

**Thadomal Shahani Engineering College**  
Bandra (W.), Mumbai - 400 050.

**CERTIFICATE**

Certify that Mr./Miss AYUSH SANJAY SHARMA  
of I.T Department, Semester V with  
Roll No. 117 has completed a course of the necessary  
experiments in the subject ADVANCE DEVOPS LAB under my  
supervision in the **Thadomal Shahani Engineering College**  
Laboratory in the year 2025 - 2026

Teacher In- Charge

*Soradat  
09/10/25*

Head of the Department

Date 09/10/25

Principal

## CONTENTS

SR. NO.	EXPERIMENTS	PAGE NO.	DATE	TEACHERS SIGN.
1.	Introduction to Advance Devops	1-3	22/07/25	7
2.	To study & perform AWS EC2 service and launch an EC2 instance	4-7	05/08/25	
3.	To study AWS S3 service & create a bucket for hosting static web application	8-11	26/08/25	
4.	To study AWS code pipeline & deploy web application using code pipeline	12-27	09/09/25	(A) go on 09/09/25
5.	To understand Kubernetes cluster architecture	28-31	09/09/25	
6.	To understand terraform lifecycle and to build, change & destroy AWS infrastructure using Terraform.	32-40	13/09/25	
7.	To perform static analysis using Sonarqube SAFT PROCESS.	41-48	15/09/25	
8.	To understand continuous monitoring using Nagios	49-52	16/09/25	
9.	To understand AWS lambda functions & create a lambda function using Python to log 'An image has been added' msg, once a file is added to S3 bucket	53-58	23/09/25	
10.	To create lambda function using Python for adding data to DynamoDB database	59-70	30/09/25	
11.	Assignment 1	71-76		
12.	Assignment 2.	77-81		

# Assignment-1

Aim- To study Advanced DevOps,its tools,features,benefits and principles.

Theory-

## What is advanced DevOps-

Advanced DevOps represents a significant step up from the foundational principles of DevOps. While traditional DevOps focuses on automating the software delivery pipeline and fostering collaboration between development and operations teams, advanced DevOps emphasizes a more comprehensive and holistic approach to software delivery, integrating a broader range of practices and technologies throughout the entire software delivery lifecycle.

## Tools Used in Advanced DevOps

There are many tools used in Advanced DevOps, grouped by their functions:

CI/CD Tools (Continuous Integration and Continuous Delivery/Deployment):

Tools like Jenkins, AWS, GitLab CI, GitHub Actions, Argo CD, Tekton, and Spinnaker automate the build, test, and deployment processes. They help push code into production more frequently and reliably.

Infrastructure as Code (IaC):

Tools like Terraform, Pulumi, AWS CloudFormation, Ansible, Chef, and Puppet allow infrastructure to be managed with code. This means servers, networks, databases, and other resources can be created and destroyed automatically using configuration files.

Containerization and Orchestration:

Docker is used to package applications into containers. Kubernetes orchestrates and manages those containers at scale. Helm is used to package Kubernetes applications, while Istio and Linkerd provide service mesh capabilities like traffic routing and security between microservices.

Monitoring and Observability:

Monitoring tools like Prometheus, Grafana, Datadog, New Relic, and the ELK Stack (Elasticsearch, Logstash, Kibana) provide real-time insight into system health, application performance, and logs.

Security and Compliance (DevSecOps):

Tools such as Snyk, Aqua Security, Trivy, and HashiCorp Vault help integrate security into the DevOps pipeline. Open Policy Agent (OPA) is used to enforce policies like access control or configuration standards automatically.

**Collaboration and Project Management:**

Platforms like Slack, Jira, Confluence, Microsoft Teams, and Trello are used to manage communication, track issues, and document development processes across teams.

### **Key Features of Advanced DevOps**

Advanced DevOps includes many features that go beyond the basics:

GitOps is a way to manage infrastructure and application deployment using Git as the single source of truth. This makes rollbacks, audits, and collaboration easier.

Progressive Delivery includes strategies like blue-green deployments, canary releases, and feature flags to minimize risk during software rollouts.

Self-Healing Systems automatically detect and recover from failures using health checks, auto-scaling, and rollback mechanisms.

Immutable Infrastructure means replacing infrastructure entirely with a new version rather than modifying the existing one. This reduces errors and makes systems more predictable.

Policy as Code allows compliance and security policies to be defined and enforced using code, ensuring consistency and reducing human error.

Chaos Engineering involves deliberately injecting failures into the system to test its resilience and improve system robustness. Tools like Chaos Monkey help simulate these failures.

AI and Machine Learning in DevOps help with predictive incident detection, smart alerts, anomaly detection, and optimizing CI/CD workflows.

### **Benefits of Advanced DevOps-**

**Faster Software Delivery:** Advanced DevOps automates many manual processes like building, testing, and deploying code. This speeds up the release cycle, enabling teams to deliver new features and bug fixes much faster than traditional methods.

**Higher Quality and Reliability:** Automated testing and continuous monitoring catch bugs and issues early. With practices like canary releases and blue-green deployments, updates can be rolled out safely with minimal risk of downtime or errors.

**Improved Collaboration:** Advanced DevOps breaks down silos between development, operations, and security teams. Everyone works together using shared tools and processes, which improves communication, reduces misunderstandings, and speeds up problem-solving.

**Scalability:** Using infrastructure as code and container orchestration, organizations can easily scale their applications and infrastructure up or down based on demand. This flexibility helps handle traffic spikes smoothly without overprovisioning resources.

**Better Security (DevSecOps):** Security becomes an integral part of the development pipeline. Automated security scanning, policy enforcement, and secrets management reduce vulnerabilities and compliance risks.

**Cost Efficiency:** By automating routine tasks and optimizing infrastructure usage, Advanced DevOps reduces manual effort and lowers infrastructure costs. Cloud resource management tools help avoid waste and optimize spending.

**Conclusion-** Advanced DevOps transforms traditional software delivery by combining automation, collaboration, and security to enable faster, more reliable releases. It leverages modern tools and practices like CI/CD, infrastructure as code, and observability to improve scalability and resilience. Ultimately, Advanced DevOps empowers organizations to innovate rapidly while maintaining high quality and strong security.

LO Mapping-LO1

## Assignment 2: To launch EC2 instance

**Aim:** To study and perform the setup of AWS EC2 service and launch an EC2 instance

**Theory:** An EC2 instance in AWS is a virtualized compute resource running on Amazon's Elastic Compute Cloud, provisioned using Xen or Nitro hypervisors. It provides configurable vCPUs, RAM, storage through EBS or instance store, and network capacity within a chosen Availability Zone. Instances are launched from Amazon Machine Images and can run Linux or Windows environments. They integrate with AWS services like VPC, IAM, and Auto Scaling for secure and scalable deployments.

### Steps:

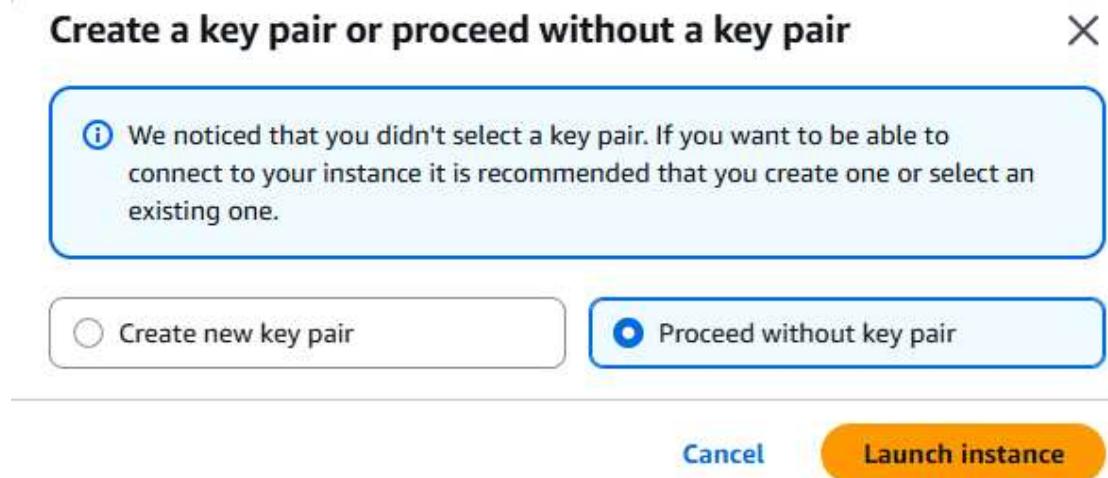
1) Sign in the AWS, Go on dashboard and select EC2, click on instances(running)

The screenshot shows the AWS EC2 dashboard in the Europe (Stockholm) region. The left sidebar includes links for Dashboard, Instances (selected), and Images. The main area displays 'Resources' with counts for Instances (running), Auto Scaling Groups, Capacity Reservations, Dedicated Hosts, Elastic IPs, Instances, Key pairs, Load balancers, Placement groups, Security groups, Snapshots, and Volumes. Below this is a 'Launch instance' button. To the right, there's a 'Service health' section and an 'EC2 cost' summary showing credits remaining (\$120 USD) and days remaining (170). A note says 'Costs in your free plan account are covered by credits.'

2) Select your desired Operating system and assign a name to your instance, then click on launch instance.

The screenshot shows the 'Launch instance' wizard. Step 1: 'Name and tags' where 'Rohit' is entered. Step 2: 'Application and OS Images (Amazon Machine Image)' where 'ubuntu' is selected. Step 3: 'Quick Start' showing options for Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and others. Step 4: 'Summary' showing 1 instance, the selected AMI (ami-042b4708b1d05f512), the t3.micro instance type, and a New security group. Step 5: 'Storage (volumes)' showing 1 volume (8 GiB). Step 6: 'Launch instance' and 'Preview code' buttons at the bottom.

3) Proceed without key pair



4) Click on the instance running and select connect option

The screenshot shows the AWS EC2 Instances page. At the top, it says "Resources" and "EC2 Global View". It lists resources: Instances (running) 1, Auto Scaling Groups 0; Capacity Reservations 0, Dedicated Hosts 0; Elastic IPs 0, Instances 0; Key pairs 0, Load balancers 0; Placement groups 0, Security groups 0; Snapshots 0, Volumes 0. Below this, the "Instances (1)" section shows details for a single instance: Name: Roh, Instance ID: i-0c3cdff7bf11208fb, Instance state: Running, Instance type: t3.micro, Status check: 3/3 checks passed, Alarm status: View alarms. The instance was last updated less than a minute ago. There are buttons for Connect, Instance state dropdown, Actions dropdown, and Launch. A search bar "Find Instance by attribute or tag (case-sensitive)" and a filter "All states" are also present.

5) Select the following and assign a username

**Connect** Info

Connect to an instance using the browser-based client.

**EC2 Instance Connect**  Session Manager  SSH client  EC2 serial console

**Instance ID**  
 i-0c3cdff7bf11208fb (Roh)

Connect using a Public IP  
Connect using a public IPv4 or IPv6 address

Connect using a Private IP  
Connect using a private IP address

Public IPv4 address  
 16.171.224.32

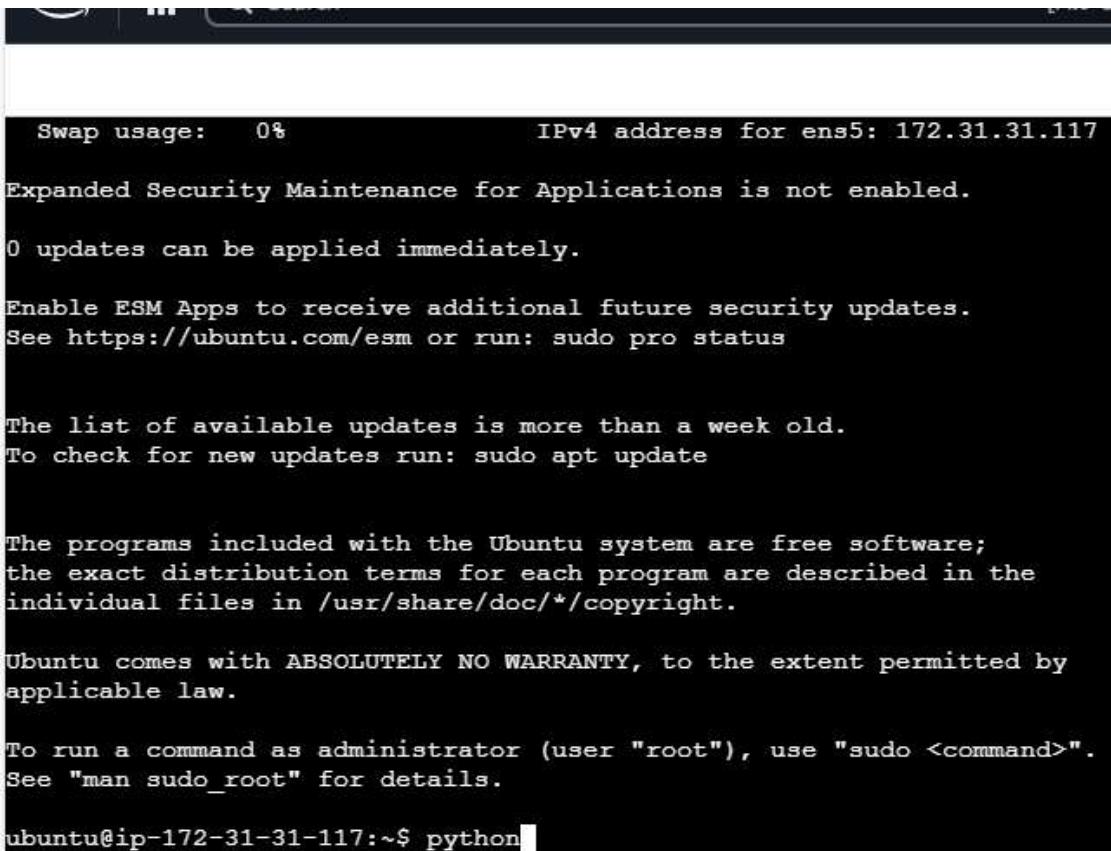
IPv6 address

**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.

ubuntu

**Note:** In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI defines a different default username.

6) EC2 instance launched



```
Swap usage:  0%          IPv4 address for ens5: 172.31.31.117
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-172-31-31-117:~$ python
```

## 7) Terminate the instance once done

The screenshot shows the AWS EC2 Instances page with one instance listed: i-0c3cdff7bf11208fb (Roh). The instance is running. A context menu is open over the instance, with the 'Actions' option selected. The 'Terminate (delete) instance' option is highlighted. Below this, the 'Terminate (delete) instance' dialog is open. It contains a warning message: '⚠️ On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.' It asks if the user is sure they want to terminate the instances. The 'Termination protection' section shows 'Disabled'. There is an option to 'Skip OS shutdown' which is unchecked. At the bottom are 'Cancel' and 'Terminate (delete)' buttons.

Instances (1/1) [Info](#)

Last updated 1 minute ago [Connect](#) [Instance state ▲](#) [Actions ▾](#) [Launch](#)

Find Instance by attribute or tag (case-sensitive)

Instance state = running [X](#) [Clear filters](#)

Name [O](#) [▼](#) | Instance ID | Instance state [▼](#) | Ins

Roh i-0c3cdff7bf11208fb  Running [Q](#) [Q](#) t3.r

Stop instance  
Start instance  
Reboot instance  
Hibernate instance  
Terminate (delete) instance

[Alarm](#) [View all](#)

i-0c3cdff7bf11208fb (Roh)

[Details](#) [Status and alarms](#) [Monitoring](#) [Security](#) [Networking](#) [Storage](#) [Tags](#)

**Terminate (delete) instance** [X](#)

⚠️ On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.

Are you sure you want to terminate these instances?

Instance ID	Termination protection
<input type="checkbox"/> i-0c3cdff7bf11208fb (Roh)	<input checked="" type="checkbox"/> Disabled

To confirm that you want to delete the instances, choose the terminate button below. Instances with termination protection enabled will not be terminated. Terminating the instance cannot be undone.

**Skip OS shutdown**  
This option skips the graceful OS shutdown process. Use only when your instance must be stopped immediately, such as during an emergency or failover.

Skip OS shutdown

[Cancel](#) [Terminate \(delete\)](#)

**Conclusion:** In conclusion, launching an EC2 instance provides a flexible and scalable way to deploy applications in the cloud. With customizable configurations and integration with other AWS services, it enables efficient resource management and reliable performance for a wide range of workloads.

**LO Mapping:** LO1

## Assignment 3: S3 Bucket

**Aim:** To study AWS S3 service and create a bucket for housing static web application.

**Theory:** An S3 bucket in AWS is a storage container used to store and organize objects such as files, images, or backups in Amazon Simple Storage Service. It offers virtually unlimited storage capacity with high durability and availability. Data in S3 can be accessed over the internet and managed with permissions for security. It also supports features like versioning, lifecycle rules, and integration with other AWS services for data processing and distribution.

### Steps:

1) Select S3 buckets in the dashboard and select on create bucket. Add a unique bucket name and proceed to create bucket

The screenshot shows two consecutive pages from the AWS S3 'Create bucket' wizard and the resulting bucket summary page.

**Create Bucket Page:**

- General configuration:**
  - AWS Region:** Europe (Stockholm) eu-north-1
  - Bucket type:** General purpose (selected)
  - Bucket name:** mybucket234342
  - Copy settings from existing bucket - optional:** Only the bucket settings in the following configuration are copied.
- Object Ownership:** Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects in the bucket.

**Bucket Summary Page:**

- General purpose buckets (1/1):** mybucket234342 (Created August 6, 2025, 06:52:57 (UTC+01:00))
- Account snapshot:** Updated daily. Storage Lens provides visibility into storage usage and activity trends.
- External access summary - new:** Updated daily. External access findings help you identify bucket permissions that allow public access or access from other AWS accounts.

## 2)Go to properties

The screenshot shows the 'Properties' tab selected in the navigation bar. Under 'Bucket overview', it displays the AWS Region (Europe (Stockholm) eu-north-1), Amazon Resource Name (ARN) (arn:aws:s3:::mybucket234342), and Creation date (August 6, 2025, 06:52:57 (UTC+01:00)). A 'Bucket Versioning' section is present with a 'Edit' button.

## 4)Enable static hosting

The screenshot shows the 'Edit static website hosting' configuration page. Under 'Static website hosting', the 'Enable' option is selected. Under 'Hosting type', the 'Host a static website' option is selected. A note at the bottom states: 'For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can enable S3 Block Public Access settings for the bucket. For more information, see Using Amazon S3 Block Public Access.'

## 5)Create a HTML file and upload it

The screenshot shows a code editor with the following HTML content:

```
<html>
<head>
<title>Sample file</title>
</head>
<body>
<h1>This is a sample HTML file</h1>
<p>This website is created for understanding the working of S3 buckets in Amazon AWS.</p>
<h2>Hello World</h2>
</body>
</html>
```

**Upload** [Info](#)

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDKs or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

**Files and folders** (1 total, 223.0 B)

All files and folders in this table will be uploaded.

Find by name

Name	Folder	Type	Size
<input checked="" type="checkbox"/> index.html	-	text/html	223.0 B

**Destination** [Info](#)

Destination  
<s3://mybucket234342>

**Upload succeeded**  
For more information, see the **Files and folders** table.

**Summary**

Destination	Succeeded	Failed
<a href="s3://mybucket234342">s3://mybucket234342</a>	1 file, 223.0 B (100.00%)	0 files, 0 B (0%)

[Files and folders](#) [Configuration](#)

**Files and folders** (1 total, 223.0 B)

Find by name

Name	Folder	Type	Size	Status	Error
<a href="#">index.html</a>	-	text/html	223.0 B	Succeeded	-

# This is a sample HTML file

This website is created for understanding the working of S3 buckets in Amazon AWS.

**Hello World**

**Conclusion:** In conclusion, creating an S3 bucket to host a static website offers a simple, cost-effective, and highly available solution for delivering content over the internet. With minimal setup, it eliminates the need for managing servers while ensuring scalability and seamless integration with other AWS services. This makes it an ideal choice for hosting lightweight, static web applications or personal projects.

**LO Mapping:** LO1

# Assignment-4:CodePipeline

**Aim:** To study AWS CodePipeline and deploy web application using CodePipeline.

**Theory:** AWS CodePipeline is a fully managed CI/CD service that automates software release workflows. It connects source control, build tools, test suites, and deployment targets into a unified pipeline. Each stage in the pipeline triggers actions based on code changes or events. This enables faster, more reliable deployments while promoting DevOps best practices.

## Steps:

1) Create a YML file and

```
buildspec.yml - Notepad
File Edit Format View Help
version: 0.2
phases:
  install:
    commands:
      - echo "No dependencies to install"
  build:
    commands:
      -echo "No build stem requires"
  post_build:
    commands:
      -echo "Deployment package ready"
artifacts:
  files:
    - '*'
```

Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).  
Required fields are marked with an asterisk (\*).

1 General

Owner \* Repository name \*

Rohitapc29 / sw  
 sw is available.

Great repository names are short and memorable. How about [effective-system](#)?

Description

0 / 350 characters

2 Configuration

## 2)Create an S3 bucket

The screenshot shows the AWS Management Console with a search bar at the top containing 's3'. The main content area displays the 'Services' section, specifically the 'S3 Scalable Storage in the Cloud' service card. Below it are cards for 'S3 Glacier Archive Storage in the Cloud' and 'AWS Snow Family Large Scale Data Transport'. A sidebar on the left provides navigation links for 'Billing and Management', 'Choose billing plan', 'Home', 'Getting Started', 'Dashboards', and 'Billing and payment'. A central panel titled 'Create bucket' contains the 'General configuration' step. It includes fields for 'Bucket name' (set to 'brightwave'), 'AWS Region' (set to 'Asia Pacific (Mumbai) ap-south-1'), and 'Bucket type' (radio button selected for 'General purpose'). Other options like 'Directory' are shown with their descriptions. At the bottom, there's a 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button and a note about copied settings.

Services

- S3 Scalable Storage in the Cloud
- S3 Glacier Archive Storage in the Cloud
- AWS Snow Family Large Scale Data Transport

Were these results helpful?

Yes  No

Create bucket

**General configuration**

**AWS Region**  
Asia Pacific (Mumbai) ap-south-1

**Bucket type** [Info](#)

General purpose  
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

Directory  
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

**Bucket name** [Info](#)  
brightwave

Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). [Learn More](#)

**Copy settings from existing bucket - optional**  
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: s3://bucket/prefix

### Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

#### **Block all public access**

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

##### **Block public access to buckets and objects granted through new access control lists (ACLs)**

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

##### **Block public access to buckets and objects granted through any access control lists (ACLs)**

S3 will ignore all ACLs that grant public access to buckets and objects.

##### **Block public access to buckets and objects granted through new public bucket or access point policies**

S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

##### **Block public and cross-account access to buckets and objects through any public bucket or access point policies**

S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

**⚠ Turning off block all public access might result in this bucket and the objects within becoming public**

AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

I acknowledge that the current settings might result in this bucket and the objects within becoming public.

### Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

#### Bucket Versioning

Disable

Enable

### Tags - optional (0)

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

No tags associated with this bucket.

[Add new tag](#)

You can add up to 50 tags.

### Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

The screenshot shows the AWS S3 console interface. On the left, a sidebar lists various bucket types: General purpose buckets, Directory buckets, Table buckets, Vector buckets, Access Grants, Access Points (General Purpose Buckets, FSx file systems), Access Points (Directory Buckets), Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and IAM Access Analyzer for S3. The main area displays two buckets under the "General purpose buckets" tab:

Name	AWS Region	Creation date
brightwave	Asia Pacific (Mumbai) ap-south-1	August 12, 2025, 14:21:31 (UTC+05:30)
kaafiuniqueaam	Asia Pacific (Mumbai) ap-south-1	August 5, 2025, 14:45:36 (UTC+05:30)

Below the buckets, there are sections for "Account snapshot" and "External access summary - new". The "Account snapshot" section indicates it is updated daily and provides visibility into storage usage. The "External access summary" section also indicates it is updated daily and helps identify bucket permissions.

The bottom of the screen shows the AWS footer with links to CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

3)Upload the yml and html files to the bucket

Amazon S3 > Buckets > brightwave > Upload

### Upload Info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDKs or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

**Files and folders (2 total, 7.9 KB)**

All files and folders in this table will be uploaded.

Find by name				<	1	>	
<input type="checkbox"/>	Name	Folder	Type	Size			
<input type="checkbox"/>	buildspec.yml.txt	-	text/plain	237.0 B			
<input type="checkbox"/>	index.html	-	text/html	7.7 KB			

**Destination Info**

Destination  
[s3://brightwave](#)

**Upload succeeded**  
For more information, see the **Files and folders** table.

ⓘ After you navigate away from this page, the following information is no longer available.

### Summary

Destination	Succeeded	Failed
s3://brightwave	2 files, 7.9 KB (100.00%)	0 files, 0 B (0%)

**Files and folders** Configuration

**Files and folders (2 total, 7.9 KB)**

Find by name							<	1	>
Name	Folder	Type	Size	Status	Error				
buildspec.yml.txt	-	text/plain	237.0 B	Succeeded	-				

**brightwave Info**

**Bucket overview**

AWS Region: Asia Pacific (Mumbai) ap-south-1  
Amazon Resource Name (ARN): arn:aws:s3:::brightwave  
Creation date: August 12, 2025, 14:21:31 (UTC+0:30)

**Bucket Versioning**

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

**Bucket Versioning**  
Enabled

**Multi-factor authentication (MFA) delete**  
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Block Public Access settings for this account  
Disabled

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

4)Edit server access logging and choose destination appropriately

## Edit server access logging Info

### Server access logging

Log requests for access to your bucket. [Learn more](#)

#### Server access logging

Disable

Enable

**Bucket policy will be updated**

When you enable server access logging, the S3 console automatically updates your bucket policy to include access to the S3 log delivery group.

#### Destination

Specify a destination bucket in the Asia Pacific (Mumbai) ap-south-1 Region. To store your logs under a particular prefix, make sure that you include a slash (/) after the name of the prefix. Otherwise, the prefix will be added to the name of your log files.

s3://brightwave

[Browse S3](#)

Format: s3://<bucket>/<optional-prefix-with-path>

#### Destination Region

Asia Pacific (Mumbai) ap-south-1

## Choose destination

X

### S3 Buckets

Buckets that are not in the same Region as your source bucket (Asia Pacific (Mumbai) ap-south-1) can't be chosen.

### Buckets (1/2)

Find buckets by name



< 1 >

Name	AWS Region	Creation date
brightwave	Asia Pacific (Mumbai) ap-south-1	August 12, 2025, 09:51:31 (UTC+01:00)
kaafuinquenaam	Asia Pacific (Mumbai) ap-south-1	August 5, 2025, 10:15:36 (UTC+01:00)

[Cancel](#)

[Choose destination](#)

Successfully edited server access logging.

X

### Server access logging

[Edit](#)

Log requests for access to your bucket. Use [CloudWatch](#) to check the health of your server access logging. [Learn more](#)

**Server access logging**  
Enabled

**Log object key format**  
[YYYY]-[MM]-[DD]-[hh]-[mm]-[ss]-[UniqueString]

**Destination bucket**  
s3://brightwave

### AWS CloudTrail data events (0) Info

[Configure in CloudTrail](#)

Configure CloudTrail data events to log Amazon S3 object-level API operations in the CloudTrail console. [Learn more](#)

**Name** ▲ | **Access** ▼

No data events

No data events to display.

[Configure in CloudTrail](#)

## 5)Upload the folder and edit bucket policy

Amazon S3 > Buckets > brightwave > T13\_54/ > Upload

Drag and drop files and folders you want to upload here, or choose Add files or Add folder.

**Files and folders (1 total, 3.0 KB)**

All files and folders in this table will be uploaded.

Find by name				
Check	Name	Folder	Type	Size
<input checked="" type="checkbox"/>	T13_54.zip	-	application/x-zip-compressed	3.0 KB

**Destination** [Info](#)

Destination  
s3://brightwave/T13\_54/ [Edit](#)

**Destination details**  
Bucket settings that impact new objects stored in the specified destination.

**Upload succeeded**  
For more information, see the [Files and folders](#) table.

**Summary**

Destination	Succeeded	Failed
s3://brightwave/T13_54/	<a href="#">1 file, 3.0 KB (100.00%)</a>	<a href="#">0 files, 0 B (0%)</a>

**Files and folders** [Configuration](#)

**Files and folders (1 total, 3.0 KB)**

Find by name						
Name	Folder	Type	Size	Status	Error	
<a href="#">T13_54.zip</a>	-	application/x-zip-compre...	3.0 KB	<a href="#">Succeeded</a>	-	

Amazon S3 > Buckets > brightwave > Edit bucket policy

arn:aws:s3:::brightwave

**Policy**

```

1 ▼ {
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Sid": "AllowAmplifyReadAccess",
6             "Effect": "Allow",
7             "Principal": {
8                 "Service": "amplify.amazonaws.com"
9             },
10            "Action": [
11                "s3:GetObject",
12                "s3>ListBucket"
13            ],
14            "Resource": [
15                "arn:aws:s3:::brightwave/",
16                "arn:aws:s3:::brightwave/*"
17            ]
18        }
19    ]
20 }

```

Successfully edited bucket policy.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAmplifyReadAccess",
      "Effect": "Allow",
      "Principal": {
        "Service": "amplify.amazonaws.com"
      },
      "Action": [
        "s3:GetObject",
        "s3>ListBucket"
      ],
      "Resource": [
        "arn:aws:s3::brightwave",
        "arn:aws:s3::brightwave/*"
      ]
    }
  ]
}
```

[Copy](#)

## 6) Host website using AWS amplify

Amazon S3 > Buckets > brightwave

Static website hosting

We recommend using AWS Amplify Hosting for static website hosting

S3 static website hosting

Hosting type

Bucket website endpoint

Bucket website endpoint

Choose create method

Start a manual deployment

Deploy your app

To manually deploy an app from Amazon S3 or a Zip file, select "Deploy without Git"

Start with a template

**Create new app**

App name: app2556 Branch name: staging

Method: Drag and drop, Amazon S3, Any URL

Zip the contents of your build output, not the top level folder

S3 location of objects to host: s3://brightwave/T13\_54/

Save and deploy

**app2556: Overview**

App ID: d162lgwmxe376o

Get to production:

- Add a custom domain
- Enable firewall protections
- Connect new branches

0 of 3 steps complete

Branches: staging (Deployed)

Domain: https://staging.d162lgwmxe376o.amplifyapp.com Last deployment: 0 minutes ago

Deploy updates \* Production branch

**BrightWave Digital** Home About Services Clients Contact

Search services... Search

Home / Welcome

# Grow Your Business with BrightWave

Expert digital marketing solutions tailored to your needs. From SEO to social media, we drive your success.

Contact Us Today

## About BrightWave Digital

At BrightWave, we combine creativity with cutting-edge strategies to help your brand thrive online. Our dedicated team works closely with clients to deliver measurable results and lasting growth.

With years of experience in SEO, PPC, content marketing, and social media management, we bring a full spectrum of services to elevate your digital presence.

7) Go to VPC and select a VPC

The screenshot shows the AWS search interface with the query 'vpc' entered in the search bar. Below the search bar, there's a sidebar with links to Services, Features, Resources, Documentation, Knowledge articles, Marketplace, Blog posts, Events, and Tutorials. The main content area displays three cards: 'VPC' (Isolated Cloud Resources), 'AWS Global View' (providing a global dashboard and search functionality), and 'AWS Firewall Manager' (central management of firewall rules). A 'Show more' link is visible at the top right of the card section. Below this, under 'Features', there are cards for 'Dashboard' (a VPC feature) and 'Route 53 VPCs' (a Route 53 feature). A 'Were these results helpful?' poll is present with 'Yes' and 'No' options. On the right side, there's a sidebar titled 'Description' with a note about no applicable features and a 'Create app' button. At the bottom, the 'VPC dashboard' is shown with sections for 'Create VPC' and 'Launch EC2 Instances'. It lists resources by region: VPCs (Mumbai 1), Subnets (Mumbai 3), Route Tables (Mumbai 1), Internet Gateways (Mumbai 1), Egress-only Internet Gateways (Mumbai 0), NAT Gateways (Mumbai 0), VPC Peering Connections (Mumbai 0), Network ACLs (Mumbai 1), Security Groups (Mumbai 1), Customer Gateways (Mumbai 0), and AWS Network Manager (Mumbai 0). The sidebar also includes 'Service Health', 'Settings' (with options for Block Public Access, Zones, and Console Experiments), 'Additional Information' (with links to VPC Documentation, All VPC Resources, Forums, and Report an Issue), and 'AWS Network Manager' information.

VPC dashboard < Your VPCs

Your VPCs (1) Info

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR	DHCP option set
vpc-0fa6b684c4f24b62f	Available	Off	172.31.0.0/16	-	-	dopt-0bf8258a

Select a VPC above

VPC dashboard < Your VPCs

Your VPCs (1/1) Info

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR	DHCP option set
vpc-0fa6b684c4f24b62f	Available	Off	172.31.0.0/16	-	-	dopt-0bf8258a

vpc-0fa6b684c4f24b62f

Details Resource map CIDRs Flow logs Tags Integrations

**Details**

VPC ID vpc-0fa6b684c4f24b62f	State Available	Block Public Access Off	DNS hostnames Enabled
DNS resolution Enabled	Tenancy default	DHCP option set dopt-0bf8258a9ea14de24	Main route table rtb-0a3162bfa404609c
Main network ACL none	Default VPC	IPv4 CIDR 172.31.0.0/16	IPv6 pool none

CloudShell Feedback

8) Go to AWS code pipeline and build a custom pipeline,choose the same S3 object and deploy

The screenshot shows two views of the AWS interface. The top view is the 'Services' search results page for 'CodePipeline'. It features a prominent 'CodePipeline' card with the subtext 'Release Software using Continuous Delivery'. Below it is a 'Resources' search bar and a callout for 'Introducing resource search'. The bottom view is the 'Choose creation option' step of the 'Create new pipeline' wizard. On the left, a sidebar lists steps from 'Step 1: Choose creation option' to 'Step 7: Review'. The main area shows a 'Category' section with four options: 'Deployment', 'Continuous Integration', 'Automation', and 'Build custom pipeline', with 'Build custom pipeline' selected. Navigation buttons 'Cancel' and 'Next' are at the bottom right.

Services

CodePipeline Release Software using Continuous Delivery

Resources / for a focused search

Introducing resource search

Enable to show cross-region resources for your account in search results. Takes less than 5 minutes to set up.

Go to Resource Explorer

Documentation

Show more

CodePipeline tutorials

User Guide

Troubleshooting CodePipeline

User Guide

AWS CodePipeline

AWS Whitepaper

Were these results helpful?

Yes No

Step 1 Choose creation option

Step 2 Choose pipeline settings

Step 3 Add source stage

Step 4 Add build stage

Step 5 Add test stage

Step 6 Add deploy stage

Step 7 Review

Category

Deployment

Continuous Integration

Automation

Build custom pipeline

Cancel Next

Step 1  
Choose creation option

Step 2  
Choose pipeline settings

Step 3  
Add source stage

Step 4  
Add build stage

Step 5  
Add test stage

Step 6  
Add deploy stage

Step 7  
Review

## Choose pipeline settings Info

Step 2 of 7

### Pipeline settings

#### Pipeline name

Enter the pipeline name. You cannot edit the pipeline name after it is created.

mywebsite

No more than 100 characters

#### Execution mode Info

Choose the execution mode for your pipeline. This determines how the pipeline is run.

- Superseded
- Queued
- Parallel

#### Service role

##### New service role

Create a service role in your account

##### Existing service role

Choose an existing service role from your account

#### Role name

AWSCodePipelineServiceRole-ap-south-1-mywebsite

Type your service role name

- Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

### ▼ Advanced settings

Configure artifact store location, encryption settings, and pipeline variables for your pipeline.

#### Artifact store

##### Default location

Create a default S3 bucket in your account.

##### Custom location

Choose an existing S3 location from your account in the same region and account as your pipeline

#### Encryption key

##### Default AWS Managed Key

Use the AWS managed customer master key for CodePipeline in your account to encrypt the data in the artifact store.

##### Customer Managed Key

To encrypt the data in the artifact store under an AWS KMS customer managed key, specify the key ID, key ARN, or alias ARN.

#### Variables

You can add variables at the pipeline level. You can choose to assign the value when you start the pipeline. [Learn more](#)

No variables defined at the pipeline level in this pipeline.

Add variable

You can add up to 50 variables.

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose creation option

Step 2 Choose pipeline settings

Step 3 Add source stage

Step 4 Add build stage

Step 5 Add test stage

Step 6 Add deploy stage

Step 7 Review

### Add source stage Info

Step 3 of 7

**Source**

**Source provider**  
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

Amazon S3

**Bucket**  
 brightwave X

**S3 object key**  
 T13\_54/T13\_54.zip

Enter the object key. You can include a file path without the delimiter character (/) at the beginning. Include the file extension. Example: SampleApp.zip

Create EventBridge rule to automatically detect source changes  
If disabled, follow AWS documentation to create an EventBridge rule for your source. [Learn more](#) ?

Enable automatic retry on stage failure

[Cancel](#) [Previous](#) [Next](#)

aws | Search [Alt+S]

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose creation option

Step 2 Choose pipeline settings

Step 3 Add source stage

Step 4 Add build stage

Step 5 Add test stage

Step 6 Add deploy stage

Step 7 Review

### Add test stage Info

Step 5 of 7

**Test**

**Test provider**  
Choose how you want to test your application or content. Choose the provider, and then provide the configuration details for that provider.

Enable automatic retry on stage failure

[Cancel](#) [Previous](#) [Skip test stage](#) [Next](#)

[CloudShell](#) [Feedback](#) © 2025, Amazon Web Services, Inc.

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 4  
Add build stage

Step 5  
Add test stage

Step 6  
Add deploy stage

Step 7  
Review

### Deploy

**Deploy provider**

Choose how you want to deploy your application or content. Choose the provider, and then provide the configuration details for that provider.

Amazon S3

**Region**

Asia Pacific (Mumbai)

**Input artifacts**

Choose an input artifact for this action. [Learn more](#)

SourceArtifact X  
Defined by: Source

No more than 100 characters

**Bucket**

Q. brightwave X

**S3 object key**

T13\_54/T13\_54.zip

Enter the object key. You can include a file path without the delimiter character (/) at the beginning. Include the file extension. Example: SampleApp.zip

Extract file before deploy  
The deployed artifact will be unzipped before deployment.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1  
Choose creation option

Step 2  
Choose pipeline settings

Step 3  
Add source stage

Step 4  
Add build stage

Step 5  
Add test stage

Step 6  
Add deploy stage

Step 7  
Review

### Review Info

Step 7 of 7

### Step 2: Choose pipeline settings

**Pipeline settings**

**Pipeline name**  
mywebsite

**Pipeline type**  
V2

**Execution mode**  
QUEUED

**Artifact location**  
A new Amazon S3 bucket will be created as the default artifact store for your pipeline

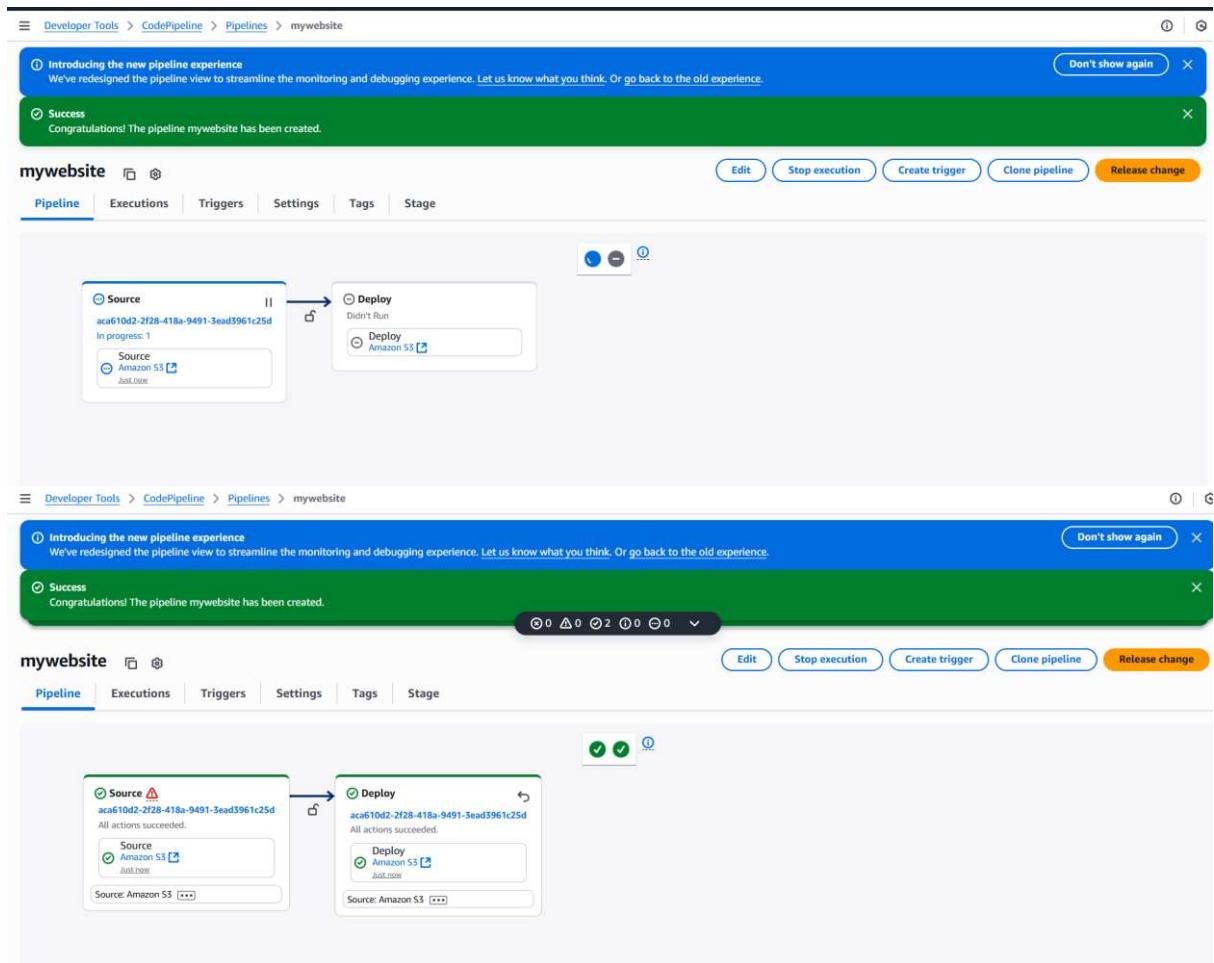
**Service role name**  
AWSCodePipelineServiceRole-ap-south-1-mywebsite

### Step 3: Add source stage

**Source action provider**

Source action provider  
Amazon S3

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#)



**Conclusion:** Thus, we learn that deploying a web application using AWS CodePipeline ensures a fully automated, reliable, and repeatable release process. It minimizes manual intervention, reduces deployment errors, and accelerates delivery cycles. By integrating source, build, and deploy stages, CodePipeline streamlines CI/CD for modern cloud-native applications.

**LO Mapping:** LO1

# ASSIGNMENT-5

**Aim:** To understand Kubernetes Cluster Architecture

**Theory:**

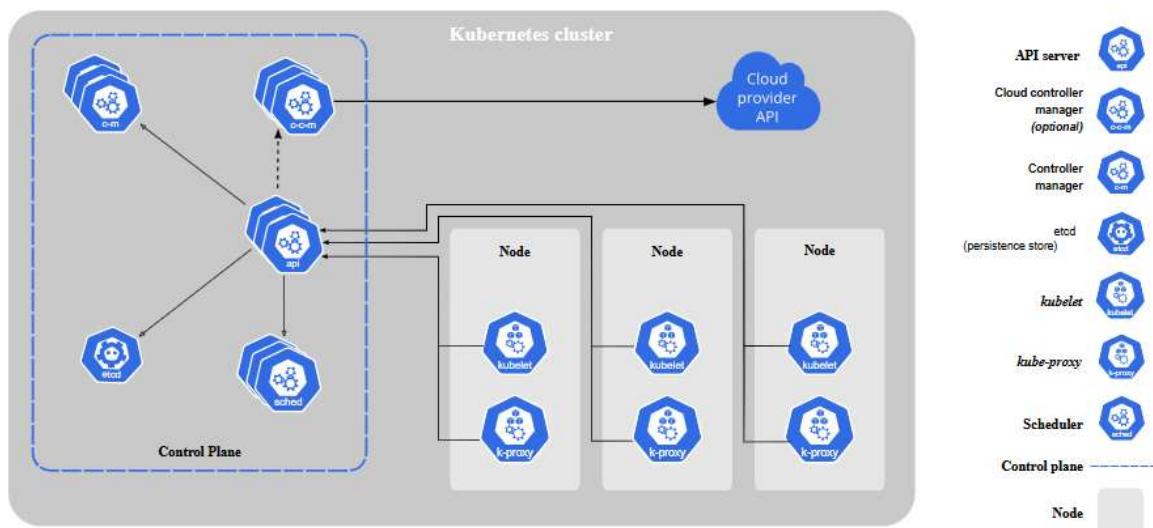
## What is Kubernetes?

Kubernetes is an open-source container orchestration system for automating deployment, scaling, and management of containerized applications. It abstracts infrastructure and provides a unified API to manage workloads across clusters of machines.

Containers are a good way to bundle and run your applications. In a production environment, you need to manage the containers that run the applications and ensure that there is no downtime. For example, if a container goes down, another container needs to start. Wouldn't it be easier if this behaviour was handled by a system?

That's how Kubernetes helps. Kubernetes provides you with a framework to run distributed systems resiliently. It takes care of scaling and failover for your application, provides deployment patterns, and more. For example: Kubernetes can easily manage a canary deployment for your system.

## Kubernetes Cluster components overview



A Kubernetes cluster consists of a control plane and one or more worker nodes. Here's a brief overview of the main components:

## **Control Plane Components**

-Manage the overall state of the cluster:

- kube-apiserver- The core component server that exposes the Kubernetes HTTP API. etcd Consistent and highly-available key value store for all API server data.
- kube-scheduler Looks for Pods not yet bound to a node, and assigns each Pod to a suitable node.
- kube-controller-manager Runs controllers to implement Kubernetes API behavior.
- cloud-controller-manager (optional) Integrates with underlying cloud provider(s).

## **Node Components**

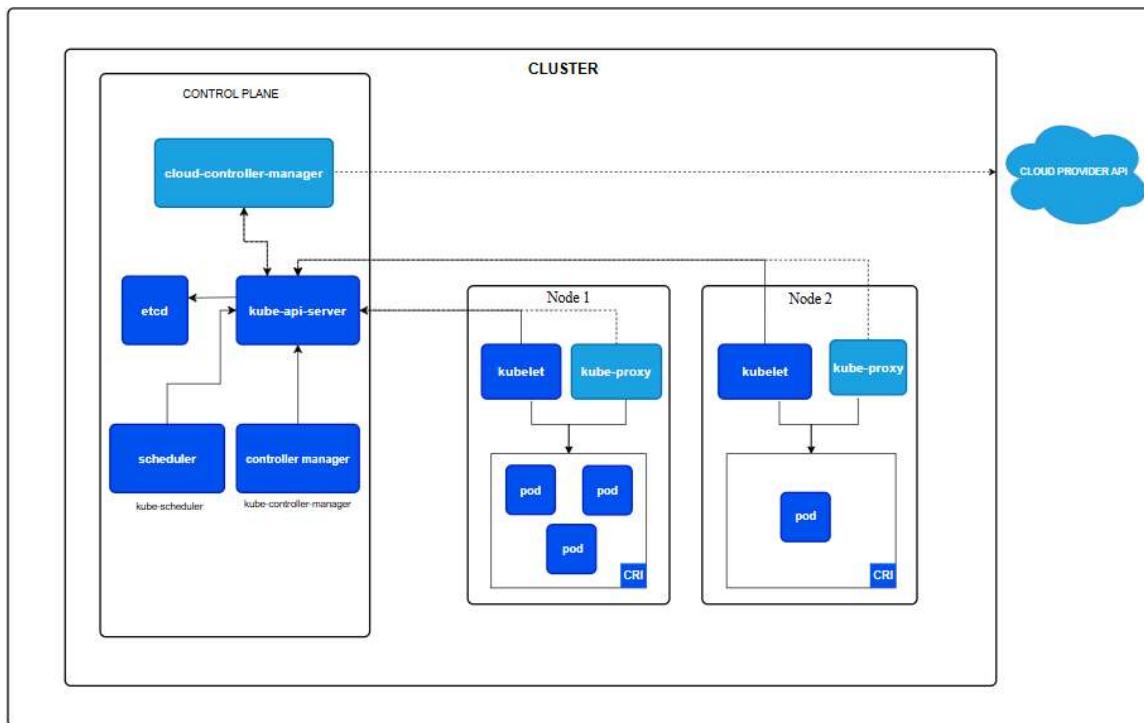
-Run on every node, maintaining running pods and providing the Kubernetes runtime environment:

- kubelet- Ensures that Pods are running, including their containers.
- kube-proxy (optional) Maintains network rules on nodes to implement Services.
- Container runtime Software- responsible for running containers.

## Kubernetes Cluster Architecture

A Kubernetes cluster consists of a control plane plus a set of worker machines, called nodes, that run containerized applications. Every cluster needs at least one worker node in order to run Pods.

The worker node(s) host the Pods that are the components of the application workload. The control plane manages the worker nodes and the Pods in the cluster. In production environments, the control plane usually runs across multiple computers and a cluster usually runs multiple nodes, providing fault-tolerance and high availability.



In the diagram, each node runs `kube-proxy` component. You need a network proxy component on each node to ensure that the Service API and associated behaviours are available on your cluster network. However, some network plugins provide their own, third party implementation of proxying. When you use that kind of network plugin, the node does not need to run `kube-proxy`. The components described are same as above

## Cluster Characteristics

- Control plane components can run on any machine but are typically isolated from user workloads.
- Production clusters often run control plane components across multiple machines for high availability.
- Worker nodes host Pods and must be registered with the control plane.

## **Advantages of Kubernetes**

### 1. Automated Scaling & Self-Healing

Kubernetes can automatically scale applications based on resource usage and restart failed containers without manual intervention.

### 2. Declarative Configuration & Version Control

Infrastructure and application state are defined using YAML/JSON manifests, enabling reproducible deployments and easy rollback.

### 3. Cloud-Agnostic & Extensible

Kubernetes runs on-premises or across public clouds (AWS, Azure, GCP) and supports plugins for networking (CNI), storage (CSI), and runtime (CRI).

## **Disadvantages of Kubernetes**

### 1. Steep Learning Curve

The architecture, terminology, and configuration syntax require significant time and expertise to master.

### 2. Complex Setup & Maintenance

Initial cluster setup, upgrades, and monitoring demand deep operational knowledge and often third-party tooling.

### 3. Resource Overhead

Running Kubernetes itself consumes CPU, memory, and storage — which can be inefficient for small-scale applications.

**Conclusion-** Thus we understand that Kubernetes is a robust, cloud-agnostic orchestration platform designed to manage containerized applications at scale. Its architecture is modular and resilient, built around a control plane that governs cluster state and worker nodes that execute workloads. Through components like kube-apiserver, etcd, kube-scheduler, and kubelet, it ensures declarative configuration, automated scaling, fault tolerance, and seamless networking

## **LO Mapping- LO2**

## Assignment-6

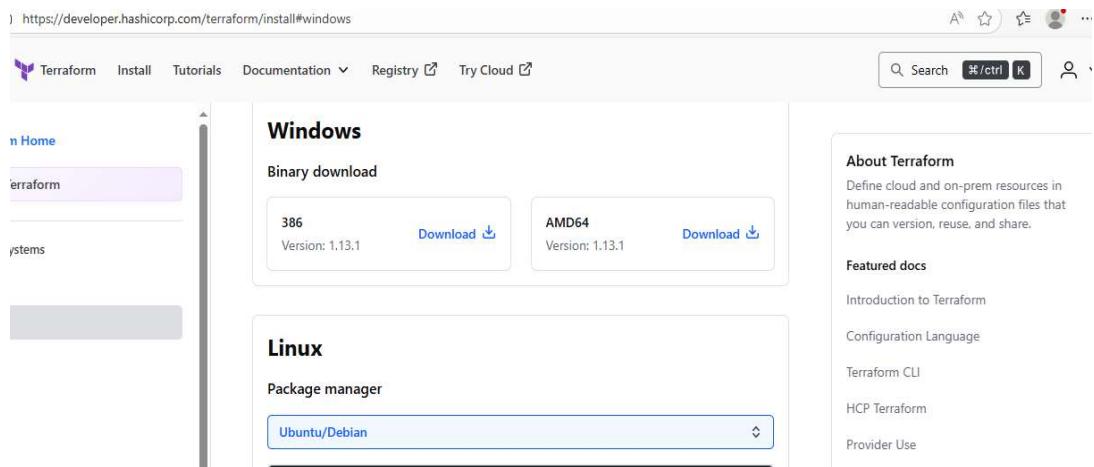
**Aim:** To understand terraform lifecycle and to build , change , and destroy AWS infrastructure using Terraform.

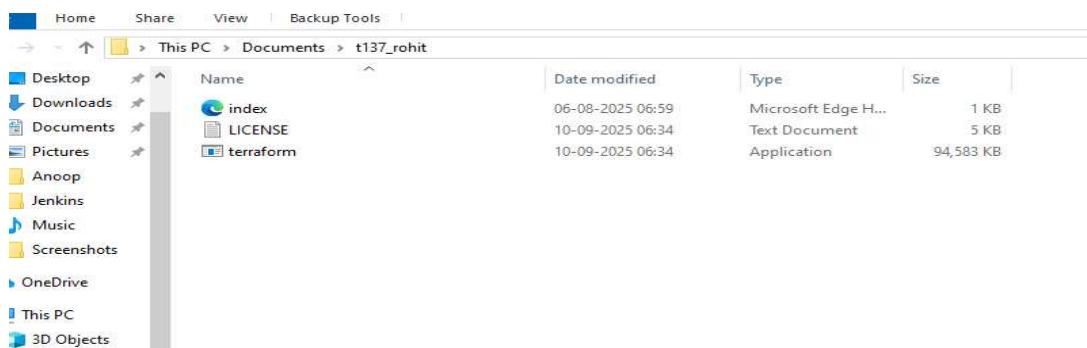
### Theory:

Terraform is an open-source Infrastructure as Code (IaC) tool developed by HashiCorp that enables users to define, provision, and manage cloud infrastructure using declarative configuration files. It supports multiple cloud providers like AWS, Azure, Google Cloud, and others, allowing for consistent infrastructure deployment across environments. Terraform uses a high-level configuration language called HCL (HashiCorp Configuration Language) to describe the desired infrastructure state, which it then translates into API calls to create and manage resources. Its execution plan feature previews changes before applying them, ensuring safe and predictable updates.

One of Terraform's key strengths is its ability to manage dependencies and maintain state. The state file records the current infrastructure configuration, enabling Terraform to detect changes and apply only the necessary updates. It also supports modules, which promote reusability and modular design, making complex infrastructure easier to manage. Terraform's provider ecosystem allows integration with a wide range of services, and its workflow—init, plan, apply—ensures a structured approach to infrastructure changes. Overall, Terraform simplifies infrastructure management, enhances scalability, and supports DevOps practices by automating provisioning and reducing manual errors.

### Steps:





A screenshot of the AWS IAM console homepage. The URL is https://eu-north-1.console.aws.amazon.com/console/home?ncf=n\_si&region=eu-north-1&src=header-signin#. The page displays the Services section with three main items:

- IAM**: Manage access to AWS resources
- IAM Identity Center**: Manage workforce user access to multiple AWS accounts and cloud applications
- Resource Access Manager**: Share AWS resources with other accounts or AWS Organizations

The sidebar on the left includes links for Services, Features, and Recents, such as Billing, DynamoDB, Lambda, IAM, and EC2.

A screenshot of the AWS IAM Users page. The URL is https://eu-north-1.console.aws.amazon.com/iam/users?region=eu-north-1. The page title is "Users (0) Info". It states: "An IAM user is an identity with long-term credentials that is used to interact with AWS in an account." A search bar is present at the top. The main table header includes columns: User name, Path, Group, Last activity, MFA, Password age, and Consol. A message at the bottom says "No resources to display".

## Specify user details

**User details**

User name

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = . @ \_ - (hyphen)

Provide user access to the AWS Management Console - *optional*  
If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

ⓘ If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keypairs, you can generate them after you create this IAM user. [Learn more](#)

[Cancel](#) [Next](#)

## Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

**Permissions options**

Add user to group  
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

Copy permissions  
Copy all group memberships, attached managed policies, and inline policies from an existing user.

Attach policies directly  
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

ⓘ Get started with groups  
Create a group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. [Learn more](#) [Create group](#)

▶ Set permissions boundary - *optional*

[Cancel](#) [Previous](#) [Next](#)

## Create user group

Create a user group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. [Learn more](#)

**User group name**  
Enter a meaningful name to identify this group.

terraformgroup

Maximum 128 characters. Use alphanumeric and '+,-,@-\_` characters.

### Permissions policies (1/1074)

Filter by Type

Policy name	Type	Use...	Description
<input checked="" type="checkbox"/> AdministratorAccess	AWS managed	None	Provides full access to AWS services
<input type="checkbox"/> AdministratorAcce...	AWS managed	None	Grants account administrative per...
<input type="checkbox"/> AdministratorAcce...	AWS managed	None	Grants account administrative per...
<input type="checkbox"/> AIOpsAssistantPolicy	AWS managed	None	Provides ReadOnly permissions re...
<input type="checkbox"/> AIOpsConsoleAdmi...	AWS managed	None	Grants full access to Amazon AI O...

[Create policy](#)

[Cancel](#) [Create user group](#)

roh29@zz > Create access key

### Access key best practices & alternatives

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

**Use case**

**Command Line Interface (CLI)**  
You plan to use this access key to enable the AWS CLI to access your AWS account.

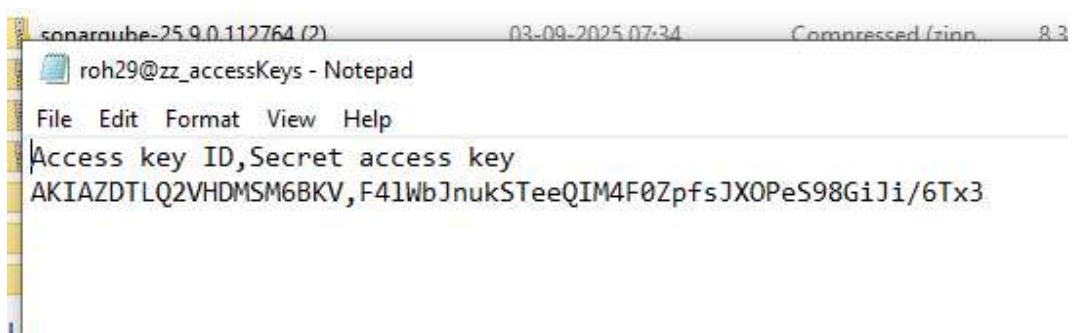
**Local code**  
You plan to use this access key to enable application code in a local development environment to access your AWS account.

**Application running on an AWS compute service**  
You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.

The screenshot shows the AWS IAM Access Key details page. At the top, there are two buttons: 'AKIAZDTLQ2VHDM6BKV' and '\*\*\*\*\* Show'. Below this, a section titled 'Access key best practices' lists four items:

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

Below the best practices, a note says 'For more details about managing access keys, see the [best practices for managing AWS access keys](#)'. At the bottom right are 'Download .csv file' and 'Done' buttons.



The screenshot shows a Windows command prompt window. The history of commands is as follows:

```
Microsoft Windows [Version 10.0.19045.6216]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lab1002>cd Desktop
C:\Users\lab1002\Desktop>mkdir myFolder
C:\Users\lab1002\Desktop>cd myFolder
C:\Users\lab1002\Desktop\myFolder>echo >.firstec2.tf
C:\Users\lab1002\Desktop\myFolder>notepad firstec2.tf
C:\Users\lab1002\Desktop\myFolder>notepad .firstec2.tf
C:\Users\lab1002\Desktop\myFolder>
```

EC2 > Instances > Launch an instance

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

Recents Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux Deepin

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI  
ami-0c4fc5dcabc9df21d (64-bit (x86), uefi-preferred) / ami-07ca8bbd22d87429f (64-bit (Arm), uefi)  
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t3.micro

Family: t3 2 vCPU 1 GiB Memory Current generation: true  
On-Demand Ubuntu Pro base pricing: 0.0143 USD per Hour  
On-Demand RHEL base pricing: 0.0396 USD per Hour  
On-Demand SUSE base pricing: 0.0108 USD per Hour  
On-Demand Linux base pricing: 0.0108 USD per Hour  
On-Demand Windows base pricing: 0.02 USD per Hour

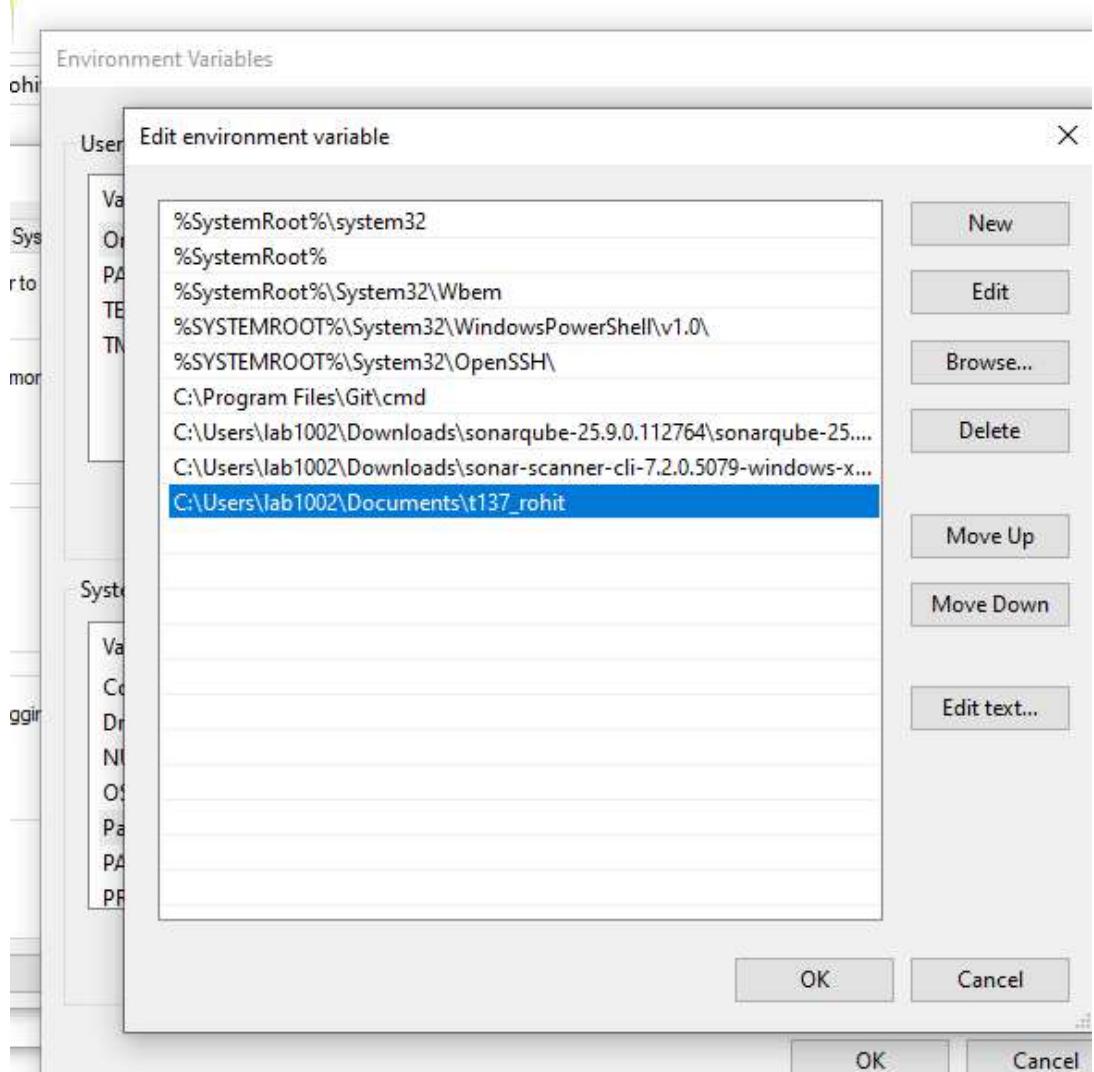
Additions to the AMI for t3.micro

\*firstec2 - Notepad

```
File Edit Format View Help
provider "aws" {
    region      = "us-east-1"
    access_key  = "AKIAZDTLQ2VHDMSM6BKV"
    secret_key  = "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"
}

resource "aws_instance" "myec2" {
    ami          = "ami-0c4fc5dcabc9df21d"
    instance_type = "t3.micro"

    tags = {
        Name = "ec2-created-from-terraform"
    }
}
```



```

Administrator: Command Prompt
Microsoft Windows [Version 10.0.19045.6216]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lab1002>cd Desktop

C:\Users\lab1002\Desktop>cd myFolder

C:\Users\lab1002\Desktop\myFolder>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v6.12.0...
- Installed hashicorp/aws v6.12.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

C:\Users\lab1002\Desktop\myFolder>

```

```

C:\Users\lab1002\Desktop\myFolder>terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are
following symbols:
+ create

Terraform will perform the following actions:

# aws_instance.myec2 will be created
+ resource "aws_instance" "myec2" {
    + ami                                = "ami-0c4fc5dcabc9df21d"
    + arn                                = (known after apply)
    + associate_public_ip_address        = (known after apply)
    + availability_zone                  = (known after apply)
    + disable_api_stop                   = (known after apply)
    + disable_api_termination           = (known after apply)
    + ebs_optimized                     = (known after apply)
    + enable_primary_ipv6               = (known after apply)
    + force_destroy                      = false
    + get_password_data                 = false
    + host_id                            = (known after apply)
    + host_resource_group_arn            = (known after apply)
    + iam_instance_profile              = (known after apply)
    + id                                 = (known after apply)
    + instance_initiated_shutdown_behavior = (known after apply)
    + instance.lifecycle

```

**Conclusion-** Thus terraform simplifies the process of building, changing, and destroying AWS infrastructure using Infrastructure as Code. By following its lifecycle -init, plan, apply, and destroy—you can manage resources efficiently, automate deployments, and maintain consistency across environments.

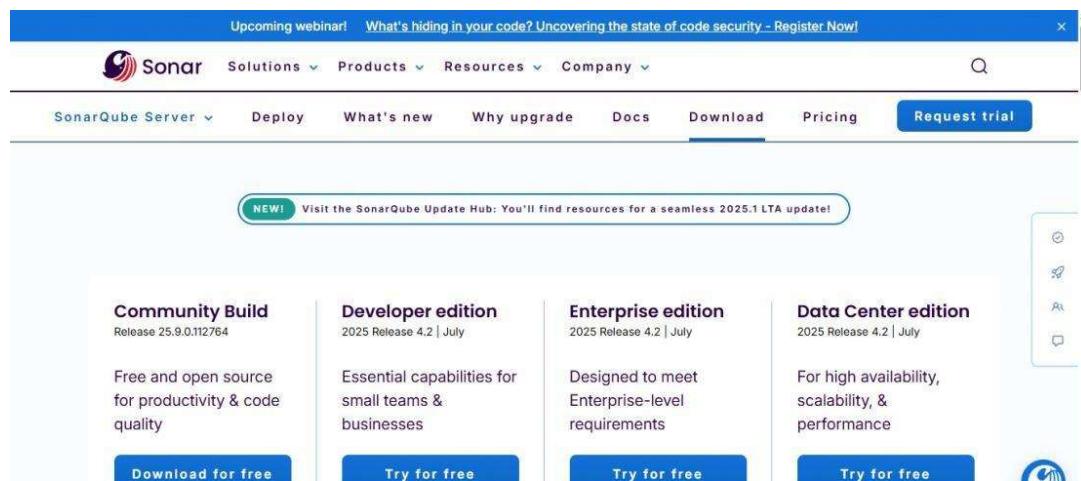
**LO Mapping:** LO3

# Assignment-7

**Aim:** To perform static analysis on python programs using SonarQube SAST process

**Theory:** SonarQube is a static application security testing (SAST) tool that analyzes source code without executing it, helping developers identify vulnerabilities early in the development cycle. It scans for issues like SQL injection, cross-site scripting (XSS), insecure API usage, and hardcoded secrets by applying rule-based checks across multiple languages. Integrated into CI/CD pipelines, SonarQube enforces security and quality gates automatically, ensuring that only safe and maintainable code gets deployed. Its dashboard provides detailed insights and remediation guidance, while IDE plugins allow developers to fix issues directly during coding. By combining static analysis with continuous integration, SonarQube strengthens code security and reduces technical debt.

## Steps:



The screenshot shows the SonarQube website homepage. At the top, there's a navigation bar with links for 'Upcoming webinar', 'What's hiding in your code? Uncovering the state of code security - Register Now!', 'Sonar' logo, 'Solutions', 'Products', 'Resources', 'Company', a search bar, and a 'Request trial' button. Below the navigation is a banner with the text 'Visit the SonarQube Update Hub: You'll find resources for a seamless 2025.1 LTA update!' and a 'NEW!' badge. The main content area features four cards representing different product editions:

- Community Build**: Release 25.9.0.112764. Description: Free and open source for productivity & code quality. Buttons: 'Download for free' (disabled), 'Try for free'.
- Developer edition**: 2025 Release 4.2 | July. Description: Essential capabilities for small teams & businesses. Buttons: 'Try for free' (disabled), 'Download only'.
- Enterprise edition**: 2025 Release 4.2 | July. Description: Designed to meet Enterprise-level requirements. Buttons: 'Try for free' (disabled), 'Download only'.
- Data Center edition**: 2025 Release 4.2 | July. Description: For high availability, scalability, & performance. Buttons: 'Try for free' (disabled), 'Download only'.

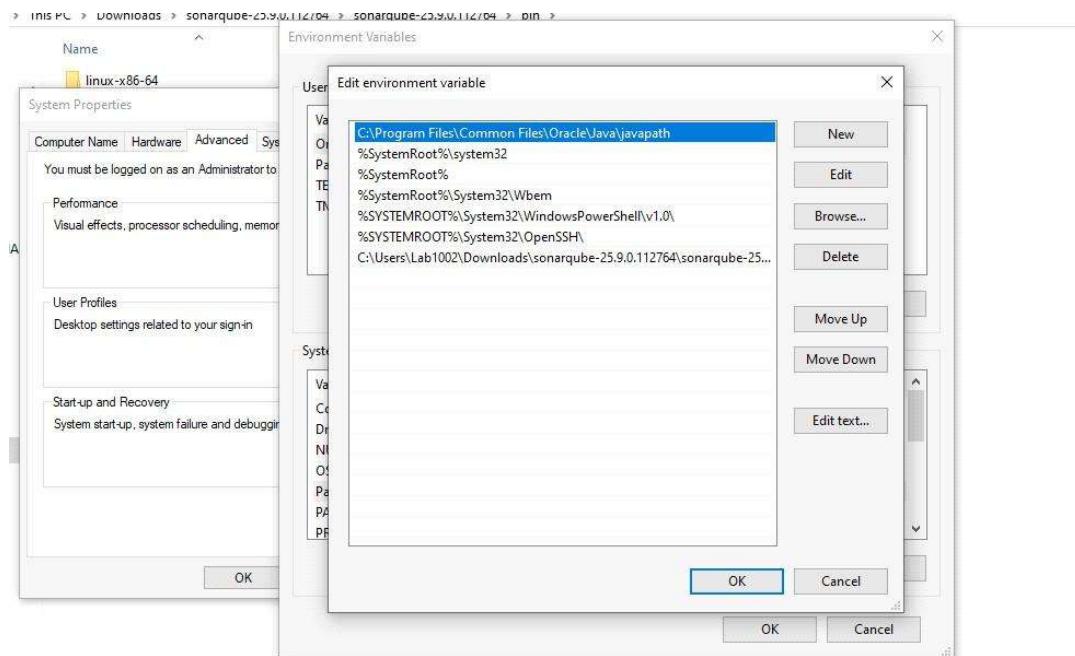
Below this, there's a 'Learn more' link and a 'Download now' button.


The screenshot shows a Windows File Explorer window with the path 'This PC > Downloads > sonarqube-25.9.0.112764 > sonarqube-25.9.0.112764 > bin'. The window displays a list of files and folders:

Name	Date modified	Type	Size
linux-x86-64	03-09-2025 06:42	File folder	
macos-universal-64	03-09-2025 06:42	File folder	
windows-x86-64	03-09-2025 06:42	File folder	
windows-license	03-09-2025 06:42	File folder	
elasticsearch	03-09-2025 06:42	File	1 KB

The left sidebar shows 'Quick access' with various desktop icons like Jenkins, Roshni Devnani, and Screenshots, and network drives like OneDrive, This PC, and Network.



```

Administrator: C:\Windows\System32\cmd.exe - StartSonar.bat

C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764>cd bin\

C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764>sonar-scanner
'sonar-scanner' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764>sonar-scanner
'sonar-scanner' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764>cd windows-x86-64

C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764>StartSonar...
Starting SonarQube...
2025.09.03 06:49:36 INFO [app][o.s.a.AppFileSystem] Cleaning or creating temp directory C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764\temp
2025.09.03 06:49:37 INFO [app][o.s.a.EsSettings] Elasticsearch listening on [HTTP: 127.0.0.1:9001, TCP: 127.0.0.1:{}]
2025.09.03 06:49:37 INFO [app][o.s.a.ProcessLauncherImpl] Launch process[ELASTICSEARCH] from [C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764\elasticsearch]: C:\Program Files\Java\jdk-21\bin\java -Xms4m -Xmx64m -XX:+UseSerialGC -Dcli.name=server -Dcli.script=.bin/elasticsearch -Dcli.libs=lib/tools/server-cli -Des.path.home=C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764\temp\conf\es -Des.distribution.type=tar -cp C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764\elasticsearch\lib\*;C:\Users\Lab1002\Downloads\sonarqube-25.9.0.112764\sonarqube-25.9.0.112764\elasticsearch\lib\launcher\* org.elasticsearch.launcher.CliToolLauncher
2025.09.03 06:49:37 INFO [app][o.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running
Standard Commons Logging discovery in action with spring-jcl: please remove commons-logging.jar from classpath in order to avoid potential conflicts

```

```

Microsoft Windows [Version 10.0.19045.6216]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Lab1002>sonar-scanner
07:00:34.901 INFO Scanner configuration file: C:\Users\Lab1002\Downloads\sonar-scanner-cli-7.2.0.5079-windows-x64\sonar-scanner-7.2.0.5079-windows-x64\bin\..\conf\sonar-scanner.properties
07:00:34.948 INFO Project root configuration file: NONE
07:00:35.075 INFO SonarScanner CLI 7.2.0.5079
07:00:35.077 INFO Windows 10 10.0 amd64

```

arguments....

**SonarScanner for Maven**  
... is recommended as the default scanner for Maven projects ...

**Installing the SonarScanner for .NET**  
... flavor used to compile the Scanner for .NET ( ...

**Running the CFamily analysis**  
... section to pick the suitable scanner variant, and refer ...

**Bitbucket Cloud integration**  
... Passing project names to the scanner Because of the nature ...

**Getting started with .NET**  
... NET Framework version of the scanner you will need ...

**Adding SonarQube Server analysis to a Jenkins job**  
can specify the corresponding SonarQube installation name configured in ...

**Adding analysis to your Azure pipeline for a .NET project**  
Adding analysis to your Azure pipeline for a .NET ...

**Verifying the analysis scope of a project**  
Verifying the analysis scope of a project This

Latest | Analyzing source code | Scanners | SonarScanner CLI

START FREE

## SonarScanner CLI

SonarScanner	Issue Tracker	Show fewer ^
<b>7.2</b>	<b>2025-07-21</b>	Restore ability to run the scanner with Java 11, update dependencies Download scanner for: Linux x64 Linux AArch64 Windows x64 macOS x64 macOS AArch64 Docker Any (Requires a pre-installed JVM) Release notes
<b>7.1</b>	<b>2025-03-21</b>	Support for SonarQube Cloud regions Download scanner for: Linux x64 Linux AArch64 Windows x64 macOS x64 macOS AArch64 Docker Any (Requires a pre-installed JVM) Release notes

### On this page

- Configuring your project
- Running SonarScanner CLI from the zip file
- Running SonarScanner CLI from the Docker image
- Scanning C, C++, or Objective-C projects
- Sample projects
- Alternatives to sonar-project.properties
- Alternate analysis directory
- Advanced configuration
- Troubleshooting

03 September 2025  
Wednesday



Log in to SonarQube

Login \*

Password \*

Go back **Log in**



Log in to SonarQube

Login \*

Password \*

Go back **Log in**

⚠️ Embedded database should be used for evaluation purposes only. It doesn't support scaling, upgrading to a new SonarQube Server version, or migration to another database engine. [Learn more](#)

ⓘ The way in which security, reliability, and maintainability counts and ratings are calculated has changed. [Learn more in SonarQube documentation](#)

 Projects Issues Rules Quality Profiles Quality Gates Administration More x

Search ? A

SonarQube™ technology is powered by [SonarSource SA](#). Community Build v25.9.0.112764 - MQR MODE

LGPL v3 Community Documentation Plugins Web API

⚠️ Embedded database should be used for evaluation purposes only. It doesn't support scaling, upgrading to a new SonarQube Server version, or migration to another database engine. [Learn more](#)

 Projects Issues Rules Quality Profiles Quality Gates Administration More x

Search ? A

## How do you want to create your project?

Do you want to benefit from all of SonarQube Community Build's features (like repository import and Pull Request decoration)?

### Create your project from your favorite DevOps platform.

First, you need to set up a DevOps platform configuration.



Are you just testing or have an advanced use-case?

Create a local project

1 of 2

## Create a local project

Project display name \* ⓘ

MyProject

Project key \* ⓘ

MyProject

Main branch name \*

main

The name of your project's default branch [Learn More](#) ⓘ[Cancel](#)[Next](#)SonarQube™ technology is  
powered by [SonarSource SA](#) ⓘ

Community Build v25.9.0.112764 • MOR MODE

2 of 2

## Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code. This helps you focus attention on the most recent changes to your project, enabling you to follow the Clean as You Code methodology. Learn more: [Defining New Code](#) ⓘ

Choose the baseline for new code for this project

 Use the global setting[Previous version](#)

Any code that has changed since the previous version is considered new code.

Recommended for projects following regular versions or releases.

 Define a specific setting for this project [Previous version](#)

Any code that has changed since the previous version is considered new code.

⚠️ Embedded database should be used for evaluation purposes only. It doesn't support scaling, upgrading to a new SonarQube Server version, or migration to another database engine. [Learn more](#)

SonarQube community Projects Issues Rules Quality Profiles Quality Gates Administration More ▾

MyProject Bind project / main ?

Overview Issues Security Hotspots Code Measures Activity Project Settings ▾ Project Information

With Jenkins With GitHub Actions With Bitbucket Pipelines

With GitLab CI With Azure Pipelines Other CI  
SonarQube Community Build integrates with your workflow no matter which CI tool you're using.

Locally  
Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI environment.

SonarQube™ technology is powered by [SonarSource SA](#) Community Build v25.9.0.112764 - MQR MODE

LGPL v3 □ Community □ Documentation □ Plugins □ Web API

⚠️ Embedded database should be used for evaluation purposes only. It doesn't support scaling, upgrading to a new SonarQube Server version, or migration to another database engine. [Learn more](#)

SonarQube community Projects Issues Rules Quality Profiles Quality Gates Administration More ▾

MyProject Bind project / main ?

Overview Issues Security Hotspots Code Measures Activity

Analysis Method / Locally

## Analyze your project

We initialized your project on SonarQube Community Build, now it's up to you to launch analyses!

1 Provide a token

Generate a project token Use existing token

Token name\* ⓘ Analyze "MyProject"

Expires in 30 days Generate

ⓘ Please note that this token will only allow you to analyze the current project. If you want to use the same token to analyze multiple projects, you need to generate a global token in your [user account](#). See the [documentation](#) for more information.

The token is used to identify you when an analysis is performed. If it has been compromised, you

Analysis Method / Locally

## Analyze your project

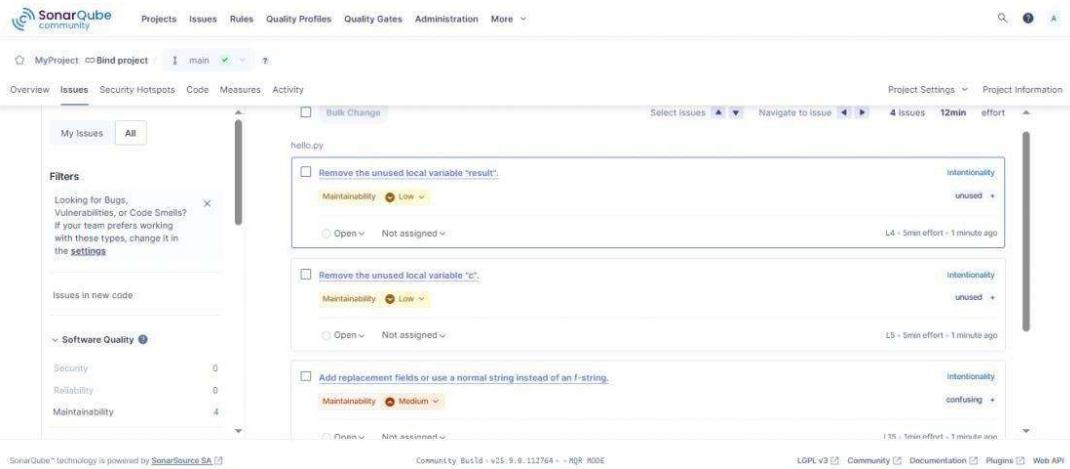
We initialized your project on SonarQube Community Build, now it's up to you to launch analyses!

### 1 Provide a token

Analyze "MyProject": sqp\_45dace35594da0440a16d0207430c53398b7ca37 

The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point in time in your [user account](#).

[Continue](#)



The screenshot shows the SonarQube community interface. At the top, there are navigation links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, and More. Below the navigation, the project 'MyProject' is selected. The main area is divided into sections: Overview, Issues, Security Hotspots, Code, Measures, and Activity. The Issues section is currently active. It shows a list of issues for the file 'hello.py'. There are three issues listed:

- Remove the unused local variable "result". (Maintainability: Low, Intentionality: unused)
- Remove the unused local variable "c". (Maintainability: Low, Intentionality: unused)
- Add replacement fields or use a normal string instead of an f-string. (Maintainability: Medium, Intentionality: confusing)

On the left side, there are filters for looking for bugs, vulnerabilities, or code smells, and sections for Software Quality (Security, Reliability, Maintainability) and Issues in new code.

```
* Any directory in the file path is named: "doc", "docs", "test" or "tests"
* Any directory in the file path has a name ending in "test" or "tests"

INFO: Start fetching files for the text and secrets analysis
INFO: Using JGit to retrieve untracked files
WARNING: Retrieving only language associated files, make sure to run the analysis inside a git repository to make inclusions specified via 'sonar.text.inclusions'
INFO: Starting the text and secrets analysis
INFO: 1 source file to be analyzed for the text and secrets analysis
INFO: 1/1 source file has been analyzed for the text and secrets analysis
INFO: Sensor TextAndSecretsSensor [text] (done) | time=4559ms
INFO: -----
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=24ms
INFO: -----
INFO: Gather SCA dependencies on project
INFO: Dependency analysis skipped
INFO: SCM Publisher No SCM system was detected. You can use the 'sonar.scm.provider' property to explicitly specify
INFO: CPD Executor Calculating CPD for 1 file
INFO: CPD Executor CPD calculation finished (done) | time=19ms
INFO: Analysis report generated in 383ms, dir size=250.5 kB
INFO: Analysis report compressed in 42ms, zip size=29.0 kB
INFO: Analysis report uploaded in 790ms
INFO: ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=MyProject
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis port
INFO: More about the report processing at http://localhost:9000/api/ce/task?id=540adf3f-4ab9-4d14-aa38-8fb43cdcab2
INFO: Analysis total time: 13.962 s
INFO: SonarScanner Engine completed successfully

C:\Users\Lab1002\Desktop\CICD>
```

```
File Edit View Insert Options Window Help
import math

def add_numbers(a,b):
    result=a+b
    c=10
    return a+b

def divide_numbers(x,y):
    return x/y

def main():
    num1=10
    num2=0
    print(f"Sum of two nos is: ",add_numbers(num1,num2))
    print(f"Div is: ", divide_numbers(num1,num2))

main()
```

**Conclusion:** Thus, we ensure that Python code undergoes early and thorough security checks using SonarQube's SAST process, minimizing vulnerabilities before runtime. We strengthen code integrity by enforcing consistent standards across the development lifecycle. Ultimately, we achieve safer, more maintainable Python applications with reduced risk and technical debt.

**LO Mapping:** LO4

# ASSIGNMENT 8

**AIM:** To understand Continuous Monitoring using Nagios.

## **THEORY:**

**Nagios Core** is a powerful open-source monitoring tool used to monitor systems, networks, and infrastructure. It provides real-time alerting and visualization for servers, services, switches, applications, and more.

### **Key Features:**

- Server & network monitoring
- Email/SMS alerts for failures
- Web interface for visualization
- Extensible with plugins and custom checks

### **Prerequisites**

- A Linux machine (Ubuntu/Debian-based system)
- Root or **sudo** privileges
- Apache web server
- Basic networking knowledge

## **STEPS:**

### 1. Update System & Install Dependencies

```
sudo apt update  
sudo apt install -y apache2 php libapache2-mod-php build-essential \  
libgd-dev unzip curl wget libssl-dev daemon make gcc
```

### 2. Create Nagios User and Group

```
sudo useradd nagios  
sudo groupadd nagcmd  
sudo usermod -aG nagcmd nagios  
sudo usermod -aG nagcmd www-data
```

### 3. Download and Compile Nagios Core

```
cd /tmp  
wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.1.tar.gz  
tar -xzf nagios-4.5.1.tar.gz  
cd nagios-4.5.1  
.configure --with-command-group=nagcmd  
make all  
sudo make install
```

```
sudo make install-commandmode  
sudo make install-init  
sudo make install-config  
sudo make install-webconf
```

#### . Set Up Web Interface Authentication

Create a login user for the web interface (e.g. `admin`):  
sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users admin  
Note: Choose a strong password and remember it.

### 5. Install Nagios Plugins

```
cd /tmp  
wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz  
tar -xzf nagios-plugins-2.3.3.tar.gz  
cd nagios-plugins-2.3.3  
.configure --with-nagios-user=nagios --with-nagios-group=nagios  
make  
sudo make install
```

### 6. Enable Required Apache Modules

```
sudo a2enmod rewrite cgi  
sudo systemctl restart apache2
```

### 7. Start Nagios Service

```
bash  
sudo systemctl enable nagios  
sudo systemctl start nagios
```

## Accessing the Nagios Web Interface

Steps:

### 1. Get your server IP:

```
ip a | grep inet
```

Example output: `192.168.4.104`

### 2. Open a browser and visit:

```
http://192.168.4.104/nagios
```

### 3. Login with:

Username: `admin`

Password: (the one you set earlier)

```
lab1002@lab1002-HP-280-G3-MT:~$ sudo apt update
Hit:1 https://download.docker.com/linux/ubuntu bionic InRelease
Hit:2 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu bionic InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:5 http://in.archive.ubuntu.com/ubuntu bionic-backports InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
47 packages can be upgraded. Run 'apt list --upgradable' to see them.
lab1002@lab1002-HP-280-G3-MT:~$ sudo apt install -y autoconf gcc make unzip apache2 php libapache2-mod-php \
> libgd-dev libmcrypt-dev libssl-dev daemon wget bc gawk dc build-essential \
> snmp libnet-snmp-perl gettext
Reading package lists... Done
Building dependency tree
Reading state information... Done
bc is already the newest version (1.07.1-2).
bc set to manually installed.
build-essential is already the newest version (12.4ubuntu1).
```

```
lab1002@lab1002-HP-280-G3-MT:~$ sudo usermod -a -G nagcmd nagios
lab1002@lab1002-HP-280-G3-MT:~$ sudo usermod -a -G nagcmd www-data
lab1002@lab1002-HP-280-G3-MT:~$ ip a | grep inet
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
    inet 192.168.4.104/24 brd 192.168.4.255 scope global dynamic noprefixroute enp4s0
        inet6 fe80::59a0:c15b:f605:a620/64 scope link noprefixroute
        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
lab1002@lab1002-HP-280-G3-MT:~$ ls /etc/apache2/sites-enabled/nagios.conf
/etc/apache2/sites-enabled/nagios.conf
lab1002@lab1002-HP-280-G3-MT:~$ sudo systemctl reload apache2
lab1002@lab1002-HP-280-G3-MT:~$ ls /usr/local/nagios/share
angularjs           includes          media
bootstrap-3.3.7     index.php       robots.txt
config.inc.php      infobox.html   side.php
contexthelp         js              spin
d3                  jsonquery.html ssi
docs                locale          stylesheets
graph-header.html   main.php       trends-form.html
histogram-form.html map-directive.html trends-graph.html
histogram-graph.html map-form.html  trends-host-yaxis.html
```

```
lab1002@lab1002-HP-280-G3-MT:~$ cat /usr/local/nagios/etc/htpasswd.users
admin:$apr1$2XweygVH$ReiuYPl/sfqIhcYq9xuMn/
lab1002@lab1002-HP-280-G3-MT:~$ sudo a2enmod rewrite cgi
Module rewrite already enabled
Module cgi already enabled
lab1002@lab1002-HP-280-G3-MT:~$ sudo systemctl restart apache2
lab1002@lab1002-HP-280-G3-MT:~$ sudo htpasswd /usr/local/nagios/etc/htpasswd.users admin
New password:
Re-type new password:
Updating password for user admin
lab1002@lab1002-HP-280-G3-MT:~$ sudo systemctl restart apache2
lab1002@lab1002-HP-280-G3-MT:~$ █
```

The screenshot shows the Nagios Core web interface at the URL 192.168.4.104/nagios/. The page includes a top navigation bar with icons for back, forward, search, and other browser functions. The main header features the Nagios logo with the word "Core" and a gear icon. A green checkmark indicates "Daemon running with PID 1437".

**General**

- Home
- Documentation

**Current Status**

- Tactical Overview
- Map (Legacy)
- Hosts
- Services
- Host Groups
- Summary
- Grid
- Service Groups
- Summary
- Grid
- Problems
- Services (Unhandled)
- Hosts (Unhandled)
- Network Outages

Quick Search:

**Reports**

- Availability
- Trends (Legacy)
- Alerts
- History
- Summary
- Histogram (Legacy)
- Notifications
- Event Log

**System**

- Comments
- Downtime
- Processor Info

**Nagios® Core™ Version 4.4.6**  
April 28, 2020  
[Check for updates](#)

**Get Started**

- Start monitoring your infrastructure
- Change the look and feel of Nagios
- Extend Nagios with hundreds of addons
- Get support
- Get training
- Get certified

**Quick Links**

- Nagios Library (tutorials and docs)
- Nagios Labs (development blog)
- Nagios Exchange (plugins and addons)
- Nagios Support (tech support)
- Nagios.com (company)
- Nagios.org (project)

**Latest News**

**Don't Miss...**

Copyright © 2010-2020 Nagios Core Development Team and Community Contributors. Copyright © 1999-2009 Ethan Galstad. See the THANKS file for more information on contributors.

Page Tour

**CONCLUSION:** This experiment successfully demonstrates the use of Nagios for continuous monitoring of IT infrastructure. It provides a clear understanding of how Nagios tracks the health and performance of servers, services, and network devices, generates time alerts, and displays system status through a web interface. Continuous monitoring with Nagios enables proactive detection of issues, reduces downtime, and ensures reliable and efficient management of critical systems and services.

**LO MAPPING: LO5**

## Assignment 9:S3 lambda

**Aim:** To understand lambda functions and create a lambda function using python to log message once file is added to S3 bucket

**Theory-** In AWS, a Lambda function is a serverless compute service that runs code in response to events without managing servers. When integrated with Amazon S3, it can automatically trigger actions—like processing files or updating databases—whenever objects are uploaded, modified, or deleted in a bucket. This enables real-time automation and streamlines workflows across cloud applications.

1)

The screenshot shows the AWS S3 console interface. At the top, the navigation bar includes 'Search' and 'Account ID: 6262-1698-Rohit'. Below the navigation bar, the path 'Amazon S3 > Buckets > mybucketrohitkiye' is visible. The main content area is titled 'mybucketrohitkiye Info'. Under the 'Objects' tab, there are 0 objects listed. A prominent 'Upload' button is located at the bottom right of the object list. The overall interface is clean and modern, typical of AWS services.

2)

The screenshot shows the AWS S3 upload interface for the 'mybucketrohitkiye' bucket. The top navigation bar shows the path 'Amazon S3 > Buckets > mybucketrohitkiye > upload'. A large central area has a placeholder text 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.' Below this, a table lists 'Files and folders (1 total, 456.0 B)'. It contains one item: 'main.py' (Type: -). There are 'Remove', 'Add files', and 'Add folder' buttons at the top right of the file list. The 'Destination' section at the bottom shows the URL 's3://mybucketrohitkiye'.

3)

☰

⌚ Upload succeeded  
For more information, see the [Files and folders](#) table.

**Summary**

Destination	Succeeded	Failed
s3://mybucketrohitiyे	<span style="color: green;">⌚ 1 file, 456.0 B (100.00%)</span>	<span style="color: gray;">⌚ 0 files, 0 B (0%)</span>

[Files and folders](#) [Configuration](#)

**Files and folders** (1 total, 456.0 B)

Name	Folder	Type	Size	Status
main.py	~	~	456.0 B	<span style="color: green;">⌚ Succeeded</span>

4)

Compute

# AWS Lambda

lets you run code without thinking about servers.

You pay only for the compute time that you consume — there is no charge when your code is not running. With Lambda, you can run code for virtually any type of application or backend service, all with zero administration.

**Get started**

Author a Lambda function from scratch or choose from one of many preconfigured examples.

[Create a function](#)

**How it works**

[Run](#) [Next: Lambda responds to events](#)

.NET | Java | **Node.js** | Python | Ruby | Custom runtime

```
1 * exports.handler = async (event) => {
2     console.log(event);
```

5)

The screenshot shows the first step of the AWS Lambda 'Create function' wizard. It features three main options:

- Author from scratch**: Start with a simple Hello World example.
- Use a blueprint**: Build a Lambda application from sample code and configuration presets for common use cases.
- Create from template**: Select a function template to get started.

## Basic information

### Function name

Enter a name that describes the purpose of your function.

Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, \_.

### Runtime | Info

Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.



### Architecture | Info

Choose the instruction set architecture you want for your function code.

- arm64
- x86\_64

### Permissions Info

By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can change the permissions when adding triggers.

6)

The screenshot shows the 'Function overview' section of the AWS Lambda console for the function 'myimageuplogg'. The function was successfully created, as indicated by the green success message at the top:

Successfully created the function **myimageuplogg**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

The function details are as follows:

- Name:** myimageuplogg
- Description:** -
- Last modified:** 24 seconds ago
- Function ARN:** arn:aws:lambda:eu-north-1:626216981838:function:myimageuplogg
- Function URL:** [Info](#)
- Code Size:** -
- Layers:** (0)
- Triggers:** + Add trigger
- Destinations:** + Add destination

7)

aws | ⚙️ | Search [Alt+S] | ☰ | 🔍 | ⓘ

☰ Lambda > Add triggers

## Add trigger

**Trigger configuration** Info

S3  
aws asynchronous storage

**Bucket**  
Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.

X C

Bucket region: eu-north-1

**Event types**  
Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual triggers cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.

All object create events X  PUT X

**Prefix - optional**  
Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters. Any [special characters](#) must be URL encoded.

e.g. *images/*

8)

aws | ⚙️ | Search [Alt+S] | ☰ | 🔍 | ⓘ | Europe

☰ Lambda > Functions > myimageuplogg

## myimageuplogg

Throttle Copy ARN Actions ▾

✓ The trigger mybucketrohitjy was successfully added to function myimageuplogg. The function is now receiving events from the trigger. X

**Function overview** Info

Diagram Template Export to Infrastructure Composer Download ▾

myimageuplogg  
 Layers (0)

S3 + Add destination

+ Add trigger

**Description**  
-

**Last modified**  
5 minutes ago

**Function ARN**  
 arn:aws:lambda:eu-north-1:6262169:81838:function:myimageuplogg

**Function URL** Info  
-

9)

**mybucketrohitkiye** [Info](#)

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

**Objects (1)** [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly them permissions. [Learn more](#)

<input type="checkbox"/> Name	Type	Last modified	Size	Storage class
<a href="#">main.py</a>	py	August 6, 2025, 07:22:20 (UTC+01:00)	456.0 B	Standard

**Upload** [Info](#)

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDKs or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

**Files and folders (1 total, 2.1 MB)**  
All files and folders in this table will be uploaded.

<input checked="" type="checkbox"/> Name	Folder	Type
<a href="#">building-2008773.jpg</a>	-	image/jpeg

**Destination** [Info](#)

10)

**Upload succeeded**  
For more information, see the [Files and folders](#) table.

**Summary**

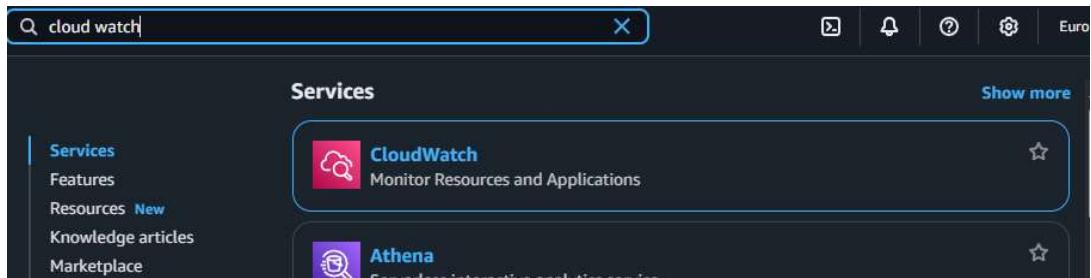
Destination	Succeeded	Failed
s3://mybucketrohitkiye	<a href="#">1 file, 2.1 MB (100.00%)</a>	<a href="#">0 files, 0 B (0%)</a>

[Files and folders](#) [Configuration](#)

**Files and folders (1 total, 2.1 MB)**

Name	Folder	Type	Size	Status	Error
<a href="#">building-2008773.jpg</a>	-	image/jpeg	2.1 MB	<a href="#">Succeeded</a>	-

11)



12)

This screenshot shows the AWS CloudWatch Log groups interface. The left sidebar includes 'CloudWatch' (selected), 'Favorites and recents', 'Dashboards', 'Operations New', 'Alarms', 'Logs', 'Log groups' (selected), and 'Log Anomalies'. The main panel shows 'Log groups (1)'. A single log group is listed: '/aws/lambda/myimageuplogg'. It has a 'Configure' button and a retention policy of 'Never expire'.

13)

This screenshot shows the detailed view of the log group '/aws/lambda/myimageuplogg'. The left sidebar shows 'CloudWatch' (selected), 'recent' (selected), 'Logs New', and 'Signals'. The main panel displays various configuration settings: 'Creation time' (2 minutes ago), 'Retention' (Never expire), 'Stored bytes' (0), 'Contributor Insights rules' (none), 'KMS key ID' (none), 'Anomaly detection' (Configure), 'Custom field indexes' (Configure), and 'Transformer' (Configure). Below this, the 'Log streams' tab is selected, showing 'Log streams (1)' with a single stream entry: '2025/08/06/[\$LATEST]385d6a86ed744e4081a49816daff3e08' (Last event time: 2025-08-06 06:31:46 UTC).

**Conclusion-** In conclusion, integrating AWS Lambda with Amazon S3 using Python offers a powerful, serverless solution for automating file-based workflows. By triggering Lambda functions on S3 events—such as object uploads or deletions—developers can process data in real time without provisioning or managing infrastructure. This approach enhances scalability, reduces operational overhead, and enables seamless integration with other AWS services, making it ideal for building efficient, event-driven cloud applications.

LO Mapping: LO6

# Assignment-10

**Aim:** To create a lambda function using python for adding data to dynamo DB database.

**Theory:** A Python lambda function for adding data to a DynamoDB table typically serves as a lightweight, serverless backend component triggered by events such as API calls or changes in other AWS services. Using the AWS SDK for Python (boto3), the lambda function initializes a DynamoDB client or resource, constructs an item with the required attributes, and performs a "put\_item" operation to insert the data into the specified DynamoDB table. This approach enables scalable, on-demand data insertion without managing servers, making it ideal for event-driven applications and real-time data processing in cloud environments.

The screenshot shows the AWS Lambda console interface. At the top, there's a search bar with 'dynamoDB' typed in. Below the search bar, the 'Services' section is visible, showing 'DynamoDB' as a managed NoSQL database. On the left sidebar, under 'Recent', 'DynamoDB' is listed. The main content area shows a step-by-step wizard for creating a new Lambda function. Step 1: Set the runtime to Python 3.9 and choose a VPC. Step 2: Select the 'HelloWorld' template and choose the 'DynamoDB' trigger. Step 3: Configure the trigger settings, selecting 'Create a new table' and choosing 'MyFirstTable'. Step 4: Review the configuration and proceed to the next step. The bottom of the screen shows the Windows taskbar and system tray.

The screenshot shows the AWS DynamoDB console. The left sidebar has 'Tables' selected under 'DynamoDB'. The main area shows a table named 'MyFirstTable' with one item: 'HelloWorld'. The table details show '1 item' and '1 partition key'. The bottom of the screen shows the Windows taskbar and system tray.

New tab Create table | Amazon DynamoDB +

https://ap-south-1.console.aws.amazon.com/dynamodbv2/home?region=ap-south-1#create-table

DynamoDB > Tables > Create table

## Create table

**Table details** Info

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

**Table name**  
This will be used to identify your table.  
  
Between 3 and 255 characters, containing only letters, numbers, underscores (\_), hyphens (-), and periods (.)

**Partition key**  
The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.  
 String  
1 to 255 characters and case sensitive.

**Sort key - optional**  
You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.  
 String  
1 to 255 characters and case sensitive.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

500312 -1.21% 11:17 14-08-2025

New tab Create table | Amazon DynamoDB +

https://ap-south-1.console.aws.amazon.com/dynamodbv2/home?region=ap-south-1#create-table

DynamoDB > Tables > Create table

Local secondary indexes	-	No
Global secondary indexes	-	Yes
Encryption key management	AWS owned key	Yes
Deletion protection	Off	Yes
Resource-based policy	Not active	Yes

**Tags**  
Tags are pairs of keys and optional values, that you can assign to AWS resources. You can use tags to control access to your resources or track your AWS spending.  
No tags are associated with the resource.

Add new tag

You can add 50 more tags.

Cancel Create table

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

NIFTY +0.06% 11:17 14-08-2025

Screenshot of the AWS DynamoDB console showing the creation of a new table.

The screenshot shows two browser windows side-by-side. Both windows have the URL <https://ap-south-1.console.aws.amazon.com/dynamodbv2/home?region=ap-south-1#tables>.

**Top Window (DynamoDB Tables):**

- Left Sidebar:** Shows navigation links for Dashboard, Tables (selected), Explore items, PartiQL editor, Backups, Exports to S3, Imports from S3, Integrations, Reserved capacity, and Settings.
- Right Content Area:** A success message box says "The demo\_table table was created successfully." Below it is a table titled "Tables (1) Info".

Name	Status	Partition key	Sort key	Indexes	Replication Regions	Deletion protection	Fav
demo_table	Active	id (\$)	-	0	0	Off	

**Bottom Window (IAM Services):**

- Left Sidebar:** Shows navigation links for Dashboard, Tables (selected), Explore items, PartiQL editor, Backups, Exports to S3, Imports from S3, Integrations, Reserved capacity, and Settings.
- Right Content Area:** A sidebar titled "Services" lists "IAM", "IAM Identity Center", and "Resource Access Manager". A sidebar titled "Features" lists "Groups" and "Roles".

**Services**

  - IAM: Manage access to AWS resources
  - IAM Identity Center: Manage workforce user access to multiple AWS accounts and cloud applications
  - Resource Access Manager: Share AWS resources with other accounts or AWS Organizations

**Features**

  - Groups: IAM feature
  - Roles: IAM feature

Screenshot of the AWS IAM Roles page (https://us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/roles).

The page shows 13 IAM roles listed in a table:

Role name	Trusted entities	Last activity
aayush117	AWS Service: lambda	-
aayush117new	AWS Service: lambda	15 days ago
AWSServiceRoleForSupport	AWS Service: support (Service-Linked)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service)	-
my_role	AWS Service: lambda	-
my_role1	AWS Service: lambda	-
my_role69	AWS Service: lambda	13 days ago
myrole	AWS Service: lambda	15 days ago
newrole	AWS Service: lambda	-

Left sidebar navigation includes: Identity and Access Management (IAM), Access management (User groups, Users, Roles, Policies, Identity providers, Account settings, Root access management), and Access reports (Access Analyzer, Resource analysis).

Screenshot of the AWS IAM Create Role page (https://us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/roles/create).

The process is at Step 1: Select trusted entity.

**Select trusted entity**

Step 1:  Select trusted entity  
 Step 2: Add permissions  
 Step 3: Name, review, and create

**Trusted entity type**

AWS service: Allow AWS services like EC2, Lambda, or others to perform actions in this account.  
 AWS account: Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.  
 Web identity: Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.  
 SAML 2.0 federation: Allows users federated with SAML 2.0 from a corporate directory to perform actions in this account.  
 Custom trust policy: Create a custom trust policy to enable others to perform actions in this account.

**Use case**

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

CloudShell Feedback

Screenshot of the AWS IAM 'Create role' wizard Step 1: Set permissions boundary - optional.

The page shows three options for setting a trust policy:

- Allow users federated by the specified external web identity provider to assume this role to perform actions in this account.
- Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy (selected): Create a custom trust policy to enable others to perform actions in this account.

**Use case**: Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

**Service or use case**: Lambda

**Choose a use case for the specified service.**

**Use case**:  
 Lambda: Allows Lambda functions to call AWS services on your behalf.

Buttons: Cancel, Next

Screenshot of the AWS IAM 'Create role' wizard Step 2: Add permissions.

Step 1: Select trusted entity (done)  
Step 2: Add permissions (selected)  
Step 3: Name, review, and create

### Add permissions Info

Permissions policies (1/1091) Info

Choose one or more policies to attach to your new role.

Filter by Type: All types, 6 matches

Policy name	Type	Description
<input checked="" type="checkbox"/>  AmazonDynamoDBFullAccess	AWS managed	Provides full access to Amazon Dynam...
<input type="checkbox"/>  AmazonDynamoDBFullAccess_v2	AWS managed	Provides full access to Amazon Dynam...
<input type="checkbox"/>  AmazonDynamoDBFullAccessWithUsageMetrics	AWS managed	This policy is on a deprecation path. Se...
<input type="checkbox"/>  AmazonDynamoDBReadOnlyAccess	AWS managed	Provides read only access to Amazon D...
<input type="checkbox"/>  AWSLambdaDynamoDBExecutionRole	AWS managed	Provides list and read access to Dynam...
<input type="checkbox"/>  AWSLambdaInvocation-DynamoDB	AWS managed	Provides read access to DynamoDB Str...

▶ Set permissions boundary - optional

Buttons: Cancel, Previous, Next

**Name, review, and create**

**Role details**

**Role name**  
Enter a meaningful name to identify this role.  
  
Maximum 64 characters. Use alphanumeric and '+-=\_,@-.' characters.

**Description**  
Add a short explanation for this role.  
  
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: \_+=\_, @-/[\{\}]!#\$%^&\*`~`

**Step 1: Select trusted entities**

**Trust policy**

```

1+ []
2+   "version": "2012-10-17",
3+   "Statement": [
4+     {
5+       "Effect": "Allow",
6+       "Action": [
7+         "sts:AssumeRole"
8+       ],
9+       "Principal": {
10+         "Service": [
11+           "lambda.amazonaws.com"
12+
13+         ]
14+
15+       }
16+     }
]

```

**Step 2: Add permissions**

**Permissions policy summary**

Policy name	Type	Attached as
<a href="#">AmazonDynamoDBFullAccess</a>	AWS managed	Permissions policy

**Step 3: Add tags**

**Add tags - optional** Info  
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.  
No tags associated with the resource.

[Add new tag](#)  
You can add up to 50 more tags.

[Cancel](#) [Previous](#) [Create role](#)

New tab Roles | IAM | Global +

https://us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#roles

IAM > Roles

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles**
- Policies
- Identity providers
- Account settings
- Root access management [New](#)

Access reports

- Access Analyzer
- Resource analysis [New](#)

CloudShell Feedback

Role demo\_role created.

Roles (14) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
aayush117	AWS Service: lambda	-
aayush117new	AWS Service: lambda	15 days ago
AWSServiceRoleForSupport	AWS Service: support (Service-Linked)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linked)	-
demo_role	AWS Service: lambda	-
my_role	AWS Service: lambda	-
my_role1	AWS Service: lambda	-

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

Type here to search

27°C Light rain 11:29 14-08-2025

The screenshot shows the AWS IAM Roles page. A green success message at the top says "Role demo\_role created." Below it, a table lists 14 roles, each with a name, trusted entity (AWS Service: lambda), and last activity date. The "demo\_role" is the most recent entry. The left sidebar shows navigation links for IAM, Access management, and Access reports. The bottom of the screen includes standard browser controls and a status bar showing weather and time.

New tab Roles | IAM | Global +

https://us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#roles

IAM > Roles

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles**
- Policies
- Identity providers
- Account settings
- Root access management [New](#)

Access reports

- Access Analyzer
- Resource analysis [New](#)

CloudShell Feedback

lambda

Services

- Lambda: Run code without thinking about servers
- CodeBuild: Build and Test Code
- AWS Signer: Ensuring trust and integrity of your code

Show more

Features

- Lambda Insights: CloudWatch feature
- Local processing: IoT Core feature

Show more

Were these results helpful?

Yes No

Manage

Temporary credentials

Temporary credentials with ease and from the enhanced security they

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

Type here to search

27°C Light rain 11:30 14-08-2025

The screenshot shows the AWS Lambda service page. A search bar at the top contains the term "lambda". The main content area displays the "Services" section, which lists Lambda, CodeBuild, and AWS Signer. Below that is the "Features" section, which lists Lambda Insights (CloudWatch feature) and Local processing (IoT Core feature). At the bottom, there's a poll asking if the results were helpful, with "Yes" and "No" buttons. The left sidebar shows the IAM navigation menu. The bottom of the screen includes standard browser controls and a status bar showing weather and time.

Screenshot of the AWS Lambda 'Create function' wizard. The first step, 'Basic information', is shown. Under 'Function name', the value 'add\_student\_data' is entered. Under 'Runtime', 'Python 3.13' is selected. Under 'Architecture', 'x86\_64' is chosen.

Screenshot of the AWS Lambda 'Create function' wizard. The second step, 'Change default execution role', is shown. Under 'Execution role', 'Use an existing role' is selected, and 'demo\_role' is chosen from the dropdown. The third step, 'Additional configurations', is partially visible at the bottom.

The screenshot shows the AWS Lambda Functions console. The top navigation bar includes tabs for 'New tab', 'add\_student\_data | Functions | La', and a '+' button. The URL is https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions/add\_student\_data?newFunction=true&tab=code. The top right corner shows 'Account ID: 2749-4944-5619' and 'Aayush'. The main content area displays the 'add\_student\_data' function overview. A green success message at the top states: 'Successfully created the function add\_student\_data. You can now change its code and configuration. To invoke your function with a test event, choose "Test".'. Below this, the function name 'add\_student\_data' is shown in a card with a 'Layers' section (0 layers). Buttons for '+ Add trigger' and '+ Add destination' are available. On the right, there are sections for 'Description', 'Last modified' (14 seconds ago), 'Function ARN' (arn:aws:lambda:ap-south-1:274949445619:function:add\_student\_data), and 'Function URL' (Info). Buttons for 'Throttle', 'Copy ARN', and 'Actions' are also present. Below the overview, a navigation bar offers links to 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. The bottom status bar shows 'CloudShell Feedback' and system information like '27°C Light rain' and '11:34 14-08-2025'.

**add\_student\_data**

Successfully created the function add\_student\_data. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

**Function overview** Info

**Description**  
-

**Last modified**  
14 seconds ago

**Function ARN**  
arn:aws:lambda:ap-south-1:274949445619:function:add\_student\_data

**Function URL** Info  
-

**Code** **Test** **Monitor** **Configuration** **Aliases** **Versions**

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

27°C Light rain 11:34 14-08-2025

The screenshot shows the AWS Lambda Functions console with the 'Code source' tab selected. The top navigation bar and status bar are identical to the previous screenshot. The main content area shows the code editor for 'lambda\_function.py' with the following content:

```
def play_game():
    while True:
        if check_winner(board, current_player):
            if is_full(board):
                print_board(board)
                print("It's a tie!")
                break
            current_player = "O" if current_player == "X" else "X"
        if __name__ == "__main__":
            play_game()
```

A test event creation dialog is open on the right, titled 'Create new test event'. It contains fields for 'Event Name' (adddata) and 'Event sharing settings' (Private selected). A note says: 'Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.' Below this, 'Shareable' is available but not selected. A 'Template - optional' dropdown is set to 'adddata' and an 'Event JSON' field is present. Buttons for 'Invoke' and 'Save' are at the bottom of the dialog.

**Code source** Info

**Create new test event**

Event Name  
adddata

Event sharing settings  
 Private  
This event is only available in the Lambda Console and to the event creator. You can configure a total of ten. [Learn more](#)

Shareable  
This event is available to IAM users within the same account who have permissions to access and use shareable events. [Learn more](#)

Template - optional  
adddata

Event JSON

Open in Visual Studio Code Upload from

EXPLORER ADD\_STUDENT\_DATA lambda\_function.py ...

DEPLOY [UNDEPLOYED CHANGES]

You have undeployed changes.

Deploy (Ctrl+Shift+U)

Test (Ctrl+Shift+T)

TEST EVENTS [SELECTED: ADDDATA]

+ Create new test event

- Private saved events

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

26°C Rain 12:01 14-08-2025

Tic Tac Toe code add\_student\_data | Functions | Lambda

https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#functions/add\_student\_data?newFunction=true&tab=code

Lambda > Functions > add\_student\_data

The test event "adddata" was successfully saved.

DEPLOY [UNDEPLOYED CHANGES]

- You have undeployed changes.

Deploy (Ctrl+Shift+U)

Test (Ctrl+Shift+T)

TEST EVENTS [SELECTED: ADDDATA]

- + Create new test event
- Private saved events
- adddata

ENVIRONMENT VARIABLES

print\_board(board)  
print(f"Player {current\_player}")  
break  
  
if is\_full(board):  
 print\_board(board)  
 print("It's a tie!")  
 break  
  
current\_player = "0" if current\_play

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

- Private
- Shareable

This event is only available in the Lambda Console and to the event creator. You can configure a total of ten. [Learn more](#)

This event is available to IAM users within the same account who have permissions to access and use shareable events. [Learn more](#)

PROBLEMS OUTPUT CODE REFERENCE LOG TERMINAL Execution Results

Function Logs:

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search 26°C Rain 12:03 14-08-2025

Screenshot of the AWS Lambda search results for "dynamoDB".

The search results page shows the following:

- Services**:
  - DynamoDB (Managed NoSQL Database)
  - Amazon DocumentDB (Fully-managed MongoDB-compatible database service)
  - CloudFront (Global Content Delivery Network)
- Features**:
  - Settings (DynamoDB feature)
  - Clusters (DynamoDB feature)

A sidebar on the left lists services like Lambda, CloudWatch, and CloudFormation. A message at the bottom asks if the results were helpful, with "Yes" and "No" buttons.

**Code preview** button is visible at the bottom.

**CloudShell Feedback** button is at the top left.

Bottom navigation bar includes CloudShell, Feedback, and links for Privacy, Terms, and Cookie preferences.

**DynamoDB Explore items** screenshot:

- Tables (1)**:
  - Any tag key dropdown
  - Any tag value dropdown
  - Find tables input field
  - demo\_table selected
- demo\_table** details:
  - Autopreview checkbox (unchecked)
  - View table details button
  - Scan or query items section:
    - Scan radio button selected
    - Query radio button
    - Select a table or index dropdown (Table - demo\_table)
    - Select attribute projection dropdown (All attributes)
  - Filters - optional section
  - Run and Reset buttons
- Completed** message: Items returned: 1 · Items scanned: 1 · Efficiency: 100% · RCU consumed: 2
- Table: demo\_table - Items returned (1)**:
  - Scan started on August 14, 2025, 12:05:06
  - Actions dropdown
  - Create item button
  - Table structure:
    - id (String)
    - 69

Bottom navigation bar includes CloudShell, Feedback, and links for Privacy, Terms, and Cookie preferences.

**Conclusion-** In conclusion, using a Python lambda function to add data to DynamoDB provides a simple, efficient, and scalable way to manage database operations within a serverless architecture. It leverages AWS's managed services to minimize infrastructure overhead while ensuring seamless integration and real-time data handling, making it an ideal solution for modern cloud-based applications.

**LO Mapping-** LO6

## Written assignment-1

Q1) What are the key features and advantages of AWS Cloud 9 ? LO1

### AWS Cloud9 – Key Features and Advantages

#### 1. Cloud-Based IDE

AWS Cloud9 is a browser-accessible development environment that eliminates the need for local setup. It includes a code editor, terminal, and debugger—all integrated into one interface. Developers can start coding instantly without installing tools or configuring environments.

#### 2. Real-Time Collaboration

Multiple users can work on the same codebase simultaneously. Cloud9 supports shared environments where teammates can see each other's cursors, chat live, and co-edit files. This is especially useful for remote teams, pair programming, and mentoring.

#### 3. Built-In Terminal with AWS CLI

The IDE includes a Linux-based terminal with pre-installed AWS CLI. Developers can run shell commands, manage AWS services, and deploy applications directly from the IDE—without switching to another tool or terminal.

#### 4. Preconfigured Development Environments

Cloud9 supports multiple programming languages like JavaScript, Python, PHP, Ruby, Go, and C++. It comes with essential tools like Git, Docker, and Node.js pre-installed. This reduces setup time and ensures consistency across teams.

#### 5. Deep Integration with AWS Services

Cloud9 is tightly integrated with AWS. Developers can create and manage EC2 instances, Lambda functions, S3 buckets, and DynamoDB tables directly from the IDE. It's ideal for building serverless applications and cloud-native solutions.

#### 6. Secure Access via IAM

Access to Cloud9 environments is controlled using AWS Identity and Access Management (IAM). This ensures secure, role-based access without exposing credentials. Developers can work safely within defined permission boundaries.

#### 7. Environment Portability and Management

Cloud9 environments can be EC2-backed or connected via SSH to existing servers. Developers can clone, pause, or terminate environments as needed. This flexibility supports both cloud-based and hybrid workflows.

## Advantages of AWS Cloud9

- **No local setup:** Code from any device with a browser.
- **Scalable:** Use AWS compute resources for heavy workloads.
- **Secure:** IAM-based access control and no credential leakage.
- **Collaborative:** Real-time editing and communication.
- **Integrated:** AWS CLI, Git, and language runtimes built-in.
- **Efficient:** Reduces context switching and setup overhead.

Q2) What is the difference between Cloud 9 and Lambda ? LO1,LO6

The difference between cloud9 and Lambda is given:

Feature	AWS Cloud9	AWS Lambda
Type	Cloud-based Integrated Development Environment (IDE)	Serverless compute service
Purpose	Write, edit, and debug code in the cloud	Run code in response to events without managing servers
Usage	Development and collaboration	Execution of backend logic, automation, APIs, etc.
Environment	Full Linux-based dev environment with terminal access	Stateless function container managed by AWS
Language Support	Multiple: JavaScript, Python, PHP, Go, etc.	Multiple: Node.js, Python, Java, Go, .NET, Ruby
Execution Model	Manual execution via terminal or editor	Event-driven execution (e.g., API call, file upload)
Persistence	Persistent workspace and file system	Ephemeral—no persistent storage between executions
Scalability	Not auto-scaled; depends on EC2 instance	Automatically scales with incoming requests

Feature	AWS Cloud9	AWS Lambda
Pricing	Based on EC2 usage and storage	Pay-per-use (requests and compute time)
Ideal For	Coding, debugging, team collaboration	Microservices, automation, backend APIs

Q3) What are the three components of AWS Lambda ? LO1,LO6

AWS Lambda is characterized by being serverless, provisioningless, and function-based. Its main use is located in the computing layer of an application which does not have a server, in addition to its main purpose is to create applications based on events and that have the possibility of being activated by various AWS events.

If the situation of having multiple concurrent events arises, AWS lambda will trigger multiple copies of the function, making Lambda a Function of Service (FaaS) type.

In AWS Lambda, the Function is the core component where the actual code resides. This code is written to perform a specific task, such as processing an image, handling an API request, or responding to a database update. The function is stateless and designed to execute quickly and efficiently. Developers write this code in supported languages like Python, Node.js, Java, or Go, and it is uploaded to AWS either directly or via deployment tools. The function is the executable logic that AWS invokes when triggered by an event.

The Configuration component defines how the function should behave during execution. It includes settings such as memory allocation, timeout duration, environment variables, IAM roles for permissions, and concurrency limits. These parameters allow developers to fine-tune performance, security, and resource usage. For example, increasing memory can improve execution speed, while setting a timeout ensures that long-running functions are terminated appropriately. Configuration also determines how the function integrates with other AWS services and manages its runtime environment.

The Event Source is the trigger mechanism that initiates the function's execution. It can be any AWS service capable of generating events, such as S3 (file uploads), DynamoDB (table updates), API Gateway (HTTP requests), or CloudWatch (scheduled tasks). Third-party services or custom applications can also act as event sources via AWS SDKs or messaging systems. While event sources are optional, they are essential for building reactive, event-driven architectures. When configured, AWS automatically invokes the function whenever the specified event occurs, enabling seamless automation and integration across cloud services.

Q4) Discuss the difference between Kubernetes and other container orchestration tools like Docker Swarm? LO1,LO2

#### Architecture & Complexity

- Kubernetes has a more complex architecture with components like etcd, kube-apiserver, kube-scheduler, and kube-controller-manager. It offers fine-grained control over deployments, networking, and storage.
- Docker Swarm is simpler and tightly integrated with Docker. It uses a manager-worker model and is easier to set up for small-scale applications.

#### Scalability & Performance

- Kubernetes is designed for large-scale, production-grade workloads. It supports auto-scaling, rolling updates, and self-healing mechanisms.
- Docker Swarm is suitable for smaller clusters and simpler use cases. It lacks advanced scaling and recovery features found in Kubernetes.

#### Load Balancing & Networking

- Kubernetes uses a robust networking model with built-in service discovery, DNS-based load balancing, and support for ingress controllers.
- Docker Swarm provides basic internal load balancing and service discovery but lacks advanced ingress and traffic routing capabilities.

#### Configuration & Management

- Kubernetes uses YAML manifests for declarative configuration, offering extensive customization and automation.
- Docker Swarm uses Docker Compose files with limited orchestration features and less flexibility in defining complex deployments.

#### Security & Role Management

- Kubernetes supports Role-Based Access Control (RBAC), secrets management, and network policies.
- Docker Swarm has basic security features like mutual TLS between nodes but lacks granular access control.

#### Ecosystem & Community

- Kubernetes has a vast ecosystem with tools like Helm, Istio, Prometheus, and ArgoCD. It's backed by the Cloud Native Computing Foundation (CNCF) and widely adopted across industries.

- Docker Swarm has a smaller community and fewer integrations. Docker Inc. has shifted focus toward Kubernetes support.

## **Written Assignment-2**

### **Q1) What are the key Terraform commands, explain ? LO1,LO3**

The key terraform commands are as shown :

- **terraform init**

This command initializes a Terraform working directory. It sets up the backend configuration, downloads required provider plugins, and prepares the environment for further Terraform operations. You should run `terraform init` once when starting a new project or after modifying the provider configuration in your `.tf` files.

- **terraform plan**

This command shows you what Terraform will do before it actually makes any changes. It compares your current infrastructure state with the desired state defined in your configuration files and outputs a detailed execution plan. This helps you verify and review changes before applying them, making it a safe step in any deployment workflow.

- **terraform apply**

This command executes the actions proposed in the plan and creates or updates infrastructure accordingly. It prompts for confirmation before proceeding unless you use the `-auto-approve` flag. After running `terraform apply`, your infrastructure will match the configuration defined in your `.tf` files.

- **terraform destroy**

This command removes all infrastructure managed by your Terraform configuration. It's used when you want to tear down your environment completely. Like `apply`, it shows a plan and asks for confirmation before executing unless you use `-auto-approve`.

- **terraform validate**

This command checks whether your Terraform configuration files are syntactically valid. It does not access remote services or check the correctness of resource arguments, but it ensures that your `.tf` files are properly structured and free of syntax errors.

- **terraform fmt**

This command formats your Terraform configuration files to follow the standard style conventions. It helps maintain clean, readable code and is especially useful when working in teams or reviewing code in version control.

- **terraform state**

This command lets you inspect and manage the Terraform state file, which tracks the current state of your infrastructure. You can use subcommands like `terraform state list` to view resources or `terraform state rm` to remove specific items from state tracking.

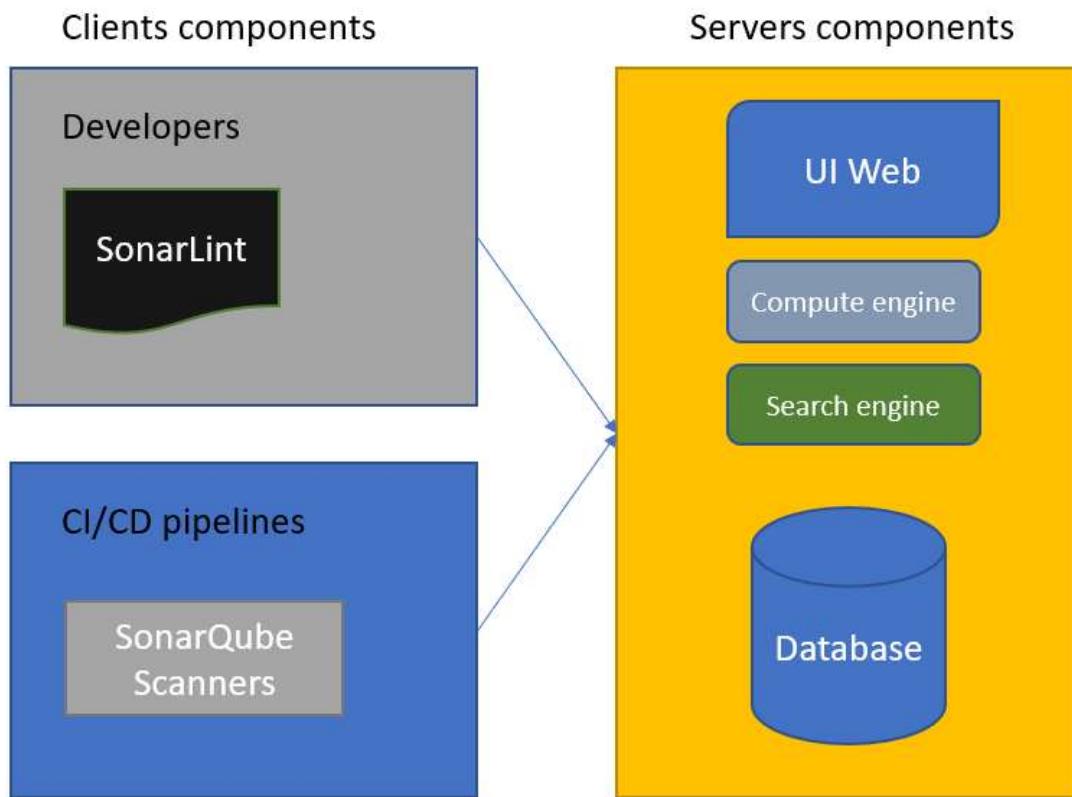
- **terraform output**

This command displays the output values defined in your configuration after running `terraform apply`. These outputs can include IP addresses, resource IDs, or any other values you've marked as outputs in your `.tf` files. It's useful for retrieving dynamic values that other tools or scripts might need.

## Q2) Explain the key components of Sonarqube's architecture? LO1,LO4

SonarQube is a client-server tool, which means that its architecture is composed of artifacts on the server side and also on the client side.

A simplified SonarQube architecture is shown in the following diagram:



**Client components** are primarily involved in the development and automated testing/deployment processes:

1. **Developers:** This area focuses on individual developer workstations or environments.
  - **SonarLint:** This is an IDE (Integrated Development Environment) extension used by developers to provide real-time feedback on code quality and security issues while they write code. It acts as a local, "shift-left" tool for static analysis.
2. **CI/CD pipelines:** This represents the continuous integration and continuous delivery/deployment automation infrastructure.
  - **SonarQube Scanners:** These are tools integrated into the CI/CD pipeline (e.g., Jenkins, GitHub Actions, GitLab CI) that perform static analysis on the entire codebase as part of the build process. They typically send their findings to a central SonarQube Server component for detailed reporting and quality gating.

**Server components** represent the core application or platform that processes requests and manages data:

- The entire server-side application is contained within the large yellow box.
- **UI Web:** This is the user interface, likely a web application, that users interact with.
- **Compute engine:** This component handles the business logic, processing, and computational tasks of the application.
- **Search engine:** This component is dedicated to indexing data and executing fast, complex search queries, often used by the UI Web or Compute engine.
- **Database:** This is the persistent storage layer for the application's data.

The client components (specifically the code/artifacts produced by Developers and analyzed by CI/CD pipelines) interact with and feed into the server environment, resulting in a process where developed and validated code is deployed to run the server component

### **Q3) What are the 3 full deployment modes that can be used for AWS? LO1,LO3**

The three full deployment modes commonly used in AWS environments, each suited to different stages of development and operational needs are:

- **Development Mode**

Development mode is used for building and testing applications in a non-production environment. Resources are typically provisioned with minimal cost and scale, allowing developers to iterate quickly. This mode often includes mock data, debugging tools, and relaxed security settings to facilitate experimentation and rapid prototyping.

- **Staging Mode**

Staging mode replicates the production environment as closely as possible but is isolated from live users. It is used for final testing, performance validation, and quality assurance before deployment. This mode helps identify issues that may not surface during development and ensures that the application behaves as expected under realistic conditions.

- **Production Mode**

Production mode is the live environment where real users interact with the application. Resources are provisioned for high availability, scalability, and security. Monitoring, logging, and backup systems are typically enabled, and infrastructure is optimized for performance and reliability. This mode represents the final and most critical stage of deployment.

#### **Q4) What are the benefits of NAGIOS in AdvDevOps ? LO1,LO5**

They key benefits of using of using Nagios are:

- **Real-time Monitoring:** Nagios provides continuous monitoring of servers, applications, services, and network devices, helping teams detect issues before they impact users.
- **Alerting and Notifications:** It sends alerts via email, SMS, or custom scripts when thresholds are breached or failures occur, enabling rapid incident response.
- **Scalability:** Nagios can monitor thousands of hosts and services using plugins and distributed architectures, making it suitable for large-scale DevOps environments.
- **Extensibility via Plugins:** With a rich ecosystem of community and custom plugins, Nagios can be tailored to monitor virtually any system metric or application.
- **Performance Graphing:** Integration with tools like PNP4Nagios or Grafana allows visualization of performance data over time, aiding in capacity planning and optimization.
- **Infrastructure Visibility:** Nagios offers a centralized dashboard that gives DevOps teams a clear view of system health, dependencies, and service status.
- **Automation Integration:** It can trigger automated remediation scripts or workflows when specific conditions are met, aligning with DevOps principles of self-healing systems.
- **Audit and Reporting:** Nagios logs all events and alerts, supporting compliance, post-mortem analysis, and continuous improvement.
- **Multi-platform Support:** It supports monitoring across Linux, Windows, Unix, and cloud platforms, making it versatile for hybrid infrastructure setups.
- **Community and Enterprise Editions:** Nagios Core is open-source and widely adopted, while Nagios XI offers enterprise-grade features for advanced DevOps teams.