

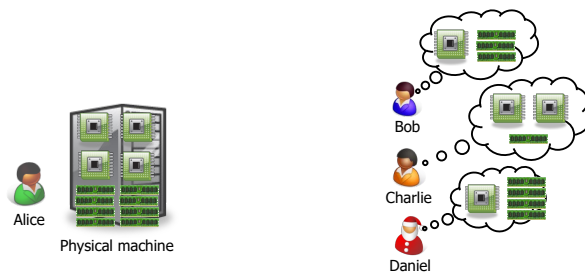
Week 2 Amazon Web Services Introduction

Dr Zhi Zhang

Overview

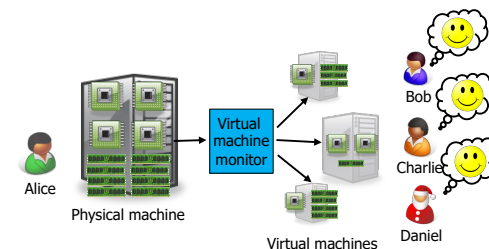
- Brief introduction to virtualization
- AWS introduction

Virtualization: the core technique of cloud computing



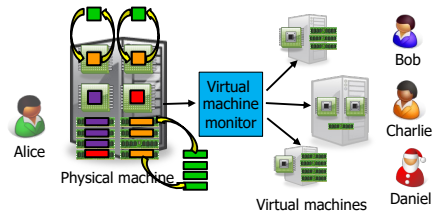
- Suppose Alice has a machine with 16 CPUs and 64 GB of memory. There are three customers:
 - Bob wants a machine with 8 CPUs and 32 GB of memory
 - Charlie wants a machine with 8 CPUs and 32 GB of memory
 - Daniel wants a machine with 4 CPUs and 8 GB of memory
- What should Alice do?

Virtualization: the core technique of cloud computing



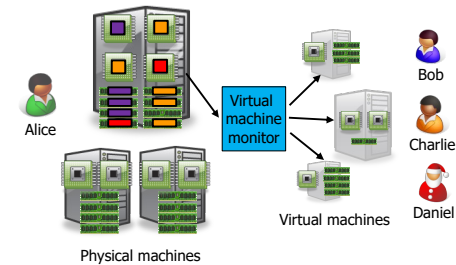
- Alice installs a virtual machine monitor (VMM) and provides each customer a **guest OS/virtual machine (VM)** with the requested resources
 - Bob is assigned with 8 vCPU and 32 GB of guest physical memory
 - Charlie is assigned with 8 vCPUs and 32 GB of guest physical memory
 - Daniel is assigned with 4 CPUs and 8 GB of guest physical memory
 - From each customer's perspective, it appears as if they owned the physical machine all by themselves (indicating VM isolation)
- Obviously, more resources are provided by virtualization.

Time sharing



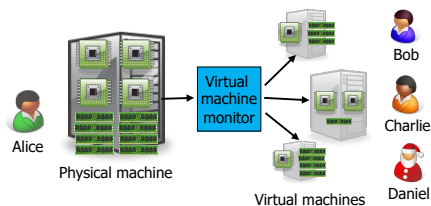
- VMs **time-share** existing hardware resources (a time-multiplexed manner)
- What if Alice serves more customers?

VM live migration



- What if the machine needs to be shut down for maintenance?
 - Alice can **migrate** the running VMs to different physical machines without interrupting any customers.

VM isolation

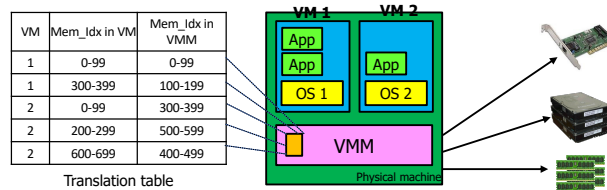


- One customer cannot access data of another customer

The benefits of virtualization

- Flexible to cloud providers, who
 - can generate VMs with different resource capabilities
 - can migrate VMs if necessary (e.g., for maintenance)
 - can increase load by overcommitting resources
- Secure to VMs
 - One VM cannot access another without permission
- Convenient to customers
 - Complete control over the virtual 'hardware' (can install their own operating systems, applications, ...)
- **But there might be performance impacts**
 - Load changes in one VM may affect the performance of another VM on the same physical machine

How is virtualization implemented?



- Resources (CPU, memory, and I/O devices) are virtualized
 - VMM ("Hypervisor") has translation tables that map virtual resources to physical resources
 - Example: VM 1 accesses memory cell #300; VMM maps this to memory cell #100.

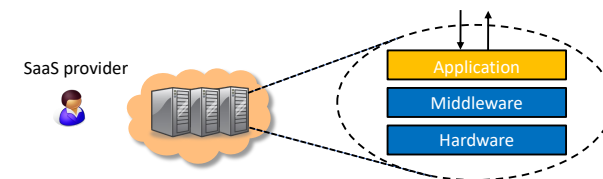
Amazon: the pioneer

- Jul 2002: Amazon Web Services launched
- Mar 2006: Amazon S3 (Simple Storage Service) launched
 - Proposed the 'pay-per-use' business model, which is now the standard in cloud computing
- Aug 2006: Amazon EC2 (Elastic Compute Cloud) launched
- Nov 2009: Microsoft Azure Beta launched
 - Microsoft's online services are gradually transitioning to Azure
- Dec 2013: Google Compute Engine launched

What services does a cloud provide?

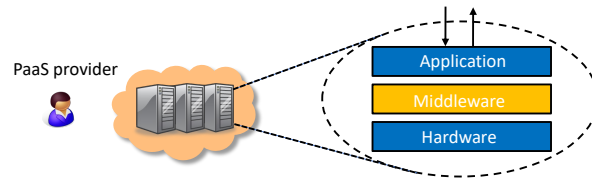
- Three common types of cloud services:
 - Software as a service (SaaS)**
 - Analogy: Restaurant. Prepares & Serves an entire meal...
 - Platform as a service (PaaS)**
 - Analogy: Take-out food. Prepares meal, but does not serve it.
 - Infrastructure as a service (IaaS)**
 - Analogy: Grocery store. Provides raw ingredients.
- Other **xaaS (anything as a service)** types have been defined, but are less common
 - Desktop, Backend,, Network, Monitoring, ...

Software as a Service (SaaS)



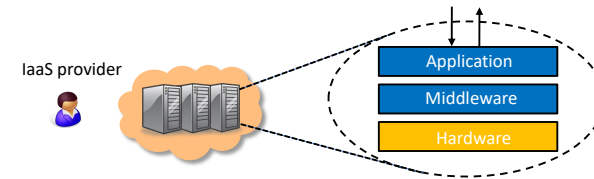
- Cloud provides an entire application
 - Email, Drive, Doc, Slides,...
 - Example: Google Workspace

Platform as a Service (PaaS)



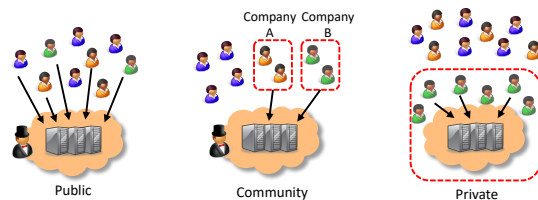
- Cloud provides middleware
 - Common Language Runtime environment
 - Example: Heroku

Infrastructure as a Service (IaaS)



- Cloud provides hardware resources
 - Virtual Machines, Virtual Memory, Virtual Disk,...
 - Examples: AWS (Amazon Web Services)

Public/community/Private clouds

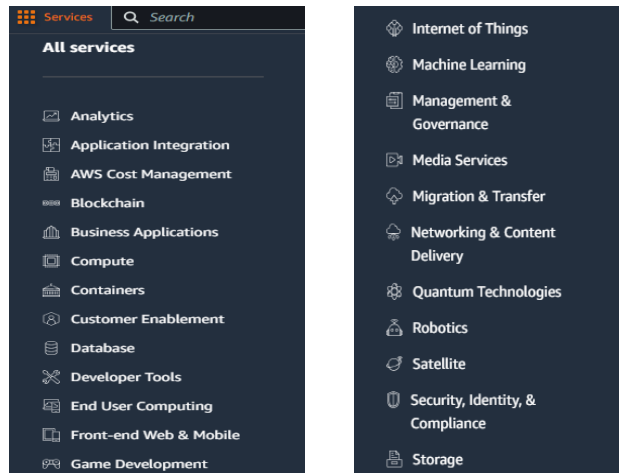


- Targeting different customers:
 - **Public cloud:** open to general public.
 - Example: AWS, Microsoft Azure, Google Compute Engine
 - **Community cloud:** shared by multiple organizations with common interests and requirements.
 - Example: Healthcare Sectors
 - **Private cloud:** exclusively used by a single organization.
 - Example: Google's internal datacenter

Practice Questions

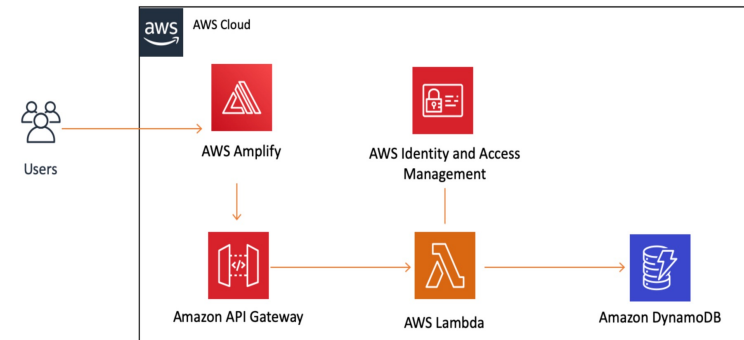
- [7 marks] Q1: What is service (XaaS) cloud computing? Describe the different categories of XaaS cloud computing can provide with specific examples of each service.
- [1 mark] Cloud computing offers a range of services, referred to as "XaaS," which stands for "Anything as a Service."
- [2 marks] Infrastructure as a Service that provides virtualized computing resources. An example is AWS.
- [2 marks] Platform as a Service that provides a runtime environment for developers to build, deploy, and manage applications. An examples is Google App Engine or Heroku.
- [2 marks] Software as a Service that delivers fully functional software applications. An example is Microsoft 365 office or Google workplace.

AWS: the leading cloud providing a wide range of services



AWS: the leading cloud providing a wide range of services

- The services satisfy different needs of customers:
 - Allowing us to design, deploy, and manage applications on the cloud, so-called application architecture.
 - e.g., build a web application



What is an application architecture?

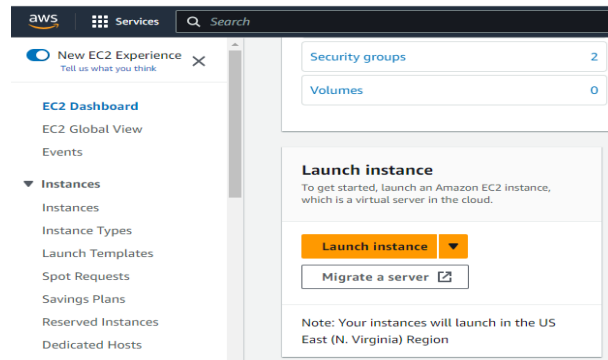
- It is the structural design and organization of a software application.
- Main types of application architectures
 - **Monolithic architecture**: combines the frontend, backend, and database functionalities into a single application.
 - e.g., wordpress: a content management system (CMS)
 - **Client/Server architecture**: an application is split into two main components: the client, which handles the user requests, and the server, which handles the processing of requests and data storage.
 - e.g., gmail
 - **Three-tier architecture**: a presentation tier (frontend), application logic tier (backend), and a database tier (backend).
 - e.g., Shopify: an e-commerce platform.
 - **Cloud-based serverless architecture**: known as **Function-as-a-Service**, which is a cloud computing model where the cloud provider provisions, maintain and scale applications.
 - e.g., AWS Lambda

Deploying application architecture

- When an architecture is ready, it can be deployed onto AWS via different services:
 - **Elastic Compute Cloud (EC2)**: an EC2 instance is a virtual machine that runs a Windows or Linux server. It is essentially running on Virtual Machine Monitor (VMM).
 - **Elastic Container Service (ECS)**: a customized environment provided by an EC2 instance.

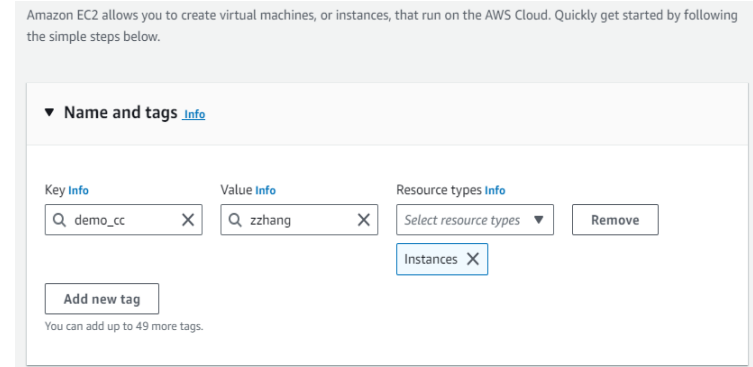
AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - Sign in to the AWS Management Console:
 - <https://aws.amazon.com/> or <https://489389878001.signin.aws.amazon.com/console>
 - Navigate to EC2 Dashboard and Click “Launch instance”



AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Name and tags”: In this step, we can add names and tags to our instance. Tags are key-value pairs that help identify and categorize instances for better management, e.g., Key: demo_cloudcompute, Value: MyEC2Instance. We can click the "Add new tag" button to add multiple tags as needed.

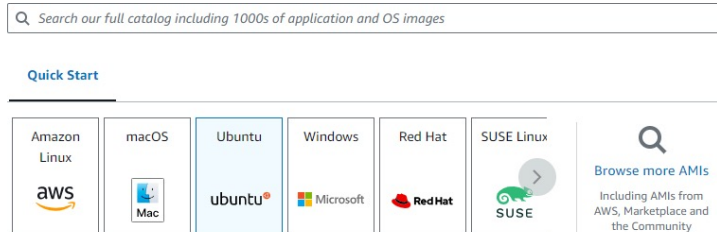


AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Application and OS Images (Amazon Machine Image)” : In this step, we select an AMI that suits our requirements. We can choose from various operating systems and pre-configured applications. The definition of an AMI is as follows:

▼ Application and OS Images (Amazon Machine Image) [Info](#)

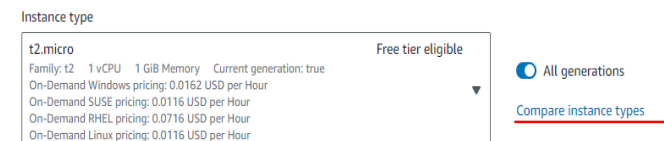
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below



AWS EC2


- Steps of creating, configuring and launching an EC2 instance :
 - “Instance type”: In this step, we select an appropriate instance type that meets our computing, memory, networking or storage needs.


▼ Instance type [Info](#)



AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Instance type”
What does Architecture mean?

Instance types (1/651) 

< 1 2 3 4 5 6 7 ... 14 > 

	Instance type ▾	vCPUs ▾	Architecture ▾	Memory (GiB) ▾	Storage (GB) ▾	Storage type ▾
<input type="radio"/>	t1.micro	1	i386, x86_64	0.612	-	-
<input type="radio"/>	t2.nano	1	i386, x86_64	0.5	-	-
<input checked="" type="radio"/>	t2.micro	1	i386, x86_64	1	-	-
<input type="radio"/>	t2.small	1	i386, x86_64	2	-	-
<input type="radio"/>	t2.medium	2	i386, x86_64	4	-	-
<input type="radio"/>	t2.large	2	x86_64	8	-	-
<input type="radio"/>	t2.xlarge	4	x86_64	16	-	-

Cancel Select instance type

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Key pair (login)” : In this step, we can select/create a key pair (SSH key) for secure access to our first instance. If we currently don't have a key pair, we can create one.

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

 [Create new key pair](#)

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Key pair (login)” :

Key pair name

Key pairs allow you to connect to your instance securely.

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type


☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair

Private key file format

☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

 When prompted, store the private key in a secure and accessible location on

Cancel

Create key pair

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Network settings” : In this step, we configure the network based on the application of our instance, e.g., connect to the instance via SSH or RDP, create security group, etc.

▼ Network settings [Info](#)

Edit

Network [Info](#)

vpc-0c9d371751c819927

Subnet [Info](#)

No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

We'll create a new security group called 'launch-wizard-1' with the following rules:

☒ Allow SSH traffic from

Helps you connect to your instance

0.0.0.0/0

☒ Allow HTTPS traffic from the internet

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Network settings”

▼ Network settings

Info

VPC - required

Info

ypc-0c9d371751c819927

172.31.0.0/16

(default) ▼

↻

Subnet

Info

No preference

▼

↻ Create new subnet

Auto-assign public IP

Info

Enable

▼

Firewall (security groups)

Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Network settings”
 - Create security group: a security group is a set of firewall rules that control the network traffic for our instance. We can add rules to allow specific traffic to reach our instance.

Security group name - *required*

demo-secgroup

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-./@#,%&[]!\$*

Description - *required*

demo-secgroup created for test 2023-07-26T03:16:33.988Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type *Info*

ssh

Protocol *Info*

TCP

Port range *Info*

22

Source type *Info*

Anywhere

Source *Info*

Q

Add CIDR, prefix list or security group

0.0.0.0/0

×

Description - *optional*

e.g. SSH for admin desktop

Remove

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Network settings”
 - Create security group

▼ Security group rule 2 (TCP, 443, 0.0.0.0/0)

Type [Info](#)
HTTPS

Protocol [Info](#)
TCP

Port range [Info](#)
443

Source type [Info](#)
Anywhere

Source [Info](#)

Description - optional [Info](#)

▼ Security group rule 3 (TCP, 80, 0.0.0.0/0)

Type [Info](#)
HTTP

Protocol [Info](#)
TCP

Port range [Info](#)
80

Source type [Info](#)
Anywhere

Source [Info](#)

Description - optional [Info](#)

⚠ 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.

[Add security group rule](#)

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Configure storage”: In this step, we specify storage volume for our instance. By default, one Elastic Block Store (EBS) volume is attached to the instance. We can add more EBS volumes if needed.

▼ Configure storage [Info](#)

Advanced

1x GiB ▼ 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

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

0 x File systems

Edit

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Configure storage”
- What is EBS about?
 - provides block level storage volumes for use with EC2 instances.
 - EBS storage is allocated in **volumes**. A volume is a 'virtual disk' (size from 1GB to 1TB)
 - Can be attached to any instance in the same Availability Zone
 - We pay for what we use with EBS.

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Configure storage”

EBS features

- **Persistent Storage:** EBS volumes persist independently from EC2 instances, indicating that when an instance is terminated, the data stored on the attached EBS volume CAN be retained.
- **High durability and availability:** EBS replicates volumes data across multiple physical servers within the same availability zone to prevent the loss of data from hardware failures.
- **Snapshots:** allows users to create point-in-time snapshots of their volumes. These snapshots serve as backups and can be used to restore volumes to a previous state.
- **Elastic volumes :** allows users to dynamically adapt their volumes when the needs of their applications change.

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Configure storage”
 - EBS setting

▼ Volume 1 (AMI Root) (Custom)

Storage type Info EBS	Device name - <i>required</i> Info /dev/sda1	Snapshot Info snap-0d3283808e9f92122
Size (GiB) Info <input type="text" value="8"/>	Volume type Info <input type="text" value="gp2"/>	IOPS Info 100 / 3000
Delete on termination Info <input type="text" value="Yes"/>	Encrypted Info <input type="text" value="Not encrypted"/>	KMS key Info <input type="text" value="Select"/>

KMS keys are only applicable when encryption is set on this volume.

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

Add new volume

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Configure storage”
 - EBS setting

▼ Volume 2 (Custom) Remove

Storage type Info EBS	Device name - <i>required</i> Info <input type="text" value="/dev/sdb"/>	Snapshot Info <input type="text" value="Select"/>
Size (GiB) Info <input type="text" value="8"/>	Volume type Info <input type="text" value="gp3"/>	IOPS Info <input type="text" value="3000"/>
Delete on termination Info <input type="text" value="No"/>	Encrypted Info <input type="text" value="Not encrypted"/>	KMS key Info <input type="text" value="Select"/>

KMS keys are only applicable when encryption is set on this volume.

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Advanced details”

▼ Advanced details [Info](#)

Purchasing option [Info](#)

☐ Request Spot Instances

Domain join directory [Info](#)

Select

Create new directory

IAM instance profile [Info](#)

Select

Create new **IAM** profile

Hostname type [Info](#)

IP name

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Advanced details”
 - Identity and Access Management (IAM) : It is a service that allows us to manage users, groups, and permissions to securely control access to AWS resources, e.g., root user and IAM user (according to the principle of least privilege).

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](#)

What is the principle of least privilege?

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Advanced details”
 - Instance purchasing options

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

- **On-Demand Instances** – Pay, by the second, for the instances that you launch.
- **Savings Plans** – Reduce your Amazon EC2 costs by making a commitment to a consistent amount of usage, in USD per hour, for a term of 1 or 3 years.
- **Reserved Instances** – Reduce your Amazon EC2 costs by making a commitment to a consistent instance configuration, including instance type and Region, for a term of 1 or 3 years.
- **Spot Instances** – Request unused EC2 instances, which can reduce your Amazon EC2 costs significantly.
- **Dedicated Hosts** – Pay for a physical host that is fully dedicated to running your instances, and bring your existing per-socket, per-core, or per-VM software licenses to reduce costs.
- **Dedicated Instances** – Pay, by the hour, for instances that run on single-tenant hardware.
- **Capacity Reservations** – Reserve capacity for your EC2 instances in a specific Availability Zone for any duration.

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Advanced details”
 - Identity and Access Management (IAM) : It is a service that allows us to manage users, groups, and permissions to securely control access to AWS resources, e.g., root user and IAM user (according to the principle of least privilege).

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](#)

What is the principle of least privilege?

A user only has those privileges which are essentially vital to perform their intended operations. Thus, root user in AWS can be regarded as the system administrator in a local OS. IAM user is a normal user in the OS.

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Advanced details”

Instance auto-recovery [Info](#)
Select ▼

Shutdown behavior [Info](#)
Stop ▼

Stop - Hibernate behavior [Info](#)
Select ▼

Termination protection [Info](#)
Select ▼

Stop protection [Info](#)
Select ▼

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Review and Launch”

▼ Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)
ami-053b0d53c279acc90

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Cancel [Launch instance](#)

AWS EC2

- Steps of creating, configuring and launching an EC2 instance :
 - “Review and Launch”

EC2 > Instances > Launch an instance

✓ **Success**
Successfully initiated launch of instance (i-0cb2a733e648d7158)

▼ Launch log

Initializing requests	✓ Succeeded
Creating security groups	✓ Succeeded
Creating security group rules	✓ Succeeded
Launch initiation	✓ Succeeded

AWS EC2

- Connecting to an EC2 instance :

Connect to your instance

Once your instance is running, log into it from your local computer.

[Connect to instance](#)

[Learn more](#)

aws Services Search [Alt+S]

```
+ Documentation: https://help.ubuntu.com
+ Management:   https://landscape.canonical.com
+ Support:       https://ubuntu.com/advantage

System information as of Wed Jul 26 06:02:45 UTC 2023

System load: 0.8798828125   Processes:            102
Usage of /:  20.6% of 7.57GB Users logged in:        0
Memory usage: 25%          IPv4 address for eth0: 172.31.22.142
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

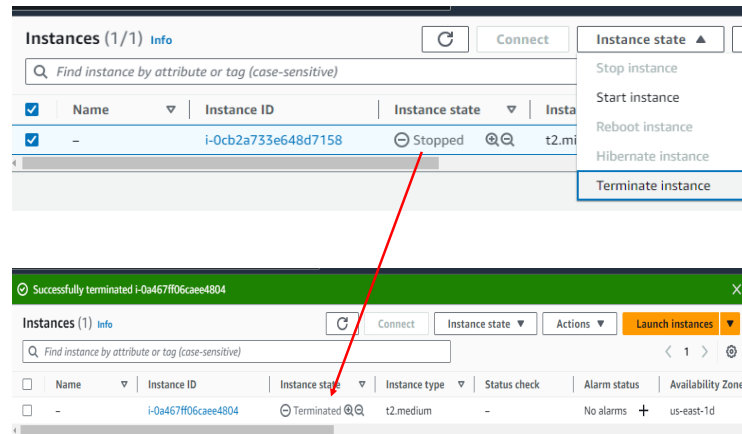
0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

i-0cb2a733e648d7158
PublicIPs: 3.85.202.9 PrivateIPs: 172.31.22.142
```

AWS EC2

- Terminating an EC2 instance :



Demo

- Create, configure and launch an instance
- Connect to an instance
- Terminate an instance

Geographical location of an instance

- AWS region:
 - There are many different geographical locations for AWS data centers around the world, known as “AWS Regions”, identified by names such as “us-west-2” (US West-Oregon).

US East (N. Virginia)	us-east-1
US East (Ohio)	us-east-2
US West (N. California)	us-west-1
US West (Oregon)	us-west-2
Asia Pacific (Mumbai)	ap-south-1
Asia Pacific (Osaka)	ap-northeast-3
Asia Pacific (Seoul)	ap-northeast-2
Asia Pacific (Singapore)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2
Asia Pacific (Tokyo)	ap-northeast-1

Geographical location of an instance

- Availability zone:
 - Each region is further divided into multiple availability zones, which are identified by additional letters such as “us-west-2a” and “us-west-2b”.
 - Each zone represents a separate data center in different physical locations within a region.
 - Each zone within a region is networked with low-latency connections

Availability Zones	
All Availability Zones for AWS Region, US East (N. Virginia).	
US East (N. Virginia) - us-east-1 Info	
Availability Zones	Status
us-east-1a (use1-az4)	Enabled by default
us-east-1b (use1-az6)	Enabled by default
us-east-1c (use1-az1)	Enabled by default
us-east-1d (use1-az2)	Enabled by default
us-east-1e (use1-az3)	Enabled by default
us-east-1f (use1-az5)	Enabled by default

Practice Questions

- [10 marks] Q2: Describe the steps which you would take on AWS and the decisions that would need to be made to create, configure and launch an EC2 Instance.
- [1 mark] Sign in to the AWS management console
- [1 mark] Navigate to EC2 Dashboard and click “Launch instance”
- [1 mark] Select/Create names and tags
- [2 marks] Select AMI and Instance type
- [1 mark] Select/Create Key pair for login
- [3 marks] Configure network settings, storage and more advanced details (e.g., specify some self-defined behaviors)
- [1 mark] Review and launch

Practice Questions

- [7 marks] Q3: Describe EBS and what features it offers.
- [1 mark] A brief introduction to EBS.
- [2 marks] One feature and its brief description.
- [2 marks] One feature and its brief description.
- [2 marks] One feature and its brief description.