

Practical 1: Infrastructure as a Service using AWS

NAME: Vedant Bohat

SAP ID: 86092300045

ROLL NO: A044

Cloud Computing Architecture:

As we know, cloud computing technology is used by both small and large organizations to store the information in cloud and access it from anywhere at any time using the internet connection.

Cloud computing architecture is a combination of service-oriented architecture and event-driven architecture.

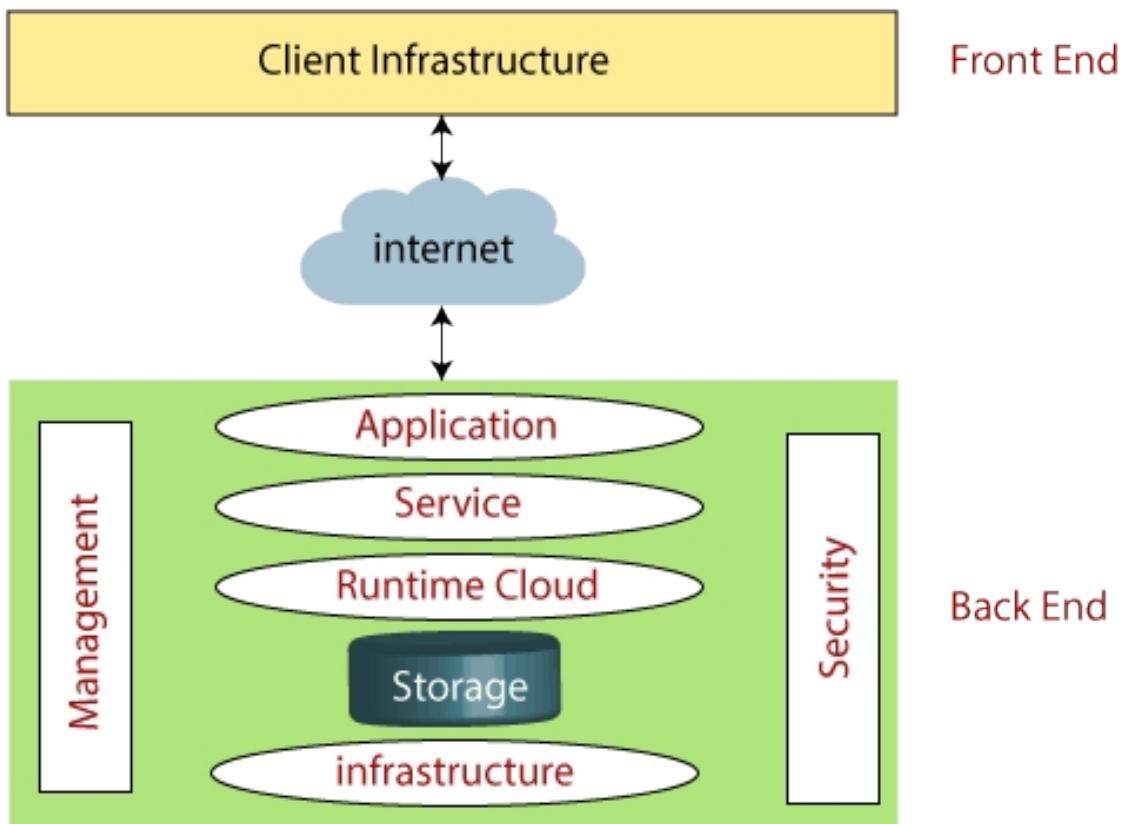
Cloud computing architecture is divided into the following two parts -

Front End

Back End

The below diagram shows the architecture of cloud computing -

Architecture of Cloud Computing



Front End

The front end is used by the client. It contains client-side interfaces and applications that are required to access the cloud computing platforms. The front end includes web servers

(including Chrome, Firefox, internet explorer, etc.), thin & fat clients, tablets, and mobile devices.

Back End

The back end is used by the service provider. It manages all the resources that are required to provide cloud computing services. It includes a huge amount of data storage, security mechanism, virtual machines, deploying models, servers, traffic control mechanisms, etc.

Both front end and back end are connected to others through a network, generally using the internet connection.

Components of Cloud Computing Architecture

There are the following components of cloud computing architecture -

1. Client Infrastructure

Client Infrastructure is a Front end component. **It provides GUI (Graphical User Interface) to interact with the cloud.**

2. Application

The application may be any software or platform that a client wants to access.

3. Service

A Cloud Services manages that which type of service you access according to the client's requirement.

Cloud computing offers the following three type of services:

i. Software as a Service (SaaS) – It is also known as cloud application services. Mostly, SaaS applications run directly through the web browser means we do not require to download and install these applications. Some important example of SaaS is given below – Example: Google Apps, Salesforce, Dropbox, Slack, Hubspot, Cisco WebEx.

ii. Platform as a Service (PaaS) – It is also known as cloud platform services. It is quite similar to SaaS, but the difference is that PaaS provides a platform for software creation, but using SaaS, we can access software over the internet without the need of any platform. Example: Windows Azure, Force.com, Magento Commerce Cloud, OpenShift.

iii. Infrastructure as a Service (IaaS) – It is also known as cloud infrastructure services. It is responsible for managing applications data, middleware, and runtime environments.

Example: Amazon Web Services (AWS) EC2, Google Compute Engine (GCE), Cisco Metapod.

4. Runtime Cloud

Runtime Cloud provides the execution and runtime environment to the virtual machines.

5. Storage

Storage is one of the most important components of cloud computing. It provides a huge amount of storage capacity in the cloud to store and manage data.

6. Infrastructure

It provides services on the host level, application level, and network level. Cloud infrastructure includes hardware and software components such as servers, storage, network devices, virtualization software, and other storage resources that are needed to support the cloud computing model.

7. Management

Management is used to manage components such as application, service, runtime cloud, storage, infrastructure, and other security issues in the backend and establish coordination between them.

8. Security

Security is an in-built back end component of cloud computing. It implements a security mechanism in the back end.

9. Internet

The Internet is medium through which front end and back end can interact and communicate with each other.

IASS:

Infrastructure as a service (IaaS) is the on-demand availability of highly scalable computing resources as services over the internet. It eliminates the need for enterprises to procure, configure, or manage infrastructure themselves, and they only pay for what they use.

IaaS in cloud computing is when you rent access to cloud infrastructure resources as individual services from a cloud service provider (CSP), including servers, virtual machines, networking resources, and storage. IaaS helps eliminate much of the complexity and costs associated with building and maintaining physical infrastructure in an on-premises data centre.

The CSP is responsible for managing and maintaining the infrastructure, so you can concentrate on installing, configuring, and managing software and keeping your data secure. IaaS providers also offer additional services, such as detailed billing management, logging, monitoring, storage resiliency, and security.

You can access IaaS resources using a pay-as-you-go basis, allowing you to only pay to consume the resources that you need. In other words, you can easily increase or decrease resources, allowing you to pay less when needed or instantly provision and scale out resources to meet new demand.

AWS:

AWS stands for Amazon Web Services, It is an expanded cloud computing platform provided by Amazon Company. AWS provides a wide range of services with a pay-as-per-use pricing model over the Internet such as Storage, Computing power, Databases, Machine Learning services, and much more. AWS facilitates for both businesses and individual users with

effectively hosting the applications, storing the data securely, and making use of a wide variety of tools and services improving management flexibility for IT resources.

AWS comes up with its own network infrastructure on establishing the datacentres in different regions mostly all over the world. Its global Infrastructure acts as a backbone for operations and services provided by AWS. It facilitates the users on creating secure environments using Amazon VPCs (Virtual Private Clouds). Essential services like Amazon EC2 and Amazon S3 for utilizing the compute and storage service with elastic scaling. It supports the dynamic scaling of the applications with the services such as Auto Scaling and Elastic Load Balancing (AWS ELB). It provides a good user-friendly AWS Management Console facilitating seamless configuration and management of AWS services to the Users. Its Architecture ensures high availability, fault tolerance making AWS as a versatile powerful Cloud Computing Platform.

The following are the some of the main fundamentals of AWS:

Regions: AWS provide the services with respective division of regions. The regions are divided based on geographical areas/locations and will establish data centres. Based on need and traffic of users, the scale of data centres is depended to facilitate users with low-latencies of services.

Availability Zones (AZ): To prevent the Data centres for the Natural Calamities or any other disasters. The Datacentres are established as sub sections with isolated locations to enhance fault tolerance and disaster recovery management.

Global Network Infrastructure: AWS ensures the reliability and scalability of services through setting up its own AWS Network Infrastructure globally. It helps in better management of data transmissions for optimized performance and security reliance.

Top AWS Services

In the rapid revolution of Cloud Computing, AWS facilitates with wide variety of services respect to the fields and needs. The following are the top AWS services that are in wide usage:

Amazon EC2(Elastic Compute Cloud): It provides the Scalable computing power via cloud allowing the users to run applications and manage the workloads over their remotely.

Amazon S3 (Simple Storage Service): It offers scalable object Storage as a Service with high durability for storing and retrieving any amount of data.

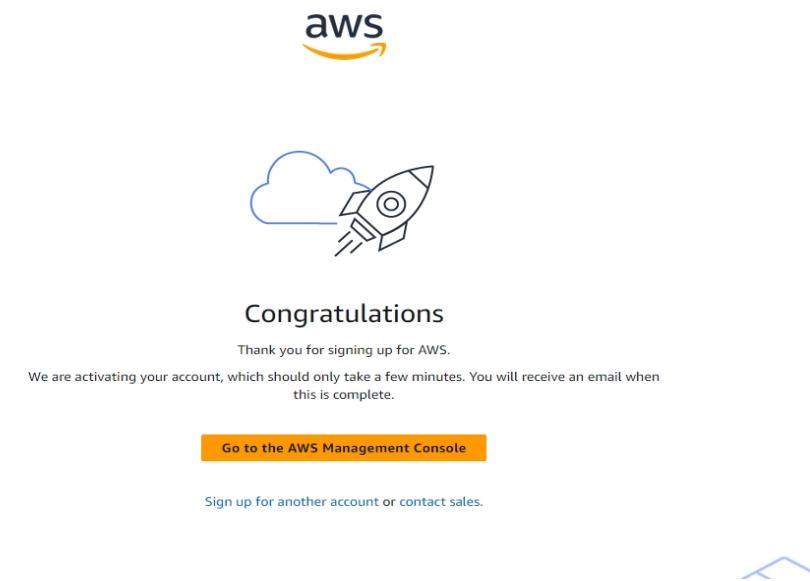
AWS Lambda: It is a service in Serverless Architecture with Function as a Service facilitating serverless computing i.e., running the code on response to the events, the background environment management of servers is handled by aws automatically. It helps the developers to completely focus on the logic of code build.

Amazon RDS (Relational Database Service): This is an aws service that simplifies the management of database providing high available relational databases in the cloud.

Amazon VPC (Virtual Private Cloud): It enables the users to create isolated networks with option of public and private expose within the AWS cloud, providing safe and adaptable configurations of their resources.

1. Implement Windows Machine using AWS EC2.

Go to aws.amazon.com and Sign Up by providing all the information.



After signing in, Login and go to EC2 Service

A screenshot of the AWS Management Console. The left sidebar shows "All services" and "Services by category". Under "Compute", there are links for EC2, Lightsail, Lambda, Batch, Elastic Beanstalk, Serverless Application, Repository, AWS Outposts, EC2 Image Builder, AWS App Runner, and AWS SimSpace Weaver. Under "Containers", there are links for Elastic Container Registry, Elastic Container Service, Elastic Kubernetes Service, Red Hat OpenShift Service on AWS. Under "Storage", there are links for S3, EFS, FSx, S3 Glacier, Storage Gateway, AWS Backup, and AWS Elastic Disaster. Under "Developer Tools", there are links for CodeStar, CodeCommit, CodeBuild, CodePipeline, Cloud9, CloudShell, X-Ray, AWS FIS, CodeArtifact, Amazon CodeCatalyst, AWS AppConfig, Amazon CodeWhisperer, Application Composer, AWS IQ, Managed Services, Activate for Startups, Support, AWS rePost Private, AWS Robotics, AWS RoboMaker, Blockchain, and Amazon Managed. Under "Machine Learning", there are links for Amazon SageMaker, Amazon Augmented AI, Amazon CodeGuru, Amazon DevOps Guru, Amazon Comprehend, Amazon Forecast, Amazon Fraud Detector, Amazon Kendra, Amazon Personalize, Amazon Polly, Amazon Rekognition, Amazon Textract, Amazon Transcribe, Amazon Translate, Amazon DeepComposer, AWS DeepLens, AWS DeepRacer, AWS Panorama, Amazon Monitron, AWS HealthLake, Amazon Lookout for Vision, Amazon Lookout for Equipment, Amazon Lookout for Metrics, Amazon Lex, Amazon Comprehend, and Medical. Under "Cloud Financial Management", there are links for AWS Marketplace, Subscriptions, AWS Billing Conductor, Billing and Cost Management, Front-end Web & Mobile, AWS Amplify, AWS AppSync, Device Farm, and Amazon Location Service. Under "Application Integration", there are links for Step Functions, Amazon AppFlow, Amazon EventBridge, Amazon MQ, Simple Notification Service, Simple Queue Service, SWF, Managed Apache Airflow, AWS B2B Data Interchange, and Business Applications. Under "Business Applications", there is a link for Amazon Connect.

On the left side bar, click on Instances

EC2 Dashboard

- Instances
- Instances
- Instance Types
- Launch Templates
- Spot Requests
- Savings Plans
- Reserved Instances
- Dedicated Hosts
- Capacity Reservations
- Images
- AMIs
- AMI Catalog
- Elastic Block Store
- Volumes
- Snapshots
- Lifecycle Manager
- Network & Security
- Security Groups
- Elastic IPs
- Placement Groups
- Key Pairs
- Network Interfaces
- Load Balancing

Resources

You are using the following Amazon EC2 resources in the Europe (Stockholm) Region:

Instances (running)	0	Auto Scaling Groups	0	Dedicated Hosts	0
Elastic IPs	0	Instances	0	Key pairs	0
Load balancers	0	Placement groups	0	Security groups	1
Snapshots	0	Volumes	0		

Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Service health

AWS Health Dashboard

Zones

Region: Europe (Stockholm)

Zone name	Zone ID
eu-north-1a	eun1-az1
eu-north-1b	eun1-az2
eu-north-1c	eun1-az3

Account attributes

Default VPC: vpc-0fb2e4fb5a1ae9b

Settings

Data protection and security
Zones
EC2 Serial Console
Default credit specification
Console experiments

Explore AWS

Amazon GuardDuty Malware Protection

Now click on Launch Instance:

EC2 Dashboard

- Instances
- Instances
- Instance Types
- Launch Templates

Instances Info

Find Instance by attribute or tag (case-sensitive)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic
No instances									
You do not have any instances in this region.									
Launch instances									

EC2 > Instances > Launch an instance

Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name: MyWindows123

Add additional tags

Application and OS Images (Amazon Machine Image)

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.

Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE, Browse more AMIs

Summary

Number of instances: 1

Software Image (AMI): Microsoft Windows Server 2022 ...read more

Virtual server type (instance type): t3.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 30 GB

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, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Launch instance

Create a New Key Pair (.pem format)

Create key pair

Key pair name
Key pairs allow you to connect to your instance securely.

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

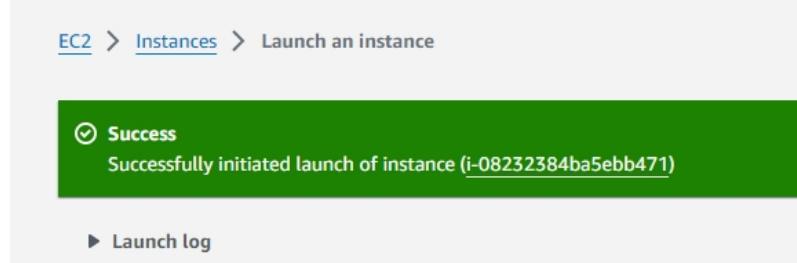
Key pair type
 RSA RSA encrypted private and public key pair
 ED25519 ED25519 encrypted private and public key pair (Not supported for Windows instances)

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 your computer. You will need it later to connect to your instance. [Learn more](#)

Create key pair

After successfully launching instance, you shall see following message.



Now select the instance and click on connect.

The screenshot shows the AWS EC2 Instances page. A single instance named "MyWindows123" is listed, showing it is running. The instance ID is "i-08232384ba5ebb471". The instance summary table includes columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4 IP, and Elastic IP. The instance is currently running in the eu-north-1b availability zone with a public IP of 51.20.1.218 and a private IP of 172.31.39.113. The Public IPv4 DNS is ec2-51-20-1-218.eu-north-1.compute.amazonaws.com. The instance type is t3.micro. The VPC ID is vpc-0fb2e4efb5a1ae9b. The instance summary also lists the instance ID, IP address, instance state, private IP address, public IP address, public DNS name, instance type, VPC ID, and elastic IP addresses.

Now, Click on RDP client and click on Get Password also download remote desktop file.

EC2 > Instances > i-08232384ba5ebb471 > Connect to instance

Connect to instance Info

Connect to your instance i-08232384ba5ebb471 (MyWindows123) using any of these options

Session Manager | **RDP client** | EC2 serial console

Instance ID
i-08232384ba5ebb471 (MyWindows123)

Connection Type

Connect using RDP client
Download a file to use with your RDP client and retrieve your password.

Connect using Fleet Manager
To connect to the instance using Fleet Manager Remote Desktop, the SSM Agent must be installed and running on the instance. For more information, see [Working with SSM Agent](#)

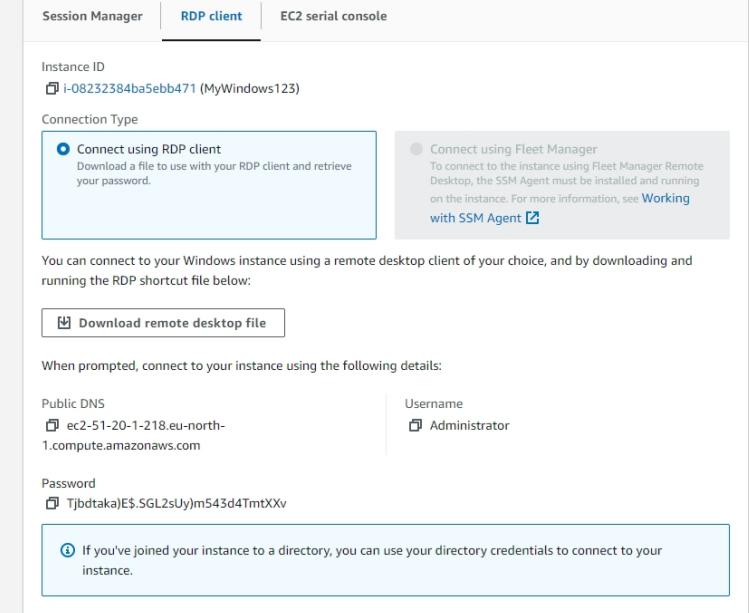
You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download remote desktop file](#)

When prompted, connect to your instance using the following details:

Public DNS ec2-51-20-1-218.eu-north-1.compute.amazonaws.com	Username Administrator
Password Tjbdtaka)E\$.SGL2sUy)m543d4TmtXXv	

If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.



After clicking on get password, upload your private key file that you created and click on decrypt password.

EC2 > Instances > i-08232384ba5ebb471 > Get Windows password

Get Windows password Info

Use your private key to retrieve and decrypt the initial Windows administrator password for this instance.

Instance ID
i-08232384ba5ebb471 (MyWindows123)

Key pair associated with this instance
WindowsKey

Private key
Either upload your private key file or copy and paste its contents into the field below.

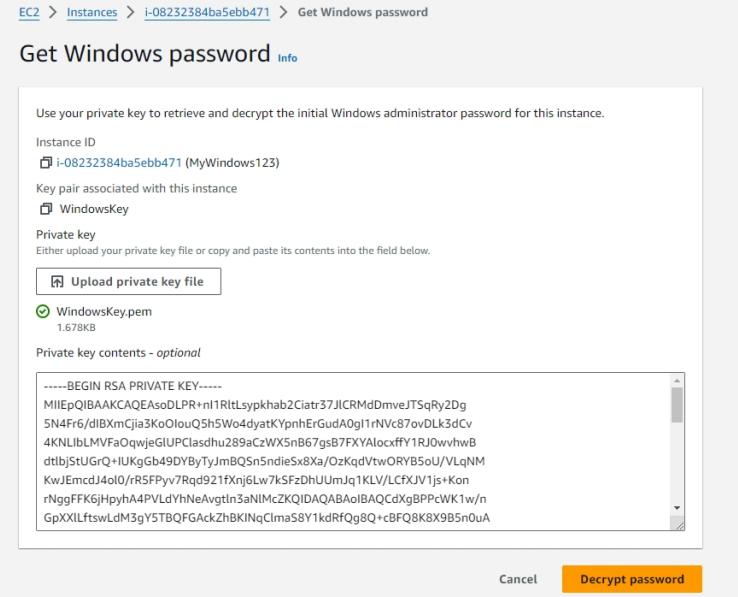
[Upload private key file](#)

WindowsKey.pem
1.678KB

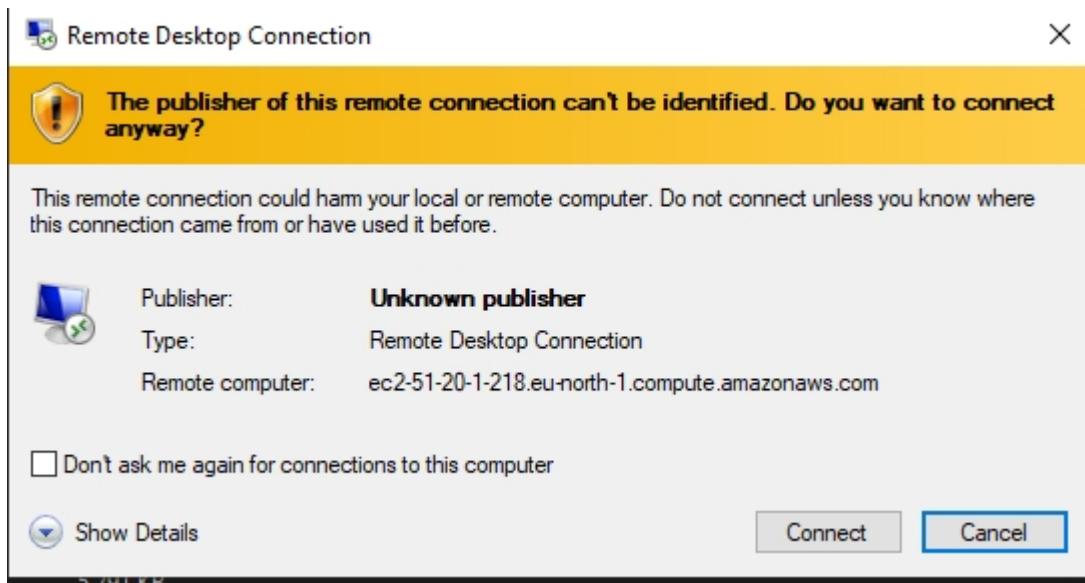
Private key contents - optional

```
-----BEGIN RSA PRIVATE KEY-----  
MIIEpQIBAAKCAQEAs0DLP+nl1RltLsyphab2Ciatr37JICRMdDmveJTSqRy2Dg  
5N4Fr6/dlBXmCja3KoOlouQ5h5Wo4dyatKyphErGudA0gl1rNVc87ovDLk3dCv  
4KNLlbLMVFa0wqejeGIUPCladhu289aCzWX5nB67gsB7FXYAlocffY1RJ0vwvhwB  
dtbjStUGrQ+IUkgGb49DYByTyJmBQSn5ndieSx8xa/OzKqdVtwORYB5oU/VLqNM  
Kw:Emcdj4ol0/r5FPyy7Rqd921fxnj6Lw7kSFzDhUUmJq1KLV/LCFXJV1js+Kon  
rNggFFK6jhpHyA4PVLDyHNeAvgtt13NmCZKQDQAABaIBAQCdXgBPcWK1w/n  
GpXXILftswLdm3gY5TBQFGAckZhBKINqClma58Y1kdRfQg8Q+cBFQ8K8X9B5n0uA
```

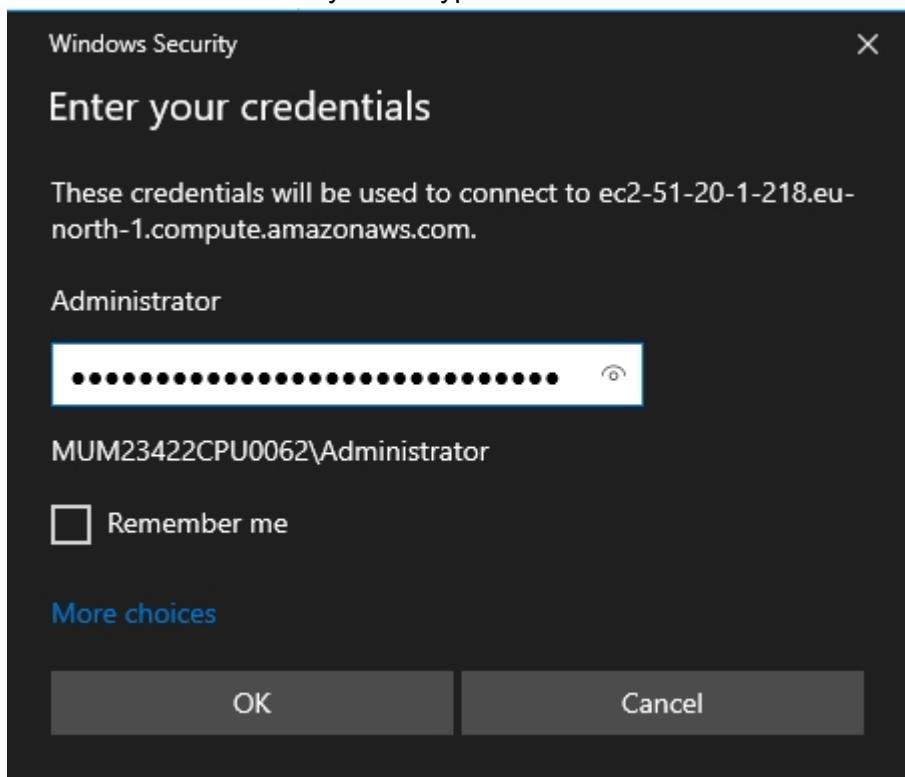
Cancel **Decrypt password**



Now open the downloaded remote desktop file.



Enter the Password that you decrypted.



Now you'll be able to see a Windows OS as follows which is a Virtual Machine



Now install python in that machine and run it in the Command Prompt

A screenshot of a Windows Command Prompt window titled 'C:\Users\Administrator\Downloads\python-3.12.1-embed-amd64\python.exe'. The window displays the following Python session:

```
Python 3.12.1 (tags/v3.12.1:2305ca5, Dec  7 2023, 22:03:25) [MSC v.1937 64 bit (AMD64)] on win32
>>> print('My name is Anil')
My name is Anil
>>> print('My name is Atharva')
My name is Atharva
>>> print('My name is Bhavesh')
My name is Bhavesh
>>> print('My name is Vinayak')
My name is Vinayak
>>>
```

The command prompt window has a dark theme and includes standard Windows UI elements like minimize, maximize, and close buttons.

After this Terminate the Instance that you created.

A screenshot of the AWS Management Console, specifically the EC2 Instances page. A green banner at the top indicates that the instance 'MyWindows123' has been successfully terminated. The main table lists one instance: 'MyWindows123' (Instance ID: i-08232384ba5ebb471), which was running and is now terminated. The table includes columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4 IP, and Elastic IP. The instance is located in the eu-north-1b availability zone with a public IP of 51.20.1.218. The AWS navigation bar at the top includes 'Services', 'Search', and the current region 'Stockholm'.

2. Implement the Ubuntu Machine using AWS EC2 and Execute the Linux Commands

- Disk Information in Human Readable form
- Create a file with your name
- Create a file with you CourseName and add a text file in it
- Display the created file
- Copy the contents of the created file in another file and print it.
- Install Chrome Browser/ Python3

Step 1: Create an EC2 instance

The screenshot shows the 'Launch an instance' step of the AWS EC2 wizard. It includes sections for 'Name and tags', 'Application and OS Images (Amazon Machine Image)', and a 'Quick Start' section with various AMI icons.

Name and tags Info

Name: Atharva

Add additional tags

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

Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux	macOS	Ubuntu	Windows	Red Hat	SUSE Li

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Create key pair

Key pair name
Key pairs allow you to connect to your instance securely.

The name can include up to 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 your computer. You will need it later to connect to your instance. [Learn more](#)

[Cancel](#) [Create key pair](#)

Connect to instance Info

Connect to your instance i-010877623d8df0676 (Atharva3) using any of these options

[EC2 Instance Connect](#) [Session Manager](#) [SSH client](#) [EC2 serial console](#)

Instance ID

Connection Type

- Connect using EC2 Instance Connect
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.
- Connect using EC2 Instance Connect Endpoint
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

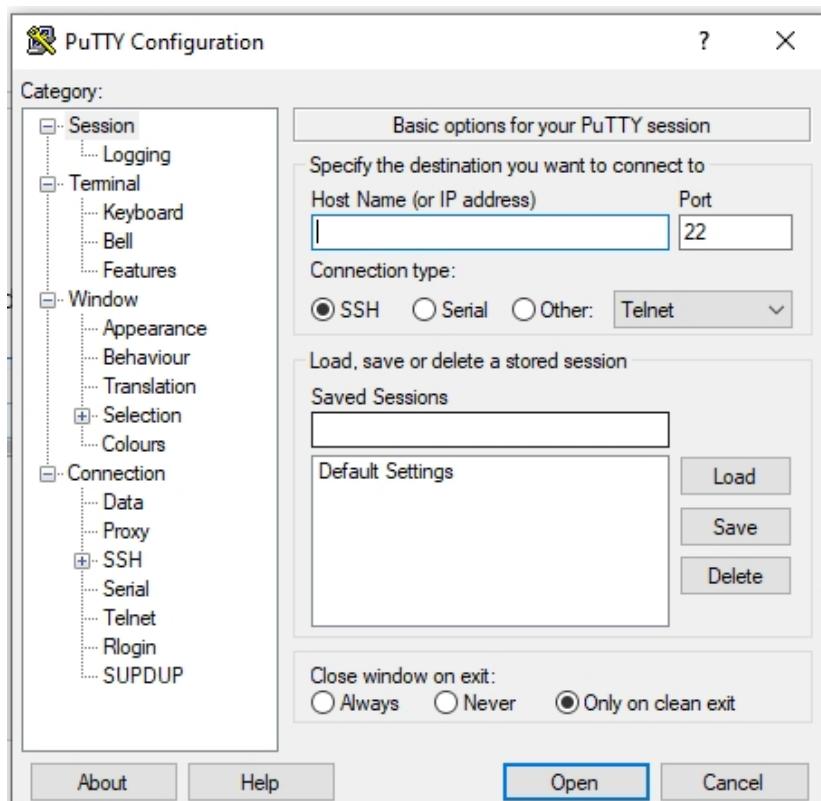
Public IP address

Username
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.

ⓘ Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

[Cancel](#) [Connect](#)

Step 2: After the instance is created open putty.exe file that we have downloaded



The screenshot shows the AWS Lambda console interface. At the top, a table lists instances: one row for 'Atharva' with Instance ID i-0a333f2fc10ee5b9d, Instance state Running, Instance type t3.micro, Status check Initializing, and Alarm status View alarms +. Below this is a large empty space. At the bottom, the instance details for 'Atharva' are shown, including its public IPv4 address 16.171.198.38, which is underlined and has an 'open address' link next to it. The 'Details' tab is selected.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
Atharva	i-0a333f2fc10ee5b9d	Running	t3.micro	Initializing	View alarms +

Instance: i-0a333f2fc10ee5b9d (Atharva)

Details Status and alarms New Monitoring Security Networking Storage Tags

▼ Instance summary Info

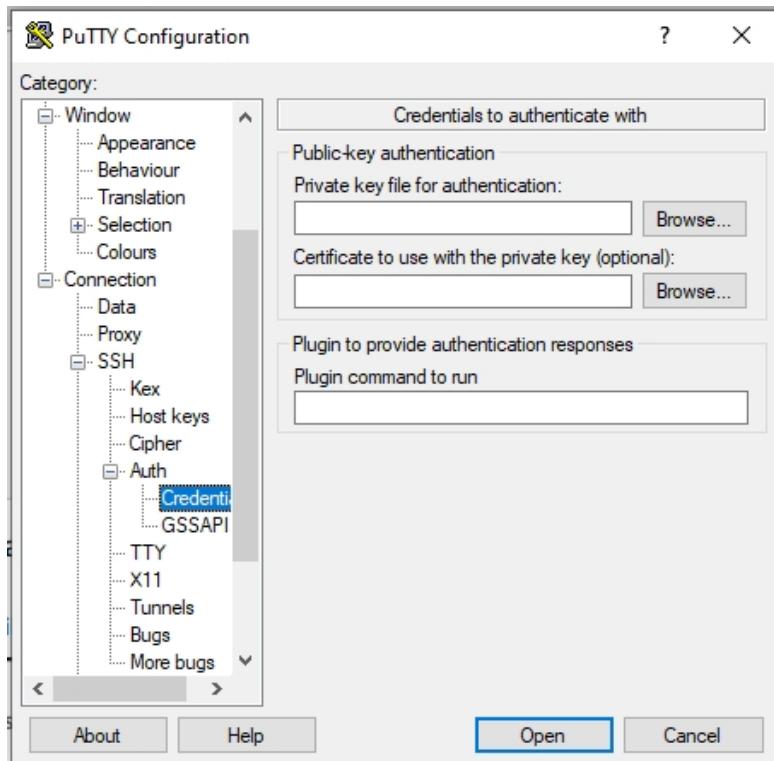
Instance ID
i-0a333f2fc10ee5b9d (Atharva)

IPv6 address

Public IPv4 address
16.171.198.38 [open address]

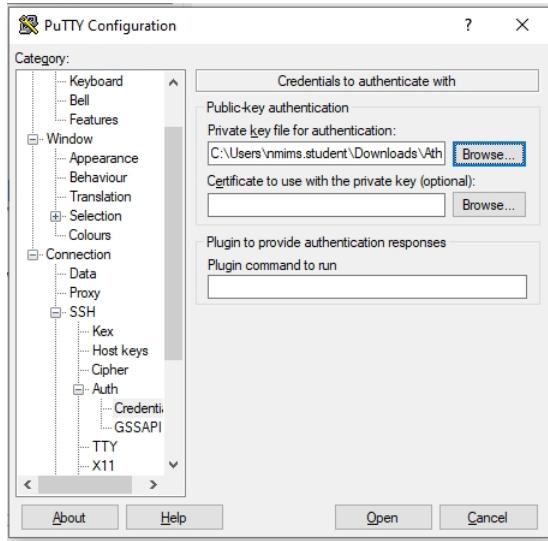
Instance state

Put this IPv4 address in putty.exe



Select browse from here

Step 3 Browse the ppk file you have downloaded



Step 4: Now a command prompt will be opened type your user name here

```
ubuntu@ip-172-31-41-166: ~
login as: ubuntu
Authenticating with public key "Atharva28"
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1017-aws x86_64)

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

System information as of Sat Jan 20 02:45:28 UTC 2024

System load: 0.0          Processes: 99
Usage of /: 20.6% of 7.57GB  Users logged in: 0
Memory usage: 21%          IPv4 address for ens5: 172.31.41.166
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

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.
```

Step 5: Now enter the commands here

```
ubuntu@ip-172-31-41-166:~$ ls
ubuntu@ip-172-31-41-166:~$ mkdir msc
ubuntu@ip-172-31-41-166:~$ ls
msc
ubuntu@ip-172-31-41-166:~$ cd msc
ubuntu@ip-172-31-41-166:~/msc$ touch cloud.txt
ubuntu@ip-172-31-41-166:~/msc$ ls
cloud.txt
ubuntu@ip-172-31-41-166:~/msc$
```

```
ubuntu@ip-172-31-41-166:~/msc$ cat>cloud.txt
Bhaveh pashte only sonapapdi loverubuntu@ip-172-31-41-166:~/msc$
ubuntu@ip-172-31-41-166:~/msc$ cat cloud.txt
Bhaveh pashte only sonapapdi loverubuntu@ip-172-31-41-166:~/msc$
ubuntu@ip-172-31-41-166:~/msc$
```

Step 6: Now install python in cmd

```
ubuntu@ip-172-31-41-166:~/msc$ sudo apt install python3
```

```
ubuntu@ip-172-31-41-166:~/msc$ sudo apt install python3
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3 is already the newest version (3.10.6-1~22.04).
python3 set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-41-166:~/msc$
```

Because the python was already installed

Step 7: Now type python3 and then you can run python code on it

```
ubuntu@ip-172-31-41-166:~$ python3
Python 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print('Hello World')
Hello World
>>>
```

Step 8 Create a file write your address and read the file

```
ubuntu@ip-172-31-41-166:~$ nano city
ubuntu@ip-172-31-41-166:~$ cat city
Atharva kulkarni kattar punekar 411033

ubuntu@ip-172-31-41-166:~$
```

Step 9: Copy content from 1 file to another

```
ubuntu@ip-172-31-41-166:~$ cp city cityl
ubuntu@ip-172-31-41-166:~$ cat cityl
Atharva kulkarni kattar punekar 411033
```

Step 10: install chrome browser in ubuntu

```
ubuntu@ip-172-31-41-166:~$ wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
--2024-01-20 03:06:02-- https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
Resolving dl.google.com (dl.google.com)... 142.250.74.142, 2a00:1450:400f:802::200e
Connecting to dl.google.com (dl.google.com)|142.250.74.142|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 105000828 (100M) [application/x-debian-package]
Saving to: 'google-chrome-stable_current_amd64.deb'

google-chrome-stable_current_amd 100%[=====] 100.14M  182MB/s   in 0.6s

2024-01-20 03:06:02 (182 MB/s) - 'google-chrome-stable_current_amd64.deb' saved [105000828/105000828]

ubuntu@ip-172-31-41-166:~$
```