

# INDEX

AMAZON EC2 (ELASTIC CLOUD COMPUTE) .....	2
ABOUT EC2: .....	2
BENEFITS OF EC2: .....	2
EC2 INSTANCE FAMILIES AND NAMES:.....	2
EC2 PRICING:.....	2
VIRTUAL MACHINES (EC2) v/s PHYSICAL SERVERS:.....	3
STEPS TO CREATE AN EC2 INSTANCE:.....	3
DOCKER v/s VIRTUAL MACHINE .....	5
ABOUT DOCKER: .....	5
ABOUT VIRTUAL MACHINE: .....	5
KEY DIFFERENCES BETWEEN DOCKER AND VIRTUAL MACHINE:.....	5

## AMAZON EC2 (ELASTIC CLOUD COMPUTE)

### ABOUT EC2:

- 1) Resizable compute capacity.
- 2) Complete control of your computing resources.
- 3) Reduced time required to obtain and boot new server instances.
- 4) Provides pay-as-you-go pricing and a broad selection of hardware and software:
  - ➔ Use Amazon Machine Images (AMIs).
  - ➔ Add or terminate instances as needed.
  - ➔ Pause and resume your instances.
- 5) Template for:
  - ➔ Storage volumes.
  - ➔ Launch permissions.
  - ➔ A block device mapping.
- 6) Examples: Application Server, Web Server, Database Server, Game Server, Mail Server, Media Server, Catalog Server, File Server.

### BENEFITS OF EC2:

- 1) Elasticity
- 2) Control
- 3) Flexibility
- 4) Integrated
- 5) Reliable
- 6) Secure
- 7) Inexpensive
- 8) Easy

### EC2 INSTANCE FAMILIES AND NAMES:

- Choosing the correct type of instance is very important for efficient use of instances and cost reduction.

INSTANCE FAMILY	USE CASES
<b>General Purpose</b> - A1, T3, T3a, T2, M6g, M5	✓ Low-traffic websites and web applications. ✓ Small databases and midsize databases.
<b>Compute Optimized</b> - C5, C5n, C4, C7g	✓ High-performance web servers. ✓ Video encoding.
<b>Memory Optimized</b> - R5, R5n, X1e, X1, z1d	✓ High-performance databases. ✓ Distributed memory caches.
<b>Storage Optimized</b> - I3, I3en, D2, H1	✓ Data warehousing. ✓ Log or data processing applications.
<b>Accelerated Computing</b> - P3, P2, Inf1, G4, G3, F1	✓ 3D visualizations. ✓ Machine learning.

### EC2 PRICING:

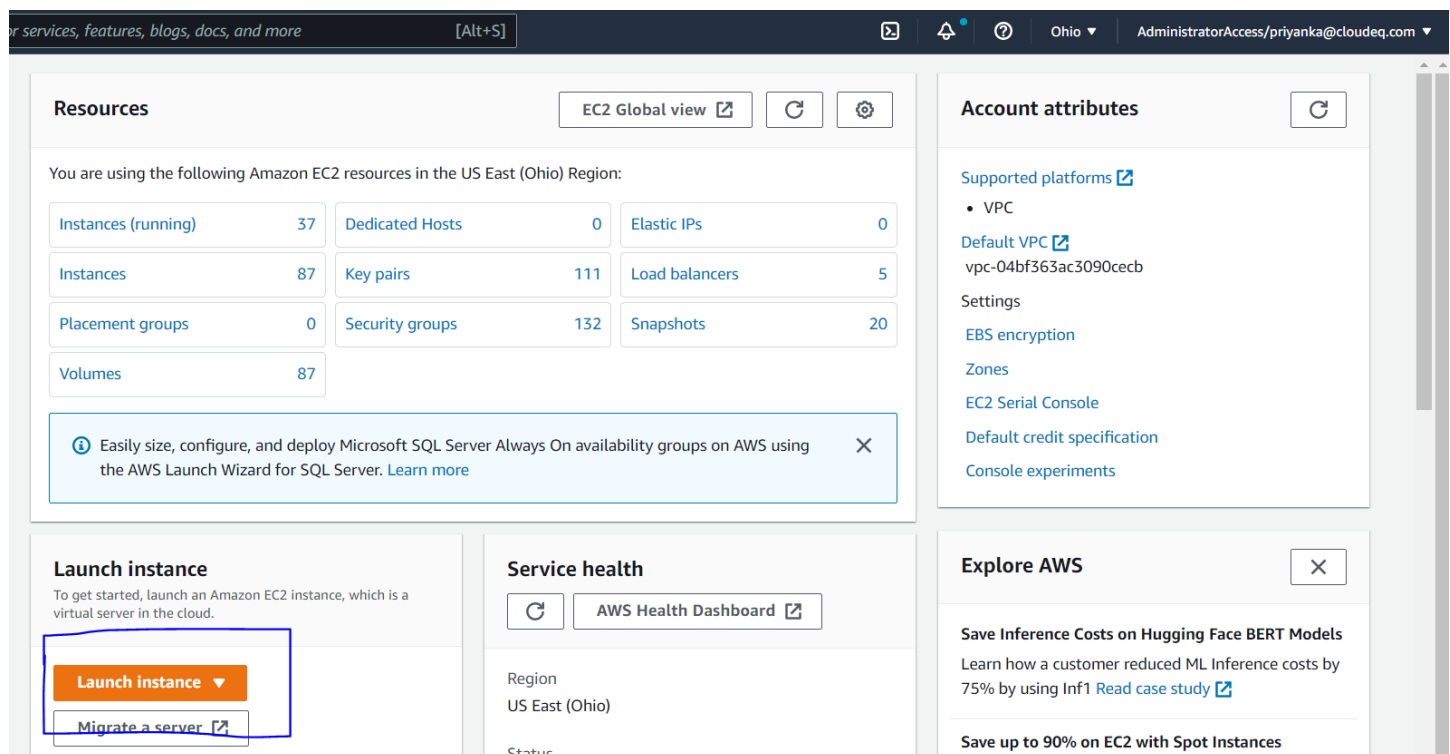
- Per-second billing for supported instances.

## VIRTUAL MACHINES (EC2) v/s PHYSICAL SERVERS:

- EC2 can solve some problems that are more difficult with an on-premises server:
  - ➔ Data-driven decisions.
  - ➔ Quick iterations.
  - ➔ Free to make mistakes.

## STEPS TO CREATE AN EC2 INSTANCE:

- ➔ Login to console.
- ➔ Search for EC2 and open it.
- ➔ Click launch instance to launch the instance.



- ➔ Give the details of the type of instance you want:
  - Name of instance.
  - The AMI you want.
  - Instance Type.
  - Choose or create your key-pair.
  - Set-up your security group or choose the default one.
  - Configure the storage.
  - Add advanced details (if you have any).
- ➔ Check the summary of your instance.

➔ Again, click launch instance to finally launch it.

Services Search for services, features, blogs, docs, and more [Alt+S] Ohio AdministratorAccess/priyanka@cloudeq.com

☐ Allow HTTP traffic from the internet  
To set up an endpoint, for example when creating a web server

☐ Allow HTTP traffic from the internet  
To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

▼ **Configure storage** Info Advanced

1x 8 GiB gp2 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

0 x File systems Edit

► **Advanced details** Info

▼ **Summary**

Number of instances Info  
1

Software Image (AMI)  
Amazon Linux 2 Kernel 5.10 AMI...read more  
ami-0f924dc71d44d23e2

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 3 million IP addresses and

Cancel Launch instance

➔ Instance is successfully launched.

Services Search for services, features, blogs, docs, and more [Alt+S] Ohio AdministratorAccess/priyanka@cloudeq.com

EC2 > Instances > Launch an instance

✓ **Success**  
Successfully initiated launch of instance (i-06b4fc52f8d3f83fb)

► Launch log

for services, features, blogs, docs, and more [Alt+S] Ohio AdministratorAccess/priyanka@cloudeq.com

EC2 > Instances > i-06b4fc52f8d3f83fb

**Instance summary for i-06b4fc52f8d3f83fb (practice-ins)** Info

Updated less than a minute ago

Instance ID  
i-06b4fc52f8d3f83fb (practice-ins)

IPv6 address  
-

Hostname type  
IP name: ip-172-31-21-72.us-east-2.compute.internal

Answer private resource DNS name  
IPv4 (A)

Auto-assigned IP address  
3.142.49.55 [Public IP]

IAM Role  
-

Public IPv4 address  
3.142.49.55 | open address

Instance state  
Running

Private IP DNS name (IPv4 only)  
ip-172-31-21-72.us-east-2.compute.internal

Instance type  
t2.micro

VPC ID  
vpc-04bf363ac3090cecb (vpc)

Subnet ID  
subnet-079a71562c143535b

Private IPv4 addresses  
172.31.21.72

Public IPv4 DNS  
ec2-3-142-49-55.us-east-2.compute.amazonaws.com | open address

Elastic IP addresses  
-

AWS Compute Optimizer finding  
Opt-in to AWS Compute Optimizer for recommendation s.  
| Learn more

Auto Scaling Group name  
-

Connect Instance state Actions

## DOCKER v/s VIRTUAL MACHINE

### ABOUT DOCKER:

- 1) Virtualization software that helps in developing, deploying, monitoring and running applications in a Docker Container with all their dependencies.
- 2) Docker Container includes all dependencies (framework, libraries etc.) to run an application in an efficient and bug-free manner.
- 3) Benefits:
  - ➔ Light-Weight.
  - ➔ Applications run in isolation.
  - ➔ Occupies less space.
  - ➔ Easily portable and highly secure.
  - ➔ Short boot-up time.

### ABOUT VIRTUAL MACHINE:

- Computing environment or software that aids developers to access an operating system via a physical machine.

### KEY DIFFERENCES BETWEEN DOCKER AND VIRTUAL MACHINE:

DOCKER	VIRTUAL MACHINE
Container-based model where containers are software packages used for executing an application on any operating system.	Not container-based model instead uses user space along with the kernel space of an OS.
Share the host kernel i.e., execution engine.	Doesn't share the host kernel i.e., uses hypervisor.
Multiple workloads can run on a single OS.	Each workload needs a complete OS or hypervisor.
High-performance as they use the same operating system with no additional software (like hypervisor)	VM uses a separate OS; it causes more resources to be used.
Start-up quickly and result in less boot-up time.	Don't start quickly and lead to poor performance.
Users can create application and store it into a container image and then can run it across any host environment.	Portability issues as it doesn't have central hub and requires more memory to store data.
Lightweight (KB/ MB).	Few GBs.
