identity and Access Management (IAM)

Importance of IAM

IAM

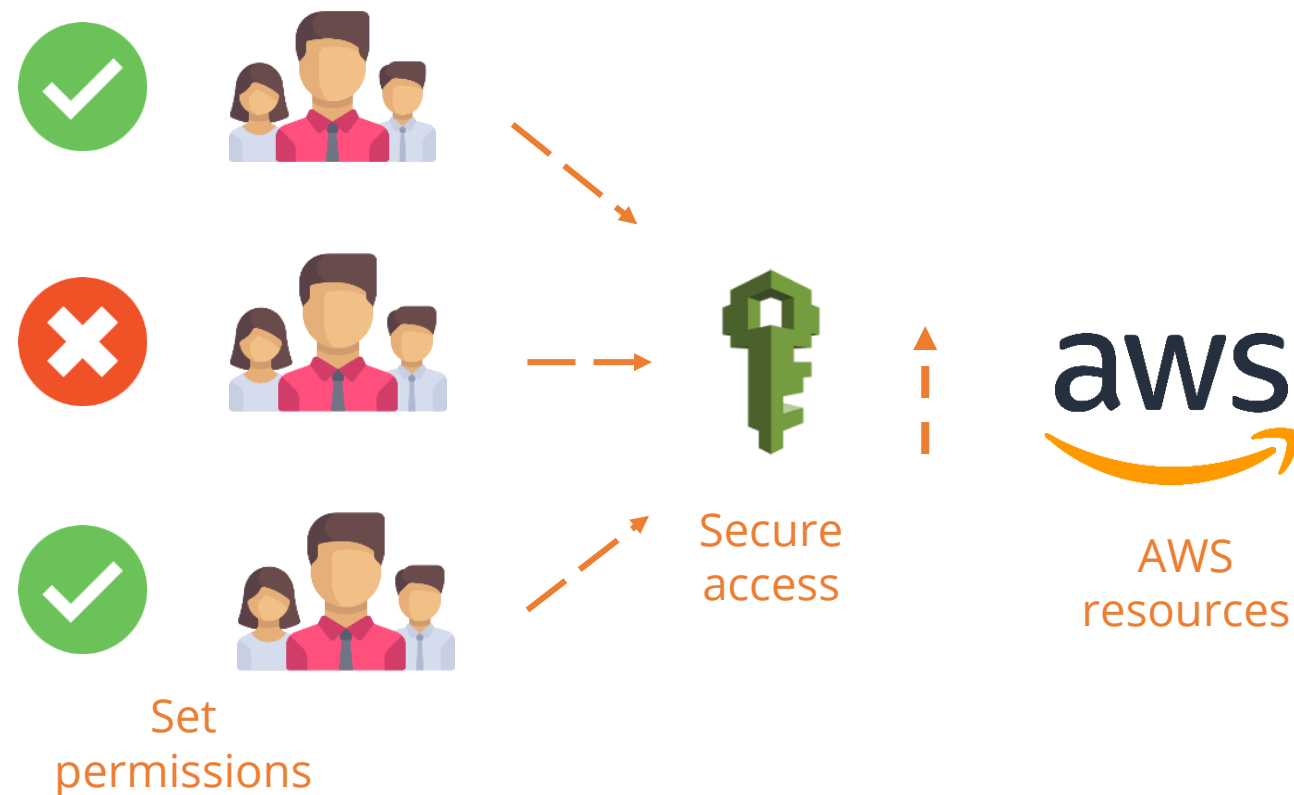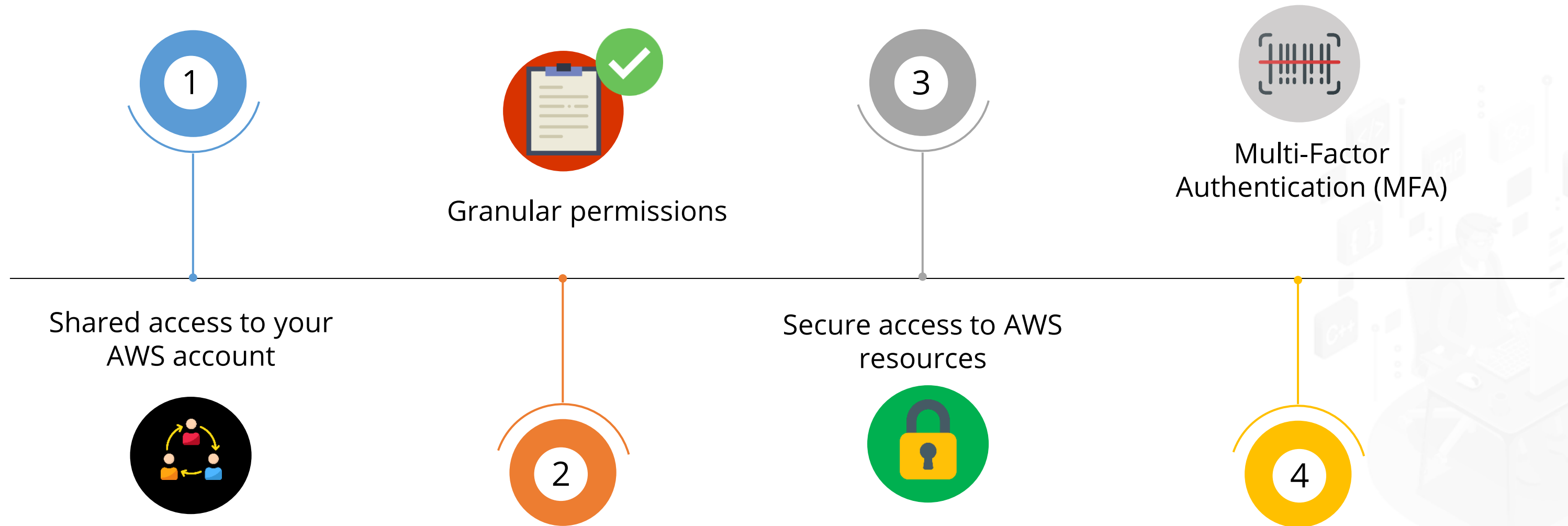- AWS Identity and Access Management is a web service for securely controlling access to AWS services.
- It enables you to create and control services for user authentication or limit access to a certain set of users on your AWS resources.
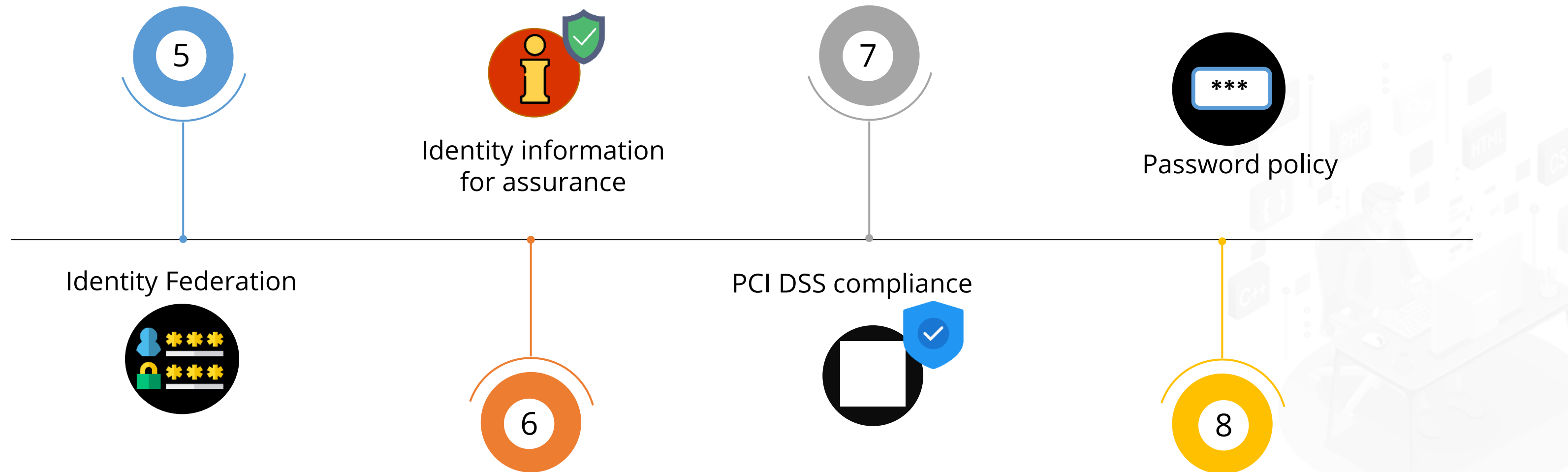

Set permissions

Secure access

AWS resources

IAM

Features of IAM

1 Shared access to your AWS account

2 Granular permissions

3 Secure access to AWS resources

4 Multi-Factor Authentication (MFA)

Features of IAM

5

Identity information for assurance

7

Password policy

Identity Federation

6

PCI DSS compliance

8

Best Practices of IAM

Use IAM users instead of your root account.

Enable AWS CloudTrail to get logs of API calls.

Manage permissions with groups.

Assign the least privileged.

Configure a strong password policy.

Rotate security credentials regularly.

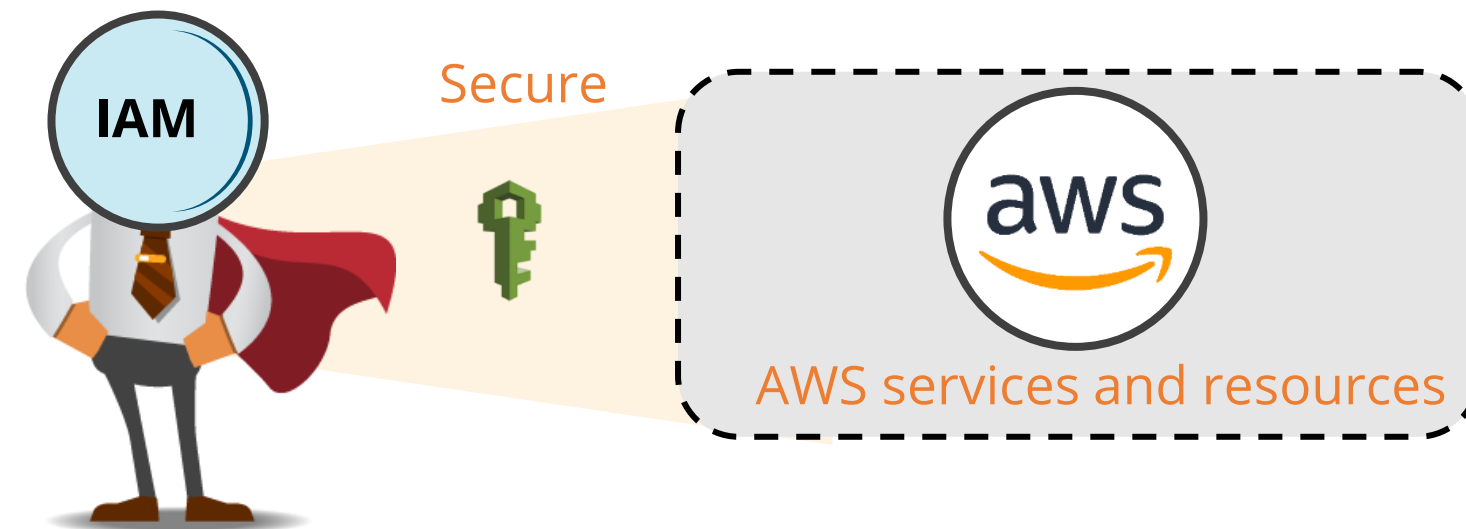Enable Multi-Factor Authentication (MFA) for privileged users.

Components of IAM

Users   Groups   Policies   Roles
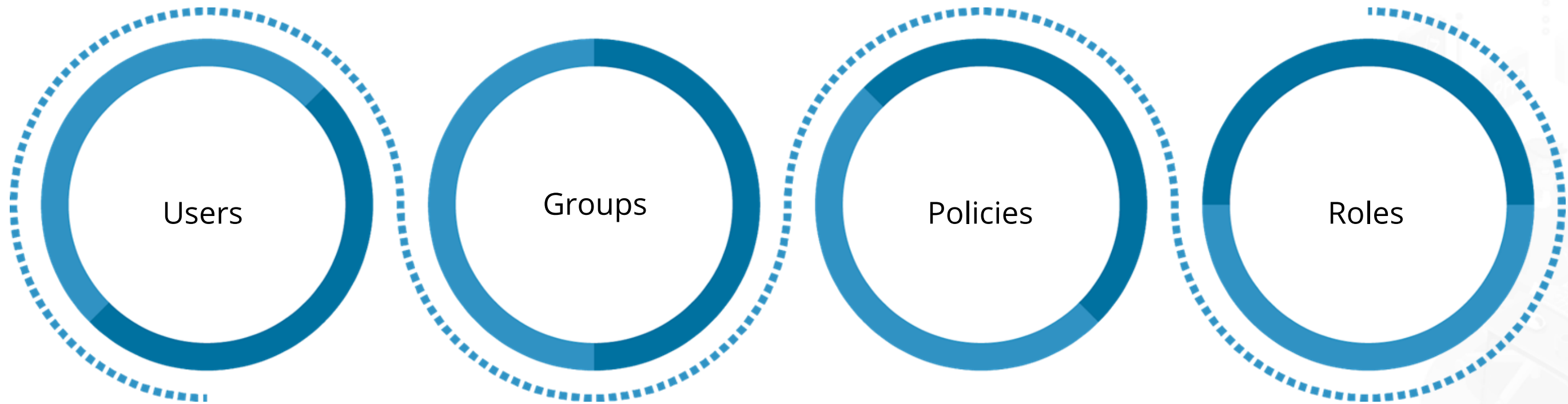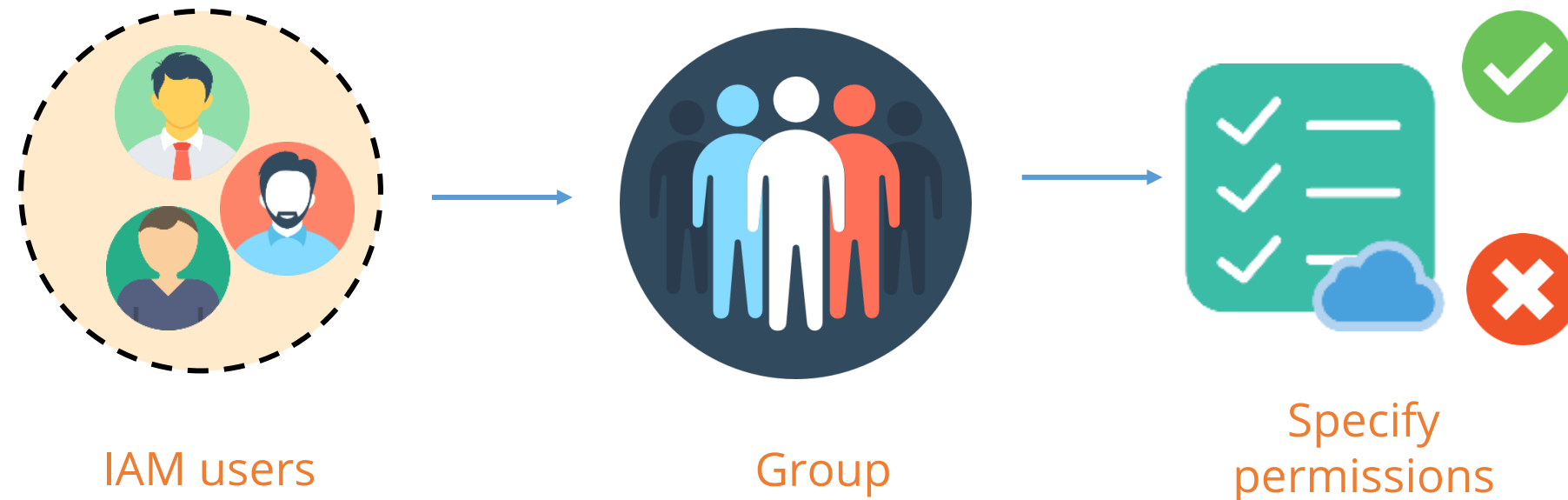
IAM Users

- With IAM, a user can securely manage access to AWS services.
- It helps you to manage the access on AWS types (Programmatic access and management console access) and AWS services.
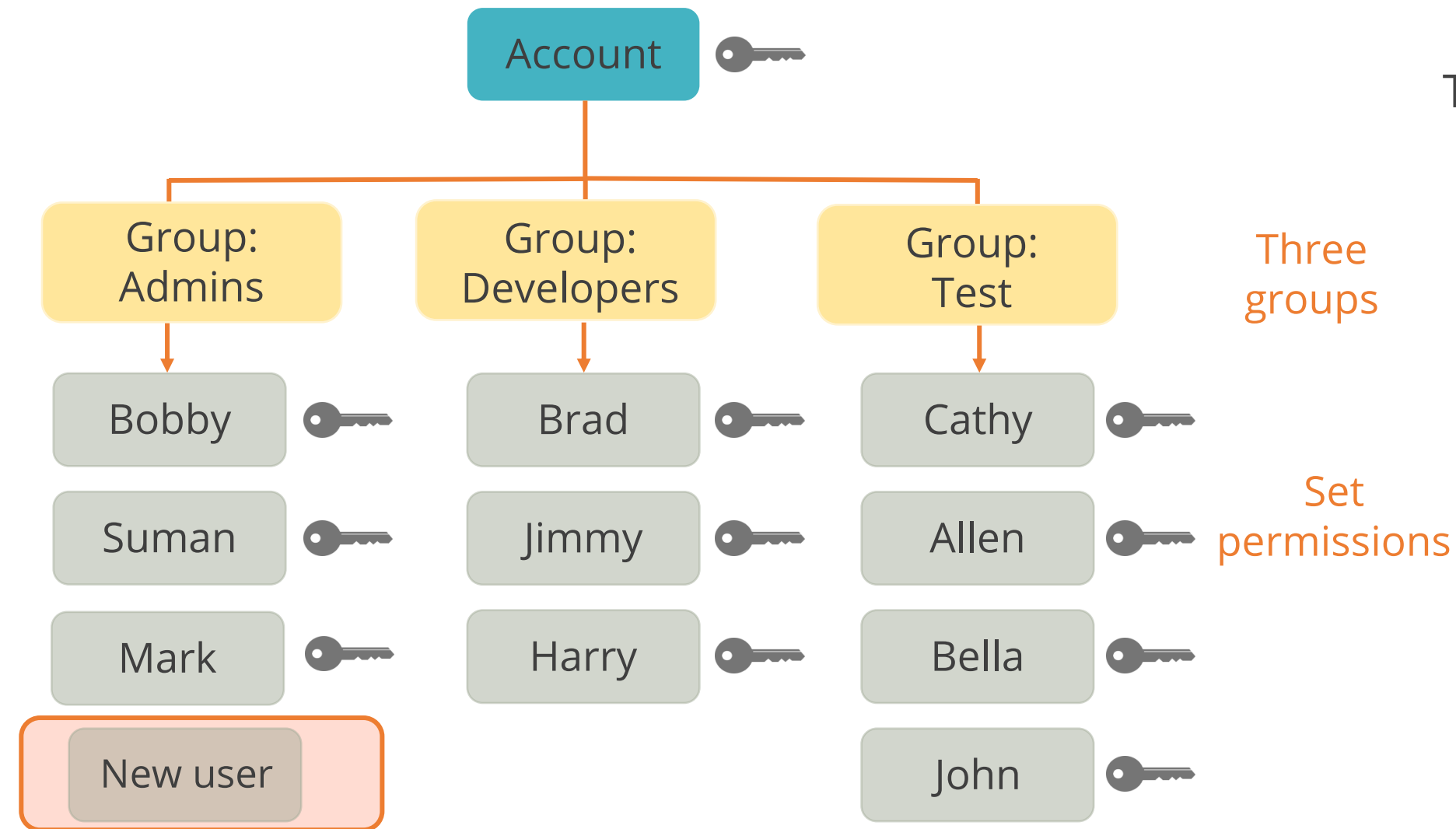
IAM

Secure

aws

AWS services and resources

Note: Each IAM user is associated with only one AWS account.

simplilearn

IAM Groups

- An IAM group is a collection of IAM users.
- You can use IAM groups to specify permissions for multiple users so that any permission applied to the group is applied to its users as well.

IAM users   Group   Specify permissions

IAM Groups

Account 🔑

Group: Admins
Group: Developers
Group: Test

Three groups

Group: Admins	Group: Developers	Group: Test
Bobby 🔑	Brad 🔑	Cathy 🔑
Suman 🔑	Jimmy 🔑	Allen 🔑
Mark 🔑	Harry 🔑	Bella 🔑
New user		John 🔑

Set permissions

This diagram is an example of groups created for a small company.

IAM

⚠️ Note: If a new user joins your organization and needs administrator privileges, you can assign the appropriate permissions by adding the user to that group.

simplilearn

IAM Policies

- IAM policies are JSON-formatted documents.

- They contain statements (permissions) which specify:

    -> What actions a principle can perform
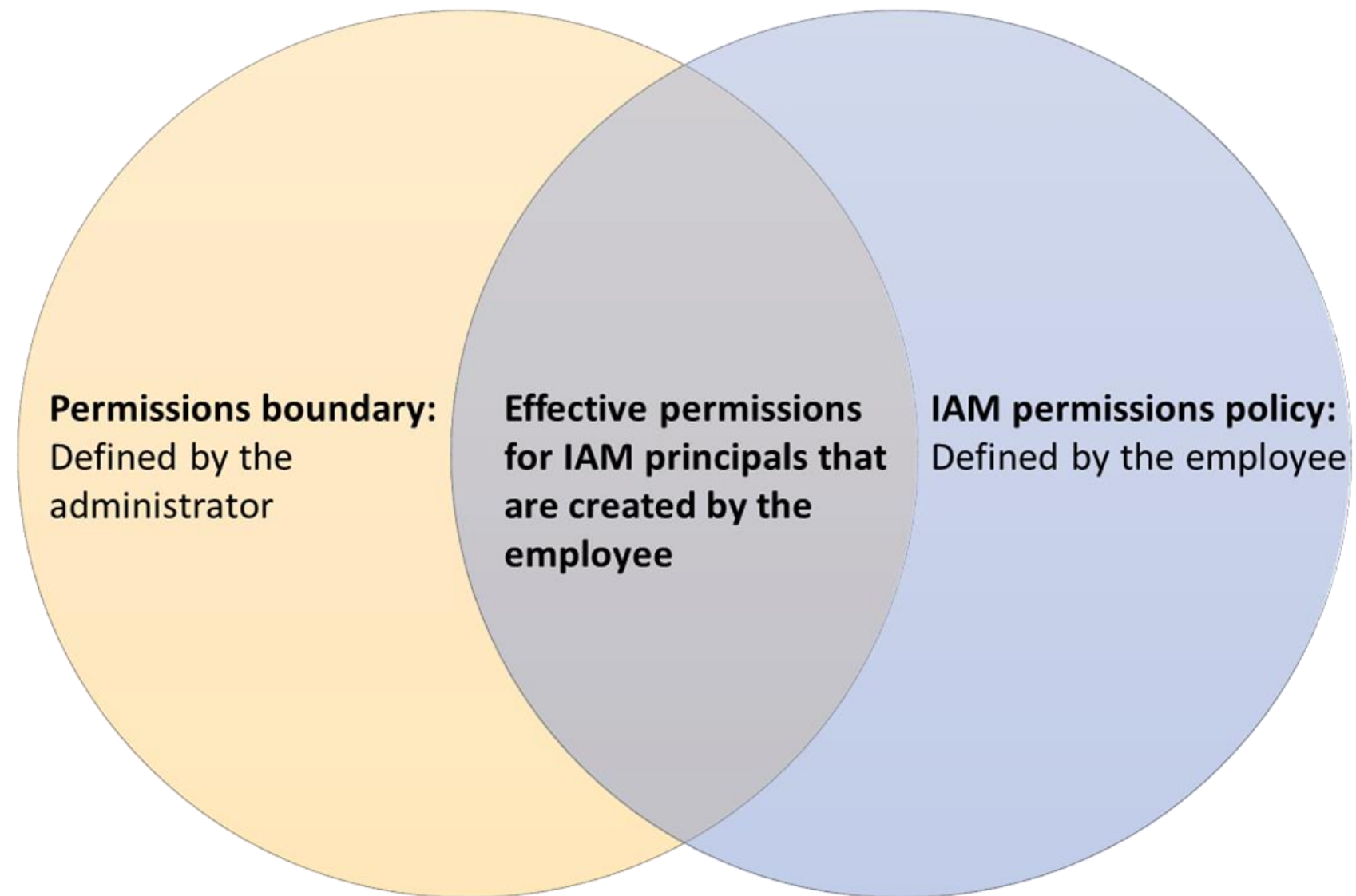
    -> Which resources can be accessed

```json
{

"Statement": [
{
"Effect": "Allow",
"Action": ["s3:Get*", "s3:List*"],
"Resource": "*"
}
]
}
```
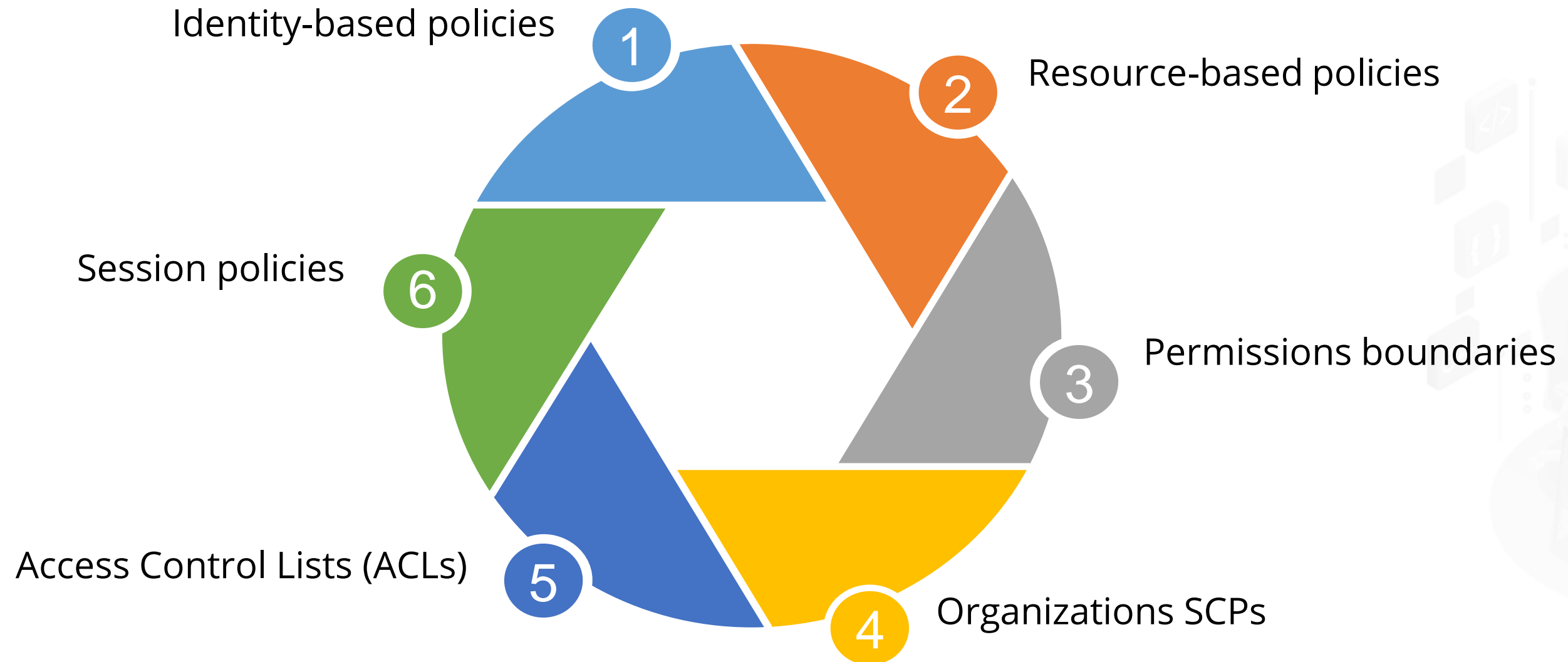
Example of an IAM user/group/role access policy
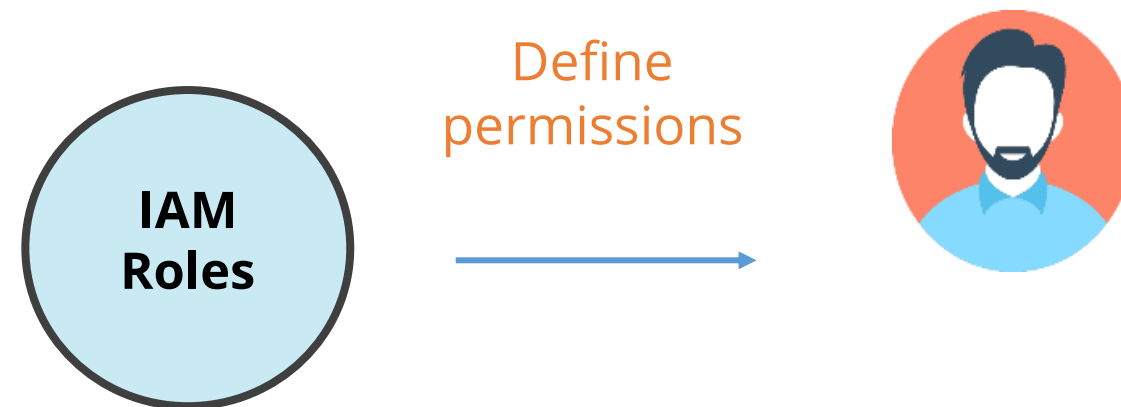
IAM Policies

Categories of access policies

Permissions boundary:
Defined by the administrator

Effective permissions for IAM principals that are created by the employee

IAM permissions policy:
Defined by the employee

IAM Policy Types

Identity-based policies
1

Resource-based policies
2

Permissions boundaries
3

Organizations SCPs
4

Access Control Lists (ACLs)
5
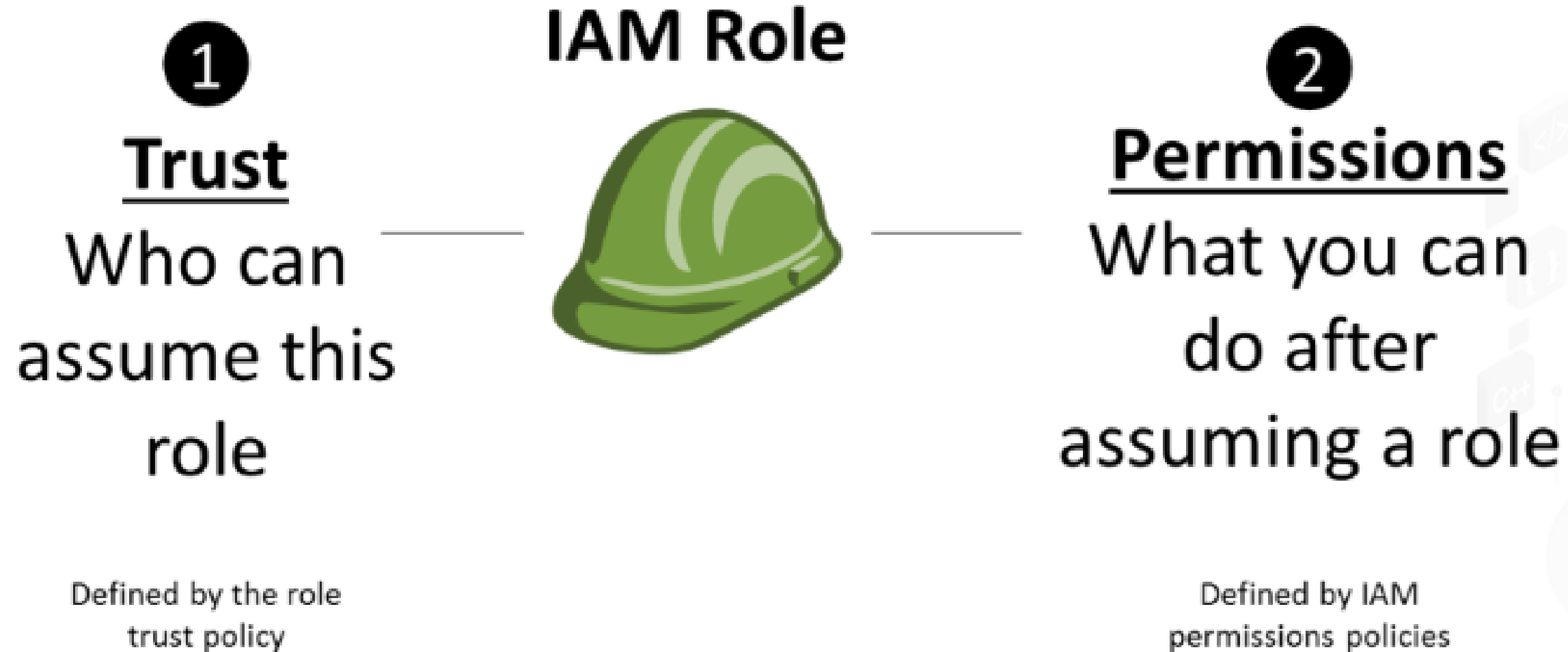
Session policies
6

simplilearn

IAM Roles

- An IAM role is a secure role that defines a set of permissions to entities that you trust.
- It is a permission policy that defines what actions are allowed and denied by an entity in AWS.
- Trusted entities like IAM users or any AWS service assume roles and are not uniquely associated with a user or group.

IAM Roles → Define permissions →

IAM Roles

IAM roles are a secure way to grant permissions to entities you trust.

IAM Role

1

Trust

Who can assume this role

Defined by the role trust policy

2

Permissions

What you can do after assuming a role

Defined by IAM permissions policies

Create an IAM User

Problem Statement:

You are given a project to create an IAM user.

ASSISTED PRACTICE

Assisted Practice: Guidelines to Create an IAM User

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Select the IAM from the AWS console.

4. Navigate to the user creation page.

5. Change access permissions.

6. Add an identifier.

Create an IAM Role

Problem Statement:

You are given a project to create an IAM role.

Assisted Practice: Guidelines to Create an IAM Role

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Create an IAM role.

4. Add AWS policy to the role.

5. Provide a key and value to the role.

Duration: 10 min.

Problem Statement:

You are given a project to create an IAM group.

ASSISTED PRACTICE

Assisted Practice: Guidelines to Create an IAM Group

1. Login to your AWS lab.

2. Free up the utilized resources.

3. Create an IAM group.

4. Assign the required policy to the group.

Duration: 10 min.

Problem Statement:

You are given a project to implement policies and permissions.

Assisted Practice: Guidelines to Understand Policies and Permissions

1. Go to AWS Management Console.

2. Choose a bucket.

3. Add policies and permissions to the bucket.

Duration: 15 min.

Problem Statement:
You are given a project to demonstrate the use of AWS to explain web hosting.

Assisted Practice: Guidelines to Perform Web Hosting

1. Create a custom docker image.

2. Deploy a Spring Boot application to AWS EKS.

Duration: 20 min.

Problem Statement:
You are given a project to demonstrate the use of AWS to host a Spring application on it.

ASSISTED PRACTICE

Assisted Practice: Guidelines to Deploy Your Application

1. Set up EKS CTL command line and dependencies.

2. Create an EKS cluster using eksctl command line.

3. DEploy qan application to AWS EKS cluster.

Key Takeaways

○ Cloud computing refers to on-demand provisioning of IT resources and applications through the Internet

○ Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud.

○ Placement groups are used to influence the placement of a group of interdependent instances to meet the needs of your workload

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

○ Amazon S3 (Simple Storage Service) provides object storage which is built for storing and recovering data from anywhere over the Internet

simplilearn

Deploy Angular Application on EC2 Using Docker

Problem Statement:

Duration: 60 min.

You are asked to deploy an Angular application on AWS EC2 instance using Docker.

Before the Next Class

You should know:

- Fundamentals of software testing

- Unit testing using Backbone.js

- Basics of Grunt

- Docker containerization

- Cloud basics and AWS concepts

# Dockerizing Angular Application

Duration: 240 min.

Project Objective:

As a MEAN Stack Developer, you have to deploy an Angular application on Docker container on an EC2 instance.

Background of the Project Statement

As the project is progressing, the management has asked you to deploy your Angular application on a cloud virtual machine using Docker, so that other developers can access and modify it if required.
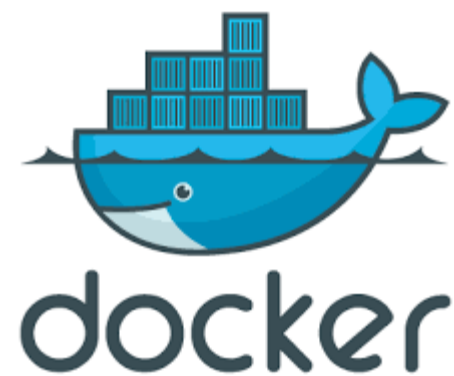
You Are Asked to Do

Following requirements should be meet:

- Source code should be tracked on GitHub repositories. You need to document the tracked files that are ignored during the final push to the GitHub repository.

- The submission of your GitHub repository link is mandatory in order to track your task.

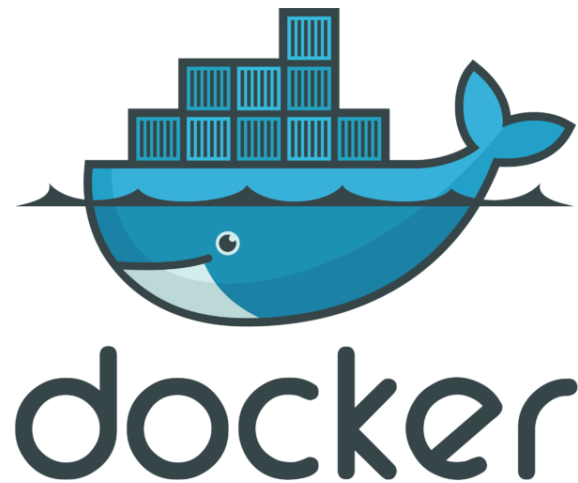- The step-by-step process involved in completing this task should be documented.

You Must Use the Following

AWS Instance

Node.js

Angular CLI

Docker

GitHub