



Pimpri Chinchwad Education Trust's
Pimpri Chinchwad College of Engineering
(PCCoE)

(An Autonomous Institute)
Affiliated to Savitribai Phule Pune
University(SPPU) ISO 21001:2018 Certified by
TUV



Course: DevOps Laboratory

Code: BIT26VS01

Name: Amar Vaijinath Chavan

PRN: 124B2F001

Assignment 13: Explore the different services of AWS (EC2, S3, AWS Lambda, RDS, ELB, ECS) and create EC2 instance in AWS.

Aim: To explore core Amazon Web Services (AWS) infrastructure and successfully provision an Ubuntu-based EC2 instance within the AWS cloud environment.

Objectives:

- To understand the utility and architecture of key AWS services.
- To navigate the AWS Management Console for cloud resource provisioning.
- To establish a secure remote connection to a cloud-based virtual server using SSH.

Theory:

1. Amazon Elastic Compute Cloud (EC2)

EC2 provides scalable, resizable computing capacity in the form of virtual servers called instances. It eliminates the need to invest in hardware upfront, allowing for faster development and deployment of applications. Users can choose various configurations of CPU, memory, and storage to suit their specific application needs.

2. Amazon Simple Storage Service (S3)

S3 is an object storage service offering industry-leading scalability, data availability, security, and performance. It is designed for 99.99999999% durability and stores data as objects within containers called "Buckets." It is widely used for backups, data archiving, and hosting static website content.

3. AWS Lambda (Serverless Compute)

Lambda is a serverless, event-driven compute service that lets you run code for virtually any type of application or backend service without provisioning or managing servers. It automatically scales your application by running code in response to triggers such as changes in data in an S3 bucket or an HTTP request via API Gateway.

4. Amazon Relational Database Service (RDS)

RDS is a managed service that makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching, and backups.

5. Elastic Load Balancing (ELB)

ELB automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses. It increases the fault tolerance of your applications by ensuring that only healthy targets receive traffic and by scaling to handle changes in request patterns.

6. Amazon Elastic Container Service (ECS)

ECS is a highly scalable, fast, container management service that makes it easy to run, stop, and manage Docker containers on a cluster. It integrates deeply with other AWS services such as IAM for security and ELB for traffic distribution, providing a robust platform for microservices.

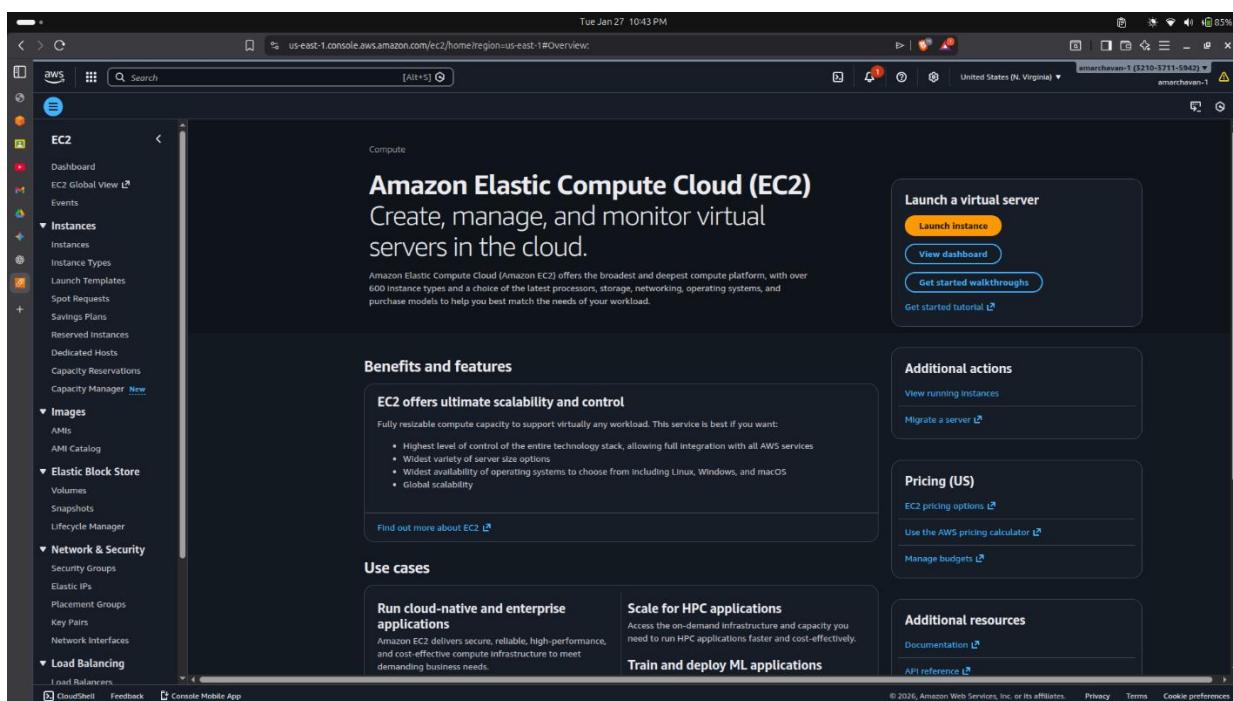
7. Key Infrastructure Components (AMI, Security Groups, and Key Pairs)

An **Amazon Machine Image (AMI)** is a template containing the software configuration (OS, application server) required to launch an instance. **Security Groups** act as virtual firewalls to control inbound and outbound traffic. **Key Pairs** utilize public-key cryptography to secure the login process; the user holds the private key (.pem), and AWS stores the public key.

Practical Procedure / Steps:

Step 1: AWS Console Navigation

- Log in to the AWS Management Console.
- Search for and select the **EC2** service to open the EC2 Dashboard.



Step 2: Launching the EC2 Instance

- Name and Tags:** Enter a name for the instance, such as Assignment13-Server.
- Application and OS Images (AMI):** Select **Ubuntu** and choose the **Ubuntu Server 24.04 LTS (HVM) 64-bit (x86)** image.
- Instance Type:** Select t3.micro, which is part of the AWS Free Tier.
- Key Pair (login):** Select "Create new key pair." Name it assignment13key, select **RSA**, and download the .pem file.

Tue Jan 27 10:50 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#launchinstances

EC2 Instances Launch an instance

Launch an instance info

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 info

Name Add additional tags

Application and OS Images (Amazon Machine Image) info

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

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

Recent [Amazon Linux](#) [macOS](#) [Ubuntu](#) [Windows](#) [Red Hat](#) [SUSE Linux](#) [Debian](#) Quick Start [Search more AMIs](#)

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-0b6c6ebcd2801a5cb (64-bit (x86)) / ami-0071cb4c41eaed0b (64-bit (Arm))
Virtualization: hvm DNA enabled: true Root device type: ebs

Description
Ubuntu Server 24.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>). Canonical, Ubuntu, 24.04, amd64 noble Image

Architecture: 64-bit (x86) AMI ID: ami-0b6c6ebcd2801a5cb Publish Date: 2025-12-12 Username: ubuntu Verified provider

Cancel [Launch instance](#) [Preview code](#)

Tue Jan 27 10:50 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#launchinstances

EC2 Instances Launch an instance

Instance type info | Get advice

Instance type [t3.micro](#) Family: t3 2 vCPU 1 GiB Memory Current generation: true Free tier eligible On-Demand Preemptible price: \$0.0139 USD per Hour On-Demand Sustained price: \$0.0154 USD per Hour On-Demand burst credits price: \$0.0001 USD per Hour On-Demand Reserved price: \$0.0001 USD per Hour On-Demand Windows base pricing: \$0.196 USD per Hour

Additional costs apply for AMIs with pre-installed software

Key pair (login) info

You can use a key pair to securely connect to your instance. Ensure that you have a key pair ready.

Key pair name - required

Create key pair

Key pair name: assignment13key 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](#)

Network settings info

Network: [Info](#) vpc-0d593587d0d6791fb Subnet: [Info](#) subnet-d05d1f75aca79fc Auto-assign public IP: [Info](#) Disable Firewall (security groups): [Info](#) A security group is a set of firewall rules that control the traffic for your instance. Add rule Create security group Select existing security group

We'll create a new security group called 'Launch-wizard-7' with the following rules:

Allow SSH traffic from Anywhere 0.0.0.0/0 Help you connect to your instance

Allow HTTPS traffic from the internet Anywhere 0.0.0.0/0 To set up an endpoint, for example when creating a web server

Cancel [Create key pair](#) [Launch instance](#) [Preview code](#)

Step 3: Network and Storage Configuration

- Network Settings:** Ensure a VPC and Subnet are selected. Enable "Auto-assign public IP" to allow external access.
- Firewall (Security Groups):** Create a new security group. Add a rule to allow SSH traffic on port 22 from My IP for secure access.
- Configure Storage:** Maintain the default 8 GiB of gp3 (General Purpose SSD) storage.

This screenshot shows the AWS EC2 Launch Instances wizard at Step 3: Network and Storage Configuration. The left panel displays network settings, including a selected VPC (vpc-0d5939897d06791fb), subnet (subnet-0aa5df25bcaca79ec), and auto-assign public IP (Enabled). It also shows the creation of a new security group named 'launch-wizard-7' with an inbound rule allowing SSH (TCP port 22) from 'My IP'. The right panel shows the summary, which includes the number of instances (1), software image (Canonical, Ubuntu, 24.04, amd64), virtual server type (t3.micro), and storage (1 volume(s) - 8 GiB). A large orange 'Launch instance' button is prominent at the bottom right.

This screenshot shows the continuation of the AWS EC2 Launch Instances wizard at Step 3. The left panel now includes an 'Advanced' section under 'Configure storage', showing a single 8 GiB gp3 volume. The right panel remains the same as the previous screenshot, displaying the summary and launch button. At the bottom right, there is a yellow box containing the number '4'.

Step 4: Establishing SSH Connection

1. Open the terminal on your local Ubuntu machine.
2. Navigate to the directory containing your .pem file and secure the file permissions: chmod 400 "assignment13key.pem"
3. Connect to the instance using its Public IPv4 address: ssh -i "assignment13key.pem" ubuntu@ec2-107-23-124-150.compute-1.amazonaws.com

The screenshot shows the AWS Management Console with the URL us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstanceDetails:instanceId=i-051ae0aab0f833a59. The left sidebar is expanded to show the EC2 service. The main content area displays the 'Instance summary for i-051ae0aab0f833a59' page. Key details include:

- Public IPv4 address:** 107.23.124.150 [open address]
- Instance state:** Running
- Private IP address:** 107.23.124.150 [Public IP]
- VPC ID:** vpc-0d5939897d06791fb
- Subnet ID:** subnet-0a5df25bcaca79ec
- Instance ARN:** arn:aws:ec2:us-east-1:321037115942:instance/i-051ae0aab0f833a59

The 'Details' tab is selected, showing sections for Instance details, Monitoring, Security, Networking, Storage, and Tags. The Monitoring section indicates 'Monitoring disabled'. The Networking section shows 'Launch time' as Tue Jan 27 2026 22:51:38 GMT+05:50 (India Standard Time) (less than a minute ago).

The screenshot shows the 'Connect' dialog box for the instance i-051ae0aab0f833a59. The 'SSH client' tab is selected. The instructions for connecting via SSH are listed:

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is assignment13key.pem.
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 assignment13key.pem
4. Connect to your instance using its Public DNS:
ec2-107-23-124-150.compute-1.amazonaws.com

A note at the bottom states: "Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username." A 'Cancel' button is visible at the bottom right.

```
Tue Jan 27 10:53 PM  
ubuntu@ip-172-31-6-188:~  
  
amar@amar-Inspiron-3501:~/Desktop/AWS-EC2-Jenkins$ chmod 400 "assignment13key.pem"  
amar@amar-Inspiron-3501:~/Desktop/AWS-EC2-Jenkins$ ssh -i "assignment13key.pem" ubuntu@ec2-107-23-124-150.compute-1.amazonaws.com  
The authenticity of host 'ec2-107-23-124-150.compute-1.amazonaws.com (64:ff9b::6b17:7c96)' can't be established.  
ED25519 key fingerprint is SHA256:EXfIwTwVplvwUjI9U3QVfkq8ossxuNroMl7hfvJejM.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added 'ec2-107-23-124-150.compute-1.amazonaws.com' (ED25519) to the list of known hosts.  
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1018-aws x86_64)  
  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/pro  
  
System information as of Tue Jan 27 17:22:45 UTC 2026  
  
System load: 0.21 Temperature: -273.1 C  
Usage of /: 25.9% of 6.71GB Processes: 116  
Memory usage: 25% Users logged in: 0  
Swap usage: 0% IPv4 address for ens5: 172.31.6.188  
  
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.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu@ip-172-31-6-188:~$ whoami  
ubuntu  
ubuntu@ip-172-31-6-188:~$ hostname -I  
172.31.6.188  
ubuntu@ip-172-31-6-188:~$
```

Conclusion

The successful completion of Assignment 13 demonstrates the efficient orchestration of cloud infrastructure using the AWS Management Console. By exploring a diverse suite of services—including EC2, S3, RDS, Lambda, ELB, and ECS—I gained an understanding of the decoupled and highly scalable nature of modern cloud environments. The practical execution involved the end-to-end provisioning of a virtual Ubuntu 24.04 server, covering critical configuration steps such as AMI selection, security group firewall rules, and key-pair management. Establishing a secure SSH connection to the cloud instance proved the ability to remotely manage Linux workloads in a production-ready environment.