Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

AY: 2023-24

Class:	TE	**Semester:**	VI
Course Code:	CSL605	**Course Name:**	Skill Based Lab course : Cloud Computing

Name of Student:	Soham Ajit Dahanukar
Roll No. :	13
Experiment No.:	1
Title of the Experiment:	To study and implement Infrastructure as a Service using AWS EC2 by creating Ubuntu Virtual Machine through PUTTY open Source terminal emulator
Date of Performance:	
Date of Submission:	

Evaluation

Performance Indicator	Max. Marks	Marks Obtained
Performance	5	
Understanding	5	
Journal work and timely submission	10	
Total	20	

Performance Indicator	Exceed Expectations (EE)	Meet Expectations (ME)	Below Expectations (BE)
Performance	4-5	2-3	1
Understanding	4-5	2-3	1
Journal work and timely submission	8-10	5-8	1-4

Checked by

Name of Faculty :

Signature :

Date

Experiment No. 1

Aim: To study and implement Infrastructure as a Service using AWS

Objective: To demonstrate the steps to create and run virtual machines inside Amazon EC2

Theory:

IAAS: Infrastructure as a service (IaaS) is a type of cloud computing service that offers essential compute, storage, and networking resources on demand, on a pay-as-you-go basis. IaaS is one of the four types of cloud services, along with software as a service (SaaS), platform as a service (PaaS), and serverless.

In an IaaS service model, a cloud provider hosts the infrastructure components that are traditionally present in an on-premises data center. This includes servers, storage and networking hardware, as well as the virtualization or hypervisor layer.

IaaS providers also supply a range of services to accompany those infrastructure components.

IaaS customers access resources and services through a wide area network (WAN), such as the internet, and can use the cloud provider's services to install the remaining elements of an application stack. For example, the user can log in to the IaaS platform to create virtual machines (VMs); install operating systems in each VM; deploy middleware, such as databases; create storage buckets for workloads and backups; and install the enterprise workload into that VM. Customers can then use the provider's services to track costs, monitor performance, balance network traffic, troubleshoot application issues and manage disaster recovery.

AWS: AWS enables you to select the operating system, programming language, web application platform, database, and other services you need. With AWS, you receive a virtual environment that lets you load the software and services your application requires. This eases the migration process for existing applications while preserving options for building new solutions.
You pay only for the compute power, storage, and other resources you use, with no long-term contracts or up-front commitments.
AWS is designed to allow application providers, ISVs, and vendors to quickly and securely host your applications – whether an existing application or a new SaaS-based application.

EC2: Amazon Elastic Compute Cloud (Amazon EC2) provides on-demand, scalable computing capacity in the Amazon Web Services (AWS) Cloud. Using Amazon EC2 reduces hardware costs so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. You can add capacity (scale up) to handle compute-heavy tasks, such as monthly or yearly processes, or spikes in website traffic. When usage decreases, you can reduce capacity (scale down) again.

Steps:

- We will start by creating an EC2 instance. Go to EC2 Console https://console.aws.amazon.com/ec2/v2/home and Click on Launch instance. Where you will find multiple AMIs, Scroll down and select Ubuntu Server

- Once you select the server, follow further steps, and launch. Make sure while launching download the PEM file as it will need in the future.

- Now go to the EC2 dashboard and look for a public IP address and Copy that as we will need for PUTTY configuration, Though you can use Public DNS as well.

- We are done with AWS configuration, now open PUTTY and follow a few steps:

- Add copied IP address in the Host Name field and name the Saved session and save it. Will keep port as it is i.e., 22 as we have only opened the 22 port as we want everything should go through the SSH tunnel; though you can open other ports as well there will be less security.
- Add SSH (PPK) key by browsing SSH -> Auth -> Private Key file for authentication -> Browse and again go to Session and Save it.
  Note: By default, AWS only provide a PEM file. We need to convert PEM to PPK. You can find multiple online tools for converting this

- Click on Open, and it will prompt a security alert just choose Yes, and you will able to see the command prompt.
  On the command prompt, just log in as Ubuntu, as we have added PPK so it won't ask for Password.
- Once you get logged into the server, you need to follow the below.
  - sudo apt update && sudo apt upgrade
  - sudo sed -i 's/^PasswordAuthentication no/PasswordAuthentication yes/' /etc/ssh/sshd_config – To set login credentials
  - sudo /etc/init.d/ssh restart
  - sudo passwd ubuntu – This will be your login password to ubuntu machine
  - sudo apt install xrdp xfce4 xfce4-goodies tightvncserver
  - echo xfce4-session> /home/ubuntu/.xsession
  - sudo cp /home/ubuntu/.xsession /etc/skel
  - sudo sed -i '0,/-1/s//ask-1/' /etc/xrdp/xrdp.ini
  - sudo service xrdp restart

- Note: In between, if this process asks for some kind of choice, then select the local version installed option.

- Now, we need to configure the tunnel on PUTTY to the route of the request.

- Open PUTTY again and go to the Connection – > SSH -> Tunnels ->

1. Add Source Port (This could be any in my case I have added 8888).
2. Add Destination Port – This will be your instance private IP address following with 3389.
3. Click on Add

- Once done, now click on the open button and you will see a Command prompt where login Id will be Ubuntu.
  Now you can check either we are listening on the same port or not using the following command: netstat -antp

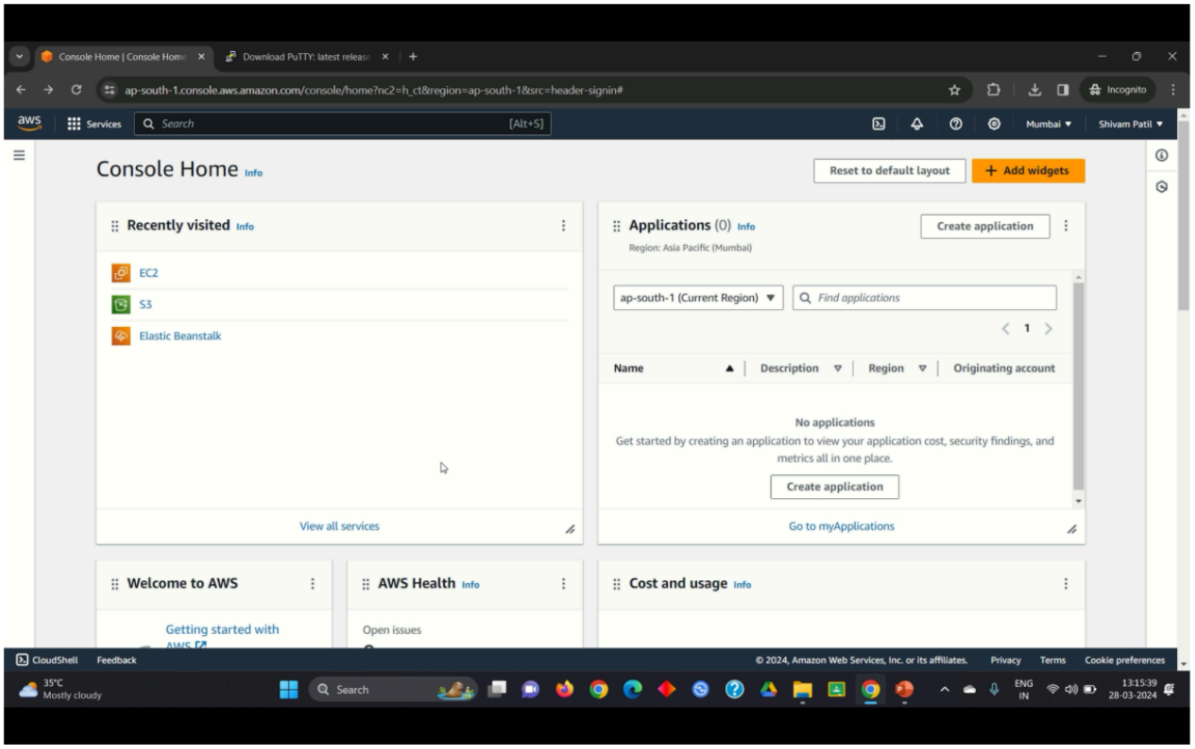
- Now we are done with the setup.

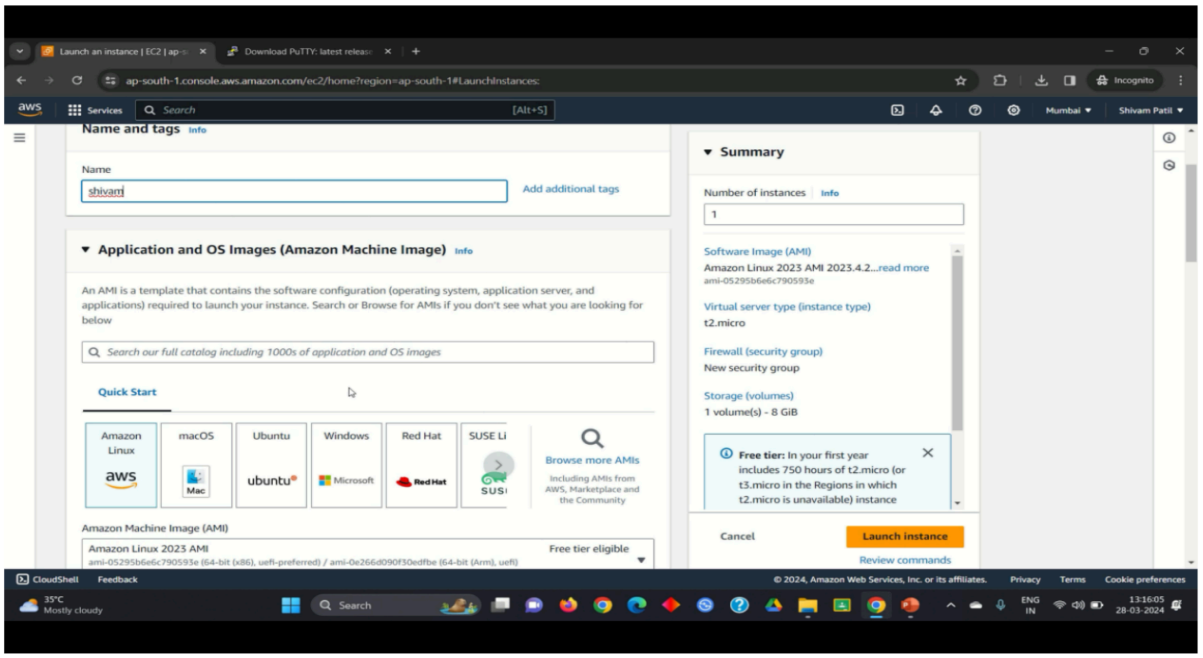
- Open Remote Desktop Connection

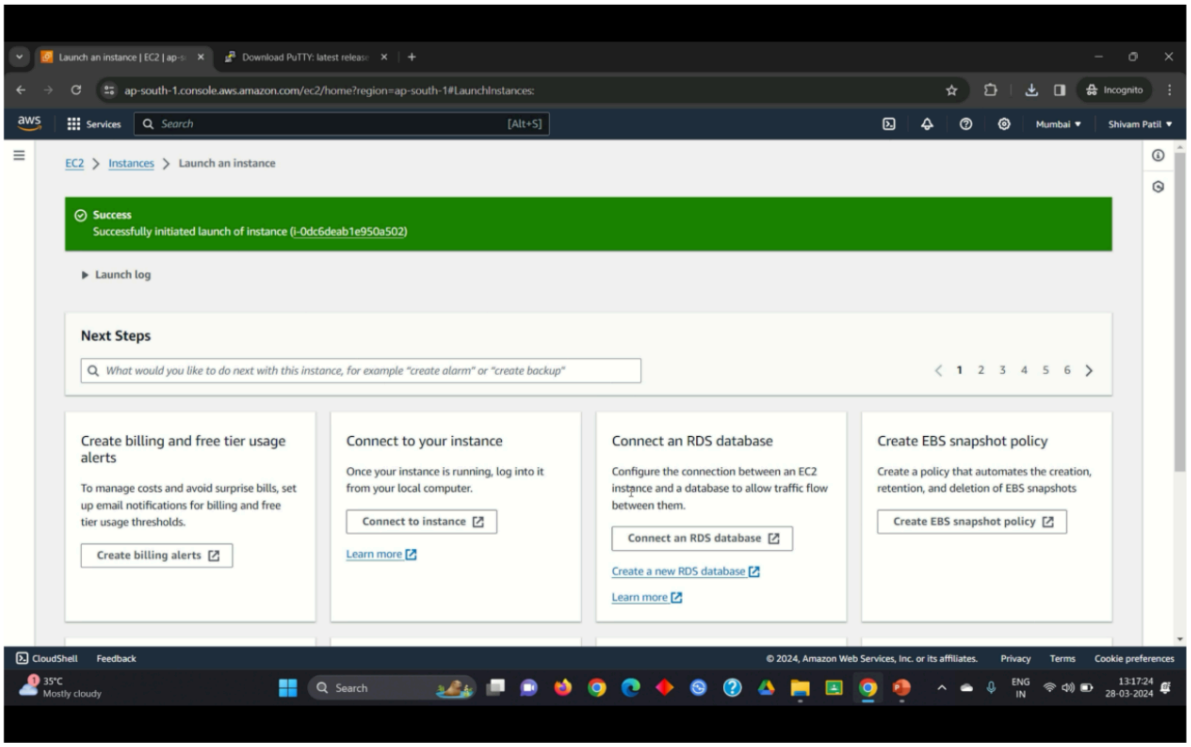
- Add localhost:8888 or 127.0.0.1:8888 and Connect

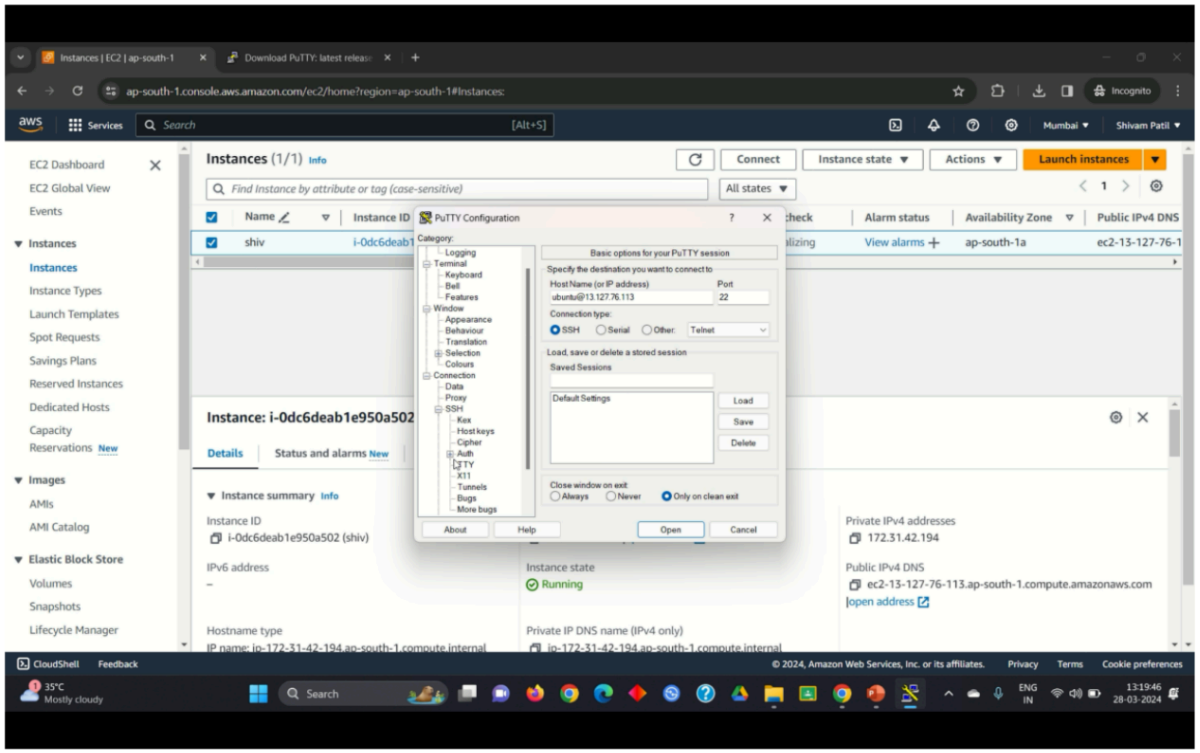
- You will be asked for a username and password; So username could be Ubuntu and Password will be added by you while installing the command. And Yeah, there you go, We have successfully logged into the Remote server.

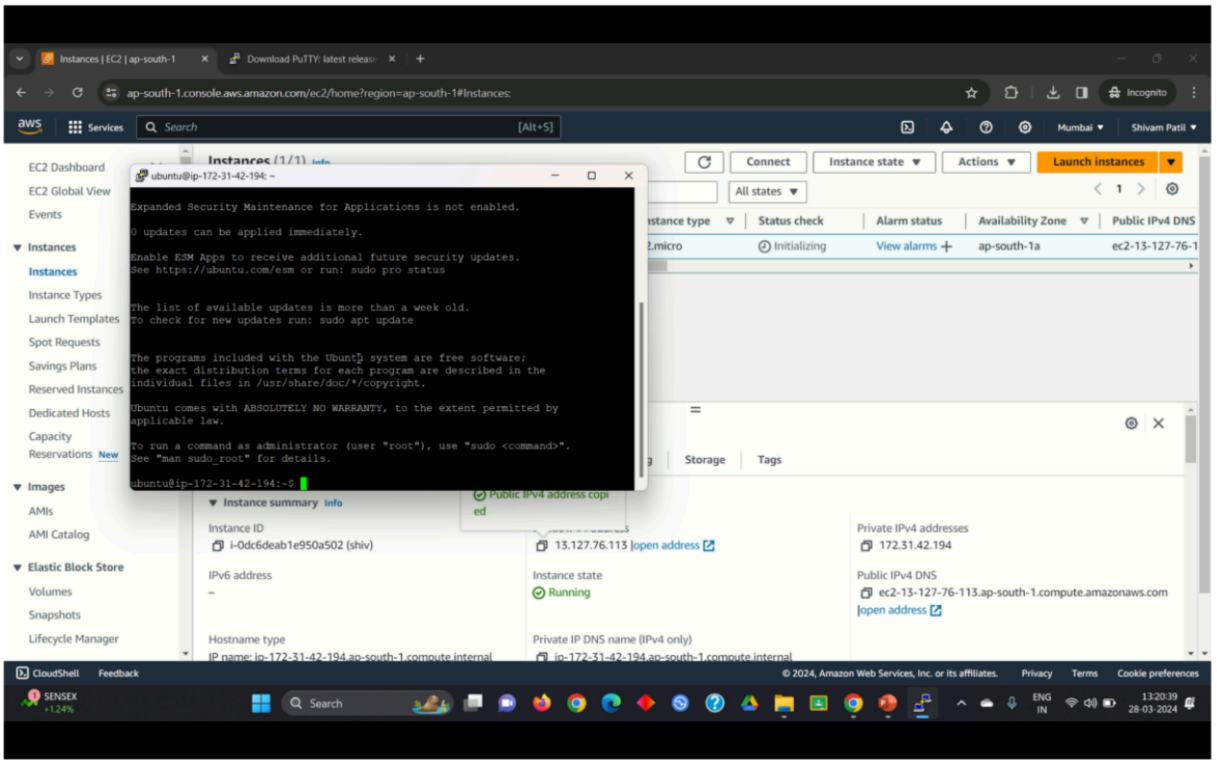
Output and Observation:






Launch an instance | EC2 | ap-s
Download PuTTY: latest releas
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:
Incognito
Services
Search
[Alt+S]
Mumbai
Shivam Patil
EC2 > Instances > Launch an instance
Success
Successfully initiated launch of instance (i-0dc6deab1e950a502)
Launch log
Next Steps
What would you like to do next with this instance, for example "create alarm" or "create backup"
1 2 3 4 5 6
Create billing and free tier usage alerts
To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.
Create billing alerts
Connect to your instance
Once your instance is running, log into it from your local computer.
Connect to instance
Learn more
Connect an RDS database
Configure the connection between an EC2 instance and a database to allow traffic flow between them.
Connect an RDS database
Create a new RDS database
Learn more
Create EBS snapshot policy
Create a policy that automates the creation, retention, and deletion of EBS snapshots
Create EBS snapshot policy
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
35°C
Mostly cloudy
Search
ENG
IN
13:17:24
28-03-2024


Instances | EC2 | ap-south-1
Download PuTTY: latest releas
ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:
Incognito
Services
Search
[Alt+S]
Mumbai
Shivam Patil
EC2 Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations New
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Instances (1/1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Alarm status
Availability Zone
Public IPv4 DNS
shiv
i-0dc6deab1
View alarms
ap-south-1a
ec2-13-127-76-1
PuTTY Configuration
Category:
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
SSH
Kex
Host keys
Cipher
Auth
TTY
X11
Tunnels
Bugs
More bugs
Basic options for your PuTTY session
Specify the destination you want to connect to
Host Name (or IP address)
Port
ubuntu@13.127.76.113
22
Connection type:
SSH
Serial
Other:
Telnet
Load, save or delete a stored session
Saved Sessions
Default Settings
Load
Save
Delete
Close window on exit:
Always
Never
Only on clean exit
About
Help
Open
Cancel
Instance: i-0dc6deab1e950a502
Details
Status and alarms New
Instance summary Info
Instance ID
i-0dc6deab1e950a502 (shiv)
IPv6 address
–
Hostname type
Instance state
Running
Private IP DNS name (IPv4 only)
Private IPv4 addresses
172.31.42.194
Public IPv4 DNS
ec2-13-127-76-113.ap-south-1.compute.amazonaws.com
open address
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
35°C
Mostly cloudy
Search
ENG
IN
13:19:46
28-03-2024



Conclusion:

EC2 (Elastic Compute Cloud) provides scalability, flexibility, and cost-effectiveness for Infrastructure as a Service (IaaS) deployments. It allows users to easily provision and manage virtual servers on-demand, enabling rapid scaling and resource optimization. Additionally, EC2 offers a wide range of instance types, operating systems, and pricing options, catering to diverse workload requirements and budget constraints.