Amazon Web Services

AWS Basics

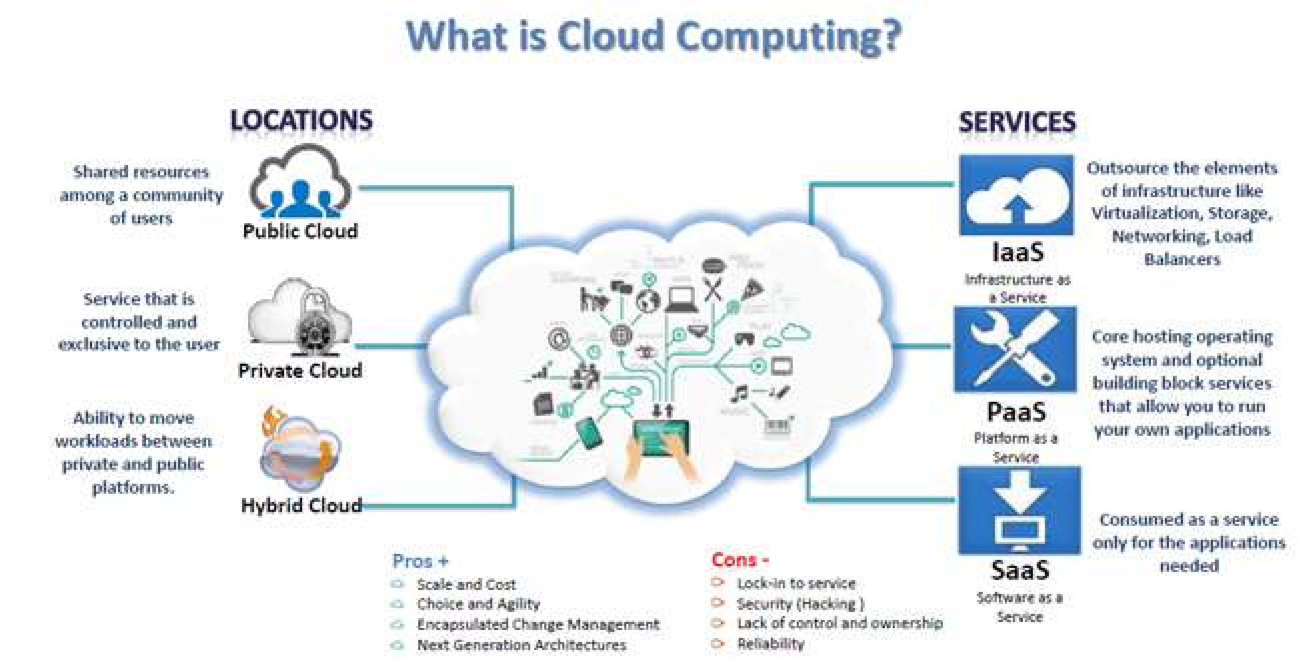
IAM, Security, CLI & EC2

Contents

- ✓ Cloud Computing Basics
- ✓ AWS Overview – Regions, AZs, Pricing
- ✓ IAM (Identity & Access Management)
- ✓ AWS Security – Security Groups & NACL
- ✓ Amazon CLI
- ✓ Amazon EC2 (Elastic Cloud Compute)

Cloud Computing

Cloud Computing is a term that indicates **delivery of computing resources** such as servers, storage, network etc. **over the internet** by a service provider and is **accessible from anywhere and at any time**.



Shared resources among a community of users

Service that is controlled and exclusive to the user

Ability to move workloads between private and public platforms.

Outsource the elements of infrastructure like Virtualization, Storage, Networking, Load Balancers

Core hosting operating system and optional building block services that allow you to run your own applications

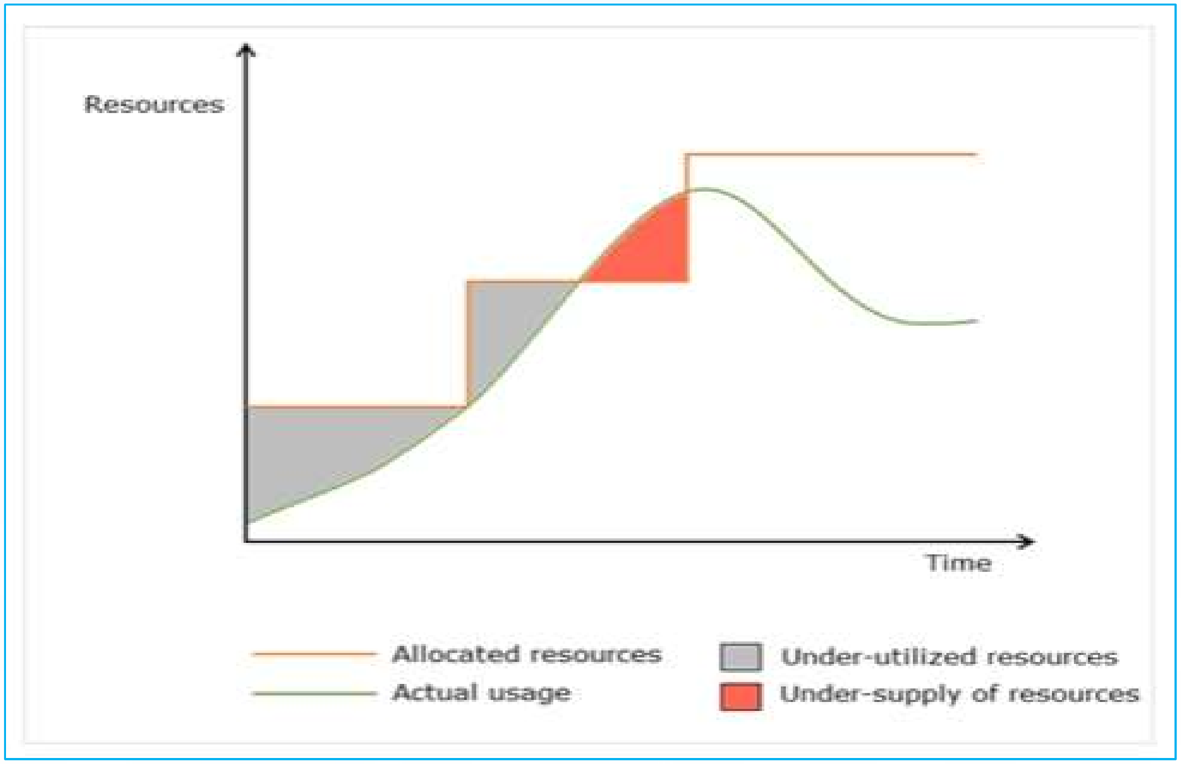
Consumed as a service only for the applications needed

Pros +
- Scale and Cost
- Choice and Agility
- Encapsulated Change Management
- Next Generation Architectures

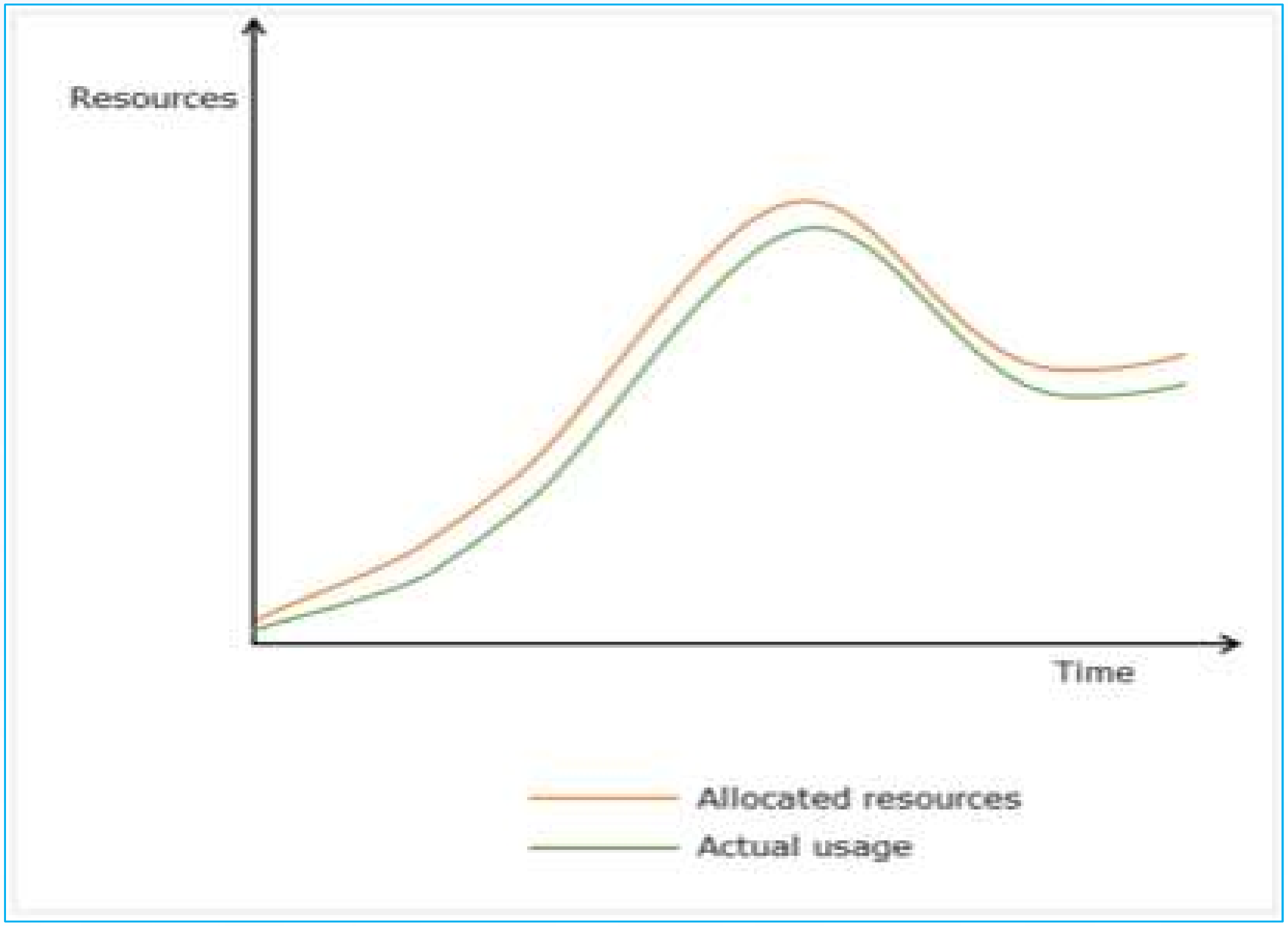
Cons -
- Lock-in to service
- Security (Hacking)
- Lack of control and ownership
- Reliability

Problems with traditional computing model

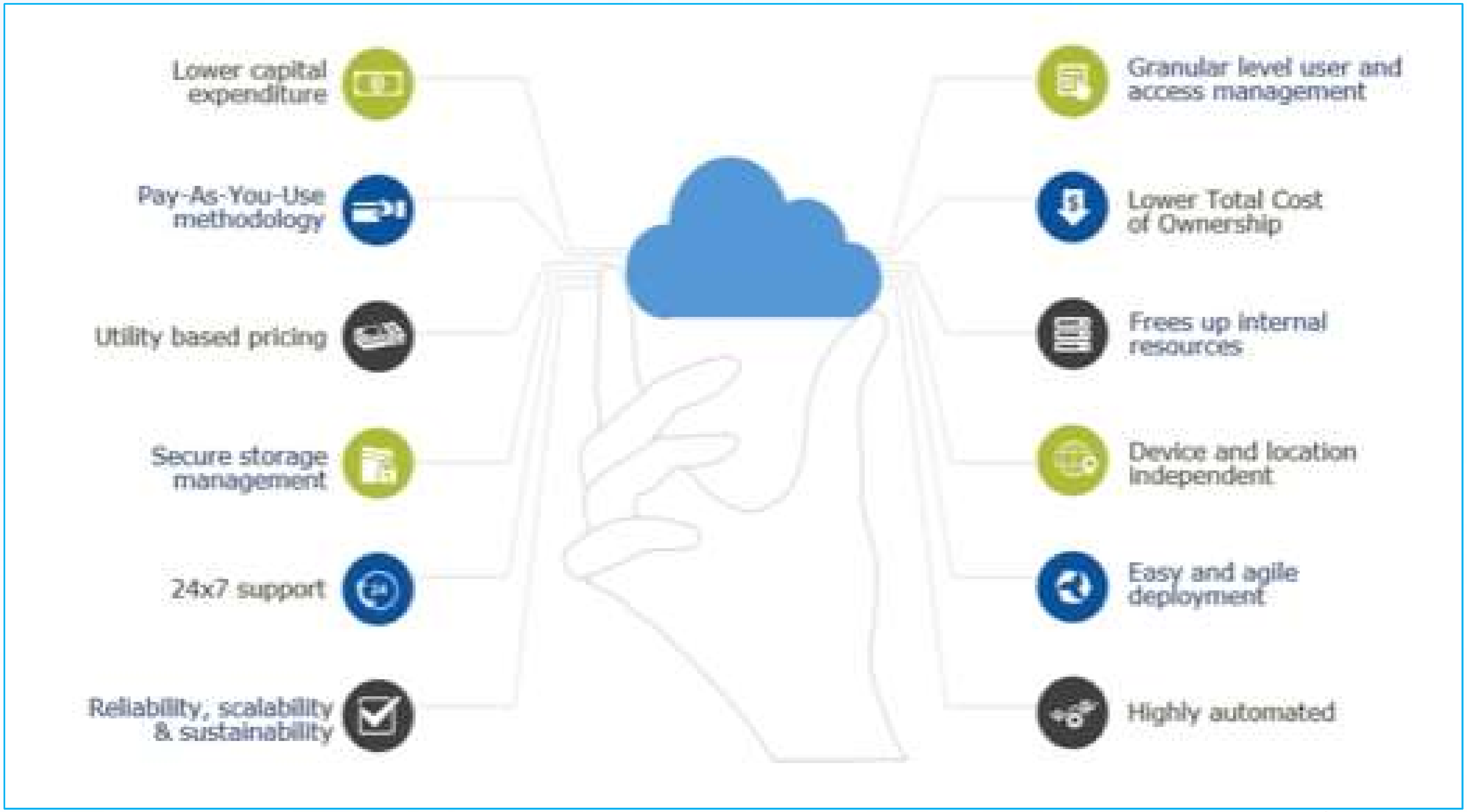
- Under-utilization or under-supply
- Not cost-effective



Runtime provisioning with Cloud

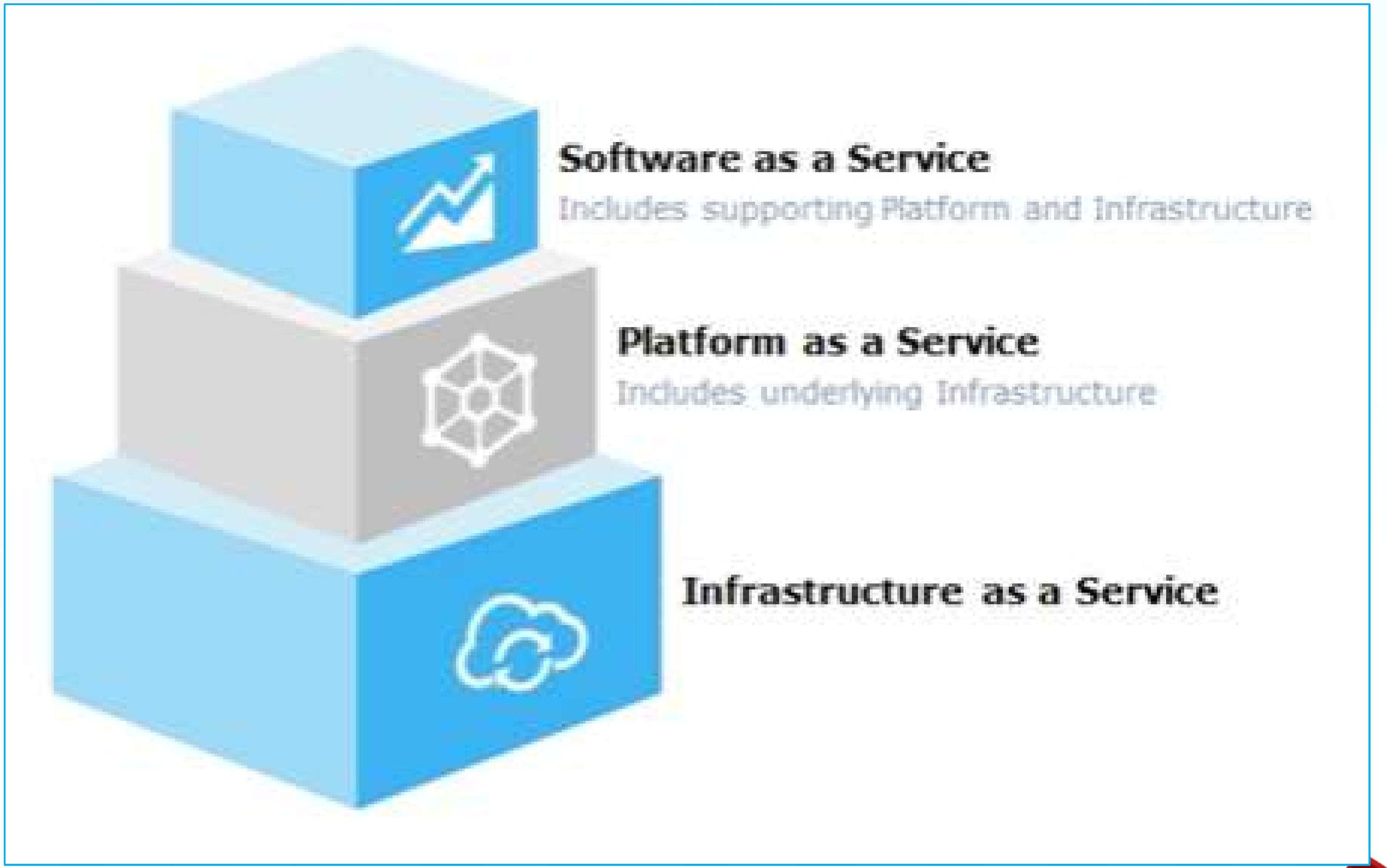


Benefits of Cloud Computing



- Lower capital expenditure
- Pay-As-You-Use methodology
- Utility based pricing
- Secure storage management
- 24x7 support
- Reliability, scalability & sustainability
- Granular level user and access management
- Lower Total Cost of Ownership
- Frees up internal resources
- Device and location independent
- Easy and agile deployment
- Highly automated

Cloud Computing Service Models



Software as a Service

Includes supporting Platform and Infrastructure

Platform as a Service

Includes underlying Infrastructure

Infrastructure as a Service

What is AWS ?

Amazon Web Services (AWS) is a comprehensive cloud computing platform that includes:

- Infrastructure as a service (IaaS)
- Platform as a service (PaaS)

AWS services offer scalable solutions for :

- compute
- storage
- databases
- analytics

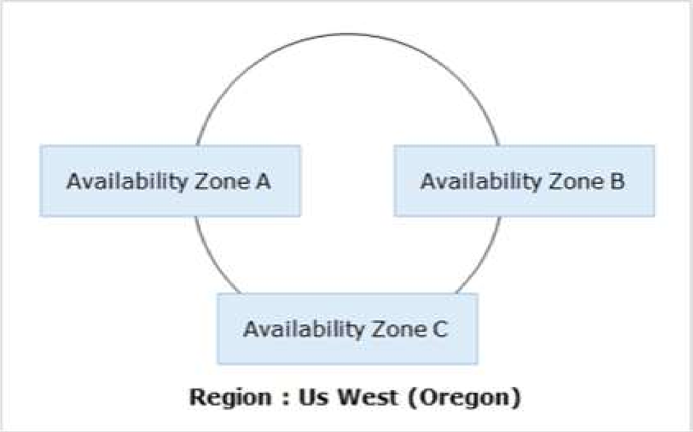
and many more…

AWS Overview

Regions & Availability Zones

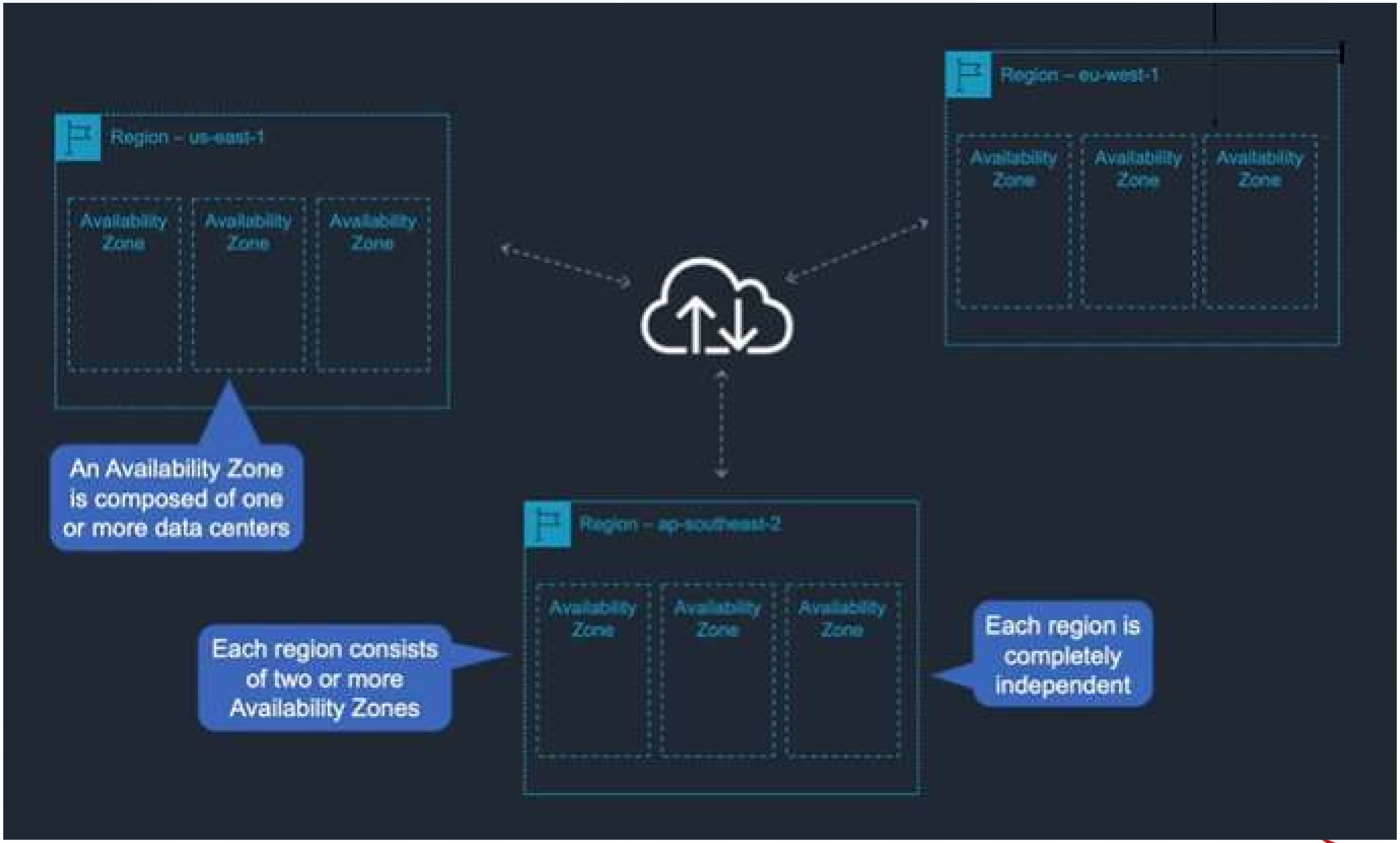
- AWS **Region** is a physical location around the world where AWS clusters data centres called Availability Zones.

- Each group of logical data centers is called an **Availability Zone**.



- Each AWS Region consists of multiple, isolated, and physically separate Availability Zones within a geographic area.
  - Each AZ has independent power, cooling, and physical security and is connected via redundant, ultra-low-latency networks.

Regions & Availability Zones



An Availability Zone is composed of one or more data centers

Each region consists of two or more Availability Zones

Each region is completely independent

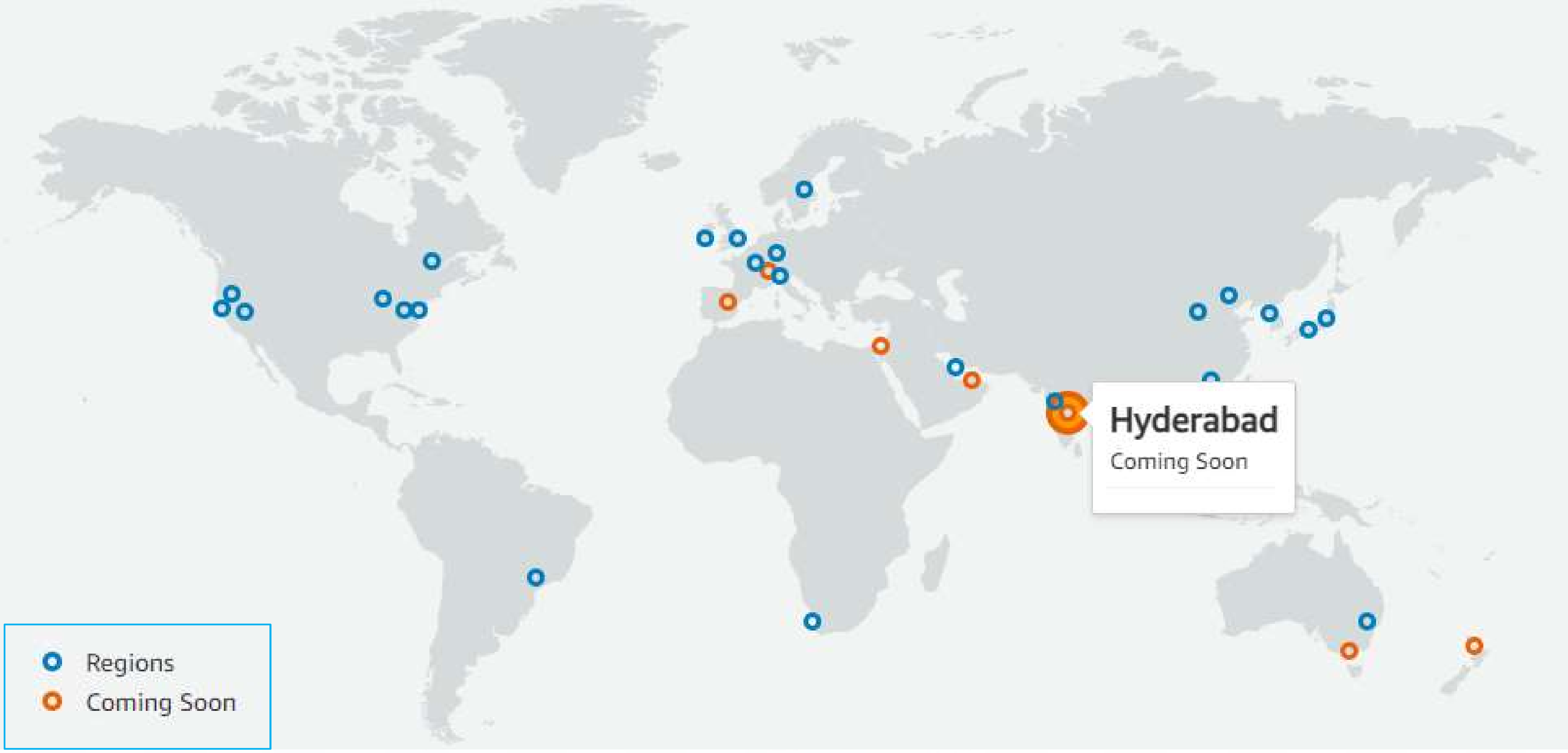
Regions & Availability Zones

- **Region** – Two or more Availability Zones
- **Availability Zone** – One or more data centers
- Each region is completely independent
- All regions are connected via high bandwidth, fully redundant network.

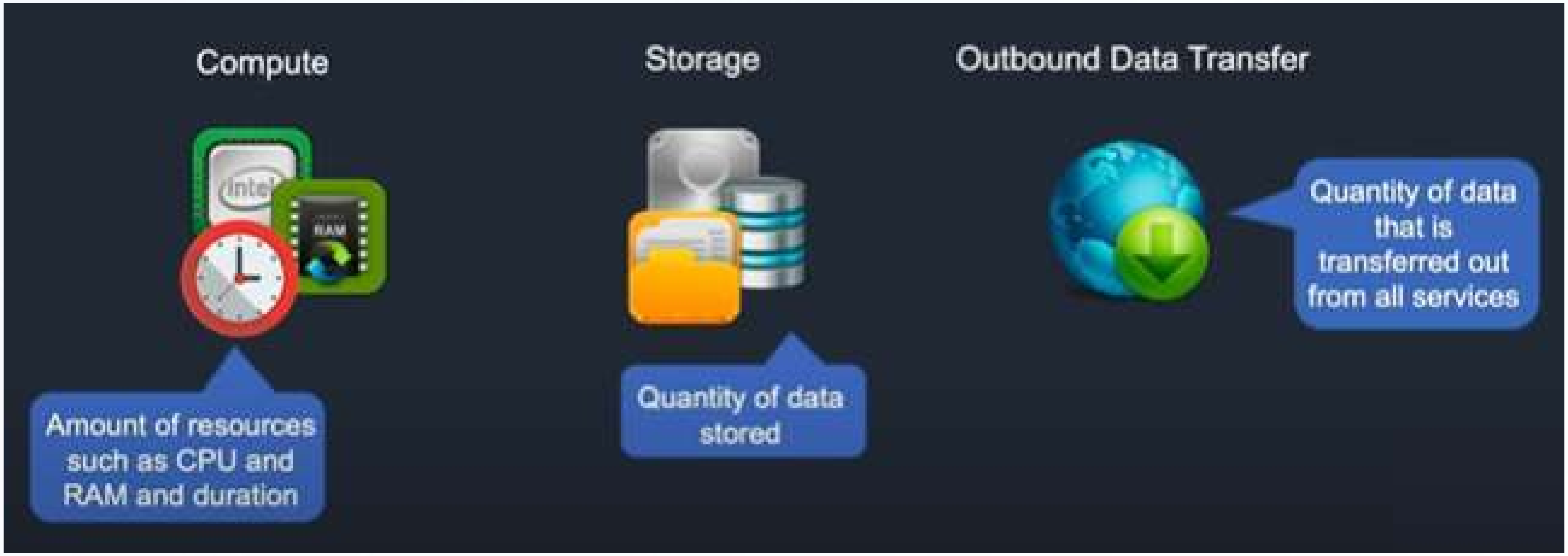
❖ As of this writing, there are 25 regions and 81 availability zones across the globe.

➢ URL: https://aws.amazon.com/about-aws/global-infrastructure/

Regions & Availability Zones



AWS Pricing


Compute
Storage
Outbound Data Transfer
Intel
RAM
Quantity of data that is transferred out from all services
Quantity of data stored
Amount of resources such as CPU and RAM and duration

- ➢ AWS Pricing URL: https://aws.amazon.com/pricing/

AWS Pricing

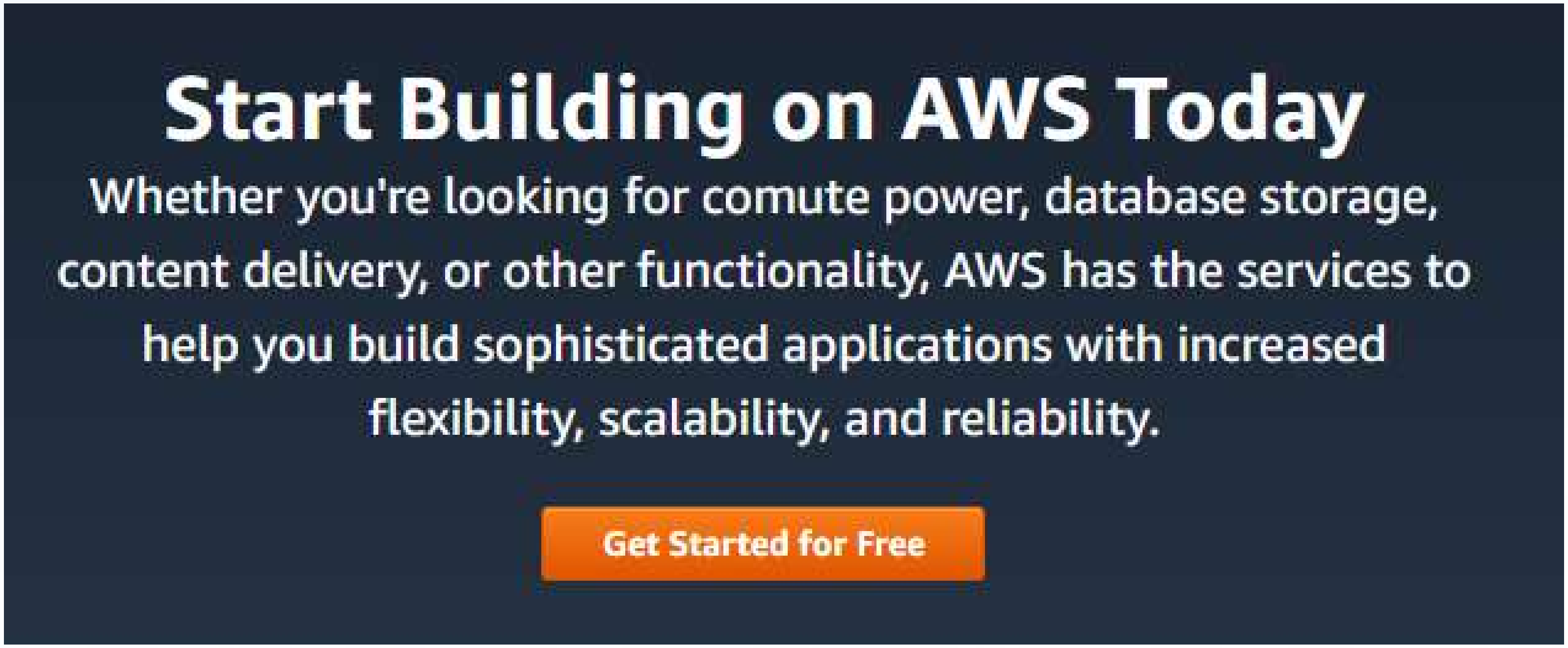
- **Compute**
  - Amount of resources (CPU & RAM)
  - Amount of time
- **Store**
  - Quantity of data you store
- **Outbound Data Transfer**
  - Quantity of data transferred outbound from all services

➢ AWS Pricing URL: https://aws.amazon.com/pricing/

Setup free-tier account

AWS Free Tier Account

URL: https://aws.amazon.com/free/



Start Building on AWS Today

Whether you're looking for comute power, database storage, content delivery, or other functionality, AWS has the services to help you build sophisticated applications with increased flexibility, scalability, and reliability.

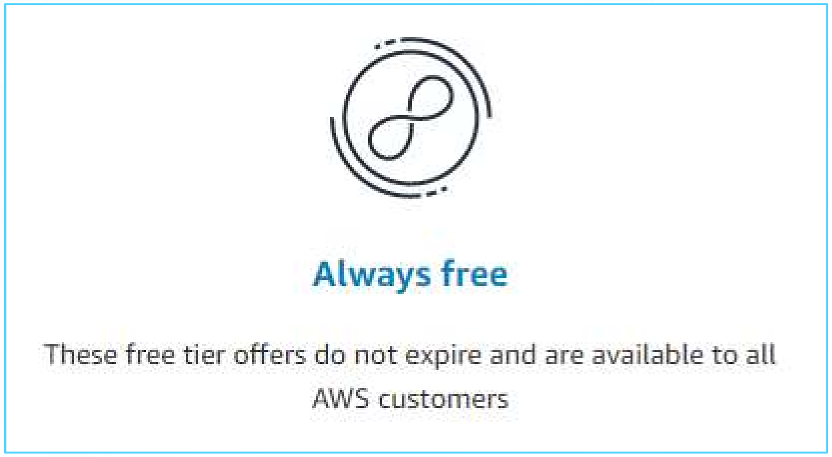
AWS Free Tier Account

Free trials

Short-term free trial offers start from the date you activate a particular service

12 months free

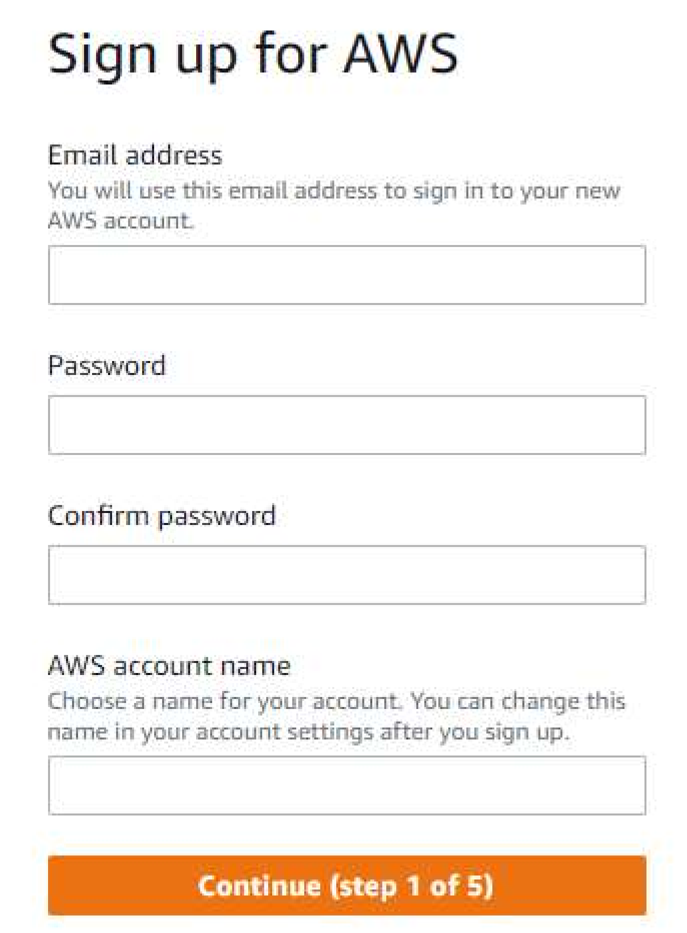
Enjoy these offers for 12-months following your initial sign-up date to AWS



Always free

These free tier offers do not expire and are available to all AWS customers

AWS Free Tier Account



Sign up for AWS

Email address

You will use this email address to sign in to your new AWS account.

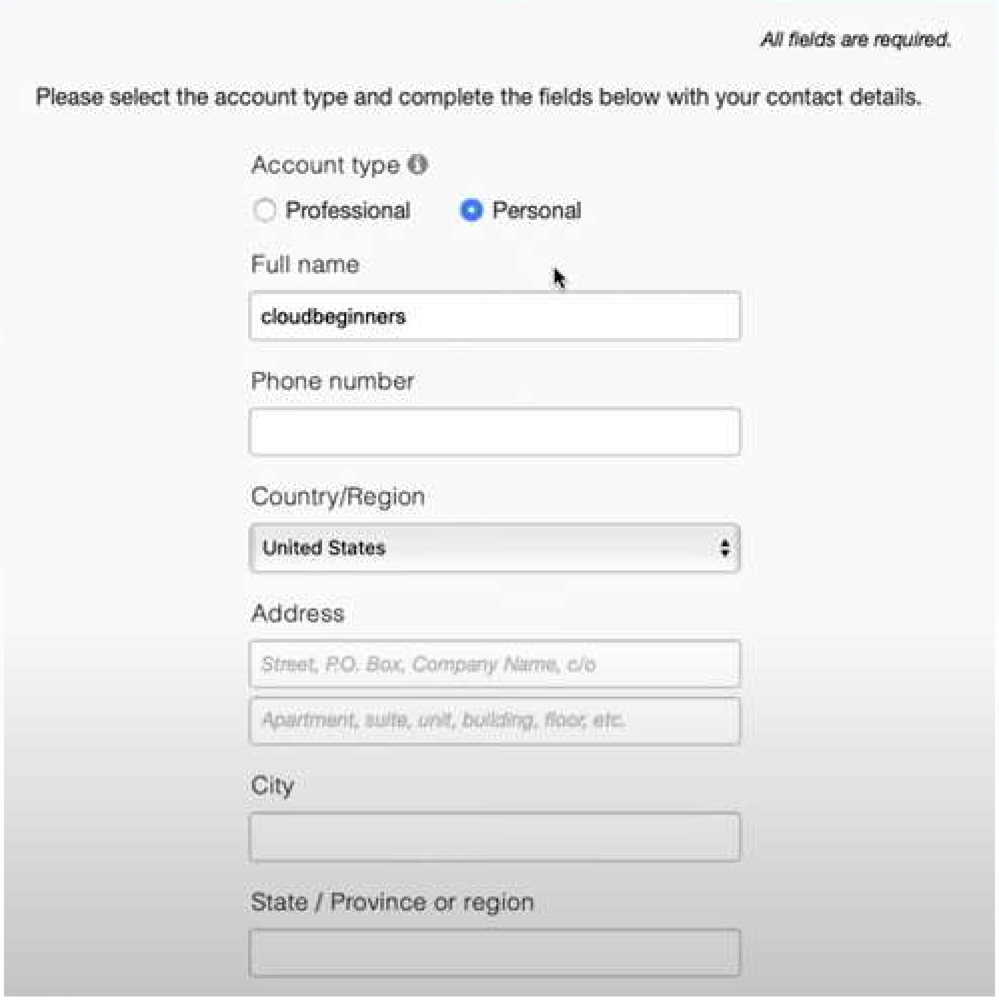
Password

Confirm password

AWS account name

Choose a name for your account. You can change this name in your account settings after you sign up.

AWS Free Tier Account



All fields are required.

Please select the account type and complete the fields below with your contact details.

Account type

○ Professional ● Personal

Full name

cloudbeginners

Phone number

Country/Region

United States

Address

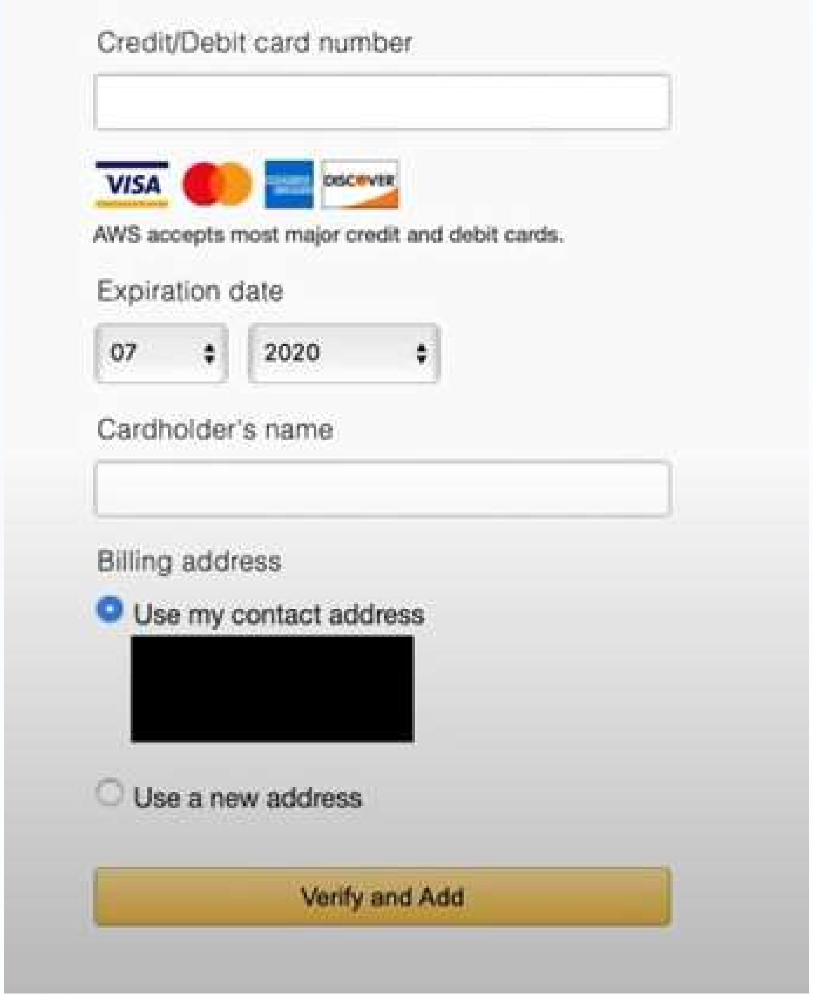
Street, P.O. Box, Company Name, c/o

Apartment, suite, unit, building, floor, etc.

City

State / Province or region

AWS Free Tier Account



Credit/Debit card number

AWS accepts most major credit and debit cards.

Expiration date

07 2020

Cardholder's name

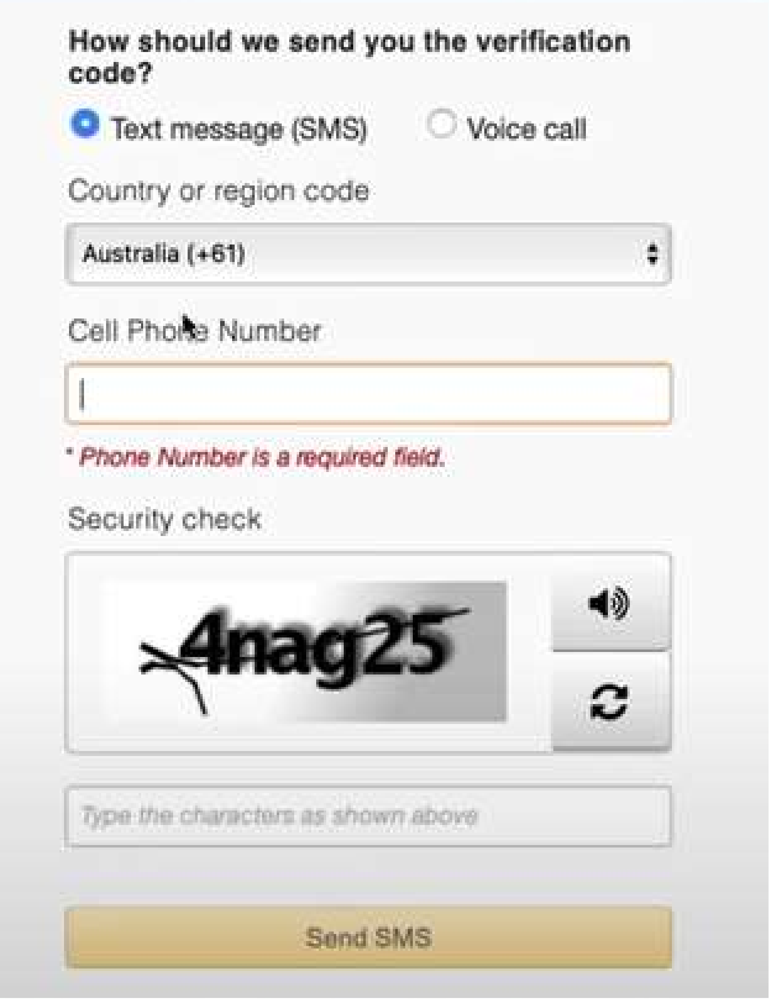
Billing address

● Use my contact address

[redacted]

○ Use a new address

AWS Free Tier Account



How should we send you the verification code?

● Text message (SMS) ○ Voice call

Country or region code

Australia (+61)

Cell Phone Number

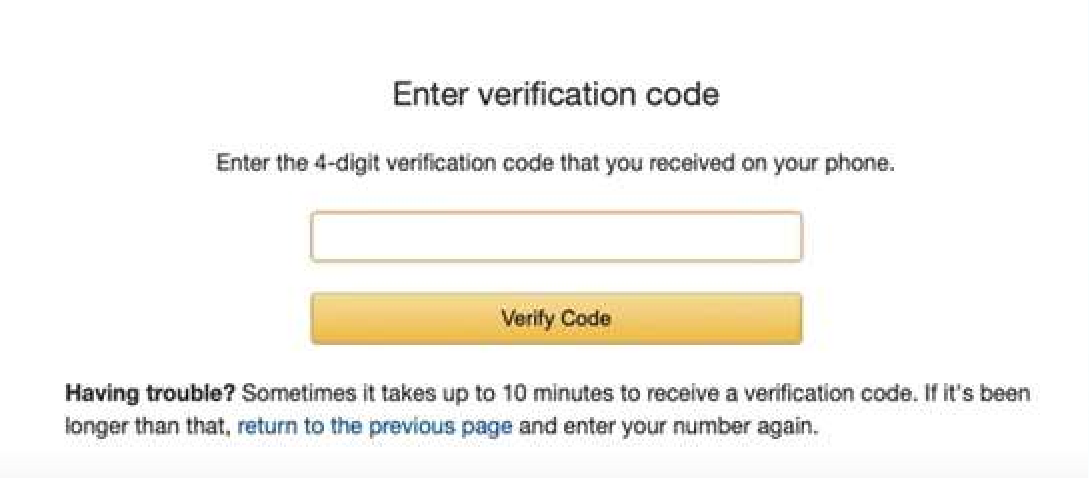
* Phone Number is a required field.

Security check

4nag25

Type the characters as shown above

AWS Free Tier Account

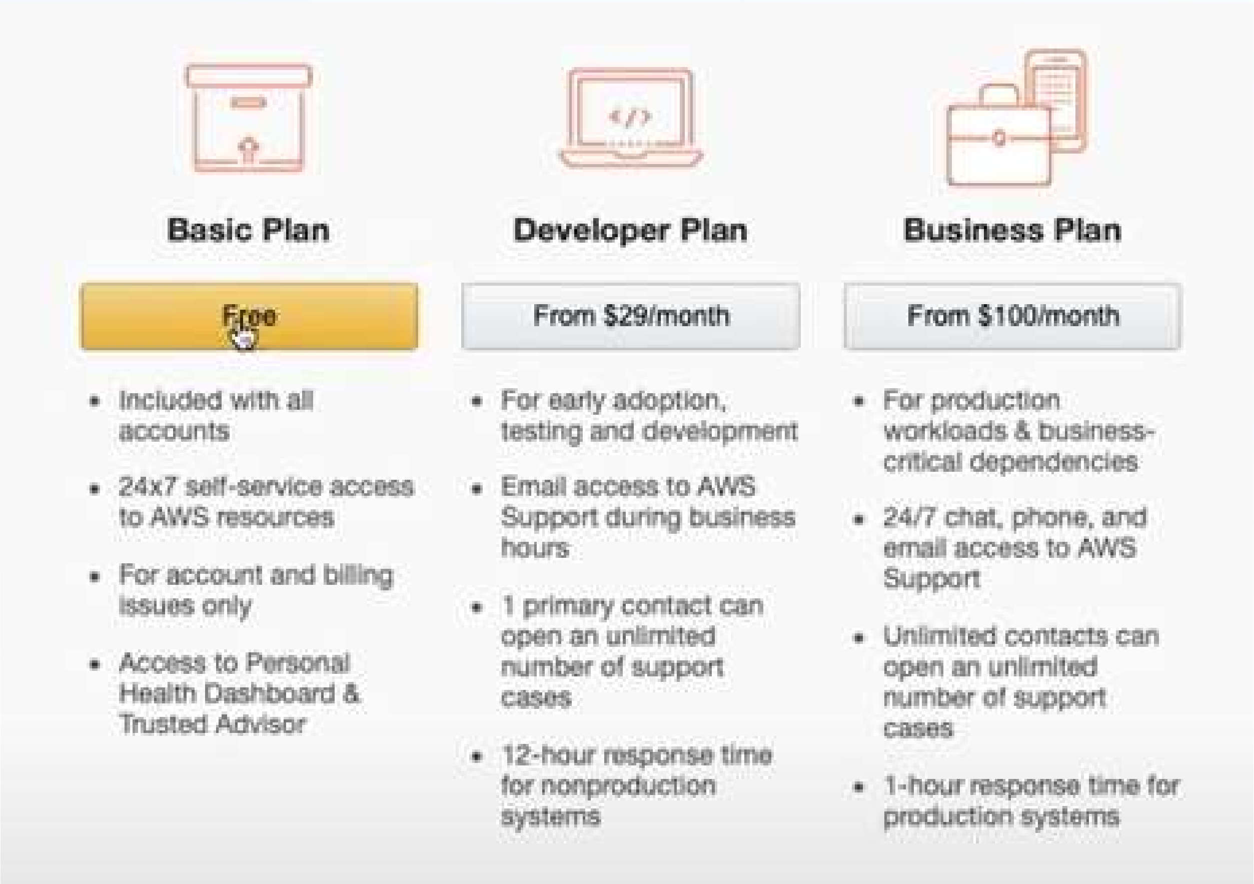


Enter verification code

Enter the 4-digit verification code that you received on your phone.

Having trouble? Sometimes it takes up to 10 minutes to receive a verification code. If it's been longer than that, return to the previous page and enter your number again.

AWS Free Tier Account



Basic Plan

Free

- Included with all accounts
- 24x7 self-service access to AWS resources
- For account and billing issues only
- Access to Personal Health Dashboard & Trusted Advisor

Developer Plan

From $29/month

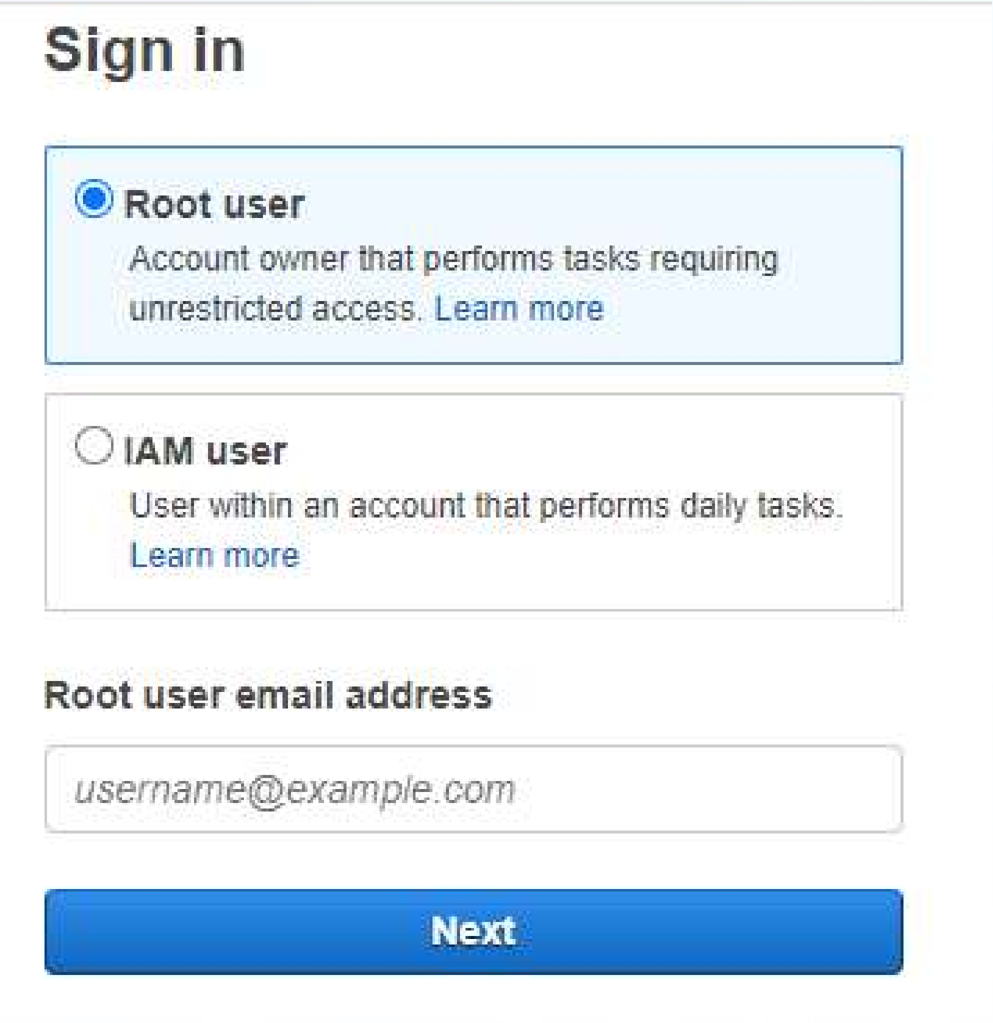
- For early adoption, testing and development
- Email access to AWS Support during business hours
- 1 primary contact can open an unlimited number of support cases
- 12-hour response time for nonproduction systems

Business Plan

From $100/month

- For production workloads & business-critical dependencies
- 24/7 chat, phone, and email access to AWS Support
- Unlimited contacts can open an unlimited number of support cases
- 1-hour response time for production systems

Sign-in to AWS account



Sign in

● **Root user**
Account owner that performs tasks requiring unrestricted access. Learn more

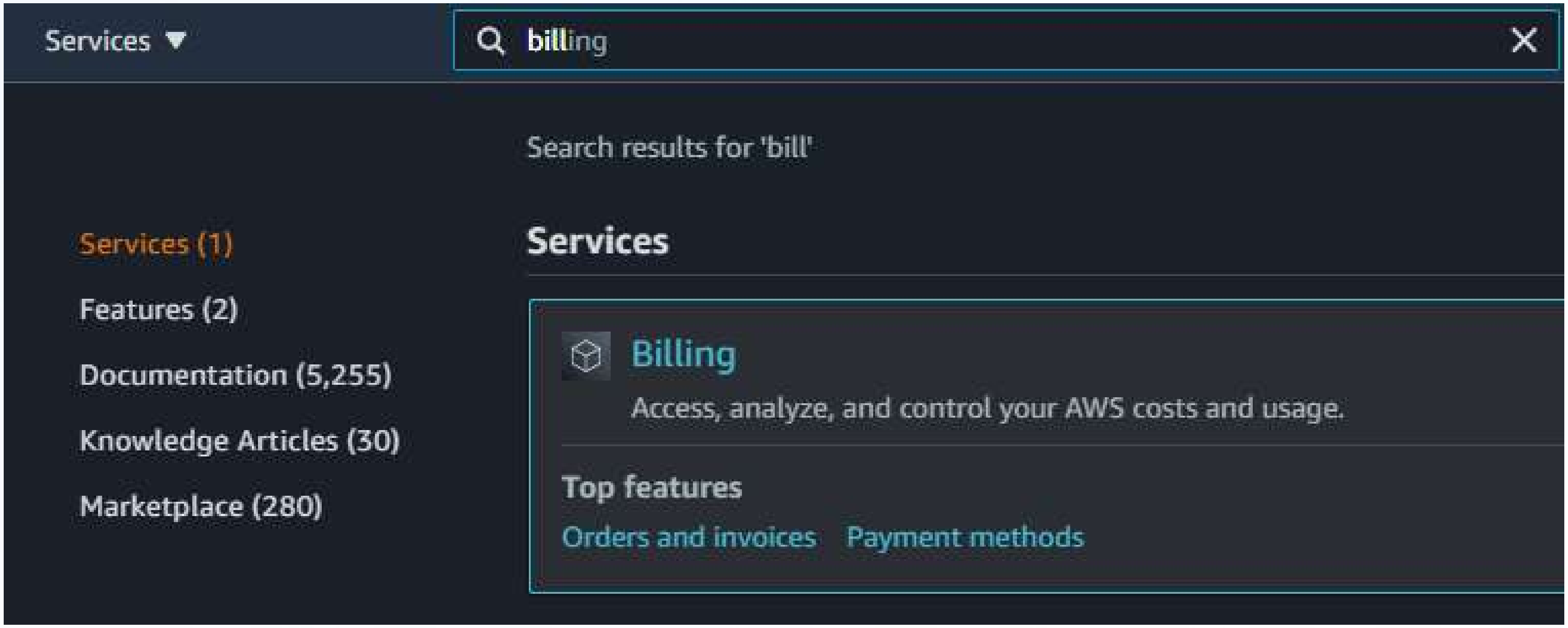
○ **IAM user**
User within an account that performs daily tasks. Learn more

Root user email address

username@example.com

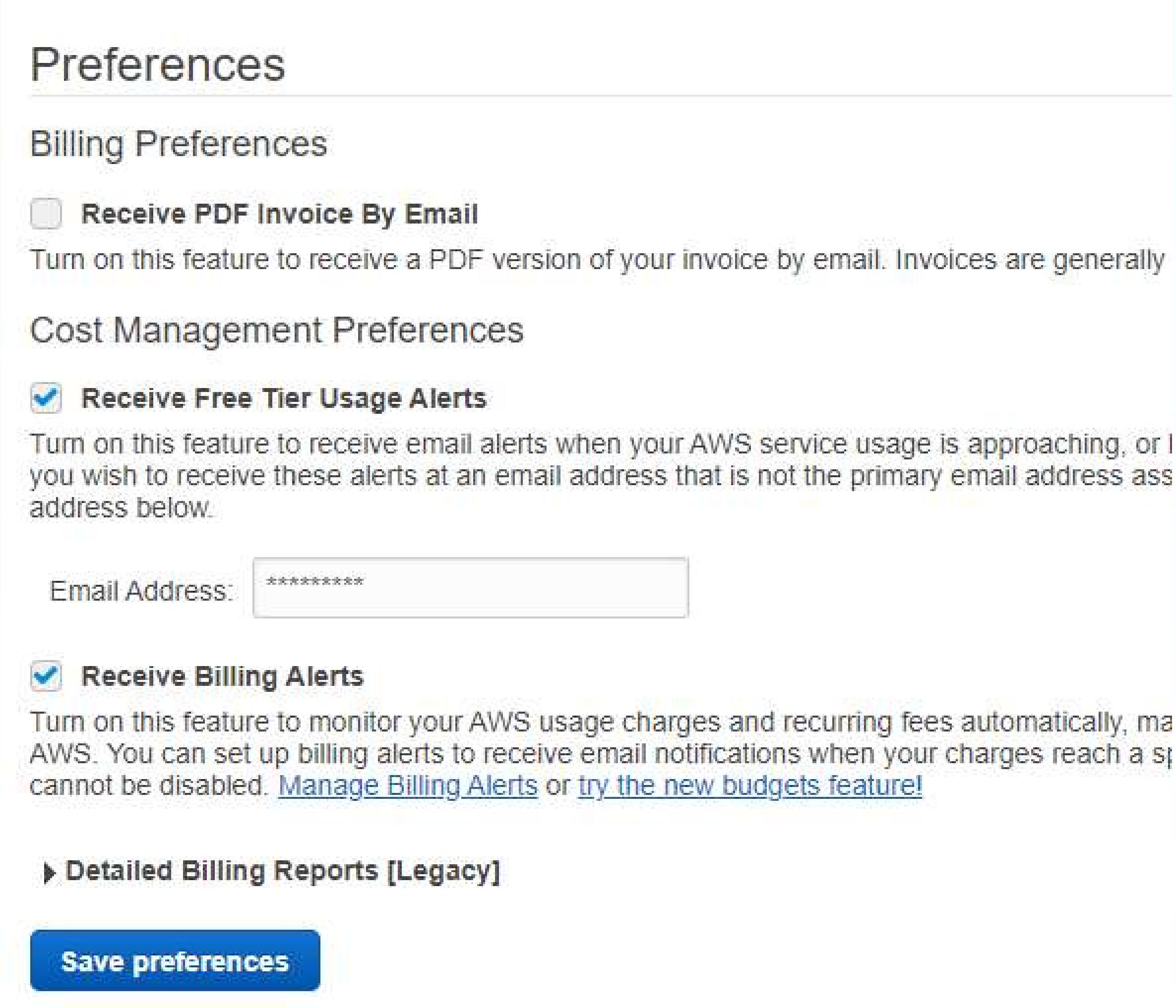
Creating a Billing Alarm

Setting up a Billing Alarm



- Click on **Billing Preferences** Option from the left menu

Setting up a Billing Alarm



Preferences

Billing Preferences

□ **Receive PDF Invoice By Email**

Turn on this feature to receive a PDF version of your invoice by email. Invoices are generally

Cost Management Preferences

☑ **Receive Free Tier Usage Alerts**

Turn on this feature to receive email alerts when your AWS service usage is approaching, or I you wish to receive these alerts at an email address that is not the primary email address ass address below.

Email Address: **********

☑ **Receive Billing Alerts**

Turn on this feature to monitor your AWS usage charges and recurring fees automatically, ma AWS. You can set up billing alerts to receive email notifications when your charges reach a sp cannot be disabled. Manage Billing Alerts or try the new budgets feature!

▸ **Detailed Billing Reports [Legacy]**

Identity and Access Management Service (IAM)

AWS IAM (Identity & Access Management)

- AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources for your users.
- IAM is used to control
  - **Identity** – who can use your AWS resources (authentication)
  - **Access** – what resources they can use and in what ways (authorization)

AWS IAM (Identity & Access Management)

User

IAM user account provides one login with its own specified permissions.

Group

Allows you to apply specified permissions to a group of users.

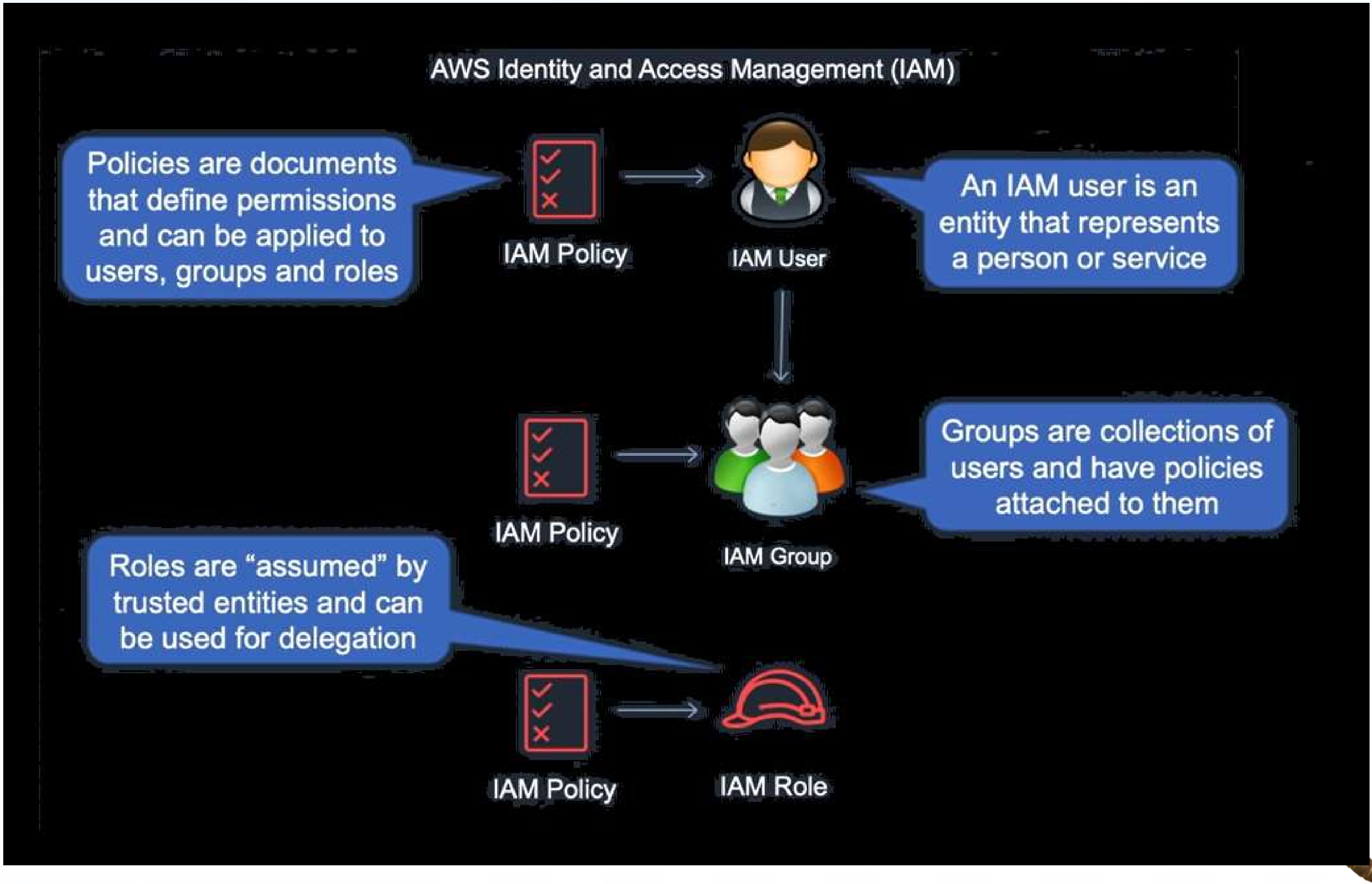
Role

Provides a simple way to delegate groups of permissions to specified users or AWS services.

Permissions

Contains permissions which specify which actions an entity can perform and on which resources.

AWS IAM (Identity & Access Management)



AWS Identity and Access Management (IAM)

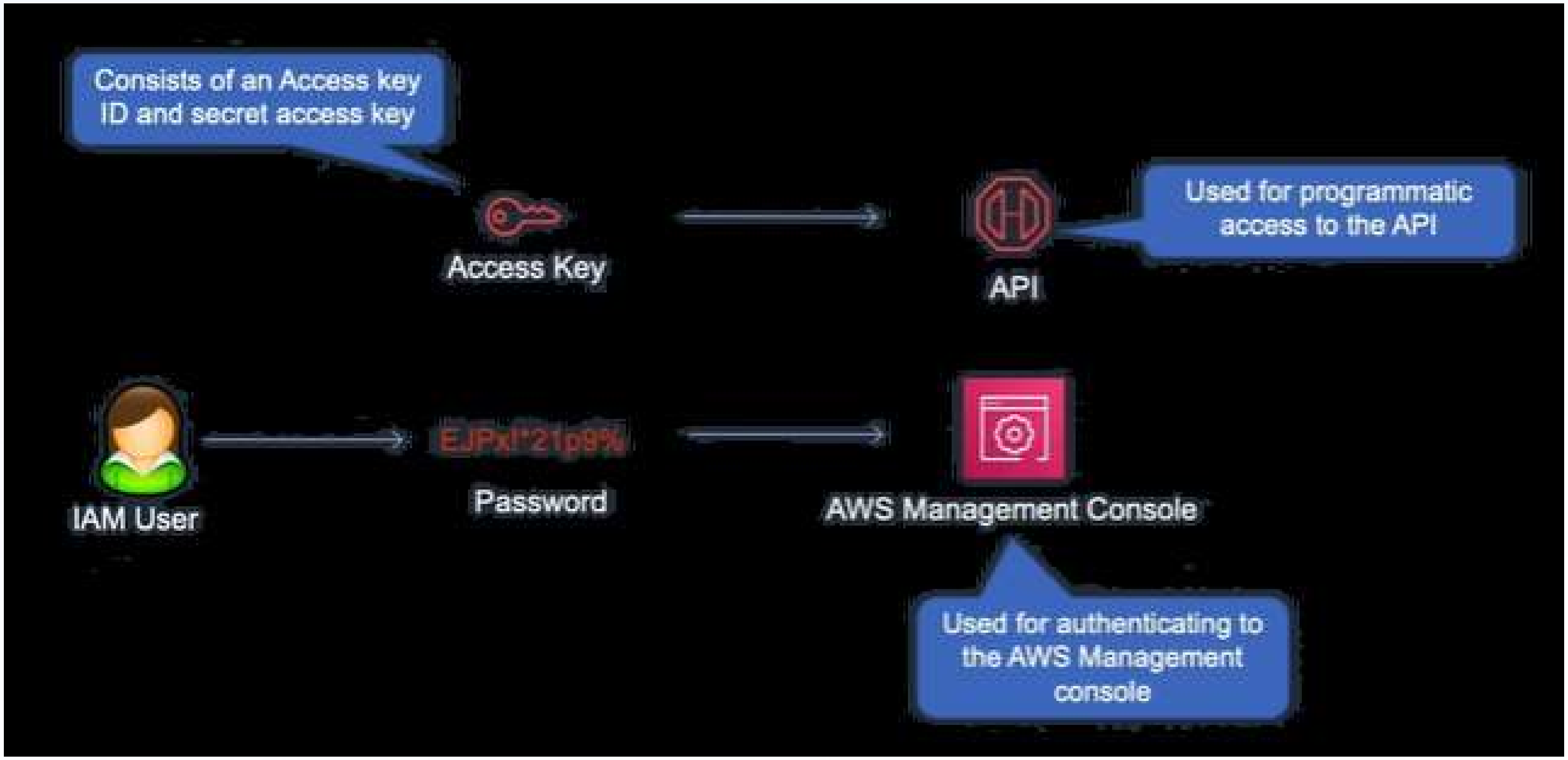
Policies are documents that define permissions and can be applied to users, groups and roles

An IAM user is an entity that represents a person or service

Groups are collections of users and have policies attached to them

Roles are “assumed” by trusted entities and can be used for delegation

Authentication Methods



Consists of an Access key ID and secret access key
Access Key
API
Used for programmatic access to the API
IAM User
EJPx!*21p9%
Password
AWS Management Console
Used for authenticating to the AWS Management console

Create IAM User & Group

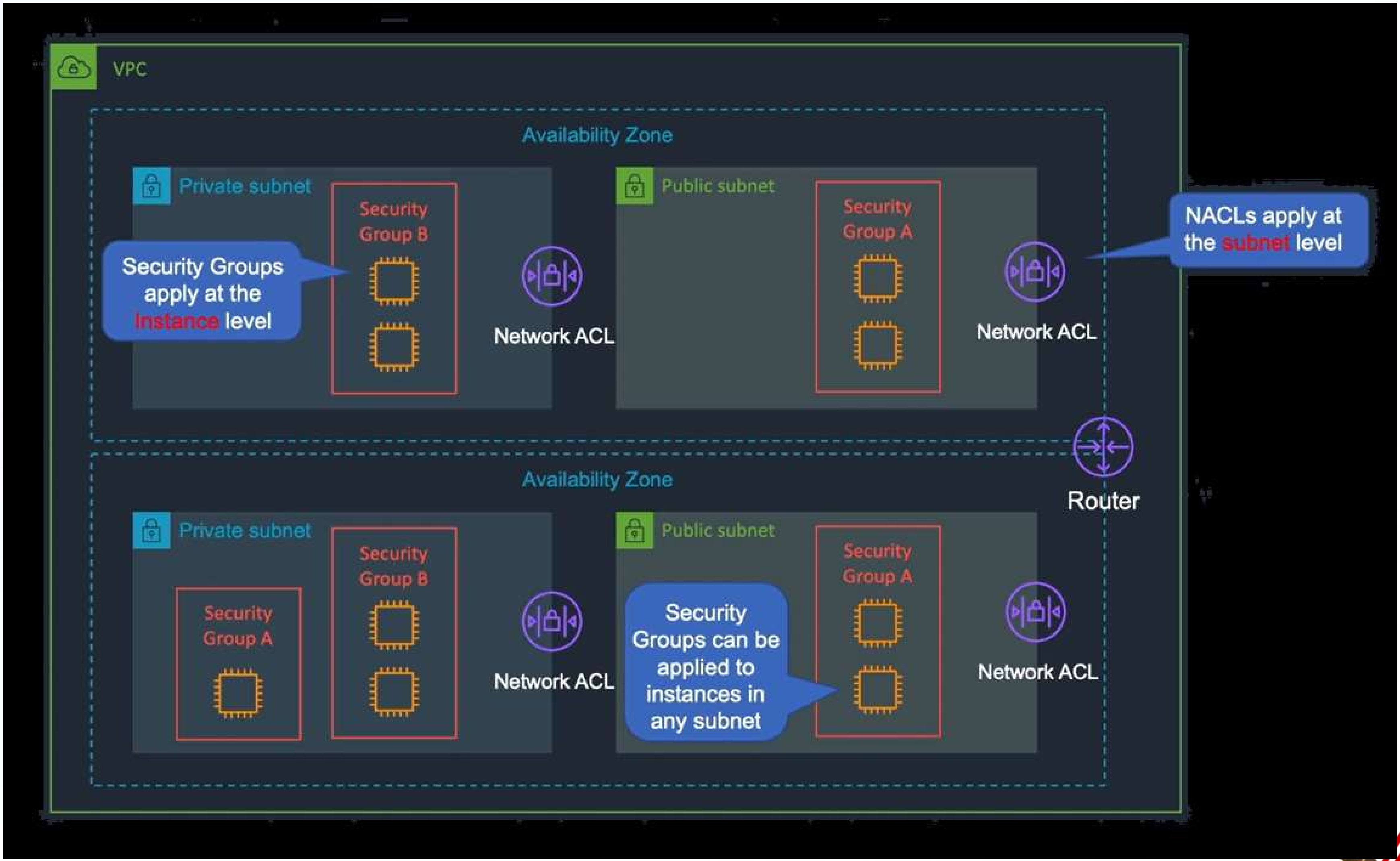
Demo

- Create a User Group
- Create a User in that group

AWS Security

Security Groups & Network Access Control Lists (NACLs)

Security Groups & NACLs



Security Group

- A **security group** acts as a virtual firewall for your EC2 instances to control incoming and outgoing traffic.
- Inbound rules control the incoming traffic to your instance, and outbound rules control the outgoing traffic from your instance.
- When you launch an instance, you can specify one or more security groups.
- If you don't specify a security group, the default security group is used.
- You can add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time. New and modified rules are automatically applied to all instances that are associated with the security group.
- Security groups are stateful firewalls

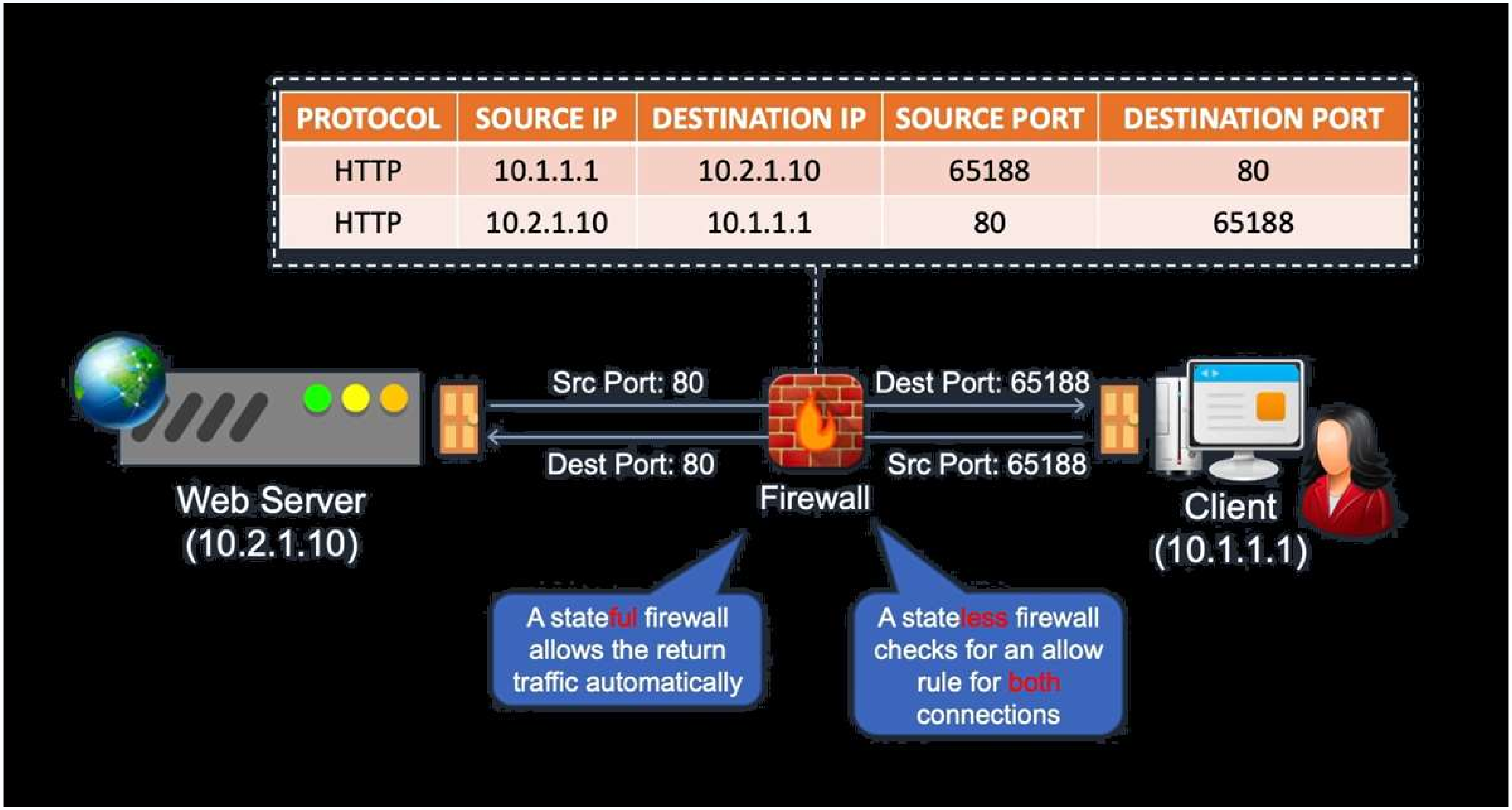
Network ACLs

- A **network access control list (ACL)** is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets.

- You might set up network ACLs with rules similar to your security groups in order to add an additional layer of security to your VPC.

- Network ACLs are stateless firewalls

Firewalls – Stateful & Stateless

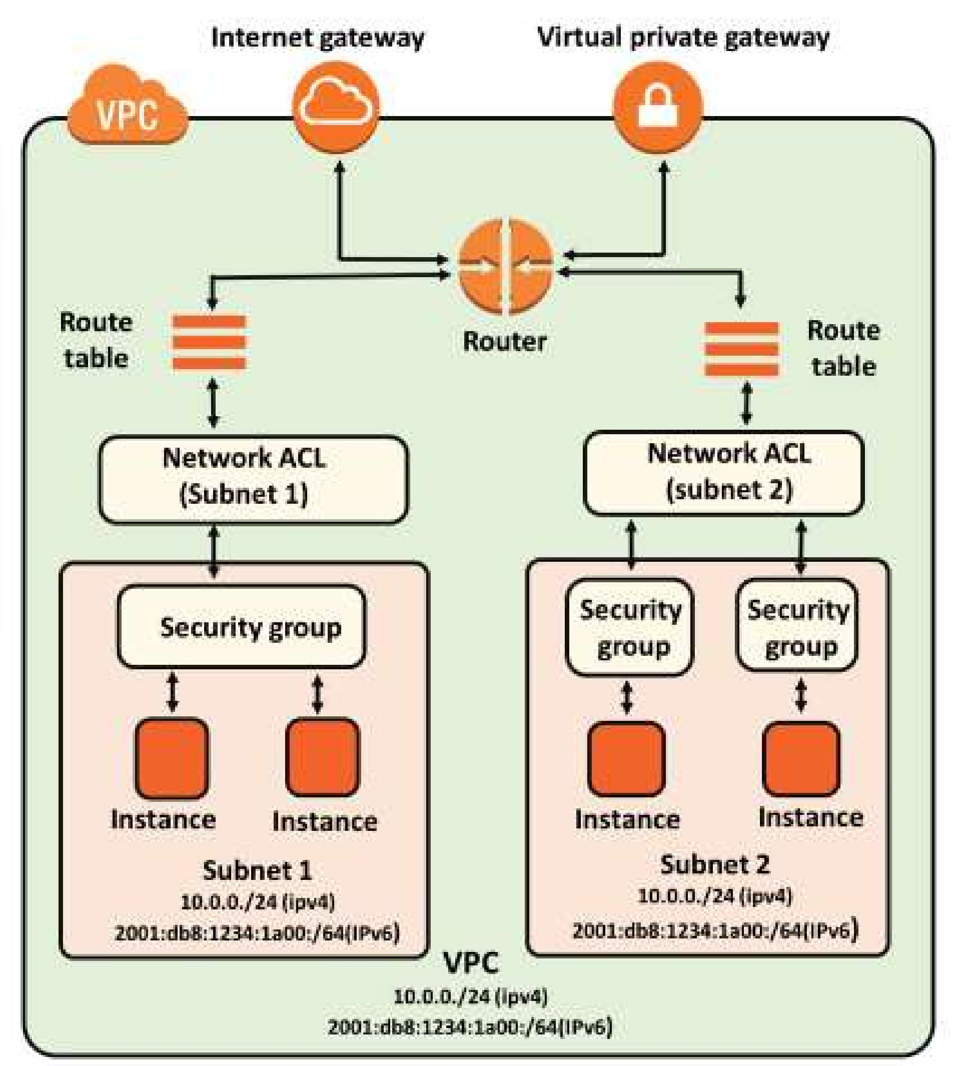


PROTOCOL	SOURCE IP	DESTINATION IP	SOURCE PORT	DESTINATION PORT
HTTP	10.1.1.1	10.2.1.10	65188	80
HTTP	10.2.1.10	10.1.1.1	80	65188

A stateful firewall allows the return traffic automatically

A stateless firewall checks for an allow rule for both connections

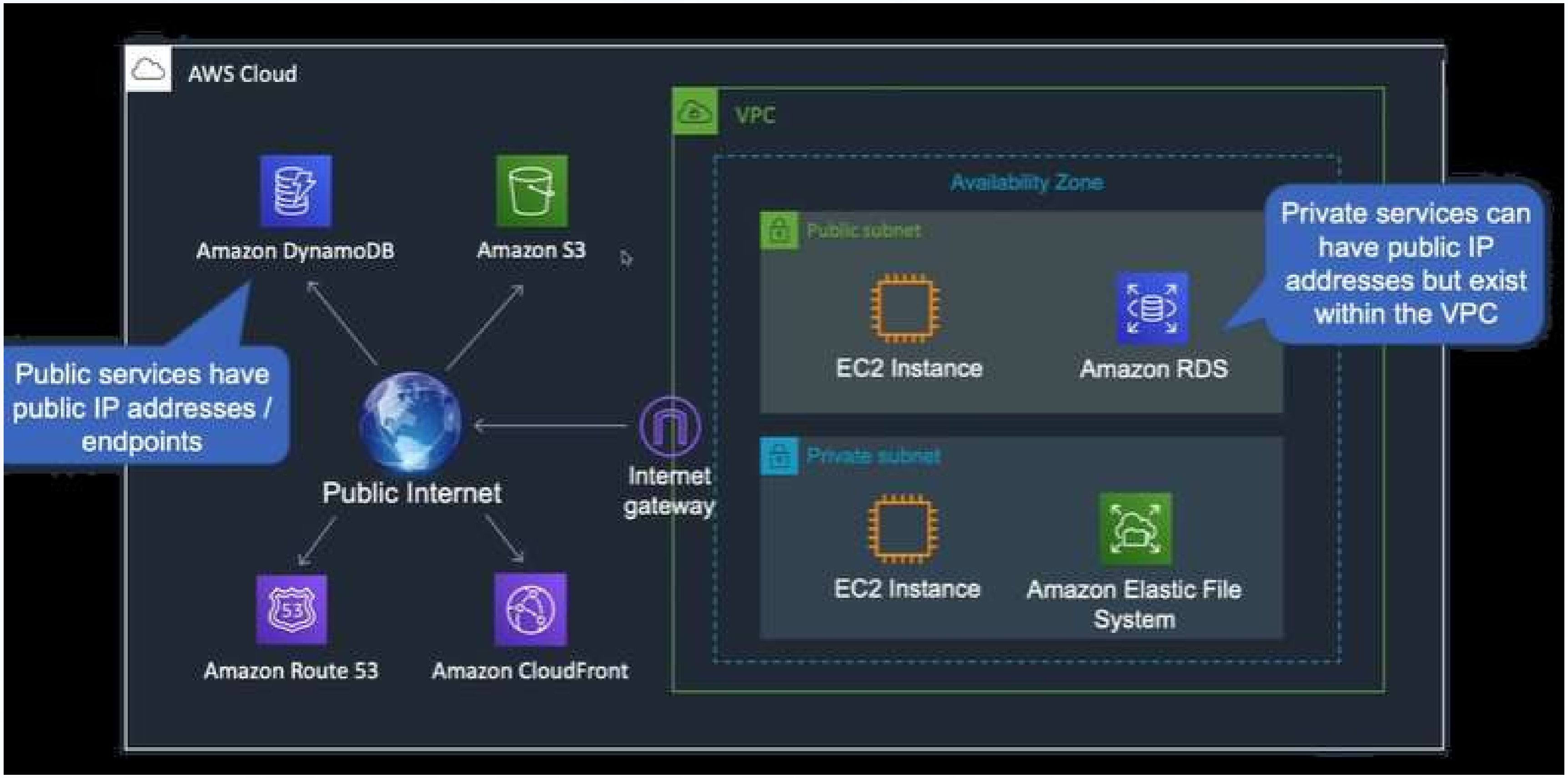
Security Group Vs. Network ACL



Security Group Vs. Network ACL

Security Group	NACL (Network Access Control List)
It supports only **allow** rules, and by default, all the rules are denied. You cannot deny the rule for establishing a connection.	It supports both **allow and deny** rules, and by default, all the rules are denied. You need to add the rule which you can either allow or deny it.
It is a **stateful** means that any changes made in the inbound rule will be automatically reflected in the outbound rule. For example, If you are allowing an incoming port 80, then you also have to add the outbound rule explicitly.	It is a **stateless** means that any changes made in the inbound rule will not reflect the outbound rule, i.e., you need to add the outbound rule separately. For example, if you add an inbound rule port number 80, then you also have to explicitly add the outbound rule.
It is associated with an EC2 instance.	It is associated with a subnet.
All the rules are evaluated before deciding whether to allow the traffic.	Rules are evaluated in order, starting from the lowest number.
Security Group is applied to an instance only when you specify a security group while launching an instance.	NACL has applied automatically to all the instances which are associated with an instance.
It is the first layer of defense.	It is the second layer of defense.

AWS Public & Private Services



AWS Command Line Interface (CLI)

Amazon CLI

- The AWS Command Line Interface (CLI) is a unified tool to manage your AWS services.
- With just one tool to download and configure, you can control multiple AWS services from the command line and automate them through scripts.
- The AWS CLI v2 offers several new features including improved installers, new configuration options such as AWS Single Sign-On (SSO), and various interactive features.

Amazon CLI

- The AWS Command Line Interface (AWS CLI) is an open source tool that enables you to interact with AWS services using commands in your command-line shell.

- With minimal configuration, the AWS CLI enables you to start running commands that implement functionality equivalent to that provided by the browser-based AWS Management Console from the command prompt in your terminal program:

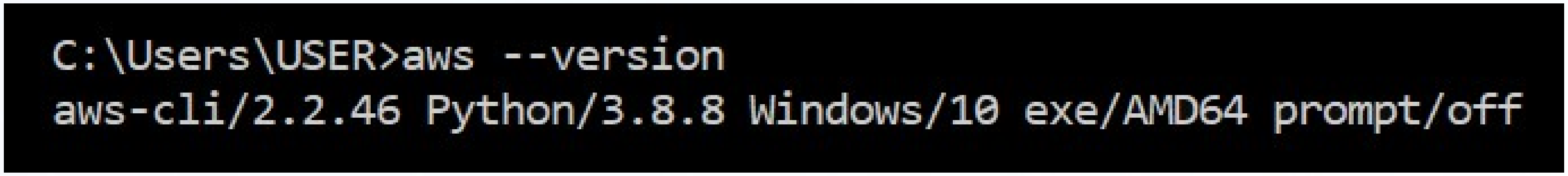
  - **Linux shells** – Use common shell programs such as bash, zsh, and tcsh to run commands in Linux or macOS.

  - **Windows command line** – On Windows, run commands at the Windows command prompt or in PowerShell.

  - **Remotely** – Run commands on Amazon Elastic Compute Cloud (Amazon EC2) instances through a remote terminal program such as PuTTY or SSH, or with AWS Systems Manager.

Install Amazon CLI

URL: https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-install.html



```
C:\Users\USER>aws --version
aws-cli/2.2.46 Python/3.8.8 Windows/10 exe/AMD64 prompt/off
```

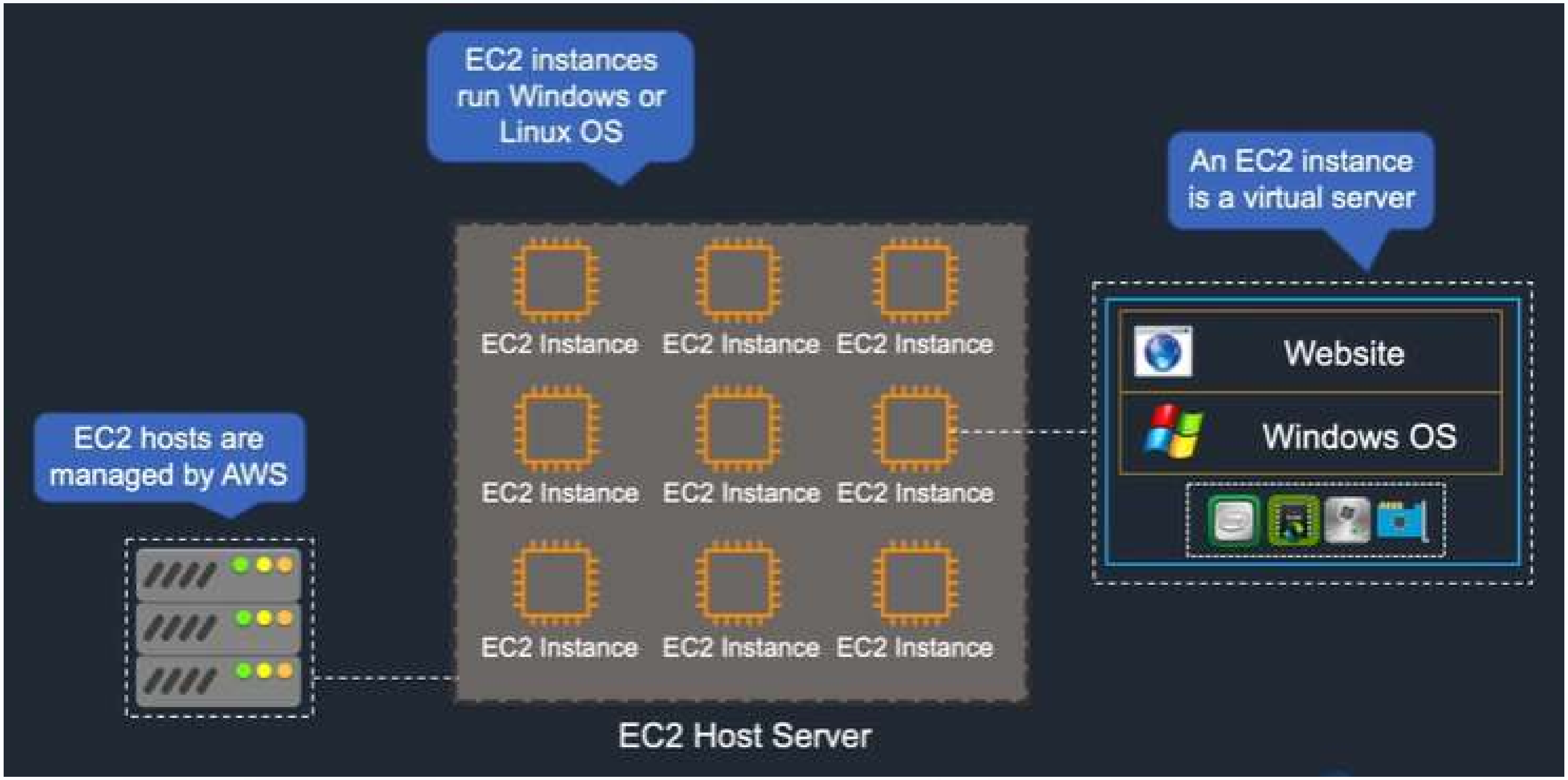
Amazon Elastic Cloud Compute (EC2)

Amazon EC2

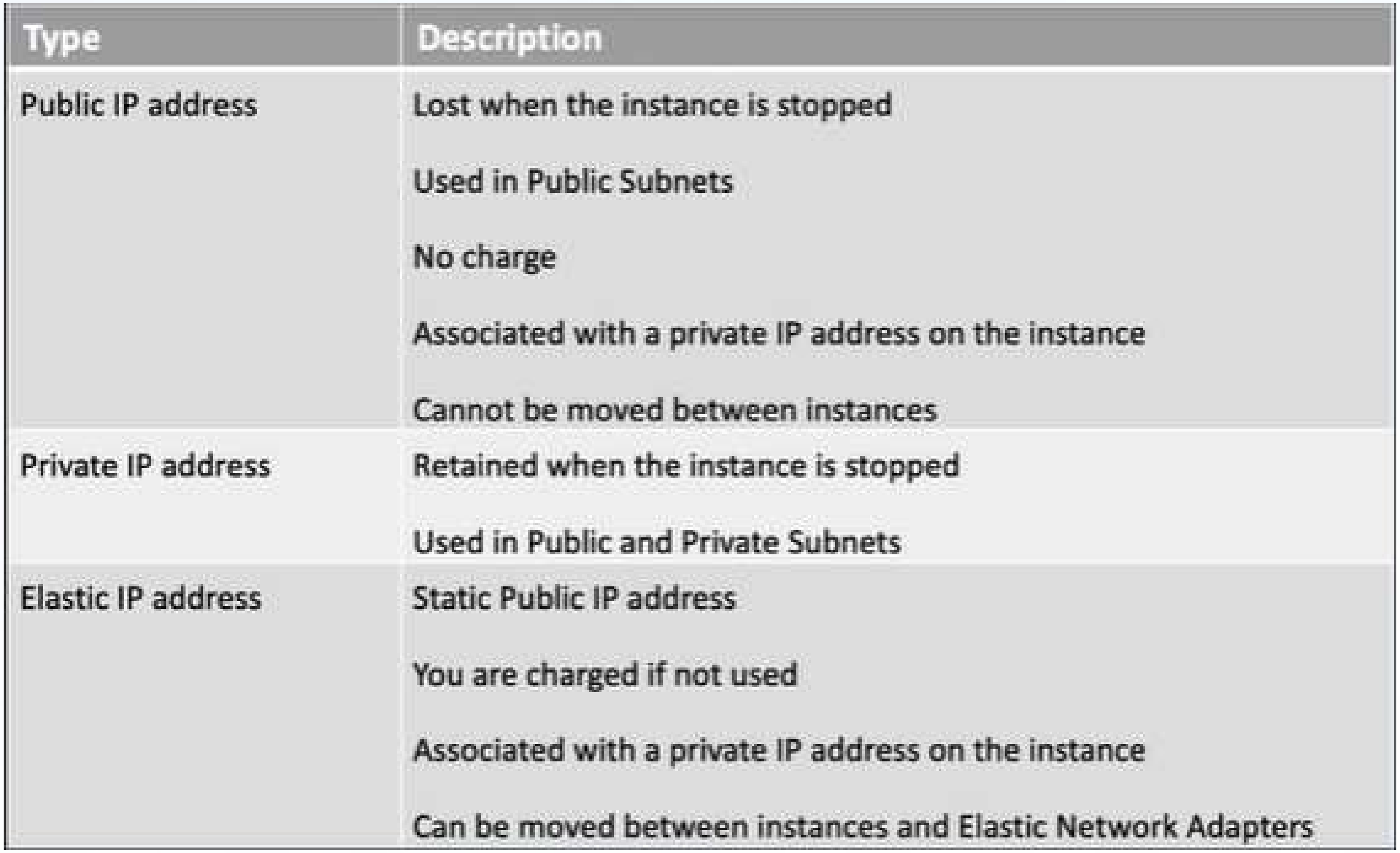
An EC2 instance is a virtual server on the cloud.

- Amazon EC2 is a web service that provides resizable compute capacity in the cloud.
- Amazon EC2 reduces the time required to obtain and boot new user instances by providing virtual machines in the cloud in minutes.
- You can scale the compute capacity up and down as per the computing requirement changes.
- Amazon EC2 allows you to pay for the capacity that you actually use.
- Amazon EC2 provides the developers with the tools to build resilient applications that isolate themselves from some common scenarios.

Amazon EC2

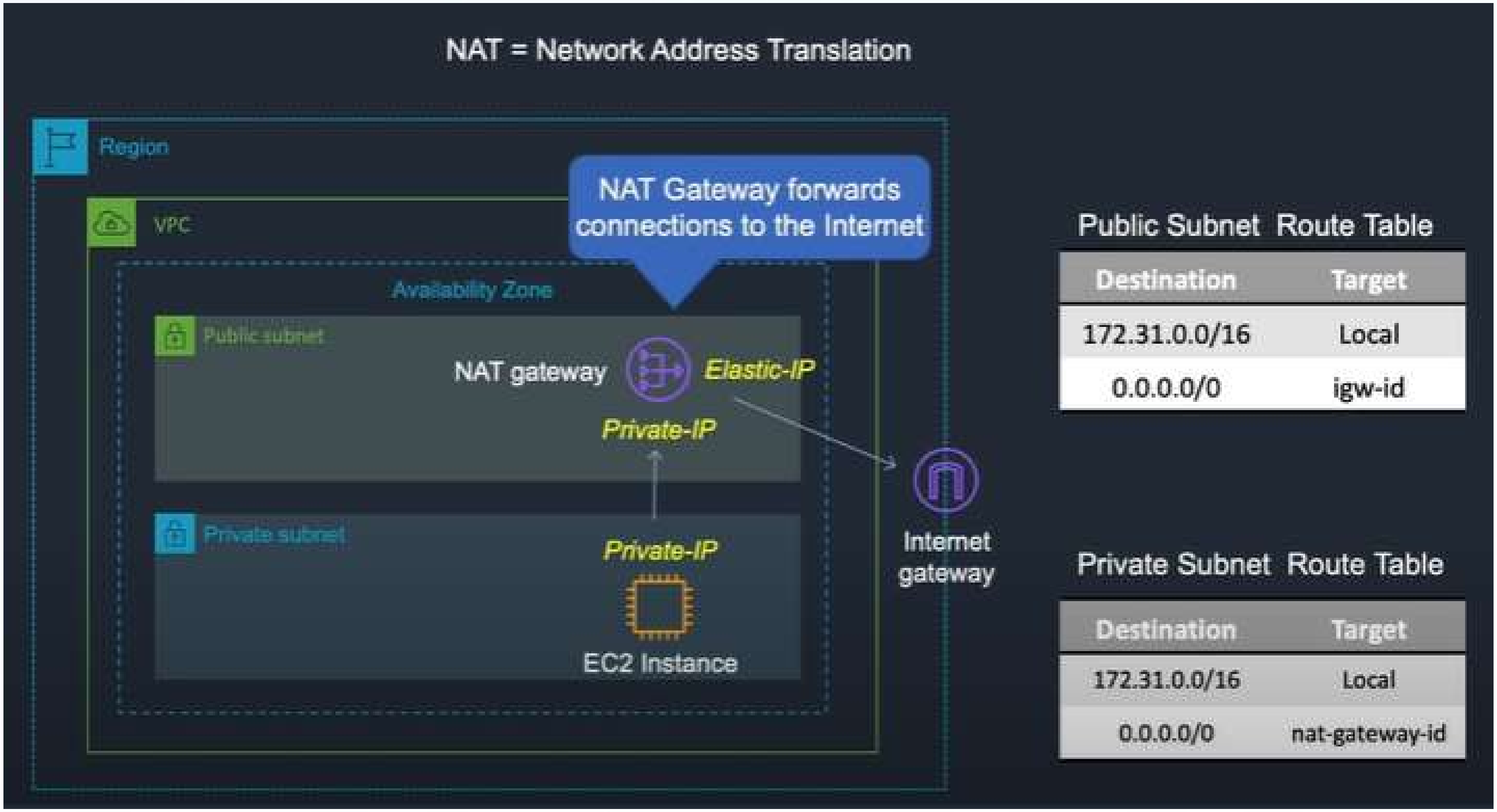

EC2 instances run Windows or Linux OS
An EC2 instance is a virtual server
EC2 Instance
EC2 Instance
EC2 Instance
EC2 Instance
EC2 Instance
EC2 Instance
EC2 Instance
EC2 Instance
EC2 Instance
Website
Windows OS
EC2 hosts are managed by AWS
EC2 Host Server

IP addresses for EC2 instances



Type	Description
Public IP address	Lost when the instance is stopped Used in Public Subnets No charge Associated with a private IP address on the instance Cannot be moved between instances
Private IP address	Retained when the instance is stopped Used in Public and Private Subnets
Elastic IP address	Static Public IP address You are charged if not used Associated with a private IP address on the instance Can be moved between instances and Elastic Network Adapters

Public & Private Subnets in EC2



NAT = Network Address Translation

Public Subnet Route Table

Destination	Target
172.31.0.0/16	Local
0.0.0.0/0	igw-id

Private Subnet Route Table

Destination	Target
172.31.0.0/16	Local
0.0.0.0/0	nat-gateway-id

Public & Private Subnets in EC2

- An EC2 instance in the public subnet has a public (non-static) IP address and can connect to the internet via a **Public subnet route table**.

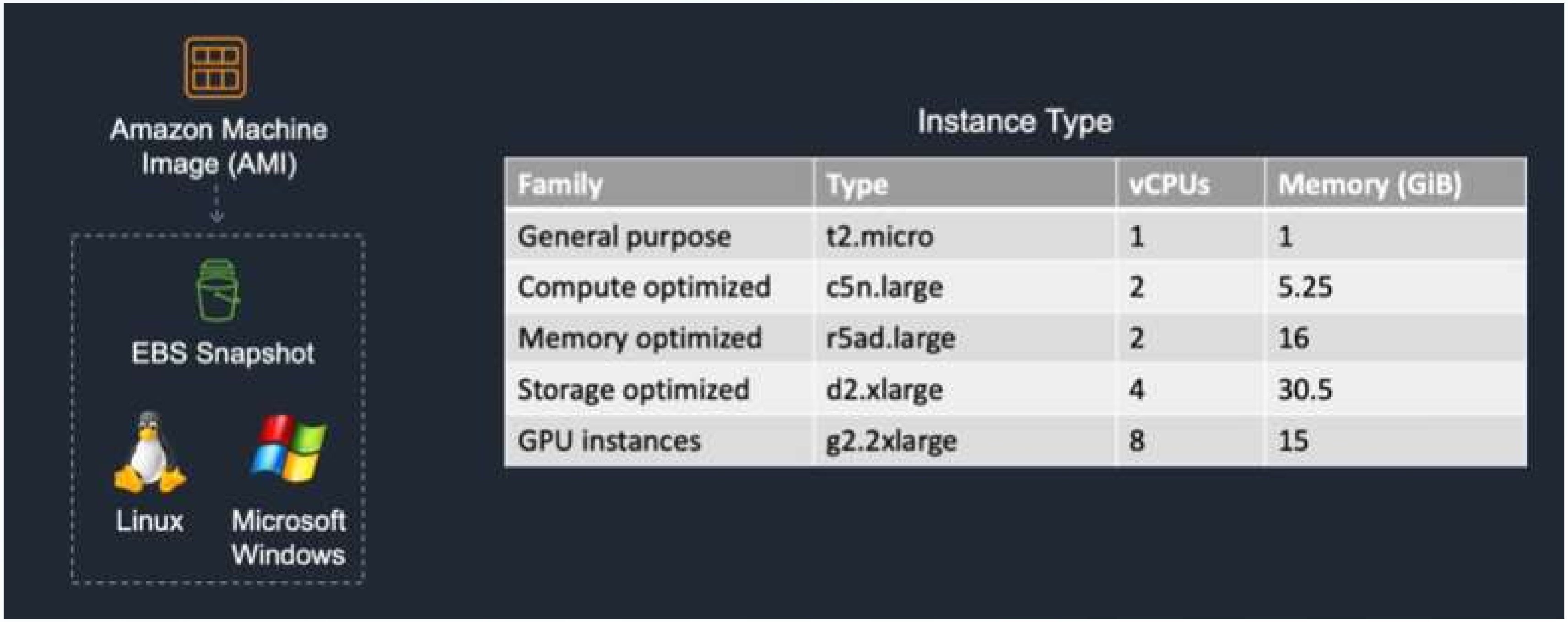
- An EC2 instance in the private subnet can only connect to other EC2 instances with the same subnet. If you want to connect an EC2 instance in the private subnet to the internet, you have to use **Private subnet route table** and implement a **NAT Gateway for an EC2 node in the public subnet.** The private EC2 instance can then connect to internet via the NAT gateway of the node in the public subnet.

EC2 Instance Types

Family	Type	vCPUs	Memory (GiB)
General Purpose	t2.micro	1	1
Compute Optimized	c5n.large	2	5.25
Storage Optimized	d2.xlarge	4	30.5
Memory Optimized	r5ad.large	2	16
Accelerated Computing Instances	g2.2xlarge	8	15

URL: https://www.amazonaws.cn/en/ec2/instance-types/

Launch an Amazon EC2 instance



Instance Type

Family	Type	vCPUs	Memory (GiB)
General purpose	t2.micro	1	1
Compute optimized	c5n.large	2	5.25
Memory optimized	r5ad.large	2	16
Storage optimized	d2.xlarge	4	30.5
GPU instances	g2.2xlarge	8	15

Launching an EC2 instance

- Select ‘Launch Instance’ from EC2 service main page

- Step 1: Choose an Amazon Machine Image (AMI)
  - NOTE: Make a note of ‘free tier eligible’ if you want that.

- Step 2: Choose an Instance Type (such as t2.micro)
  - NOTE: Make a note of ‘free tier eligible’ if you want that.

- Step 3: Configure Instance Details
  - Your instance will be created with a VPC network and a public IP

- Step 4: Add Storage
- Step 5: Add Tags
- Step 6: Configure Security Group

- Step 7: Review Instance Launch
  - Create a Key-pair.
    - A key-pair is used to access an instance. It consists of a public key and private key. The public-key is kept by AWS and ***the private key (.pem file) is to be downloaded and kept securely***. You need to provide this to authenticate an instance.

Connecting to an EC2 instance

To connect to an EC2 instance from Windows machine, we generally use PuTTY and PuTTYgen tools

Step 1: Create a private-key (.ppk file) using PuTTYgen tool

- Open the **.pem file** (private key downloaded while creating an EC2 instance) and **save it as a text file**.
- Open PuTTYgen and click Load button
- Select the above file (that is saved as text file)
- Click on **Save Private Key** button
- Save it as **.ppk file**

Connecting to an EC2 instance

To connect to an EC2 instance from Windows machine, we generally use PuTTY and PuTTYgen tools

Step 2: Connect to the EC2 instance using the private key generated in the previous step

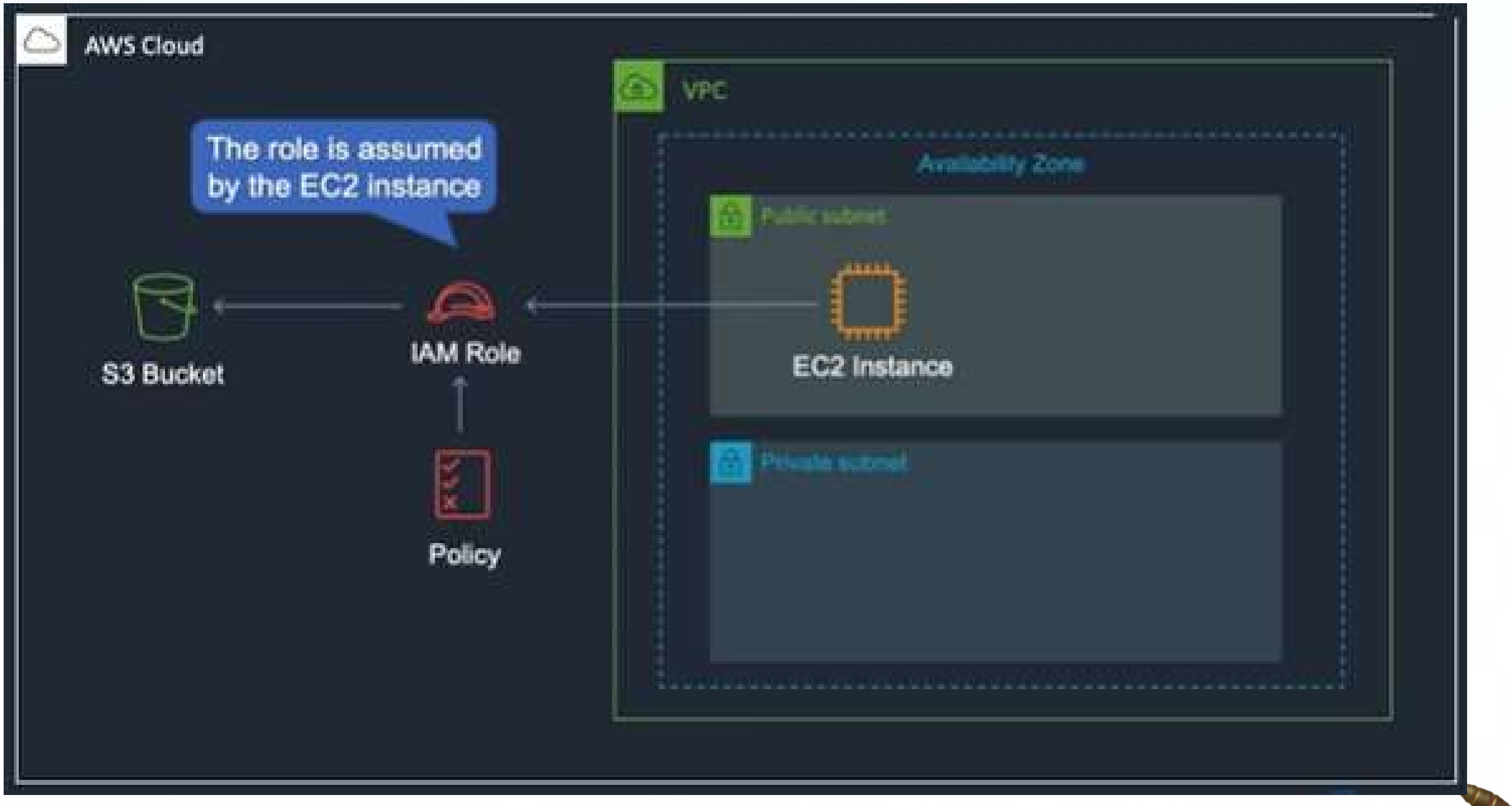
- Open PuTTY and paste the **Public IPv4 DNS** of your EC2 instance. You can copy this from the EC2 AWS Management Console as shown below: (use ec2-user as user name)

ec2-user@ec2-18-222-222-90.us-east-2.compute.amazonaws.com

- Go to Connection >> SSH >> Auth option.
- Select the private key you saved in step 1 and click Open
- This opens an EC2 instance terminal.

Adding an IAM role to an EC2 instance

Adding a proper IAM role to EC2 allows EC2 instance to have



Adding an IAM role to an EC2 instance

Adding an IAM role allows EC2 instances to call AWS services on your behalf.

Step 1: Create an IAM Role

- Go to IAM >> Roles page and click on Create Role button.
- Select EC2 as Use case and click on Next: Permissions button
- Type S3 in Filter Policies textbox.
- Select AmazonS3ReadOnlyAccess and click on Next: Tags
- Add any tags if you want and click on Next: Review
- Give a Role Name and Click on Create Role button.

Adding an IAM role to an EC2 instance

Adding an IAM role allows EC2 instances to call AWS services on your behalf.

Step 2: Attach the Role to an EC2 instance

- Select your EC2 instance
- Click ***Actions*** menu option
- Selection ***Security*** >> ***Modify IAM Role*** option
- Select the role created in the previous step and ***Save***

THANK YOU