

Experiment 1

AIM : TO understand the benefits of cloud infra and setup AWS Cloud9 IDE, launch AWS Cloud9 IDE and perform collaboration.

THEORY:

Amazon Web Services (AWS) is a comprehensive cloud computing platform provided by Amazon. It offers wide range of cloud-based services, including computer power, storage and databases, machine learning and more.

Amazon EC2 is a core service within AWS that provides scalable virtual servers known as EC2 instances. These instances are designed to handle various workloads from basic web application to high-performance computing tasks. EC2 allows users to launch and manage virtual servers in the cloud providing flexibility in terms of computing power and operating system. An EC2 instance is a virtual server within the Amazon EC2 service. It represents a single unit of computing capacity in the AWS cloud.

Conclusion: Thus, successfully understood the benefits of cloud infrastructure and setup AWS Cloud9 IDE.

~~DATE TO~~

EC2 Instance

Instance launching:

The screenshot shows the AWS CloudShell interface. At the top, there's a green success message: "Successfully initiated launch of instance (i-0ad8f9e1684f03be3)". Below this, a link "Launch log" is visible. The main area of the CloudShell displays the terminal output of an Ubuntu 24.04 LTS system. The output includes system information, security updates, and a note about ESM apps. It also shows the instance ID (i-0ad8f9e1684f03be3) and its public and private IP addresses.

```

Success
Successfully initiated launch of instance (i-0ad8f9e1684f03be3)

▶ Launch log

Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1009-aws x86_64)

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

System information as of Tue Aug  6 16:03:51 UTC 2024

System load: 0.15      Temperature:          -273.1 C
Usage of /: 22.8% of 6.71GB  Processes:        109
Memory usage: 34%       Users logged in:    0
Swap usage:  0%         IPv4 address for ens5: 172.31.39.135

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-172-31-39-135:~$ i-0ad8f9e1684f03be3 (youtube-sk)
Public IPs: 13.60.181.111 Private IPs: 172.31.39.135

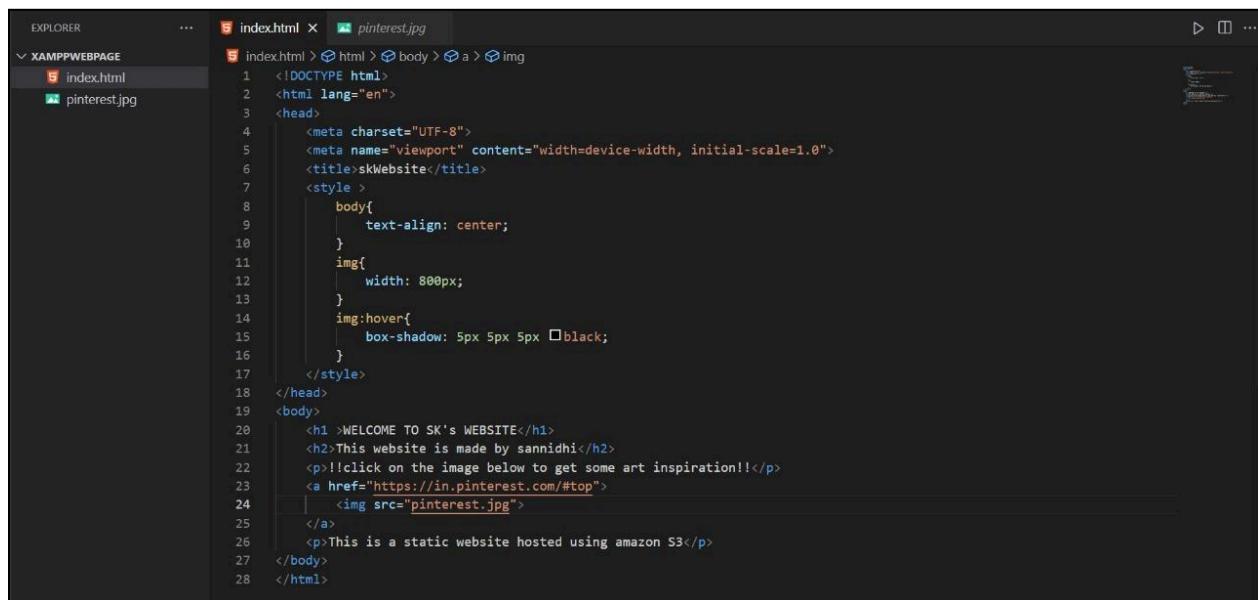
```

Instance termination:

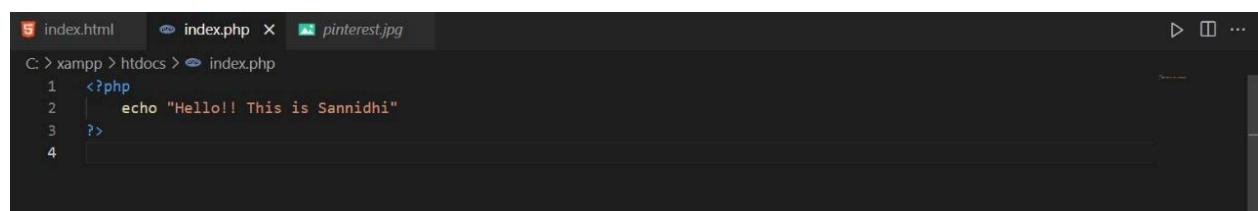
The screenshot shows the AWS EC2 Instances page. A green success message at the top states "Successfully initiated termination of i-0ad8f9e1684f03be3". The main table lists two instances: "youtube-sk" (terminated, t3.micro, eu-north-1b) and "AAR-OCD" (running, t3.micro, eu-north-1b). A modal window titled "Select an instance" is open at the bottom, listing the same two instances. The left sidebar shows the navigation menu for the EC2 service.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IP
youtube-sk	i-0ad8f9e1684f03be3	Terminated	t3.micro	-	View alarms	eu-north-1b	-	-
AAR-OCD	i-049e288da9522f193	Running	t3.micro	2/2 checks passed	View alarms	eu-north-1b	ec2-16-171-26-99.eu-n...	16.171.2

A) To develop a website and host it on local machine on a VM



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>skWebsite</title>
    <style>
        body{
            text-align: center;
        }
        img{
            width: 800px;
        }
        img:hover{
            box-shadow: 5px 5px 5px black;
        }
    </style>
</head>
<body>
    <h1>WELCOME TO SK's WEBSITE</h1>
    <h2>This website is made by sannidhi</h2>
    <p>!!click on the image below to get some art inspiration!!</p>
    <a href="https://in.pinterest.com/#top">
        
    </a>
    <p>This is a static website hosted using amazon S3</p>
</body>
</html>
```

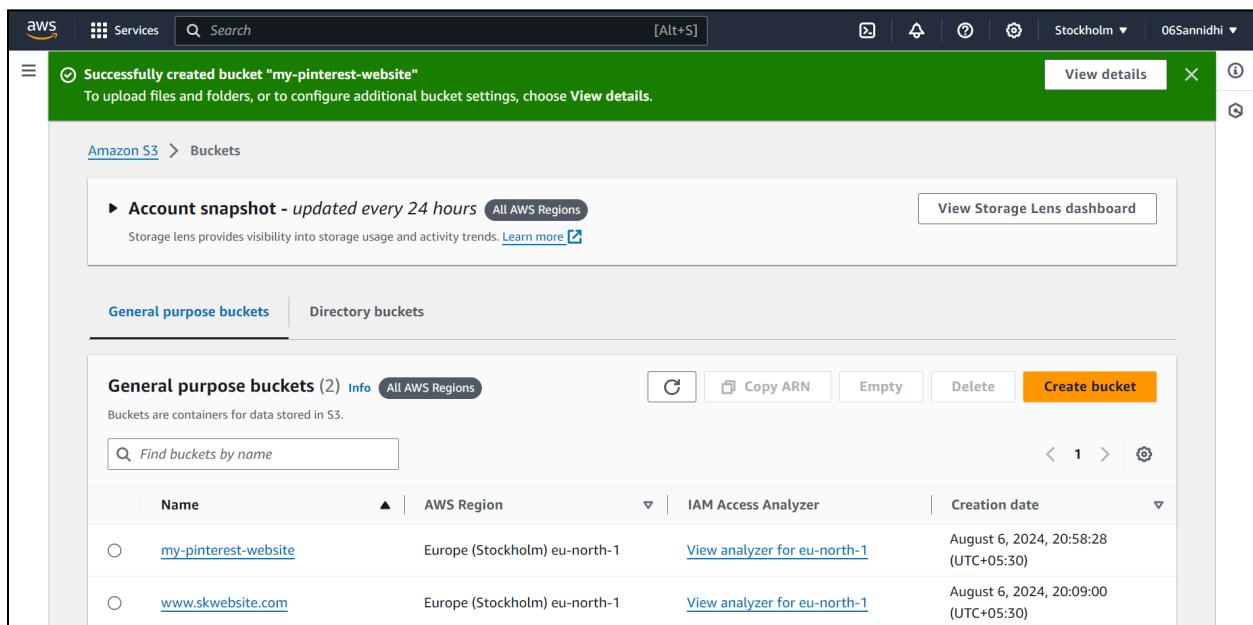


```
<?php
echo "Hello!! This is Sannidhi"
?>
```



B) Hosting a static website on Amazon S3

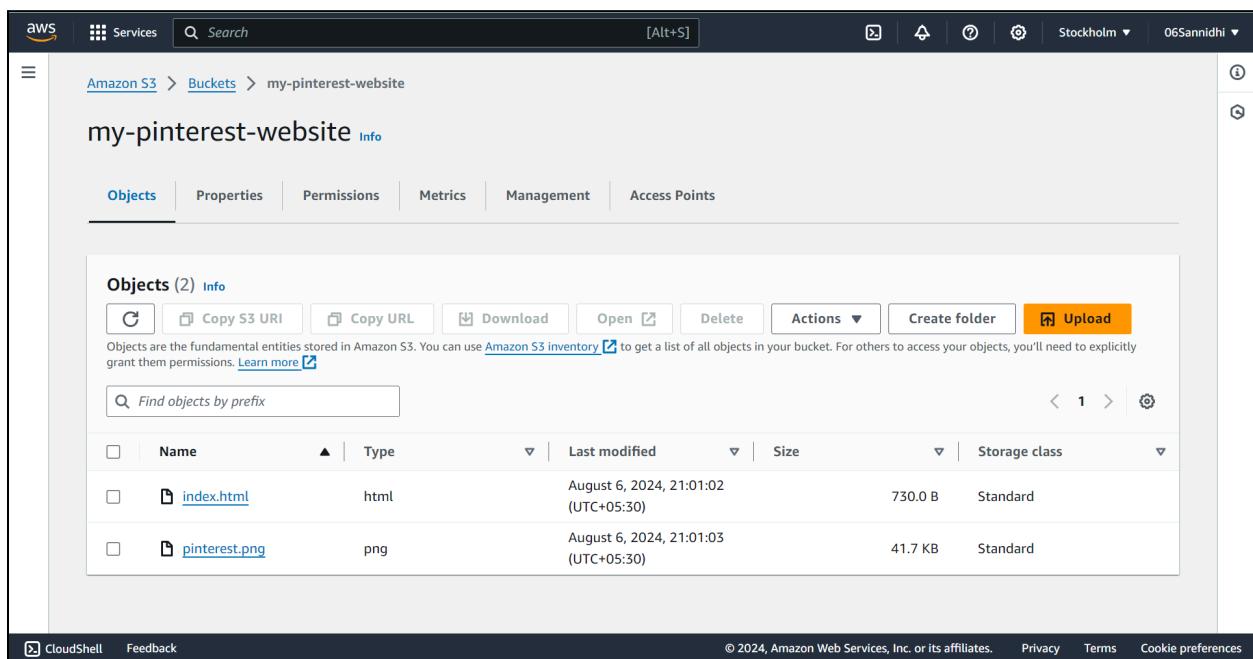
Bucket creation



The screenshot shows the AWS S3 Buckets page. A green success banner at the top states: "Successfully created bucket 'my-pinterest-website'. To upload files and folders, or to configure additional bucket settings, choose View details." Below the banner, the page displays an account snapshot and a list of general purpose buckets. The list shows two buckets: "my-pinterest-website" and "www.skwebsite.com". Both buckets were created on August 6, 2024, at 20:58:28 (UTC+05:30). The "Create bucket" button is visible at the top right of the list.

Name	AWS Region	IAM Access Analyzer	Creation date
my-pinterest-website	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 6, 2024, 20:58:28 (UTC+05:30)
www.skwebsite.com	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 6, 2024, 20:09:00 (UTC+05:30)

Uploading files



The screenshot shows the AWS S3 Bucket Objects page for the "my-pinterest-website" bucket. The page header shows the path: Amazon S3 > Buckets > my-pinterest-website. The "Objects" tab is selected. The page displays two objects: "index.html" and "pinterest.png". Both files were uploaded on August 6, 2024, at 21:01:02 (UTC+05:30). The file sizes are 730.0 B and 41.7 KB respectively, and both are stored in the Standard storage class.

Name	Type	Last modified	Size	Storage class
index.html	html	August 6, 2024, 21:01:02 (UTC+05:30)	730.0 B	Standard
pinterest.png	png	August 6, 2024, 21:01:03 (UTC+05:30)	41.7 KB	Standard

Accessing Index.html file

The screenshot shows the AWS S3 console interface. A single object named "index.html" is selected. The "Properties" tab is active, displaying various metadata fields:

- Owner:** 044ce036f5675977ad8c12b54b02d6ce6e702323f1199f39492eb54b0f9b7f
- AWS Region:** Europe (Stockholm) eu-north-1
- Last modified:** August 6, 2024, 21:01:02 (UTC+05:30)
- Size:** 730.0 B
- Type:** html
- Key:**

On the right side, corresponding values are listed:

- S3 URI:** s3://my-pinterest-website/index.html
- Amazon Resource Name (ARN):** arn:aws:s3:::my-pinterest-website/index.html
- Entity tag (Etag):** 0c78d2dc56cf4c881f79ebf5058fdfa8
- Object URL:** https://my-pinterest-website.s3.eu-north-1.amazonaws.com/index.html

At the bottom, there are links for CloudShell, Feedback, and a footer with copyright information and links to Privacy, Terms, and Cookie preferences.

Enabling static website hosting

The screenshot shows the "Edit static website hosting" configuration page for the "my-pinterest-website" bucket. The "Static website hosting" section is active, with the "Enable" option selected. The "Hosting type" section shows "Host a static website" selected, with a note explaining that the bucket endpoint will be used as the web address. Below this, there is a callout box with instructions about making content publicly readable via S3 Block Public Access settings. The "Index document" section is partially visible at the bottom.

At the bottom, there are links for CloudShell, Feedback, and a footer with copyright information and links to Privacy, Terms, and Cookie preferences.

Enabling object ownership

Screenshot of the AWS S3 console showing the 'Edit Object Ownership' page for a bucket named 'my-pinterest-website'.

The 'Object Ownership' section shows two options:

- ACLs disabled (recommended)**: All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.
- ACLs enabled**: Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

A warning message states: "⚠ We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a bucket policy instead of ACLs to share data with users outside of your account simplifies permissions management and auditing."

A note below says: "⚠ Enabling ACLs turns off the bucket owner enforced setting for Object Ownership". It explains that once turned off, access control lists (ACLs) and their associated permissions are restored. Access to objects that you do not own will be based on ACLs and not the bucket policy.

I acknowledge that ACLs will be restored.

Footer: CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, Cookie preferences

Hosted Website

Screenshot of a web browser displaying a static website hosted on Amazon S3. The URL is 'my-pinterest-website.s3.eu-north-1.amazonaws.com/index.html'.

WELCOME TO SK's WEBSITE

This website is made by sannidhi

!click on the image below to get some art inspiration!!

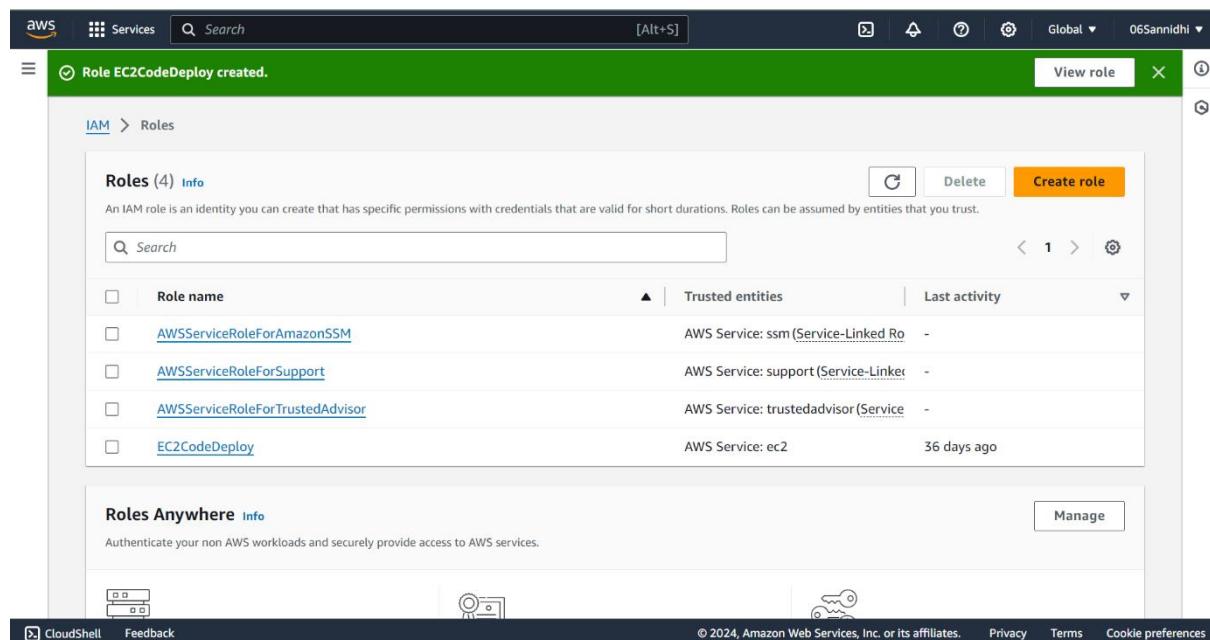
This is a static website hosted using amazon S3

EXPERIMENT-2

Aim

To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

Implementation



The screenshot shows the AWS IAM Roles page. At the top, a green banner indicates that the 'EC2CodeDeploy' role has been created. Below this, the 'Roles (4)' section is displayed, listing four roles: 'AWSServiceRoleForAmazonSSM', 'AWSServiceRoleForSupport', 'AWSServiceRoleForTrustedAdvisor', and 'EC2CodeDeploy'. The 'EC2CodeDeploy' role is highlighted. The page also includes sections for 'Roles Anywhere' and navigation links like CloudShell and Feedback.

Role name	Trusted entities	Last activity
AWSServiceRoleForAmazonSSM	AWS Service: ssm (Service-Linked Role)	-
AWSServiceRoleForSupport	AWS Service: support (Service-Linked Role)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linked Role)	-
EC2CodeDeploy	AWS Service: ec2	36 days ago

Step 2 of 3

Add permissions Info

Permissions policies (1) Info

The type of role that you selected requires the following policy.

Policy name	Type
AWSCodeDeployRole	AWS managed

► Set permissions boundary - *optional*

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

[Feedback](#) Privacy Terms Cookie preferences
© 2024, Amazon Web Services, Inc. or its affiliates.

Step 3 of 3

Name, review, and create

Role details

Role CodeDeployRole created. [View role](#)

Roles (5) 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
AWSServiceRoleForAmazonSSM	AWS Service: ssm (Service-Linked Role)	-
AWSServiceRoleForSupport	AWS Service: support (Service-Linked Role)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linked Role)	-
CodeDeployRole	AWS Service: codedeploy	-
EC2CodeDeploy	AWS Service: ec2	36 days ago

Roles Anywhere Info

Authenticate your non AWS workloads and securely provide access to AWS services.

[Manage](#)

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

The screenshot shows the AWS EC2 "Launch an instance"成功 (Success) page. A green success message box at the top right contains the text "Successfully initiated launch of instance (i-03cf3c5c7c35f9559)". Below this, a "Launch log" link is visible. A "Next Steps" section follows, containing a search bar and a numbered list from 1 to 12. Two cards are present: one for creating billing alerts and another for connecting to the instance.

Success
Successfully initiated launch of instance (i-03cf3c5c7c35f9559)

▶ Launch log

Next Steps

What would you like to do next with this instance, for example "create alc"

1 2 3 4 5 6 7 ... 12 >

Create billing and free tier usage alerts
To manage costs and avoid surprise bills, set up email notifications for

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

The screenshot shows the AWS Developer Tools "Create application" configuration page. It includes fields for "Application name" (set to "AAR-CICD"), "Compute platform" (set to "EC2/On-premises"), and "Tags". A "Create application" button is highlighted in orange at the bottom right. The footer features standard AWS navigation links: Feedback, Privacy, Terms, and Cookie preferences, along with a copyright notice: © 2024, Amazon Web Services, Inc. or its affiliates.

Application configuration

Application name
Enter an application name
AAR-CICD
100 character limit

Compute platform
Choose a compute platform
EC2/On-premises

Tags

Add tag

Create application

Cancel

Feedback Privacy Terms Cookie preferences

© 2024, Amazon Web Services, Inc. or its affiliates.

AWS Services Search [Alt+S] N. Virginia 06Sannidhi

Developer Tools **CodeDeploy**

- Source • CodeCommit
- Artifacts • CodeArtifact
- Build • CodeBuild
- Deploy • CodeDeploy**
 - Getting started
 - Deployments
 - Applications
 - Application**
 - Settings
 - Deployment configurations
 - On-premises instances
 - Pipeline • CodePipeline
 - Settings

[Create deployment group](#)

Application

Application
AAR-CICD
Compute type
EC2/On-premises

Deployment group name

Enter a deployment group name
AAR-CICD-DP
100 character limit

Service role

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

AWS Services Search [Alt+S] N. Virginia 06Sannidhi

Success Deployment group created

Developer Tools > CodeDeploy > Applications > AAR-CICD > AAR-CICD-DP

AAR-CICD-DP

Deployment group details

Deployment group name AAR-CICD-DP	Application name AAR-CICD	Compute platform EC2/On-premises
Deployment type In-place	Service role ARN arn:aws:iam::022499028207:role/CodeDeployRole	Deployment configuration CodeDeployDefault.AllAtOnce
Rollback enabled False	Agent update scheduler Learn to schedule update in AWS Systems Manager	

Environment configuration: Amazon EC2 instances

Key	Value
-----	-------

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

AWS Services Search [Alt+S] N. Virginia 06Sannidhi

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Choose pipeline settings Info

Step 1 of 5

Pipeline settings

Pipeline name
Enter the pipeline name. You cannot edit the pipeline name after it is created.
AAR-CICD-PIPELINE
No more than 100 characters

Pipeline type
ⓘ You can no longer create V1 pipelines through the console. We recommend you use the V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model.

Execution mode
Choose the execution mode for your pipeline. This determines how the pipeline is run.
 Superseded
A more recent execution can overtake an older one. This is the default.
 Queued (Pipeline type V2 required)
Executions are processed one by one in the order that they are queued.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS Services Search N. Virgin 06Sannid

Developer Tools > ... > Create new pipeline

Add source stage Info

Step 2 of 5

Source

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

Cancel Previous Next

Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the 'Create a connection' page for creating a GitHub App connection in AWS Developer Tools. The connection name is set to 'AAR-CICD-GIT'. There is an optional 'Tags' section and a prominent orange 'Connect to GitHub' button.

Create a connection [Info](#)

Create GitHub App connection [Info](#)

Connection name

AAR-CICD-GIT

► Tags - *optional*

[Connect to GitHub](#)

Feedback Privacy Terms Cookie preferences
© 2024, Amazon Web Services, Inc. or its affiliates.

The screenshot shows the 'Connect to GitHub' page where a GitHub connection is being set up. The connection name is 'AAR-CICD-GIT'. It includes an optional 'App installation' section where a GitHub app ID '53719785' is entered, and a 'Tags' section. A large orange 'Connect' button is at the bottom.

Connect to GitHub

GitHub connection settings [Info](#)

Connection name

AAR-CICD-GIT

App installation - *optional*
Install GitHub App to connect as a bot. Alternatively, leave it blank to connect as a GitHub user, which can be used in AWS CodeBuild projects.

53719785 or [Install a new app](#)

► Tags - *optional*

[Connect](#)

Feedback Privacy Terms Cookie preferences
© 2024, Amazon Web Services, Inc. or its affiliates.

The screenshot shows the AWS Lambda Connection settings page. At the top, there is a green success message: "Connection AAR-CICD-GIT created successfully. You can now close the pop-up window". Below this, the connection details are listed:

- Name:** AAR-CICD-GIT
- Provider:** GitHub
- Status:** Available
- Arn:** arn:aws:codeconnections:us-west-1:022499028207:connection/b2bb4c05-a9ce-454d-bf9e-87dc637b64e3

At the bottom of the page, there are links for Feedback, Privacy, Terms, and Cookie preferences, along with the copyright notice: © 2024, Amazon Web Services, Inc. or its affiliates.

The screenshot shows the AWS Lambda Create new pipeline page. The connection section displays a message: "Ready to connect" with a checkmark icon, indicating that the GitHub connection is ready for use. The pipeline configuration fields include:

- Repository name:** Skailaje/aws_cicd_pipline_codedeploy
- Default branch:** main
- Output artifact format:** CodePipeline default

At the bottom of the page, there are links for Feedback, Privacy, Terms, and Cookie preferences, along with the copyright notice: © 2024, Amazon Web Services, Inc. or its affiliates.

The screenshot shows the AWS CodePipeline 'Create new pipeline' wizard at Step 4 of 5. The title is 'Add deploy stage' with an 'Info' link. A note says 'You cannot skip this stage' because pipelines must have at least two stages. The 'Deploy' section shows 'Deploy provider' set to 'AWS CodeDeploy', 'Region' set to 'US East (N. Virginia)', and 'Input artifacts' with a 'Learn more' link. The bottom navigation bar includes 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences'.

The screenshot shows the AWS CodePipeline 'Create new pipeline' wizard at Step 5 of 5. The title is 'Review' with an 'Info' link. The left sidebar lists steps: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review). The main area shows 'Step 1: Choose pipeline settings' with a summary table:

Pipeline settings	
Pipeline name	AAR-CICD-PIPELINE
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-us-east-1-AAR-CICD-PIPELINE

The bottom navigation bar includes 'CloudShell', 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences'.

The screenshot shows the AWS CloudWatch Pipeline interface. At the top, there's a navigation bar with 'aws' logo, 'Services' dropdown, 'Search' bar, and 'N. Virginia' region. A success message 'Congratulations! The pipeline AAR-CICD-PIPELINE has been created.' is displayed. Below it, a breadcrumb trail shows 'Developer Tools > CodePipeline > Pipelines > AAR-CICD-PIPELINE'. On the right, there's a button to 'Create a notification rule for this pipeline'. The main section is titled 'AAR-CICD-PIPELINE' with tabs for 'Notify', 'Edit', 'Stop execution', 'Clone pipeline', and a prominent orange 'Release change' button. Below this, it says 'Pipeline type: V2 Execution mode: QUEUED'. A green box highlights the 'Source' stage, which is 'Succeeded'. The pipeline execution ID is listed as '0e01dd62-40e8-4275-a16e-238370807ad0'. The 'Source' stage details show a GitHub commit: '652a456e' with a link, and a note 'Source: Removed changes [***]'. On the far right, there are two circular icons: a green checkmark and a blue circle with a checkmark.

The screenshot shows the AWS EC2 Security Groups page. The top navigation bar includes the AWS logo, a services menu, a search bar, and account information for N. Virginia and user 06Sannidhi. The main content area shows the selected security group: sg-05817123e270261ba - default. The 'Details' section provides summary information:

Security group name default	Security group ID sg-05817123e270261ba	Description default VPC security group	VPC ID vpc-0c8b47d0500b122d2
Owner 022499028207	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Below the details, there are tabs for 'Inbound rules' (selected), 'Outbound rules', and 'Tags'. The 'Inbound rules' section displays one rule:

Inbound rules (1)

Name	Security group rule...	IP version	Type	Protocol	Port range
(empty)	(empty)	(empty)	(empty)	(empty)	(empty)

Actions: Refresh, Manage tags, Edit inbound rules.

Inbound rules

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0e4ef7c38fdcb99c2	All traffic	All	All	Cus... <input type="text" value="sg-05817123e270261ba"/>	<input type="button" value="Delete"/>
-	HTTP	TCP	80	An... <input type="text" value="0.0.0.0/0"/>	<input type="button" value="Delete"/>
-	SSH	TCP	22	An... <input type="text" value="0.0.0.0/0"/>	<input type="button" value="Delete"/>

[Add rule](#)

⚠️ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Inbound security group rules successfully modified on security group (sg-05817123e270261ba | default)

[Details](#)

[EC2](#) > [Security Groups](#) > sg-05817123e270261ba - default

[Actions](#)

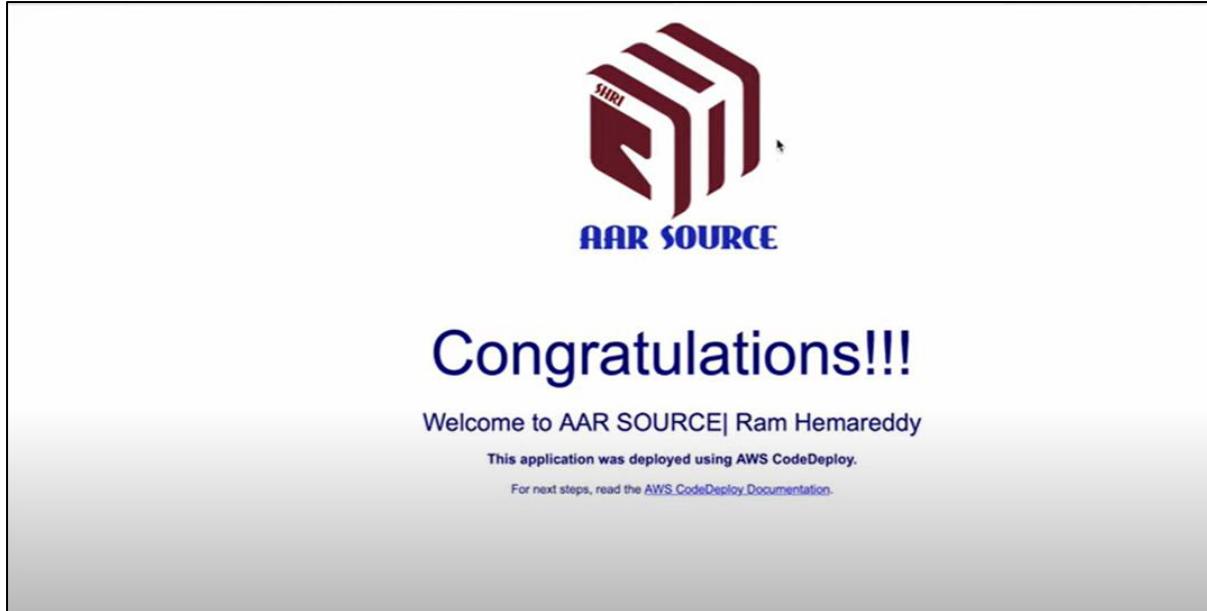
Details

Security group name default	Security group ID sg-05817123e270261ba	Description default VPC security group	VPC ID vpc-0c8b47d0500b122d2
Owner 022499028207	Inbound rules count 3 Permission entries	Outbound rules count 1 Permission entry	

[Inbound rules](#) [Outbound rules](#) [Tags](#)

Inbound rules (3) [Manage tags](#) [Edit inbound rules](#)

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)



The screenshot shows the AWS CodePipeline console. At the top, there is a green success message: "Success Congratulations! The pipeline AAR-CICD-PIPELINE has been created." Below this, the pipeline name "AAR-CICD-PIPELINE" is displayed, along with its type ("V2") and execution mode ("QUEUED"). The pipeline's stages are listed: "Source" (GitHub, Version 2) which has succeeded, and "Build" (AWS Lambda) which is pending. On the right side, there are two green checkmarks indicating successful stages.

Conclusion:

Successfully Build Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

EXPERIMENT 3

Aim

To understand the Kubernetes Cluster Architecture, install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms.

Implementation

- 1) Create EC2 Ubuntu Instances on AWS. (Master and Worker)

The screenshot shows the AWS EC2 Instances page. At the top, there are filters for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. Below the filters, a table lists two instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
-	i-069817ca75a9f4775	Running	t2.micro	-	No alarms	us-east-2c
-	i-058527c844d81c235	Running	t2.micro	-	No alarms	us-east-2c

Below the table, a large gray box prompts "Select an instance".

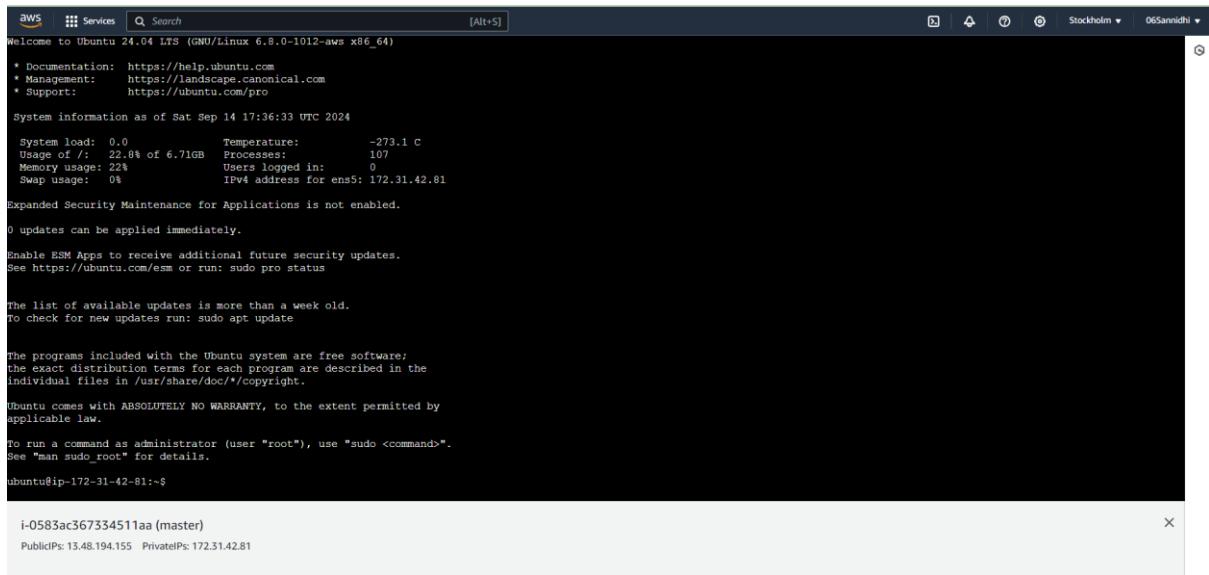
- 2) Edit the Security Group Inbound Rules to allow SSH

The screenshot shows the AWS Security Groups - Edit inbound rules page. The URL is EC2 > Security Groups > sg-0276ba92a6dd0d222 - launch-wizard-1 > Edit inbound rules. The page title is "Edit inbound rules". It displays a table of inbound rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-087a1d87c91ab1a06	All traffic	All	All	Anywhere...	0.0.0.0/0

At the bottom, a message says "You have not made any changes." and includes "Cancel", "Preview changes", and "Save rules" buttons.

3) AWS CLI for master and worker instances



```

AWS Services Search [Alt+S]
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

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

System information as of Sat Sep 14 17:36:33 UTC 2024

System load: 0.0 Temperature: -273.1 C
Usage of /: 22.8% of 6.71GB Processes: 107
Memory usage: 22% Users logged in: 0
Swap usage: 0% IPv4 address for ens5: 172.31.42.81

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

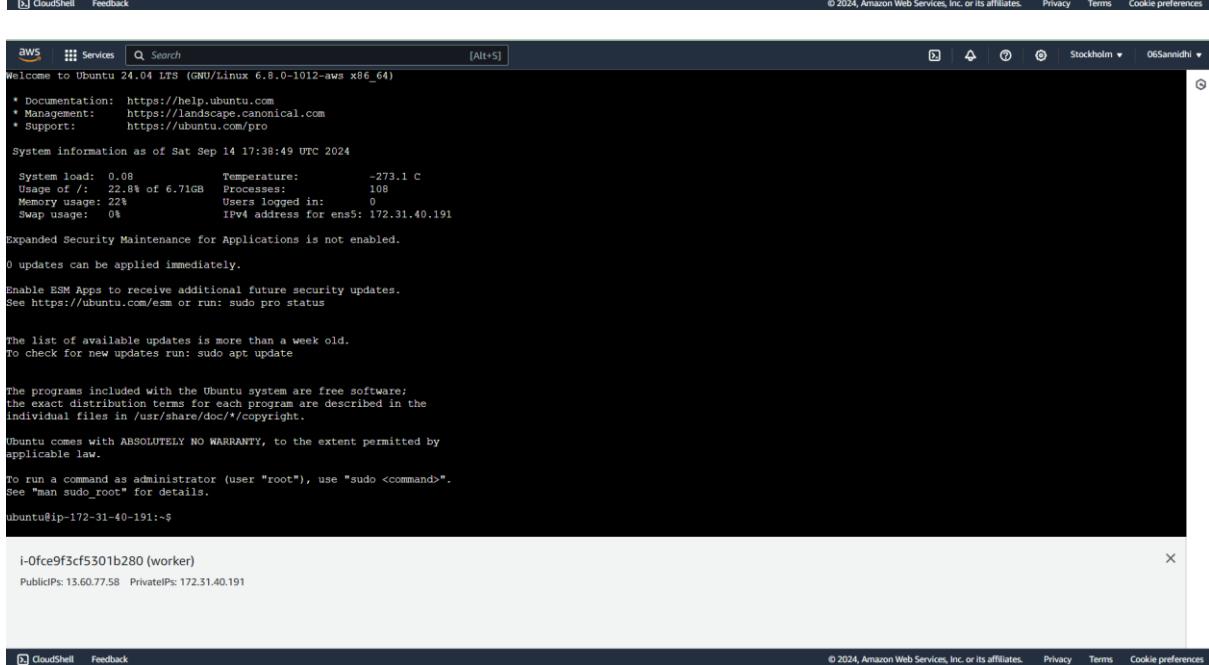
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

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-172-31-42-81:~$
```

i-0583ac367334511aa (master)
Public IPs: 13.48.194.155 Private IPs: 172.31.42.81



```

AWS Services Search [Alt+S]
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

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

System information as of Sat Sep 14 17:38:49 UTC 2024

System load: 0.08 Temperature: -273.1 C
Usage of /: 22.8% of 6.71GB Processes: 108
Memory usage: 22% Users logged in: 0
Swap usage: 0% IPv4 address for ens5: 172.31.40.191

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-172-31-40-191:~$
```

i-0fce9f3cf5301b280 (worker)
Public IPs: 13.60.77.58 Private IPs: 172.31.40.191

4) Assign Unique Hostname for Each Server Node

```
$ sudo hostnamectl set-hostname master
$ sudo hostnamectl set-hostname worker-1
```

Set up Docker (both master and worker)

5) Install Docker

```
$ sudo apt-get update
```

```
aws Services Q Search [Alt+S] Stockholm 06Samnordi
ubuntuMaster node:~ $ sudo apt-get update
Hit:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble security InRelease
Reading package lists... Done
ubuntuMaster node:~ $ sudo apt-get install -y apt-transport-https ca-certificates curl gpg
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
gpg is already the newest version (2.4.4-2ubuntu17).
gpg set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  curl libcurl3-3deb4 libcurl3-gnutls4
  libcurl3-gnutls4 is already the newest version (2.14.0-1ubuntu1.0).
  libcurl3-gnutls4 set to manually installed, 0 to remove and 130 not upgraded.
Need to get 904 kB of archives.
After this operation, 39.9 kB of additional disk space will be used.
Get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 apt-transport-https all 2.7.14build2 [3974 B]
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 curl amdgpu 0.5.0-2ubuntu10.3 [227 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl4-3deb4 libcurl4-3deb4-gnutls4 amd64 2.14.0-1ubuntu1.0 [341 kB]
Get:4 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl3-3deb4-gnutls4 amd64 2.14.0-1ubuntu10.3 [333 kB]
Fetched 904 kB in (26.3 MB/s)
Selecting previously unselected package apt-transport-https.
Reading database... 69100 files and directories currently installed.
Preparing to unpack .../apt-transport-https_2.7.14build2_all.deb...
Unpacking apt-transport-https (2.7.14build2) ...
Preparing for unpack .../curl_0.5.0-2ubuntu10.3_amd64.deb...
Unpacking curl (0.5.0-2ubuntu10.3) over (0.5.0-2ubuntu10.1) ...
Preparing to unpack .../libcurl4-3deb4_0.5.0-2ubuntu10.3_amd64.deb...
Unpacking libcurl4-3deb4 (0.5.0-2ubuntu10.3) over (0.5.0-2ubuntu10.1) ...
Preparing to unpack .../libcurl3-3deb4-gnutls4_0.5.0-2ubuntu10.3_amd64.deb...
Unpacking libcurl3-3deb4-gnutls4 (0.5.0-2ubuntu10.3) over (0.5.0-2ubuntu10.1) ...
Setting up apt-transport-https (2.7.14build2) ...
```

```
$ sudo apt-get install docker.io
```

```
aws ━ Services Q Search [Alt+S] Stockholm 06Sandhvi
ubuntu@berkerli:~$ sudo apt-get install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
bridge-utils containerd dns-root-data dhcpcsvc-base docker.io pigz runc ubuntu-fan
Suggested packages:
ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-buildx
docker-compose-v2 docker-doc fuse zfs-fuse | zfsutils
The following NEW packages will be installed:
bridge-utils containerd dns-root-data dhcpcsvc-base docker.io pigz runc ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 133 not upgraded.
Need to get 76.8 MB of archives.
After this operation, 289 MB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 pigz amd64 2.8-1 [65.6 kB]
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 bridge-utils amd64 1.7.1-lubuntu2 [33.9 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 runc amd64 1.1.12-0ubuntu3.1 [8599 kB]
Get:4 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 containerd amd64 1.7.12-0ubuntu4.1 [38.6 MB]
Get:5 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 containerd-bin amd64 1.7.12-0ubuntu4.1 [4450 B]
Get:6 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 docker.io amd64 2.9.0-2ubuntu0.1 [37.4 kB]
Get:7 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 runc amd64 24.0.7-0ubuntu4.1 [29.1 MB]
Get:8 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]
Fetched 76.0 MB in 1s (95.3 MB/s)
Preconfiguring packages...
Extracting previously unselected package pigz.
(Reading database ... 47/41 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.8-1_amd64.deb ...
Unpacking pigz (2.8-1) ...
Selecting previously unselected package bridge-utils.
Preparing to unpack .../1-bridge-utils_1.7.1-lubuntu2_amd64.deb ...
Unpacking bridge-utils (1.7.1-lubuntu2) ...
Selecting previous unselected package runc.
Preparing to unpack .../2-runc_1.1.12-0ubuntu3.1_amd64.deb ...
Unpacking runc (1.1.12-0ubuntu3.1) ...
Selecting previous unselected package containerd.
Preparing to unpack .../3-containerd_1.7.12-0ubuntu4.1_amd64.deb ...

```

```
$ docker --version
```

```
aws Services Q Search [Alt+S] Stockholm 06Sandhu
ubuntu@master-node:~$ docker --version
Command 'docker' not found, but can be installed with:
sudo snap install docker           # version 24.0.5, or
sudo apt install podman-docker     # version 4.9.3+dfsg-1ubuntu0.1
sudo apt install docker              # version 24.0.5-0ubuntu1
      * sudo ln -s docker podman for additional versions.
ubuntu@master-node:~$ sudo apt get update
E: Invalid operation
ubuntu@master-node:~$ sudo apt get update
[...]
http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
get:3 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
get:5 http://security.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
get:6 http://security.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
get:7 http://security.ubuntu.com/ubuntu noble/main amd64 Components [3671 kB]
get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [351 kB]
get:9 http://security.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
get:10 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
get:11 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [123 kB]
get:12 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-mn [109 kB]
get:13 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 kB]
get:14 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [502 kB]
get:15 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [123 kB]
get:16 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [10344 kB]
get:17 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [366 kB]
get:18 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [150 kB]
get:19 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-mn [150 kB]
get:20 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [45.0 kB]
get:21 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [14.3 kB]
get:22 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [61.4 kB]
get:23 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 c-n-f Metadata [61.4 kB]
get:24 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [14.4 kB]
get:25 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [3608 kB]
get:26 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [212 kB]
get:27 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [532 kB]
get:28 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 kB]
```

6) Start and Enable Docker

```
$ sudo systemctl enable docker
```

```
$ sudo systemctl status docker
```

```
$ sudo systemctl start docker
```

```
aws | Services | Q | 🌐 | ⓘ | ⓘ | Stockholm | 06Sannidhi
ubuntu@master-node:~$ sudo systemctl enable docker
ubuntu@master-node:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
    Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
    Active: active (running) since Sat 2024-09-14 17:48:43 UTC; 5min ago
      TriggeredBy: • docker.socket
        Docs: https://docs.docker.com
     Main PID: 3126 (dockerd)
        Tasks: 9
       Memory: 36.2M (peak: 37.6M)
         CPU: 359ms
        CGroup: /system.slice/docker.service
                └─3126 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerrd.s>

Sep 14 17:48:42 master-node systemd[1]: Starting docker.service - Docker Application Co>
Sep 14 17:48:42 master-node dockerd[3126]: time="2024-09-14T17:48:42.845702248Z" level=>
Sep 14 17:48:42 master-node dockerd[3126]: time="2024-09-14T17:48:42.846619650Z" level=>
Sep 14 17:48:42 master-node dockerd[3126]: time="2024-09-14T17:48:42.941239211Z" level=>
Sep 14 17:48:43 master-node dockerd[3126]: time="2024-09-14T17:48:43.182968694Z" level=>
Sep 14 17:48:43 master-node dockerd[3126]: time="2024-09-14T17:48:43.287256246Z" level=>
Sep 14 17:48:43 master-node dockerd[3126]: time="2024-09-14T17:48:43.289235690Z" level=>
Sep 14 17:48:43 master-node dockerd[3126]: time="2024-09-14T17:48:43.343273615Z" level=>
Sep 14 17:48:43 master-node systemd[1]: Started docker.service - Docker Application Con>

ubuntu@master-node:~$
```

```
aws | Services | Q | 🌐 | ⓘ | ⓘ | Stockholm | 06Sannidhi
ubuntu@worker1:~$ sudo systemctl enable docker
ubuntu@worker1:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
    Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
    Active: active (running) since Sat 2024-09-14 17:50:57 UTC; 5min ago
      TriggeredBy: • docker.socket
        Docs: https://docs.docker.com
     Main PID: 2705 (dockerd)
        Tasks: 9
       Memory: 71.9M (peak: 73.2M)
         CPU: 332ms
        CGroup: /system.slice/docker.service
                └─2705 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerrd.s>

Sep 14 17:50:56 worker1 systemd[1]: Starting docker.service - Docker Application Contai>
Sep 14 17:50:56 worker1 dockerd[2705]: time="2024-09-14T17:50:56.897274813Z" level=info>
Sep 14 17:50:56 worker1 dockerd[2705]: time="2024-09-14T17:50:56.903217401Z" level=info>
Sep 14 17:50:56 worker1 dockerd[2705]: time="2024-09-14T17:50:56.990653034Z" level=info>
Sep 14 17:50:57 worker1 dockerd[2705]: time="2024-09-14T17:50:57.234623704Z" level=info>
Sep 14 17:50:57 worker1 dockerd[2705]: time="2024-09-14T17:50:57.312373541Z" level=info>
Sep 14 17:50:57 worker1 dockerd[2705]: time="2024-09-14T17:50:57.313406238Z" level=info>
Sep 14 17:50:57 worker1 systemd[1]: Started docker.service - Docker Application Contain>
Sep 14 17:50:57 worker1 dockerd[2705]: time="2024-09-14T17:50:57.352453690Z" level=info>
lines 1-21/21 (END)
```

7) Install Kubernetes(both master and worker node)

```
$ sudo apt-get update
```

```
$ sudo apt-get install -y apt-transport-https ca-certificates curl
```

```
aws Service Q Search [Alt+S] Stockholm 06Samnichi
ubuntu@master-node:~$ sudo apt-get update
Hit:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... done
ubuntu@master-node:~$ sudo apt-get install -y apt-transport-https ca-certificates curl gpg
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates is set to manually installed.
curl is already the newest version (2.4.4-2ubuntu17).
gpg is already the newest version (2.3.0-1ubuntu1.1).
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  curl libcurl3-64-gnutls libcurl4-openssl4
Upgrading curl (2.4.4-2ubuntu17) to remove and 130 not upgraded.
0 upgraded, 0 newly installed, 0 to remove and 130 not upgraded.
Need to get 904 kB of archives.
After this operation, 39.9 kB of additional disk space will be used.
get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 apt-transport-https all 2.7.14build2 [3974 B]
get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 curl amd64 8.5.0-2ubuntu10.3 [227 kB]
get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl4-64 amd64 8.5.0-2ubuntu10.3 [341 kB]
get:4 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl3-64-gnutls amd64 8.5.0-2ubuntu10.3 [333 kB]
Fetched 904 kB in 0s (26.3 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 60109 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.7.14build2_all.deb ...
Unpacking apt-transport-https (2.7.14build2) ...
Preparing to unpack .../curl_8.5.0-2ubuntu10.3_amd64.deb ...
Preparing to unpack .../curl_8.5.0-2ubuntu10.1.deb ...
Unpacking curl (8.5.0-2ubuntu10.3) over (8.5.0-2ubuntu10.1) ...
Preparing to unpack .../libcurl4-64_amd64_8.5.0-2ubuntu10.3_amd64.deb ...
Unpacking libcurl4-64_amd64 (8.5.0-2ubuntu10.3) over (8.5.0-2ubuntu10.1) ...
Unpacking libcurl3-64_gnutls_amd64_8.5.0-2ubuntu10.3_amd64.deb ...
Unpacking libcurl3-64_gnutls_8.5.0-2ubuntu10.3_amd64 (8.5.0-2ubuntu10.1) ...
Setting up apt-transport-https (2.7.14build2) ...
```

```
$ sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg
```

```
$ echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg]  
https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee  
/etc/apt/sources.list.d/kubernetes.list
```

```
aws Services Search [Alt+5] Stockholm 06Sandinhi
ubuntu@master-node:~$ curl -fsSL https://pkgs.k8s.io/core/stable:v1.31/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg
ubuntu@master-node:~$ echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core/stable:v1.31/deb/ ' | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core/stable:v1.31/deb/
ubuntu@master-node:~$
```

```
aws Services Search [Alt+S] Stockholm - 08Samohi
ubuntu@control-1:~$ curl -f https://pkgs.k8s.io/core/stable/v1.31/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg
ubuntu@worker1:~$ echo "deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core/stable/v1.31/deb/ /" | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core/stable/v1.31/deb/
ubuntu@worker1:~$
```

```
$ sudo apt-get update
```

```
$ sudo apt-get install -y kubelet kubeadm kubectl
```

```
$ sudo apt-mark hold kubelet kubeadm kubectl
```

```

aws [■] Server [Q] Search [Alt+5]
ubuntu@kailaje-OptiPlex-5090:~$ sudo apt-get update
...
aws [■] Server [Q] Search [Alt+5]
Preparing cri-tools (1.31.1-1.1) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../2-kubelet_1.31.1-1.1_amd64.deb ...
Unpacking kubelet (1.31.1-1.1) ...
Selecting previously unselected package kube-proxy.
Preparing to unpack .../3-kubelet_1.31.1-1.1_amd64.deb ...
Unpacking kubelet (1.31.1-1.1) ...
Selecting previously unselected package kubelet-cni.
Preparing to unpack .../4-kubelet-cni_1.31.1-1.1_amd64.deb ...
Unpacking kubelet-cni (1.31.1-1.1) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../5-kubelet_1.31.1-1.1_amd64.deb ...
Unpacking kubelet (1.31.1-1.1) ...
Setting up kubelet (1.31.1-1.1) ...
Setting up kubelet-cni (1.31.1-1.1) ...
Setting up kubelet (1.31.1-1.1) ...
Setting up kubelet (1.31.1-1.1) ...
Setting up kubelet (1.31.1-1.1) ...
Processing triggers for man-db (2.12.0-4ubuntu2) ...
Scanning processes...
Scanning links images...
Running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.

No VM drivers are running outdated hypervisor (qemu) binaries on this host.
Downloaded files: sudo apt-get-mark held kubelet kubelet-cni
ubuntu@kailaje-OptiPlex-5090:~$
```

Kubernetes Deployment (master only)

- 8) Begin Kubernetes Deployment

```
$ sudo swapoff -a
```

- 9) Initialize Kubernetes on Master Node

```
$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16--ignore-preflight-errors=all
```

```

ubuntu@master-node:~$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16 --ignore-preflight-errors=all
[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
[preflight] This node (ip: 172.31.42.81) has less memory than the minimum required (1914 MB) is less than the minimum 1700 MB
[WARNING FileExisting-socat]: socat not found in system path
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your internet connection
[preflight] You can also perform this action beforehand using "kubeadm config images pull"
[WARNING FileExisting-registry]: registry:80/pause:3.8" as the CRI sandbox image.
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [kubernetes.kubernetes.default.svc kubernetes.default.svc.cluster.local master-node] and IPs [10.96.0.1 172.31.42.81]
[certs] Generating "apiserver-kubelet-client" certificate and key
[certs] Generating "front-proxy-ca" certificate and key
[certs] Generating "front-proxy-client" certificate and key
[certs] Generating "etcd/ca" certificate and key
[certs] Generating "etcd/server" certificate and key
[certs] etcd/server serving cert is signed for DNS names [localhost master-node] and IPs [172.31.42.81 127.0.0.1 ::1]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [localhost master-node] and IPs [172.31.42.81 127.0.0.1 ::1]
[certs] Generating "etcd/healthcheck-client" certificate and key
[certs] Generating "etcd-peer" certificate and key
[certs] Generating "sa" key and public key
[kubeconfig] Using kubeconfig folder "/etc/kubernetes"
[kubeconfig] Writing "admin.conf" kubeconfig file
[kubeconfig] Writing "super-admin.conf" kubeconfig file
[kubeconfig] Writing "kubelet.conf" kubeconfig file
[kubeconfig] Writing "controller-manager.conf" kubeconfig file
[kubeconfig] Writing "scheduler.conf" kubeconfig file
[etcd] Creating static Pod manifest for local etcd in "/etc/kubernetes/manifests"
```

- 10) Deploy Pod Network to Cluster

```
$ mkdir -p $HOME/.kube
```

```
$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
```

```
$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
$ kubectl apply -f
```

<https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml>

```

ubuntu@master-node:~$ mkdir -p $HOME/.kube
ubuntu@master-node:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
ubuntu@master-node:~$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
ubuntu@master-node:~$ kubectl apply -f https://github.com/flannel-io/flannel/releases/latest/download/kube-flannel.yml
namespace/kube-flannel created
serviceaccount/flannel created
clusterrole.rbac.authorization.k8s.io/flannel created
clusterrolebinding.rbac.authorization.k8s.io/flannel created
configmap/kube-flannel-cfg created
daemonset.apps/kube-flannel-ds created
```

```
$ kubectl get pods --all-namespaces
```

NAME	READY	STATUS	RESTARTS	AGE
kube-flannel kube-flannel-ds-cx5fj	1/1	Running	5 (37s ago)	25m
kube-flannel kube-flannel-ds-nhvpz	1/1	Running	6 (37s ago)	33m
kube-system coredns-7c656fcfc9-5fhcg	1/1	Running	2 (87s ago)	44m
kube-system coredns-7c656fcfc9-msq7b	1/1	Running	3 (87s ago)	44m
kube-system etcd-master-node	1/1	Running	3 (87s ago)	44m
kube-system kube-apiserver-master-node	1/1	Running	3 (87s ago)	44m
kube-system kube-controller-manager-master-node	1/1	Running	8 (87s ago)	44m
kube-system kube-proxy-gn4ls	1/1	Running	11 (69s ago)	25m
kube-system kube-proxy-zg8m	1/1	Running	21 (68s ago)	44m
kube-system kube-scheduler-master-node	1/1	Running	8 (87s ago)	44m

Join Worker Node to Cluster (on worker node)

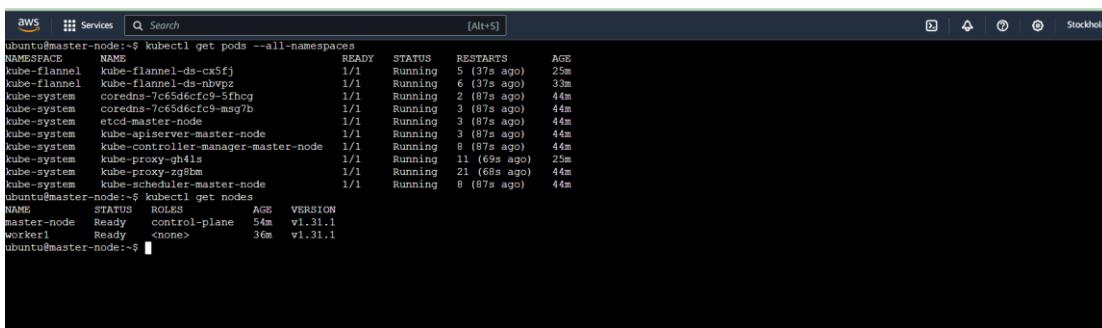
```
11) $ kubeadm join 172.31.42.81:6443 --token 1evaro.2xdo7fco07hizfja --
   discovery-token-ca-cert-hash
   sha256:be87f981c53e5ea6471eef94af12957cbdd291711150e58ceaf6a56b0f760d5e
   768d5e --ignore-preflight-errors=all
```

```
ubuntu@worker1:~$ kubeadm join 172.31.42.81:6443 --token 1evaro.2xdo7fco07hizfja --discovery-token-ca-cert-hash sha256:be87f981c53e5ea6471eef94af12957cbdd291711150e58ceaf6a56b0f760d5e --ignore-preflight-errors=all
[preflight] Running pre-flight checks
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-check] Waiting for the healthy kubelet at http://127.0.0.1:10248/healthz. This can take up to 4m0s
[kubelet-check] The kubelet is healthy after 500.209646ms
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap

This node has joined the cluster:
* Certificate signing request was sent to apiserver and a response was received.
* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.
```

12) \$ kubectl get nodes (on master node)



```
aws Services Search [Alt+S]
ubuntu@master-node:~$ kubectl get pods --all-namespaces
NAMESPACE     NAME                               READY   STATUS    RESTARTS   AGE
kube-flannel  kube-flannel-ds-cx5fj            1/1    Running   5 (37s ago)  25m
kube-flannel  kube-flannel-pod-69wpg           1/1    Running   6 (77s ago)  33m
kube-system   coredns-7c63dcfc9-5fhcg          1/1    Running   2 (60s ago)  44m
kube-system   coredns-7c63dcfc9-msg7b          1/1    Running   3 (97s ago)  44m
kube-system   etcd-master-node                 1/1    Running   3 (87s ago)  44m
kube-system   kube-apiserver-master-node       1/1    Running   3 (87s ago)  44m
kube-system   kube-controller-manager-master-node 1/1    Running   8 (87s ago)  44m
kube-system   kube-proxy-qh1ls                  1/1    Running   11 (69s ago)  25m
kube-system   kube-proxy-zgbm                  1/1    Running   21 (68s ago)  44m
kube-system   kube-scheduler-master-node       1/1    Running   8 (87s ago)  44m
ubuntu@master-node:~$ kubectl get nodes
NAME        STATUS   ROLES      AGE   VERSION
master-node Ready    control-plane  54m   v1.31.1
worker1     Ready    <none>     36m   v1.31.1
ubuntu@master-node:~$
```

we now have a Kubernetes cluster running across AWS EC2 Instances. This cluster can be used to further deploy applications and their loads being distributed across these machines.

Conclusion

Successfully understood the Kubernetes cluster architecture and deployed a Kubernetes cluster on Linux machines/cloud platforms, demonstrating seamless setup and orchestration.

EXPERIMENT 4

Aim

To install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy Your First Kubernetes Application.

Implementation

- 1) In your Kubernetes cluster check if all nodes are connected.

```
$ kubectl get nodes
```

```
ubuntu@master-node:~$ kubectl get nodes
NAME     STATUS   ROLES      AGE     VERSION
master-node   Ready    control-plane   54m    v1.31.1
worker1       Ready    <none>    36m    v1.31.1
ubuntu@master-node:~$
```

- 2) Create deploy.yaml file

```
$ sudo nano deploy.yaml
```

- 3) Copy nginx-deployment.yaml file from Kubernetes site and paste in above file.

- 4) Create deployment

```
$ kubectl apply -f
```

- 5) Check if deployment is created and all pods are running

```
$ kubectl get deploy.po
```

- 6) Expose the app to internet using load balancer service.

```
$ kubectl expose deployment.apps/nginx-deployment --type= "load balancer"
```

- 7) Kubectl get svc

The image shows two terminal windows side-by-side. The left terminal window displays the command-line interface for managing a Kubernetes cluster. It includes a command history, a search bar, and a status bar indicating the user is on an AWS master node. The right terminal window is a code editor showing a YAML configuration file named 'deploy.yaml'. This file defines a 'Deployment' object with a single pod ('nginx-deployment') and its corresponding 'Service' object ('nginx-service'). The Service is configured with a 'LoadBalancer' type, which is being exposed to the internet. Both terminals show the output of the executed commands, such as deployment creation and service exposure.

```
aws Services Q Search [Alt+S]
ubuntu@master-node:~$ sudo nano deploy.yaml
ubuntu@master-node:~$ sudo nano deploy.yaml
ubuntu@master-node:~$ kubectl create -f deploy.yaml
deployment.apps/nginx-deployment created
ubuntu@master-node:~$ kubectl get deploy
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   3/3     3           116s
ubuntu@master-node:~$ kubectl expose deployment.apps/nginx-deployment --type="LoadBalance
ubuntu@master-node:~$ kubectl expose deployment.apps/nginx-deployment \
> --type="LoadBalancer"
Command 'Kubectl' not found, did you mean:
: command 'Kubectl' from snap kubectl (1.31.1)
See 'snap info <snapname>' for additional versions.
ubuntu@master-node:~$ kubectl expose deployment.apps/nginx-deployment \
> --type="LoadBalancer"
service/nginx-deployment exposed
ubuntu@master-node:~$ kubectl get svc
NAME        CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes  ClusterIP      10.96.0.1      <none>       443/TCP      79m
nginx-deployment  LoadBalancer  10.98.23.236  <pending>   80:32324/TCP  48s
ubuntu@master-node:~$
```

```
aws Services Q Stockholm 06Sannidhi
GNU nano 7.2
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
          ports:
            - containerPort: 80
```

- 8) deployed our Nginx server on our EC2 instance.



Conclusion

Successfully installed Kubectl and executed commands to manage the Kubernetes cluster, deploying the first application seamlessly.

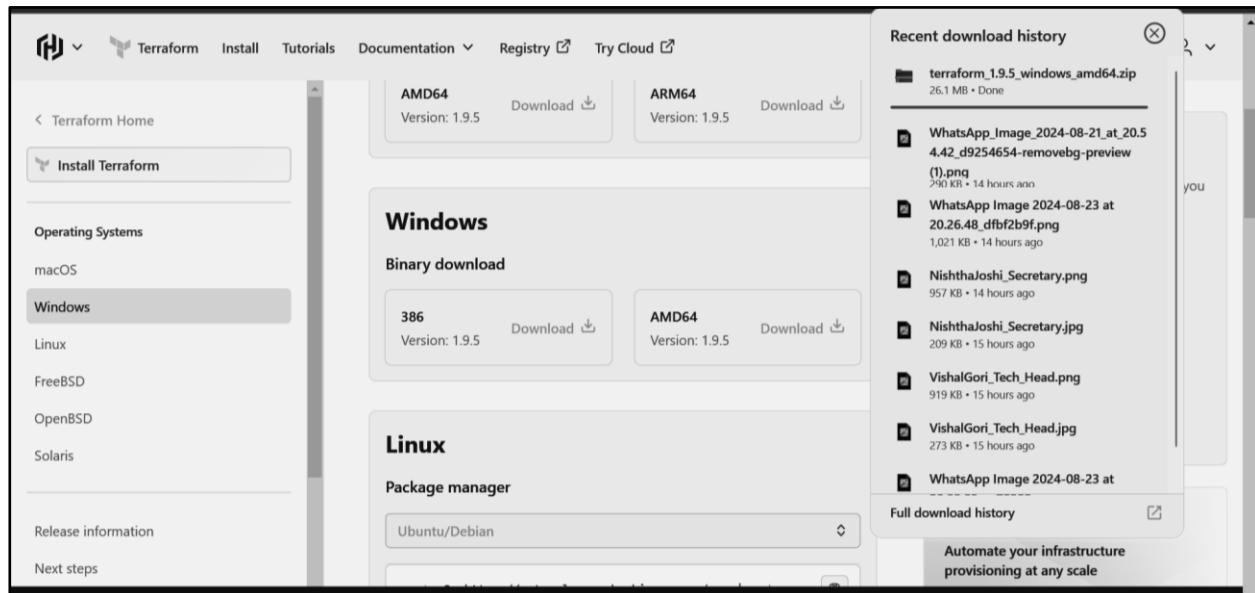
EXPERIMENT 5

Aim:

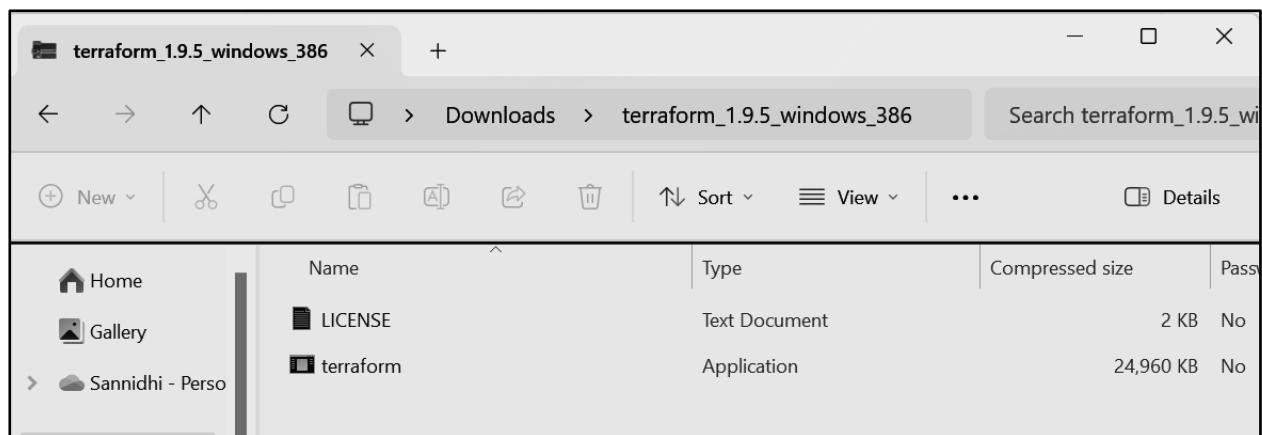
To understand terraform lifecycle, core concepts /terminologies and install it on a Linux Machine and Windows

Implementation:

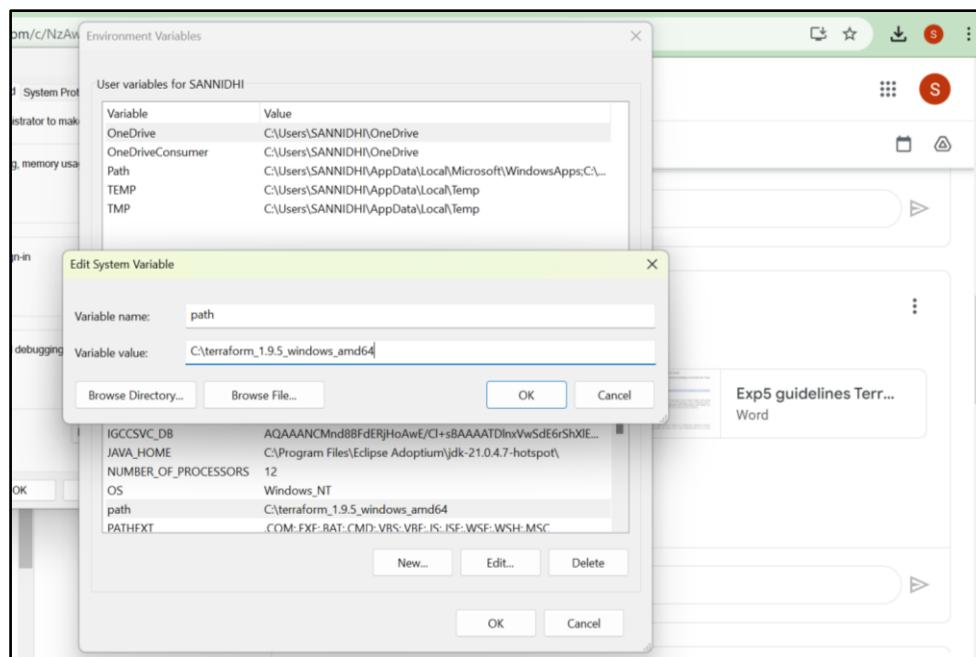
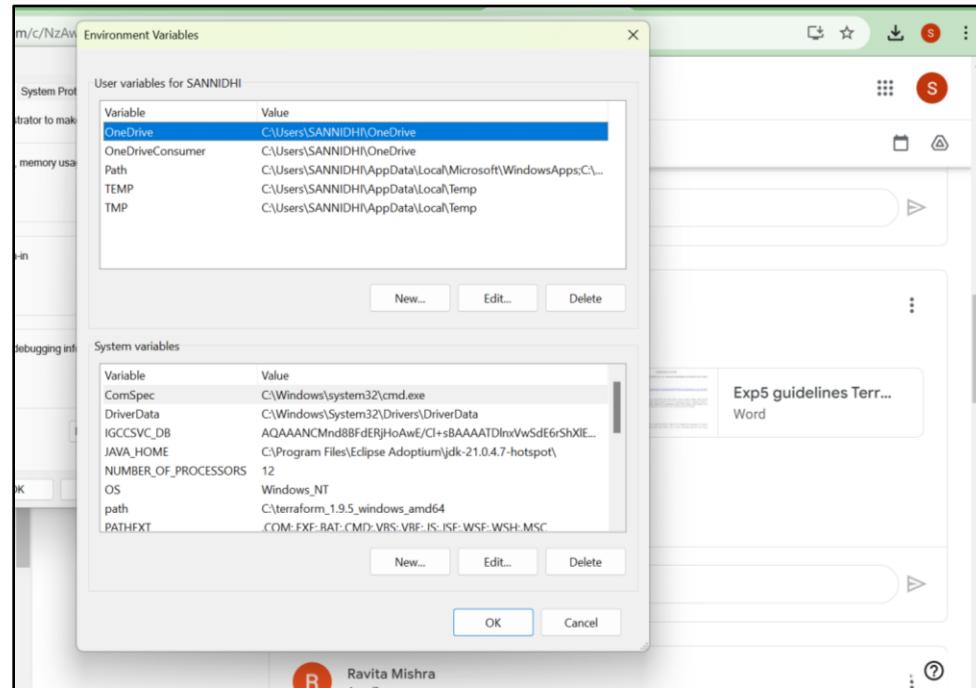
Installing Terraform



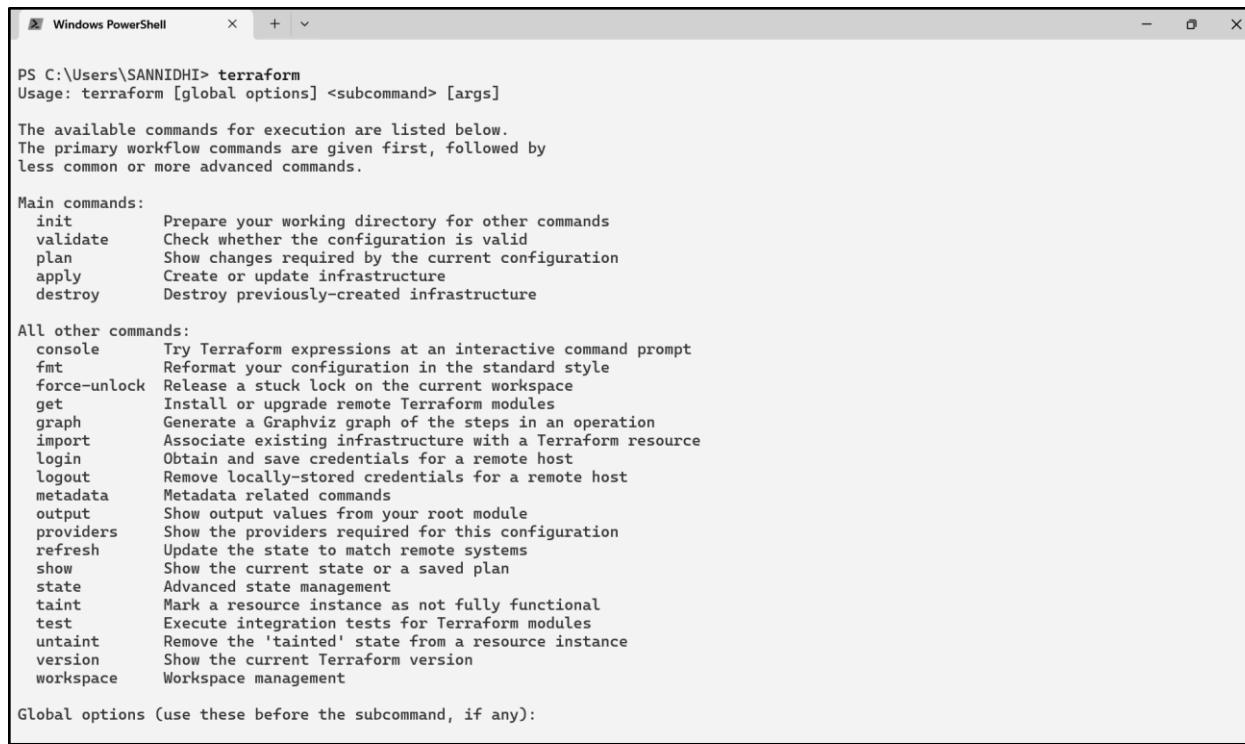
Extract the downloaded setup file Terraform.exe in C:\Terraform directory



Set the System path for Terraform in Environment Variables



Terraform in PowerShell and its functionality



```
PS C:\Users\SANNIDHI> terraform
Usage: terraform [global options] <subcommand> [args]

The available commands for execution are listed below.
The primary workflow commands are given first, followed by
less common or more advanced commands.

Main commands:
  init      Prepare your working directory for other commands
  validate   Check whether the configuration is valid
  plan      Show changes required by the current configuration
  apply      Create or update infrastructure
  destroy    Destroy previously-created infrastructure

All other commands:
  console    Try Terraform expressions at an interactive command prompt
  fmt        Reformat your configuration in the standard style
  force-unlock Release a stuck lock on the current workspace
  get        Install or upgrade remote Terraform modules
  graph      Generate a Graphviz graph of the steps in an operation
  import    Associate existing infrastructure with a Terraform resource
  login      Obtain and save credentials for a remote host
  logout     Remove locally-stored credentials for a remote host
  metadata   Metadata related commands
  output     Show output values from your root module
  providers Show the providers required for this configuration
  refresh   Update the state to match remote systems
  show       Show the current state or a saved plan
  state      Advanced state management
  taint      Mark a resource instance as not fully functional
  test       Execute integration tests for Terraform modules
  untaint   Remove the 'tainted' state from a resource instance
  version   Show the current Terraform version
  workspace Workspace management

Global options (use these before the subcommand, if any):
```



```
Microsoft Windows [Version 10.0.22631.3880]
(c) Microsoft Corporation. All rights reserved.

C:\Users\SANNIDHI> terraform --version
Terraform v1.9.5
on windows_amd64

C:\Users\SANNIDHI>
```

Conclusion:

Successfully understood the Terraform lifecycle, core concepts, and terminologies, and completed the installation on Windows machines.

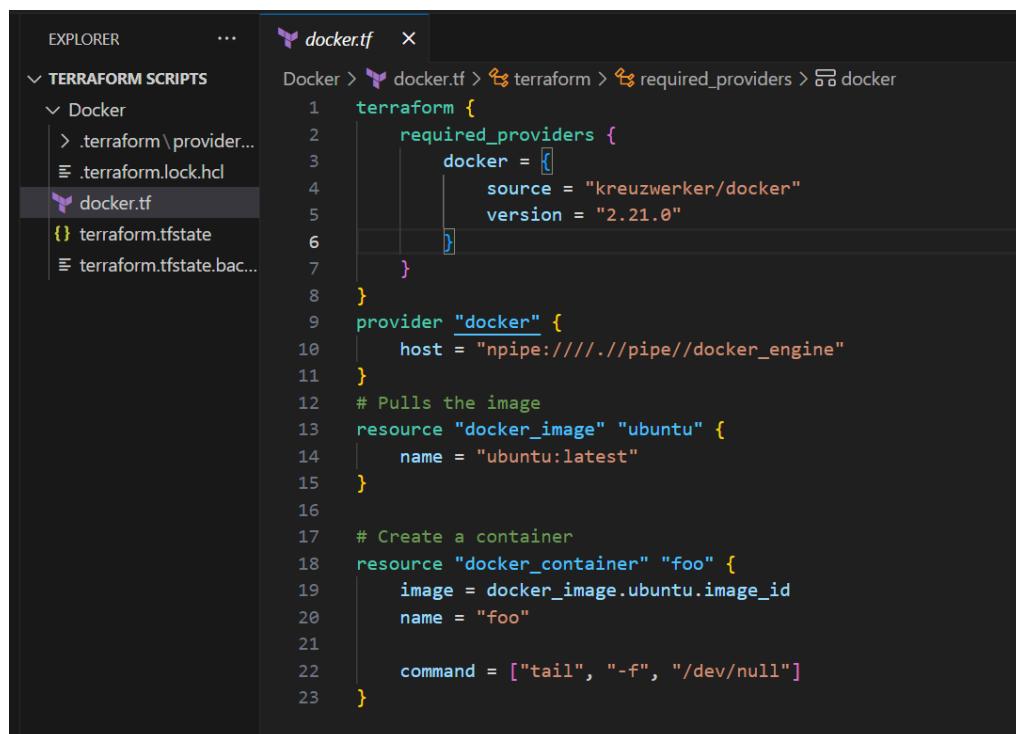
EXPERIMENT – 6

Aim

To Build, change, and destroy AWS infrastructure Using Terraform.

Implementation:

- 1) create a new folder named ‘Docker’ in the ‘TerraformScripts’ folder. Then create a new docker.tf file



```

EXPLORER      ...
TERRAFORM SCRIPTS
  Docker
    .terraform\provider...
    .terraform.lock.hcl
    docker.tf
    {} terraform.tfstate
    terraform.tfstate.bac...

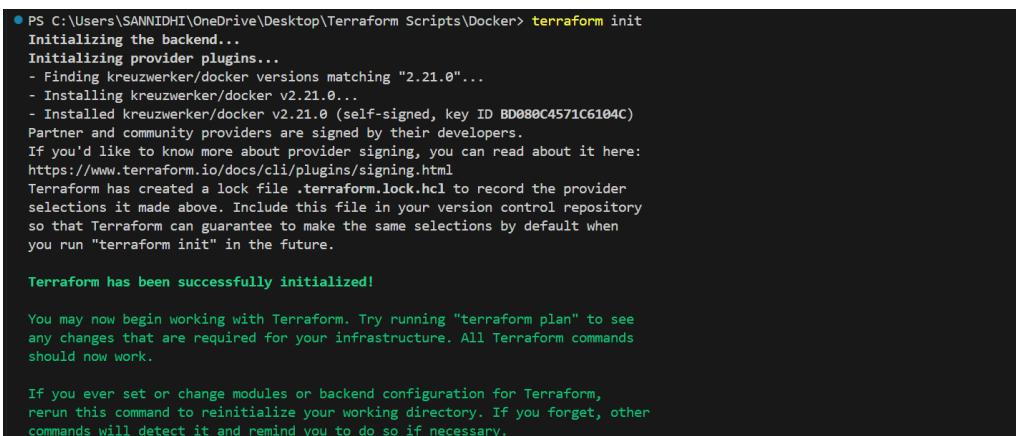
```

```

docker.tf  X
Docker > docker.tf > terraform > required_providers > docker
1  terraform {
2    required_providers {
3      docker = [
4        {
5          source = "kreuzwerker/docker"
6          version = "2.21.0"
7        }
8      }
9      provider "docker" {
10        host = "npipe://./pipe/docker_engine"
11      }
12      # Pulls the image
13      resource "docker_image" "ubuntu" {
14        name = "ubuntu:latest"
15      }
16
17      # Create a container
18      resource "docker_container" "foo" {
19        image = docker_image.ubuntu.image_id
20        name = "foo"
21
22        command = ["tail", "-f", "/dev/null"]
23    }

```

- 2) Execute Terraform Init command to initialize the resources



```

PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker> terraform init
Initializing the backend...
Initializing provider plugins...
- Finding kreuzwerker/docker versions matching "2.21.0"...
- Installing kreuzwerker/docker v2.21.0...
- Installed kreuzwerker/docker v2.21.0 (self-signed, key ID BD080C4571C6104C)
  Partner and community providers are signed by their developers.
  If you'd like to know more about provider signing, you can read about it here:
  https://www.terraform.io/docs/cli/plugins/signing.html
  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.

```

3) Execute Terraform plan to see the available resources

```
PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker> terraform plan

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

Terraform will perform the following actions:

# docker_container.foo will be created
+ resource "docker_container" "foo" {
    + attach          = false
    + bridge          = (known after apply)
    + command         = (known after apply)
    + container_logs = (known after apply)
    + entrypoint      = (known after apply)
    + env             = (known after apply)
    + exit_code       = (known after apply)
    + gateway         = (known after apply)
    + hostname        = (known after apply)
    + id              = (known after apply)
    + image           = (known after apply)
    + init            = (known after apply)
    + ip_address      = (known after apply)
    + ip_prefix_length = (known after apply)
    + ipc_mode        = (known after apply)
    + log_driver      = (known after apply)
    + logs            = false
    + must_run        = true
    + name            = "foo"
    + network_data    = (known after apply)
    + read_only       = false
    + read_only       = false
    + remove_volumes = true
    + restart         = "no"
    + rm              = false
    + runtime          = (known after apply)
    + security_opts   = (known after apply)
    + shm_size         = (known after apply)
    + start            = true
    + stdin_open       = false
    + stop_signal      = (known after apply)
    + stop_timeout     = (known after apply)
    + tty              = false

    + healthcheck (known after apply)
    + labels (known after apply)
}

# docker_image.ubuntu will be created
+ resource "docker_image" "ubuntu" {
    + id          = (known after apply)
    + image_id    = (known after apply)
    + latest      = (known after apply)
    + name        = "ubuntu:latest"
    + output      = (known after apply)
    + repo_digest = (known after apply)
}

Plan: 2 to add, 0 to change, 0 to destroy.
```

- 4) Execute Terraform apply to apply the configuration, which will automatically create and run the Ubuntu Linux container based on our configuration.

```
PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker> terraform apply
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]

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

Terraform will perform the following actions:

# docker_container.foo will be created
+ resource "docker_container" "foo" {
  + attach           = false
  + bridge           = (known after apply)
  + command          = [
    + "tail",
    + "-f",
    + "/dev/null",
  ]
  + container_logs   = (known after apply)
  + entrypoint       = (known after apply)
  + env              = (known after apply)
  + exit_code         = (known after apply)
  + gateway          = (known after apply)
  + hostname         = (known after apply)
  + id               = (known after apply)
  + image             = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a"
  + init              = (known after apply)
  + ip_address        = (known after apply)
  + ip_prefix_length = (known after apply)
  + ipc_mode          = (known after apply)
  + log_driver        = (known after apply)
  + logs              = false
  + must_run          = true
  + name              = "foo"
  + network_data      = (known after apply)
  + read_only          = false
  + remove_volumes    = true
  + restart            = "no"
  + rm                = false
  + runtime           = (known after apply)
  + security_opts     = (known after apply)
  + shm_size          = (known after apply)
  + start              = true
  + stdio_open         = false
  + stop_signal        = (known after apply)
  + stop_timeout       = (known after apply)
  + tty                = false

  + healthcheck (known after apply)

  + labels (known after apply)
}

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

Enter a value: yes

docker_container.foo: Creating...
docker_container.foo: Creation complete after 0s [id=d8d3a7916204fe136ac6bd973622991daf1a338aa6bef3f22f7aa6efefb10e98]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker>
```

- 5) Docker images, After Executing Apply step

```
PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker> docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   edbfe74c41f8  5 weeks ago  78.1MB
PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker>
```

6) Execute Terraform destroy to delete the configuration, which will automatically delete the Ubuntu Container

```
PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker> terraform destroy
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_container.foo: Refreshing state... [id=d8d3a7916204fe136ac6bd973622991daf1a338aa6bef3f22f7aa6efefb10e98]

Terraform used the selected providers to generate the following
execution plan. Resource actions are indicated with the following
symbols:
- destroy

Terraform will perform the following actions:

# docker_container.foo will be destroyed
- resource "docker_container" "foo" {
    - attach           = false -> null
    - command          = [
        - "tail",
        - "-f",
        - "/dev/null",
    ] -> null
    - cpu_shares       = 0 -> null
    - dns              = [] -> null
    - dns_opts         = [] -> null
    - dns_search       = [] -> null
    - entrypoint       = [] -> null
    - env              = [] -> null
    - gateway          = "172.17.0.1" -> null
    - group_add        = [] -> null
    - hostname         = "d8d3a7916204fe136ac6bd973622991daf1a338aa6bef3f22f7aa6efefb10e98" -> null
    - id               = "d8d3a7916204fe136ac6bd973622991daf1a338aa6bef3f22f7aa6efefb10e98"
    - image             = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
    - init              = false -> null
    - ip_address        = "172.17.0.2" -> null
    - ip_prefix_length = 16 -> null
    - ipc_mode          = "private" -> null
    - links             = [] -> null
    - log_driver        = "json-file" -> null
    - log_opts          = {} -> null
    - logs              = false -> null
    - max_retry_count   = 0 -> null
    - memory            = 0 -> null
    - memory_swap       = 0 -> null
    - must_run          = true -> null
    - name              = "foo" -> null
    - network_data      = [
        - {
            - gateway          = "172.17.0.1"
            - global_ipv6_prefix_length = 0
            - ip_address        = "172.17.0.2"
            - ip_prefix_length  = 16
            - network_name      = "bridge"
            # (2 unchanged attributes hidden)
        }
        - read_only         = false -> null
        - remove_volumes   = true -> null
        - restart           = "no" -> null
        - rm                = false -> null
        - runtime            = "runc" -> null
        - security_opts     = [] -> null
        - shm_size          = 64 -> null
        - start              = true -> null
        - stdio_open         = false -> null
        - stop_timeout       = 0 -> null
        - storage_opts      = {} -> null
        - sysctls            = {} -> null
        - tmpfs              = {} -> null
        - tty                = false -> null
        # (8 unchanged attributes hidden)
    ]
}

# docker_image.ubuntu will be destroyed
- resource "docker_image" "ubuntu" {
    - id               = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest" -> null
    - image_id         = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
    - latest           = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
    - name              = "ubuntu:latest" -> null
    - repo_digest      = "ubuntu@sha256:8a37d68f4f73ebf3d4efafbcf66379bf3728902a8038616808f04e34a9ab63ee" -> null
}

Plan: 0 to add, 0 to change, 2 to destroy.

Do you really want to destroy all resources?
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

docker_container.foo: Destroying... [id=d8d3a7916204fe136ac6bd973622991daf1a338aa6bef3f22f7aa6efefb10e98]
docker_container.foo: Destruction complete after 1s
docker_image.ubuntu: Destroying... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_image.ubuntu: Destruction complete after 0s

Destroy complete! Resources: 2 destroyed.
```

7) Docker images After Executing Destroy step

```
Destroy complete! Resources: 2 destroyed.  
● PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker> docker images  
REPOSITORY      TAG          IMAGE ID   CREATED     SIZE  
○ PS C:\Users\SANNIDHI\OneDrive\Desktop\Terraform Scripts\Docker>
```

Conclusion

Successfully implemented Terraform to build, modify, and destroy AWS infrastructure, demonstrating the tool's ability to automate and streamline infrastructure management with consistency and efficiency.

EXPERIMENT 7

Aim:

To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

Implementation:

1. Open up Jenkins Dashboard on localhost, port 8080
2. Run SonarQube in a Docker container using command:

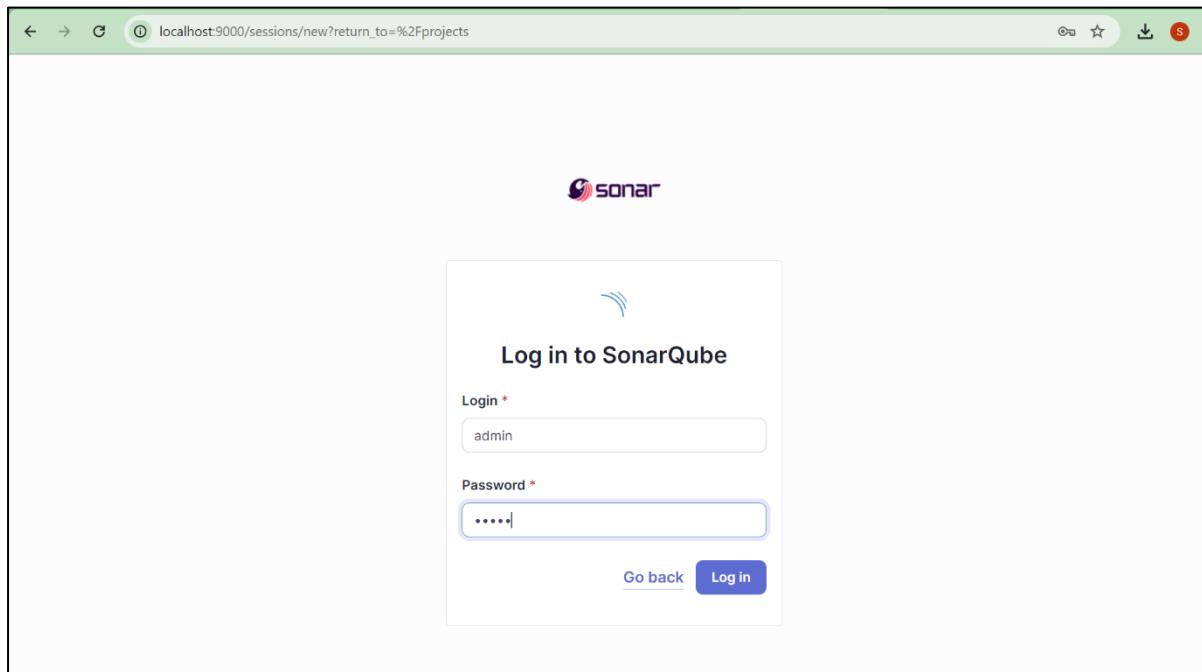
```
docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
```

```
C:\Users\SANNIDHI>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
3792724ca79359d2a71b0612ce535e0afe353d9df28039abd3323e61eee6f9f0
```

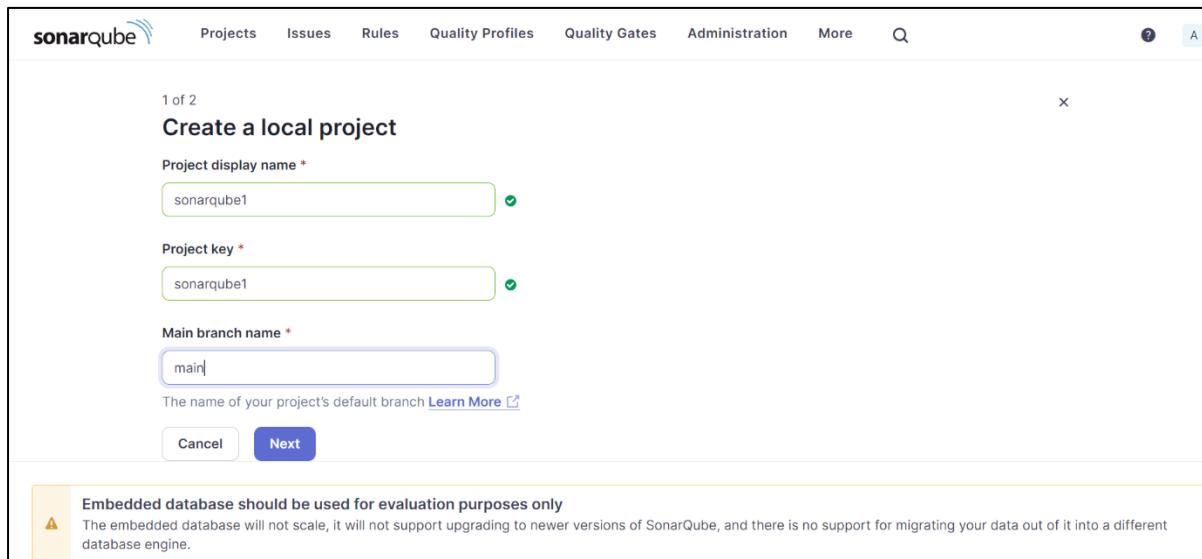
```
C:\Users\SANNIDHI>
```

Name	Image	Status	Port(s)	CPU (%)	Last started	Actions
minikube	gcr.io/k8s-minikube/kicbas	Exited (130)	0.22 Show all ports (5)	0%	9 hours ago	View Edit Delete
sonarqube	sonarqube:latest	Running	9000:9000 View	3.55%	56 minutes ago	View Edit Delete

3. Once the container is up and running, you can check the status of SonarQube at localhost port 9000.
4. Login to SonarQube using username admin and password admin.



5. Create a manual project in SonarQube with the name sonarqube1.



Setup the project and come back to Jenkins Dashboard.

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.
Recommended for projects following regular versions or releases.

6. Go to Manage Jenkins and search for SonarQube Scanner for Jenkins and install it.

Jenkins

Search (CTRL+K)

Sannidhi Kailaje log out

Dashboard > Manage Jenkins > Plugins

Plugins

sonarqu

Name	Status
SonarQube Scanner for Jenkins 2.17.2	Enabled

Updates Available plugins Installed plugins Advanced settings Download progress

REST API Jenkins 2.462.2

7. Go to Manage Jenkins > Configure System.

- Scroll down to the SonarQube Servers section and enter the required details:
 - Name: Any name you prefer.
 - Server URL: <http://localhost:9000>
 - Server Authentication Token: (Generate this token in SonarQube under My Account > Security > Generate Tokens).
 - Add Jenkins: Select Kind - Secret Text > Secret (Paste Generated Token)

The screenshot shows the Jenkins 'Configure System' page. In the 'SonarQube installations' section, a new server named 'sonarqube1' is being configured. The 'Name' field contains 'sonarqube1', the 'Server URL' field contains 'http://localhost:9000', and the 'Secret text' field contains a generated token. The 'Save' and 'Apply' buttons are visible at the bottom.

The screenshot shows the SonarQube 'Administrator' interface with the 'Security' tab selected. Under 'Generate Tokens', a new token named 'sonarqube1' is being created for 'Project Analysis Token' in project 'sonarqube1' to expire in '30 days'. Below, a table lists tokens, showing none have been generated yet.

Name	Type	Project	Last use	Created	Expiration
No tokens					

8. Search for SonarQube Scanner under Global Tool Configuration. Choose the latest configuration and choose Install automatically.

The screenshot shows the Jenkins Global Tool Configuration page. Under 'SonarQube Scanner installations', a new configuration is being created named 'sonarqube1'. The 'Install automatically' checkbox is checked. Under 'Install from Maven Central', the version 'SonarQube Scanner 6.2.0.4584' is selected. There are 'Save' and 'Apply' buttons at the bottom.

9. After the configuration, create a New Item in Jenkins, choose a freestyle project.

- Under Source Code Management, choose Git and enter the repository URL:
 - https://github.com/shazforiot/MSBuild_firstproject.git

The screenshot shows the Jenkins Configure screen for a new Freestyle Project named 'sonarqube.project'. Under 'Source Code Management', 'Git' is selected. The 'Repository URL' is set to 'https://github.com/shazforiot/MSBuild_firstproject.git'. The 'Branch Specifier' is set to '+master'. There are 'Save' and 'Apply' buttons at the bottom.

10. Under Build-> Execute SonarQube Scanner, enter these Analysis properties. Mention the SonarQube Project Key, Login, Password, Source path and Host URL.

The screenshot shows the Jenkins 'Configuration' screen for a job named 'sonarqube_project'. On the left, under 'Build Steps', the 'Execute SonarQube Scanner' step is selected. The 'Analysis properties' field contains the following configuration:

```
sonar.projectKey=sonarqube1
sonar.login=sop_80cd15f57d09d92e40c65692c2a30ebbf0a24l
sonar.sources=HelloWorldCore
sonar.host.url=http://localhost:9000
```

Below the analysis properties, there are fields for 'Additional arguments' and 'JVM Options', both currently empty. At the bottom of the configuration panel are 'Save' and 'Apply' buttons.

11. Go to http://localhost:9000/<user_name>/permissions and allow Execute Permissions to the Admin user.

The screenshot shows the SonarQube 'Administration' page under the 'Security' tab. It lists four groups: 'sonar-administrators', 'sonar-users', 'Anyone DEPRECATED', and 'Administrator admin'. For each group, checkboxes are provided for 'Administer System', 'Administer', 'Execute Analysis', and 'Create'. The 'Administrator admin' row has checked boxes for 'Administer System', 'Administer', 'Execute Analysis', and 'Create'. A note at the bottom states: 'Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.'

12. Run The Build.

sonarqube_project

SonarQube Quality Gate

sonarqube1 Passed

server-side processing: Success

Permalinks

- Last build (#2), 16 min ago
- Last stable build (#2), 16 min ago
- Last successful build (#2), 16 min ago
- Last failed build (#1), 31 min ago
- Last unsuccessful build (#1), 31 min ago
- Last completed build (#2), 16 min ago

Build History

#	Date	Status
#3	Oct 5, 2024, 9:49 PM	Passed
#2	Oct 5, 2024, 9:32 PM	Passed
#1	Oct 5, 2024, 9:18 PM	Passed

Atom feed for all Atom feed for failures

```

Started by user Sannidhi Kailaje
Running as SYSTEM
Building on the built-in node in workspace C:\ProgramData\Jenkins\workspace\sonarqube_project
The recommended git tool is: NONE
No credentials specified
> C:\Program Files\Git\bin\git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\workspace\sonarqube_project\.git # timeout=10
Fetching changes from the remote Git repository
> C:\Program Files\Git\bin\git.exe config remote.origin.url https://github.com/shazforiot/MSBuild_firstproject.git # timeout=10
Fetching upstream changes from https://github.com/shazforiot/MSBuild_firstproject.git
> C:\Program Files\Git\bin\git.exe --version # timeout=10
> git --version # 'git' version 2.46.2.windows.1'
> C:\Program Files\Git\bin\git.exe fetch -tags --force --progress -- https://github.com/shazforiot/MSBuild_firstproject.git
+refs/heads/*refs/remotes/origin/* # timeout=10
> C:\Program Files\Git\bin\git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision f2bc042c04c6e72427c380bcae6d6fee7b49adf (refs/remotes/origin/master)
> C:\Program Files\Git\bin\git.exe config core.sparsecheckout # timeout=10
> C:\Program Files\Git\bin\git.exe checkout -f f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
Commit message: "updated"
> C:\Program Files\Git\bin\git.exe rev-list --no-walk f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
[sonarqube_project] $ C:\ProgramData\Jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube1\bin\sonar-scanner.bat -
Dsonar.host.url=http://localhost:9000 ***** -Dsonar.projectKey=sonarqube1 -Dsonar.login=sgp_80db15f37d09d92e40c65692c2a30ebbf0a24b -
Dsonar.host.url=http://localhost:9000 -Dsonar.sources=HelloWorldCore -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\workspace\sonarqube_project
21:33:02.628 WARN Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
21:33:02.632 INFO Scanner configuration file: C:\ProgramData\Jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube1\bin..\conf\sonar-scanner.properties
21:33:02.632 INFO Project root configuration file: NONE
21:33:02.665 INFO SonarScanner CLI 6.2.0.4584
21:33:02.665 INFO Java 17.0.12 Eclipse Adoptium (64-bit)

```

EXPERIMENT 8

Aim:

Create a Jenkins CICD Pipeline with SonarQube / GitLab Integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web /Java / Python application.

Implementation:

Prerequisites

- Jenkins installed on your machine.
- Docker installed to run SonarQube.
- SonarQube installed via Docker.

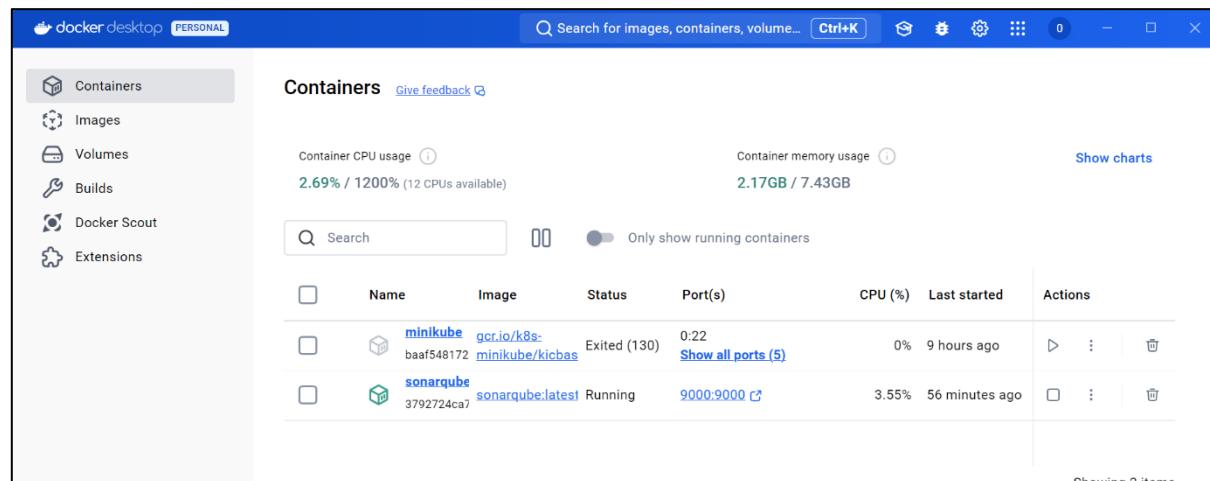
1. Open up Jenkins Dashboard on localhost, port 8080

2. Run SonarQube in a Docker container using command:

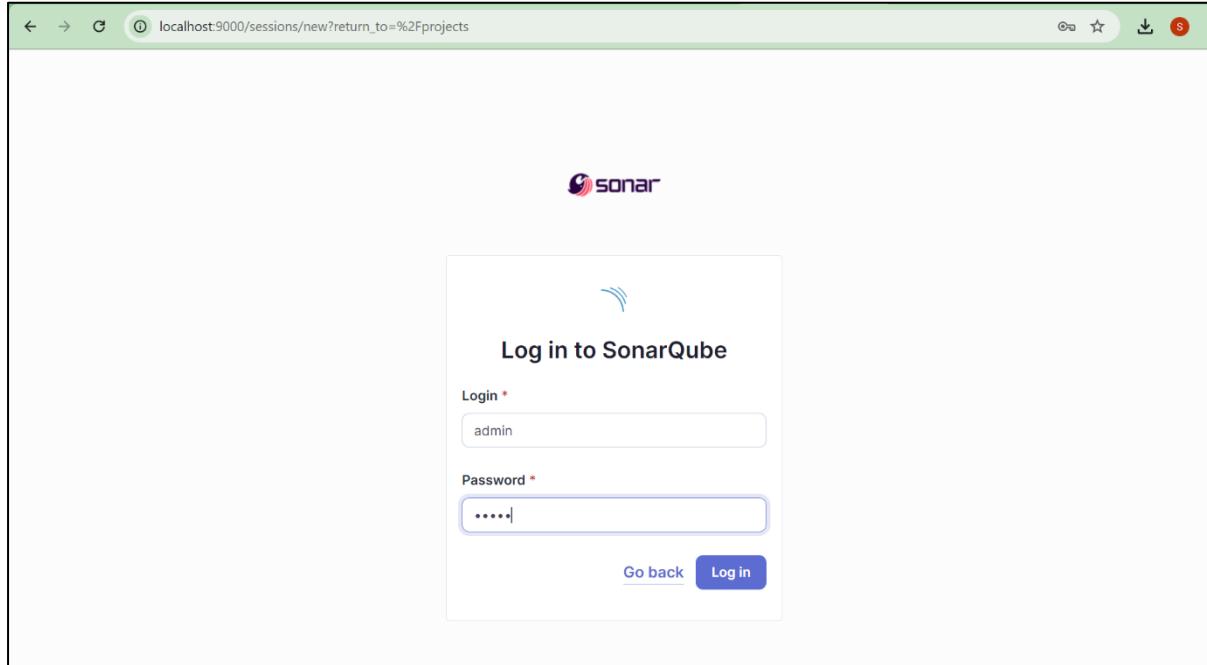
```
docker run -d --name sonarqube -e
SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000
sonarqube:latest
```

```
C:\Users\SANNIDHI>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CH
ECKS_DISABLE=true -p 9000:9000 sonarqube:latest
3792724ca79359d2a71b0612ce535e0afe353d9df28039abd3323e61eee6f9f0
```

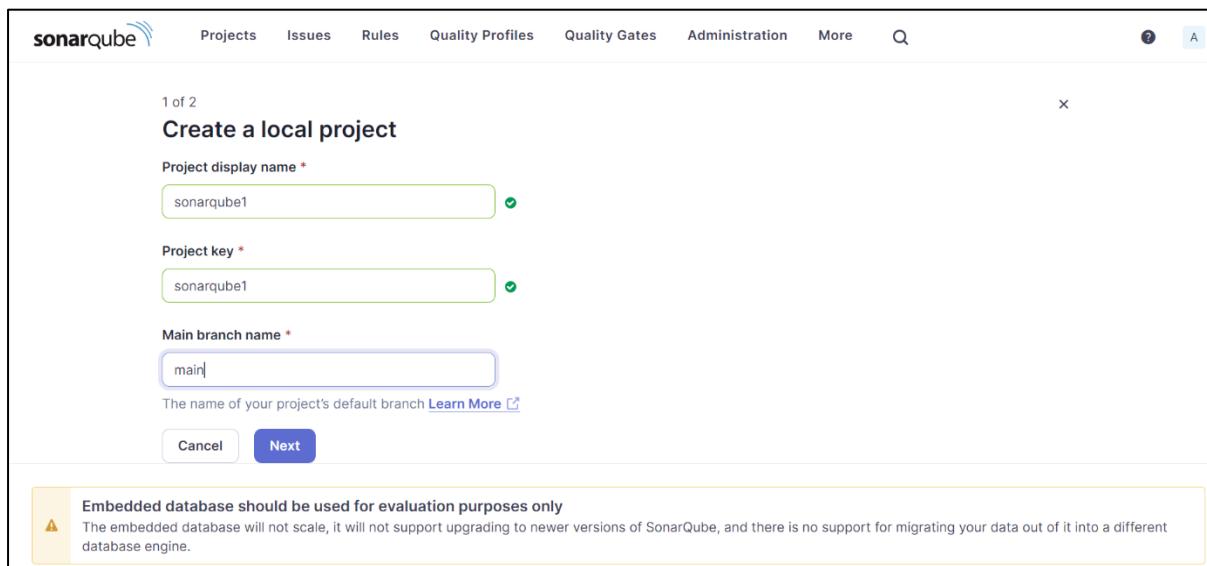
```
C:\Users\SANNIDHI>
```



3. Once the container is up and running, you can check the status of SonarQube at localhost port 9000.
4. Login to SonarQube with your credentials.



5. Create a manual project in SonarQube with the name sonarqube1.

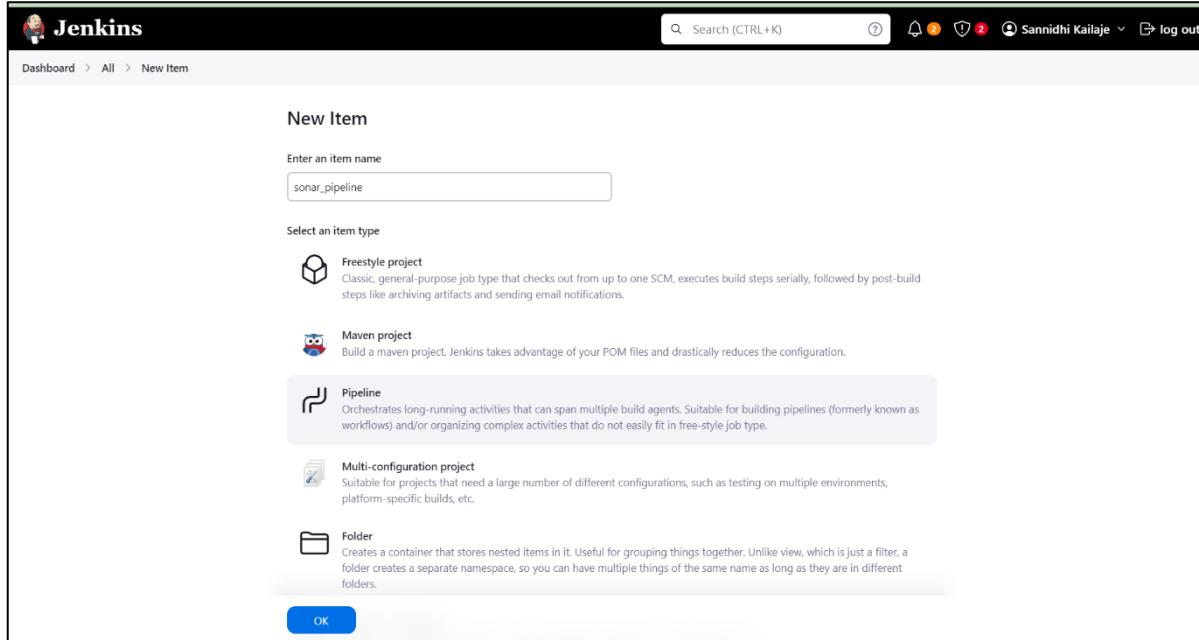


6. Generate SonarQube Token

- Go to My Account > Security > Generate Tokens.
- Copy the generated token for later use

7. Create a Jenkins Pipeline

- Go to Jenkins Dashboard, click New Item, and select Pipeline.



8. Under Pipeline Script, enter the following script:

```

docker network create sonarnet
node {
    stage('Cloning the GitHub Repo') {
        git 'https://github.com/shazforiot/GOL.git'
    }
    stage('SonarQube analysis') {
        withSonarQubeEnv('sonarqube') {
            sh """
                docker run --rm --network host \
                -e SONAR_HOST_URL=http://<ip_address>:9000 \
                -e SONAR_LOGIN=admin \
                -e SONAR_PASSWORD=<Sonarqube_password> \
                -e SONAR_PROJECT_KEY=sonarqube-test \
                -v ${WORKSPACE.replace('\\', '/')}:usr/src \
                sonarsource/sonar-scanner-cli \
                -Dsonar.projectKey=sonarqube-test \
                -Dsonar.exclusions=vendor/**,resources/**,**/*.java \
                -Dsonar.login=admin \
                -Dsonar.password=<Sonarqube_password>
            """
        }
    }
}

```

Dashboard > sonarpipe > Configuration

Configure

Advanced Project Options

General

Advanced Project Options

Pipeline

Pipeline

Definition

Pipeline script

Script ?

```
stage('Cloning the GitHub Repo') {  
    git 'https://github.com/shafiorot/GOL.git'  
}  
stage('SonarQube analysis') {  
    withSonarQubeEnv('sonarqube') {  
        sh ""  
        docker run -m 1000m --network host  
        -e SONAR_HOST_URL=http://192.168.1.103:9000 \  
        -e SONAR_LOGIN=admin  
        -e SONAR_PASSWORD=sonarqube@144  
        -e SONAR_PROJECT_KEY=sonarqube-test \  
        -v ${!NONSPACE_REPLACE.replace("\\\\","\\")}/usr/src \  
        sonar-source:sonar-scanner:2.11  
        -Dsonar.sources=src/main/java,test \  
        -Dsonar.exclusions=vendor/**,resources/**,*/*.java \  
        -Dsonar.login=admin \  
        -Dsonar.password=sonarqube@144  
    }  
}
```

Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

9. Run the Pipeline

- Save the pipeline and click Build Now.
 - Monitor the console output for any errors

Dashboard > sonarpipe >

[Status](#) [sonarpipe](#) [Add description](#) [Disable Project](#)

[Changes](#) [Build Now](#) [Configure](#) [Delete Pipeline](#)

[Full Stage View](#) [Stages](#) [Rename](#) [Pipeline Syntax](#)

[Build History](#) [trend](#) [Filter...](#)

[#11](#) Sep 22, 2024, 1:27AM

[#10](#) Sep 22, 2024, 1:26AM

Atom feed for all Atom feed for failures

Stage View

Cloning the GitHub Repo	SonarQube analysis
2s	5min 25s
1s	10min 37s
2s	12s failed

Average stage times:
(Average full run time: ~10min 41s)

#11 Sep 22 No Changes 01:27

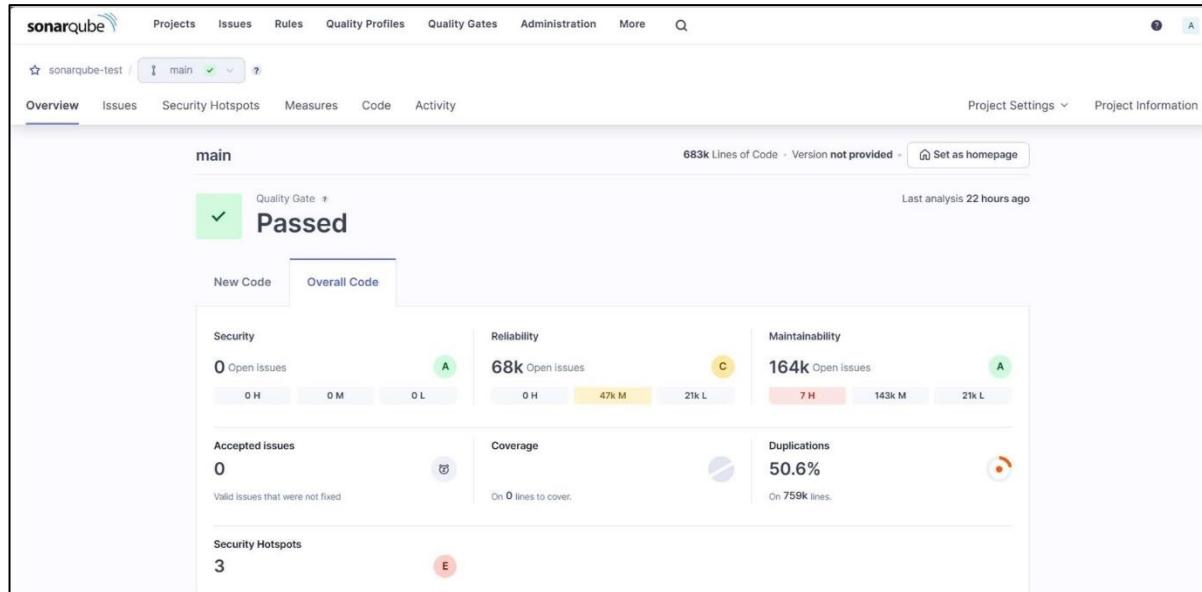
#10 Sep 22 No Changes 01:26

Permalinks

- Last build (#11), 11 min ago
- Last stable build (#11), 11 min ago
- Last successful build (#11), 11 min ago
- Last failed build (#10), 12 min ago
- Last unsuccessful build (#10), 12 min ago

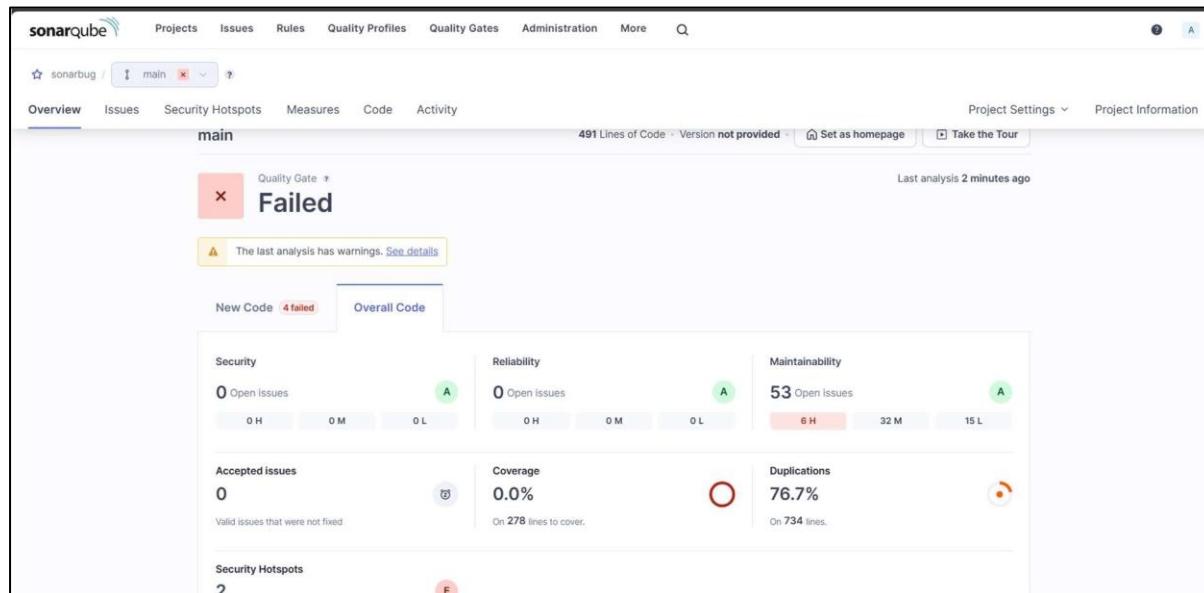
10. Check SonarQube for Analysis Results

- Go to your SonarQube dashboard and check the project for issues such as bugs, code smells, and security vulnerabilities.



11. Checking SonarQube for Analysis Results of a Code File with Bugs, Code Smells, Security Vulnerabilities, Cyclomatic Complexities and Duplicates.

- Overview –



- Issues –

The screenshot shows the SonarQube Issues page for the project 'sonarbug' under the 'main' branch. The 'Issues' tab is selected. On the left, there are filters for 'My Issues' and 'All'. The main area displays three code smell issues found in 'sonarbug.js':

- Function 'calculateInvoiceTotal' has too many parameters (8). Maximum allowed is 7.** (Adaptability, brain-overload) Status: Open, Assignee: Not assigned. L2 - 20min effort - 44 minutes ago - ⚡ Code Smell - ⚡ Major
- Expected a 'for-of' loop instead of a 'for' loop with this simple iteration.** (Consistency, clumsy) Status: Open, Assignee: Not assigned. L5 - 5min effort - 44 minutes ago - ⚡ Code Smell - ⚡ Minor
- Refactor this function to reduce its Cognitive Complexity from 36 to the 15 allowed.** (Adaptability, brain-overload) Status: Open, Assignee: Not assigned. L41 - 26min effort - 44 minutes ago - ⚡ Code Smell - ⚡ Critical

At the bottom, a note says: "⚠️ Embedded database should be used for evaluation purposes only".

- Security Hotspot (Security Vulnerabilities) –

The screenshot shows the SonarQube Security Hotspots page for the project 'sonarbug' under the 'main' branch. The 'Security Hotspots' tab is selected. It shows 0.0% security hotspots reviewed. There are two security hotspots listed:

- Review priority: High**
- Authentication**: Review this potentially hardcoded credential.
- Review this potentially hardcoded credential.**

On the right, a detailed view of a specific hotspot in 'sonarbug.js' is shown. The code snippet includes a hardcoded password:

```

    ...
    }
    // =====
    // Security Vulnerabilities Continued
    // =====
    // Security vulnerability: Hardcoded credentials
    function connectDatabase() {
        // Security issue: Hardcoded credentials should not be in source code
        const username = "admin";
        const password = "password123";
    }
    ...

```

A callout box highlights the hardcoded password 'password123' with the text: "Review this potentially hardcoded credential."

● Codesmells –

The screenshot shows the SonarQube interface for the project 'sonarbug' under the 'Issues' tab. On the left, a sidebar displays various quality metrics and filters. The main area lists three specific code smell issues found in 'sonarbug.js':

- Function 'calculateInvoiceTotal' has too many parameters (8). Maximum allowed is 7.** (Adaptability, brain-overload) - L2 - 20min effort - 16 minutes ago - Code Smell - Major
- Expected a 'for-of' loop instead of a 'for' loop with this simple iteration.** (Consistency, clumsy) - L5 - 5min effort - 16 minutes ago - Code Smell - Minor
- Refactor this function to reduce its Cognitive Complexity from 36 to the 15 allowed.** (Adaptability, brain-overload) - L41 - 26min effort - 16 minutes ago - Code Smell - Critical

A note at the bottom states: "Embedded database should be used for evaluation purposes only".

This screenshot shows a detailed view of a code smell issue in 'sonarbug.js'. The left sidebar shows the issue summary: 'Function 'calculateInvoiceTotal' has too many parameters (8). Maximum allowed is 7.' Below this, there are three additional bullet points related to the same issue:

- Expected a 'for-of' loop instead of a 'for' loop with this simple iteration.
- Refactor this function to reduce its Cognitive Complexity from 36 to the 15 allowed.
↳ 15 locations
- Function 'processPayment' has too many parameters (8). Maximum allowed is 7.

The right side of the screen shows the code editor with the problematic code highlighted. The code is as follows:

```

1 pranav... // Code Smell: Duplicate Code, Large Functions, Unused Variables, and Long Parameter Lists
2 pranav... function calculateInvoiceTotal(items, discount, taxRate, shipping, isGift, useCoupon, couponCode, isInternational) {
3 pranav...     let total = 0;
4 pranav...     for (let i = 0; i < items.length; i++) {
5 pranav...         // Bug: Incorrect price calculation
6 pranav...         total += items[i].price * items[i].quantity * 1.1; // Wrong multiplier
7 pranav...
8 pranav...

```

The specific line causing the issue is highlighted: 'total += items[i].price * items[i].quantity * 1.1; // Wrong multiplier'. A tooltip above this line reads: 'Function 'calculateInvoiceTotal' has too many parameters (8). Maximum allowed is 7.'

At the bottom, a note says: "Embedded database should be used for evaluation purposes only".

● Cyclomatic Complexity –

The screenshot shows the SonarQube interface for the project 'sonarbug' under the 'main' branch. The 'Measures' tab is selected. In the center, the 'Cyclomatic Complexity' section displays a score of 173. A prominent message at the top of this section reads: "Refactor this function to reduce its Cognitive Complexity from 36 to the 15 allowed." Below this message, there is a note: "Cognitive Complexity of functions should not be too high javascript:S3776". The code editor on the right shows a portion of 'sonarbug.js' with several cyclomatic complexity violations highlighted in red. These violations include: "Code Smell: Duplicate Code, Large Functions, Unused Variables, and Long Parameter Lists" (line 1), "Bug: Incorrect price calculation" (line 8), "Code Smell: Duplicate Code" (line 11), "Bug: Magic number used, no explanation" (line 14), "Bug: Shipping is always applied even if it's a gift" (line 21), "Extra charge for gift wrapping" (line 24), and "Code Smell: This check is overly complicated" (line 28). Annotations like 'New Code' and 'Uncovered code' are visible near the end of the code block.

● Cognitive Complexity –

The screenshot shows the SonarQube interface for the project 'sonarbug' under the 'main' branch. The 'Issues' tab is selected. In the center, the 'Cognitive Complexity' section displays a score of 36. A message at the top says: "Refactor this function to reduce its Cognitive Complexity from 36 to the 15 allowed." Below this, there is a note: "Cognitive Complexity of functions should not be too high javascript:S3776". The code editor on the right shows a portion of 'validatedOrder.js'. Several cognitive complexity violations are highlighted in red, particularly nested if statements. A callout box contains the message: "Refactor this function to reduce its Cognitive Complexity from 36 to the 15 allowed." Annotations like 'Clean code attribute' and 'Software qualities impacted' are visible on the right side.

- Duplications –

The screenshot shows the SonarQube interface for a project named 'sonarbug'. The 'Measures' tab is selected, specifically the 'Duplications' section. On the left, there's a sidebar with 'Overview' and 'New Code' sections. The 'New Code' section shows a 'Density' of 77.7% and 'Duplicated Lines' count of 543. The main panel displays a code editor for 'sonarbug > sonarbug.js' with line numbers 1 through 27. Several lines of code are highlighted with red and green annotations, indicating code smells and bugs respectively. A tooltip for line 11 reads: 'Duplicated Lines (%) on New Code 77.7%'.

```

1 // Code Smell: Duplicate Code, Large Functions, Unused Variables, and Long Parameter Lists
2 function calculateInvoiceTotal(items, discount, taxRate, shipping, isGift, useCoupon, couponCode, isInternational) {
3     let total = 0;
4
5     for (let i = 0; i < items.length; i++) {
6         // Bug: Incorrect price calculation
7         total += items[i].price * items[i].quantity * 1.1; // wrong multiplier
8
9     }
10
11    // Code Smell: Duplicate Code
12    if (discount > 0) {
13        total -= total * (discount / 100);
14    }
15
16    if (useCoupon && couponCode === 'SAVE20') {
17        total -= 20; // Magic number used, no explanation
18    }
19
20    total += total * (taxRate / 100); // Adding tax
21
22    // Bug: Shipping is always applied even if it's a gift
23    total += shipping;
24
25    if (isGift) {
26        total += 5; // Extra charge for gift wrapping
27    }

```

Conclusion:

In this experiment, we performed a static analysis of the code to detect bugs, code smells, and security vulnerabilities on our sample codes.

EXPERIMENT NO. 9

Aim:

To Understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

Implementation:

1. Create an Amazon Linux EC2 Instance

- Name it nagios-host.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Images, AMIs, AMI Catalog, and Elastic Block Store. The main content area shows a table with one row for the instance 'nagios-host'. The table columns include Name, Instance ID, Instance state, Instance type, Status check, and Alarm status. The instance is listed as 'Running' with the ID i-05899dd5e5b12359c. At the top right, there are buttons for 'Launch instances' and other actions. Below the table, a modal window titled 'Select an instance' is open, showing the same instance details. The bottom of the screen includes standard AWS footer links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

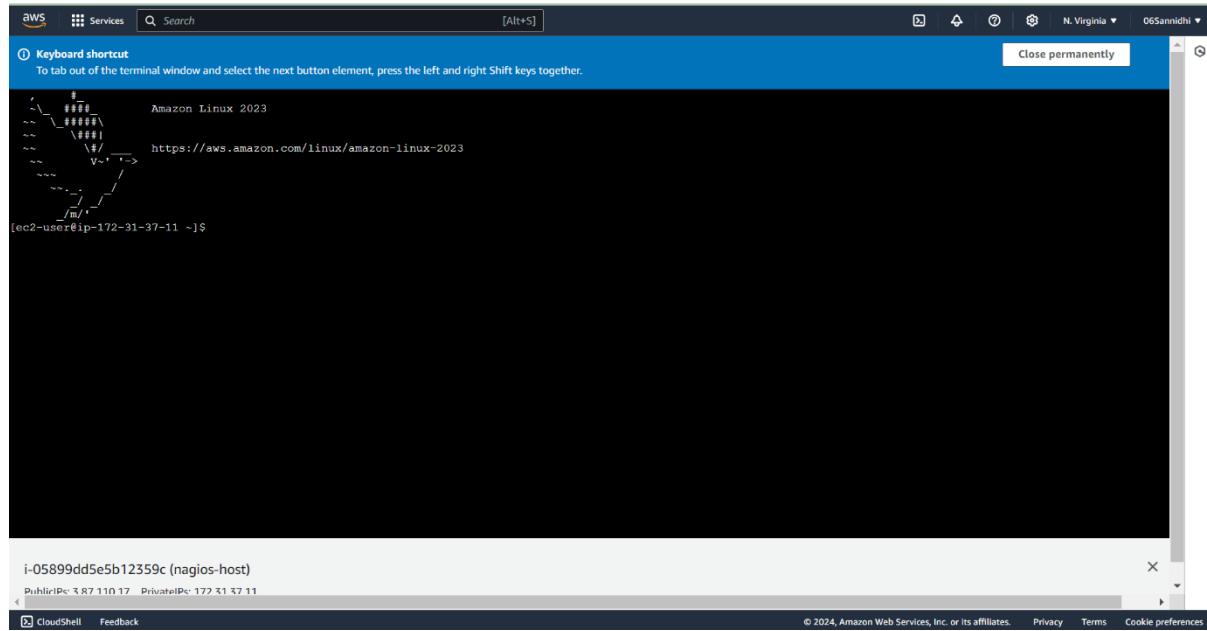
2. Configure Security Group

- Ensure HTTP, HTTPS, SSH, and ICMP are open from everywhere.
- Edit the inbound rules of the specified Security Group.

The screenshot shows the 'Edit inbound rules' page for a security group. The top header says 'Edit inbound rules' with a 'Info' link. Below it, a note says 'Inbound rules control the incoming traffic that's allowed to reach the instance.' The main area is titled 'Inbound rules' with an 'Info' link. It contains a table with columns: Security group rule ID, Type, Protocol, Port range, Source, and Description - optional. There are seven rows in the table, each representing an inbound rule. The rules allow traffic from anywhere (0.0.0.0/0) on various ports and protocols. The last rule is 'All traffic' on port 22. Each row has a 'Delete' button at the end. The bottom of the screen includes standard AWS footer links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

3. Connect to Your EC2 Instance

- SSH into your EC2 instance or use EC2 Instance Connect from the browser



4. Update Package Indices and Install Required Packages

Commands -

```
sudo yum update
sudo yum install httpd php
sudo yum install gcc glibc glibc-common
sudo yum install gd gd-devel
```

```
[ec2-user@ip-172-31-80-22 ~]$ sudo yum update -y
Last metadata expiration check: 0:09:18 ago on Thu Sep 26 08:41:50 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-80-22 ~]$ sudo yum install -y httpd php
Last metadata expiration check: 0:09:40 ago on Thu Sep 26 08:41:50 2024.
Dependencies resolved.
```

Package	Architecture	Version
---------	--------------	---------

```
Complete!
[ec2-user@ip-172-31-80-22 ~]$ sudo yum install -y gcc glibc glibc-common
sudo yum install -y gd gd-devel
Last metadata expiration check: 0:09:57 ago on Thu Sep 26 08:41:50 2024.
Package glibc-2.34-52.amzn2023.0.11.x86_64 is already installed.
Package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Dependencies resolved.
```

Package	Architecture	Version
Installing:	x86_64	11.4.1-2.amzn2023.0.2
Installing dependencies:		

5. Create a New Nagios User

Commands -

```
sudo adduser -m nagios
sudo passwd nagios
```

```
Complete!
[ec2-user@ip-172-31-37-11 ~]$ sudo useradd nagios
[ec2-user@ip-172-31-37-11 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
Sorry, passwords do not match.
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
Sorry, passwords do not match.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-37-11 ~]$
```

6. Create a New User Group

Commands -

```
sudo groupadd nagcmd
```

```
[ec2-user@ip-172-31-37-11 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-37-11 ~]$
```

i-05899dd5e5b12359c (nagios-host)

PublicIPs: 54.91.123.99 PrivateIPs: 172.31.37.11

The screenshot shows a terminal window titled 'aws Services Search [Alt+S]'. It displays the command 'sudo groupadd nagcmd' being run and its output. Below the terminal is a scrollable list of installed packages on the system. The terminal output is as follows:

```
[ec2-user@ip-172-31-37-11 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-37-11 ~]$
```

The package list includes many standard Linux utilities and libraries, such as brotli, cairo, fontconfig, freetype, glib, graphite2, harfbuzz, libxml2, libxslt, libxrender, libffi, libjpeg-turbo, libpng, libsepol, libwebp, libxcb, pcre2, sysprof, and xz.

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.9-4.amzn2023.0.2.x86_64
cairo	1.17.6-2.amzn2023.0.3.x86_64
fontconfig-devel	2.13.9-1.amzn2023.0.2.x86_64
freetype	2.13.9-1.amzn2023.0.2.x86_64
glib	2.33.1-1.amzn2023.0.3.x86_64
graphite2	1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	dev-2.74.7-699.amzn2023.0.1.x86_64
langpacks-core	for-en-3.0-21.amzn2023.0.4.noarch
libxml2	1.7.2-3.amzn2023.0.4.x86_64
libxslt	1.7.2-3.amzn2023.0.4.x86_64
libxrender	0.9.10-14.amzn2023.0.2.x86_64
libffi	dev-3.4.4-1.amzn2023.0.1.x86_64
libjpeg-turbo	2.1.4-2.amzn2023.0.5.x86_64
libpng	2.1.6.37-10.amzn2023.0.6.x86_64
libsepol	dev-3.4-3.amzn2023.0.3.x86_64
libwebp	1.2.4-1.amzn2023.0.6.x86_64
libxcb	dev-1.13.1-7.amzn2023.0.2.x86_64
pcre2	utf16-10.40-1.amzn2023.0.3.x86_64
sysprof	capture-devel-3.40.1-2.amzn2023.0.2.x86_64
xz	devel-5.2.5-9.amzn2023.0.2.x86_64
bzip2-devel	-1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11	-compat-1.7.2-3.amzn2023.0.4.noarch
libXau	-1.0.9-6.amzn2023.0.2.x86_64
libXpm	-3.5.15-2.amzn2023.0.3.x86_64
libXt	-1.2.0-4.amzn2023.0.2.x86_64
libicu	-67.1-7.amzn2023.0.3.x86_64
libjpeg-turbo-devel	-2.1.4-2.amzn2023.0.5.x86_64
libpng-devel	-2.1.6.37-10.amzn2023.0.6.x86_64
libtiff	-4.4.0-4.amzn2023.0.18.x86_64
libwebp	-dev-1.2.4-1.amzn2023.0.6.x86_64
libxml2-devel	-2.10.4-1.amzn2023.0.6.x86_64
pcre2	-utf32-10.40-1.amzn2023.0.3.x86_64
xml-common	-0.6.3-56.amzn2023.0.2.noarch
xz-devel	-1.2.11-33.amzn2023.0.5.x86_64

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel	-10.40-1.amzn2023.0.3.x86_64
pixman	-0.40.0-1.amzn2023.0.3.x86_64
xorg-x11proto-devel	-2021.4-1.amzn2023.0.2.noarch

At the bottom of the terminal, it says 'Complete!' followed by the prompt '[ec2-user@ip-172-31-37-11 ~]\$'.

The scrollable list of packages is as follows:

Package	Version
brotli-devel	1.0.8-6.amzn2023.0.2.x86_64
fontconfig	-2.13.94-2.amzn2023.0.2.x86_64
freetype	-2.13.9-1.amzn2023.0.2.x86_64
glib	-2.33.1-5.amzn2023.0.3.x86_64
graphite2	-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz	-7.0.0-2.amzn2023.0.1.x86_64
libICE	-1.0.10-6.amzn2023.0.2.x86_64
libX11-devel	-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel	-1.0.9-6.amzn2023.0.2.x86_64
libXpm-devel	-3.5.15-2.amzn2023.0.3.x86_64
libXt-devel	-2.37.4-1.amzn2023.0.4.x86_64
libicu-devel	-67.1-7.amzn2023.0.3.x86_64
libmount-devel	-2.37.4-1.amzn2023.0.4.x86_64
libselinux-devel	-3.4-5.amzn2023.0.2.x86_64
libtiff-devel	-4.4.0-4.amzn2023.0.18.x86_64
libxcb	-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel</td	

7. Add Users to the Group

Commands -

```
sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
```

```
[ec2-user@ip-172-31-37-11 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-37-11 ~]$ sudo usermod -aG nagcmd nagios
[ec2-user@ip-172-31-37-11 ~]$ sudo usermod -aG nagcmd apache
```

8. Create a Directory for Nagios Downloads

Commands -

```
mkdir ~/downloads
cd ~/downloads
```

```
[ec2-user@ip-172-31-37-11 ~]$ mkdir ~/downloads
[ec2-user@ip-172-31-37-11 ~]$ cd ~/downloads
```

9. Download Nagios and Plugins Source Files

Commands -

Wget [https://assets.nagios.com/downloads/nagioscore/releases/nagios-](https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz)

[4.4.6.tar.gz](https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz)

wget <https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz>

```
[ec2-user@ip-172-31-37-11 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz[ec2-user@ip-172-31-37-11 downloads]$ wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz--2024-10-08 13:47:31-- https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gzResolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00:f03c:92ff:fe7:45ceConnecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.HTTP request sent, awaiting response... 200 OKLength: 11333414 (11M) [application/x-gzip]Saving to: 'nagios-4.4.6.tar.gz' 0%[=====] 0 --.-KB/s    nagi...nagios-4.4.6.t...4.4.6.tar.gz 22%[=====>] 271.62K 1.31MB/s    nagi...nagios-4.4.6.t...4.4.6.tar.gz 62%[=====>] 2.45M 6.03MB/s    nagi...nagios-4.4.6.t...4.4.6.tar.gz 97%[=====>] 10.59M 13.0MB/s    nagi...nagios-4.4.6.t...4.4.6.tar.gz 100%[=====>] 10.81M 13.3MB/s    nagi...nagios-4.4.6.t...2024-10-08 13:47:33 (13.3 MB/s) - 'nagios-4.4.6.tar.gz' saved [11333414/11333414]--2024-10-08 13:47:33-- https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gzResolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:443... connected.HTTP request sent, awaiting response... 200 OKLength: 2782610 (2.7M) [application/x-gzip]Saving to: 'nagios-plugins-2.3.3.tar.gz' 0%[=====] 0 --.-KB/s    nagi...nagios-plug...-2.3.3.tar.gz 10%[=====>] 296.00K 1.43MB/s    nagi...nagios-plug...-2.3.3.tar.gz 59%[=====>] 1.59M 3.91MB/s    nagi...nagios-plug...-2.3.3.tar.gz 100%[=====>] 2.65M 6.10MB/s    in 0.4s 2024-10-08 13:47:33 (6.10 MB/s) - 'nagios-plugins-2.3.3.tar.gz' saved [2782610/2782610]
```

10. Extract the Nagios Source File

Commands -

```
tar zxvf nagios-4.4.6.tar.gz
cd nagios-4.4.6
```

```
[ec2-user@ip-172-31-37-11 downloads]$ tar zxvf nagios-4.4.6.tar.gz
cd nagios-4.4.6
nagios-4.4.6/
nagios-4.4.6/.gitignore
nagios-4.4.6/.travis.yml
nagios-4.4.6/CONTRIBUTING.md
nagios-4.4.6/Changelog
nagios-4.4.6/INSTALLING
nagios-4.4.6/LEGAL
nagios-4.4.6/LICENSE
nagios-4.4.6/Makefile.in
nagios-4.4.6/README.md
nagios-4.4.6/THANKS
nagios-4.4.6/UPGRADING
nagios-4.4.6/aclocal.m4
nagios-4.4.6/autoconf-macros/
nagios-4.4.6/autoconf-macros/.gitignore
nagios-4.4.6/autoconf-macros/CHANGELOG.md
nagios-4.4.6/autoconf-macros/LICENSE
nagios-4.4.6/autoconf-macros/LICENSE.md
nagios-4.4.6/autoconf-macros/README.md
nagios-4.4.6/autoconf-macros/add_group_user
nagios-4.4.6/autoconf-macros/ax_nagios_get_distrib
nagios-4.4.6/autoconf-macros/ax_nagios_get_files
nagios-4.4.6/autoconf-macros/ax_nagios_get_inetd
nagios-4.4.6/autoconf-macros/ax_nagios_get_init
nagios-4.4.6/autoconf-macros/ax_nagios_get_os
nagios-4.4.6/autoconf-macros/ax_nagios_get_paths
```

11. Run the Configuration Script

Commands -

```
./configure --with-command-group=nagcmd
```

```
AWS Services Search [Alt+S]
config.status: creating lib/iobroker.h
Creating sample config files in sample-config/ ...

*** Configuration summary for nagios 4.4.6 2020-04-28 ***:
General Options:
-----
  Nagios executable: nagios
  Nagios user/group: nagios,nagios
  Command user/group: nagios,nagcmd
  Event Broker: yes
  Install $prefix: /usr/local/nagios
  Install $includedir: /usr/local/nagios/include/nagios
  Lock file: /var/nagios.lock
  Check result directory: /usr/local/nagios/var/spool/checkresults
  Init directory: /lib/systemd/system
  Apache conf.d directory: /etc/httpd/conf.d
    Mail program: /bin/mail
    Host OS: linux-gnu
    IOBroker Method: epoll

Web Interface Options:
-----
  HTML URL: http://localhost/nagios/
  CGI URL: http://localhost/nagios/cgi-bin/
  Traceroute (used by WAP): /usr/bin/traceroute

Review the options above for accuracy. If they look okay,
type 'make all' to compile the main program and CGIs.
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$
```

12. Compile the Source Code

Commands: - make all

```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ make all
cd ./base && make
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/base'
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o broker.o broker.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nebmods.o nebmods.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ../common/shared.o ../common/
shared.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o query-handler.o query-handl
er.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
In function `get_wproc_list',
  inlined from `get_worker' at workers.c:277:12:
workers.c:253:17: warning: `%s' directive argument is null [-Wformat-overflow=]
  253 |           log_debug_info(DEBUGL_CHECKS, 1, "Found specialized worker
(s) for '%s'", (slash && *slash != '/') ? slash : cmd_name);
  |
  ~~~~~~
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o checks.o checks.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o config.o config.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o commands.o commands.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o events.o events.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o flapping.o flapping.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o logging.o logging.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o macros-base.o ../common/macro
s.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o netutils.o netutils.c
```

*** Support Notes *****

If you have questions about configuring or running Nagios,
please make sure that you:

- Look at the sample config files
- Read the documentation on the Nagios Library at:
<https://library.nagios.com>

before you post a question to one of the mailing lists.
Also make sure to include pertinent information that could
help others help you. This might include:

- What version of Nagios you are using
- What version of the plugins you are using
- Relevant snippets from your config files
- Relevant error messages from the Nagios log file

For more information on obtaining support for Nagios, visit:

<https://support.nagios.com>

Enjoy.

```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ ]
```

13. Install Binaries, Init Script, and Sample Config Files

Commands -

```
./sudo make install
sudo make install-init
sudo make install-config
sudo make install-commandmode
```

```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ sudo make install
sudo make install-init
sudo make install-config
sudo make install-commandmode
cd ./base && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagiosstats /usr/local/nagios/bin
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/cgi'
```

```
*** Config files installed ***
```

```
Remember, these are *SAMPLE* config files. You'll need to read
the documentation for more information on how to actually define
services, hosts, etc. to fit your particular needs.
```

```
/usr/bin/install -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw
```

```
*** External command directory configured ***
```

```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ ]
```

14. Edit the Config File to Change the Email Address

Commands -

```
sudo nano /usr/local/nagios/etc/objects/contacts.cfg
```

- Change the email address in the contacts.cfg file to your preferred email.

```
# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the
# 'generic-contact' template which is defined elsewhere.

define contact {
    contact_name      nagiosadmin          ; Short name of user
    use               generic-contact       ; Inherit default values from
    alias             Nagios Admin        ; Full name of user
    email             06sannidhi@gmail.com; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****

}

#####
# CONTACT GROUPS
#####

# We only have one contact in this simple configuration file, so there is
# no need to create more than one contact group.

define contactgroup {
    contactgroup_name   admins
    alias              Nagios Administrators
    members            nagiosadmin
}

generic-contact template (defined above)
```

15. Configure the Web Interface

Commands -

```
sudo make install-webconf
```

```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ sudo nano /usr/local/nagios/etc/objects/contacts.cfg
```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ sudo make install-webconf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/httpd/conf.d/nagios.conf
if [0 -eq 1]; then \
 ln -s /etc/httpd/conf.d/nagios.conf /etc/apache2/sites-enabled/nagios.conf
; \
fi

*** Nagios/Apache conf file installed ***

[ec2-user@ip-172-31-37-11 nagios-4.4.6]$
```

## 16. Create a Nagios Admin Account

Commands -

```
sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

You will be prompted to enter and confirm the password for the nagiosadmin user.

```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$
```

## 17. Restart Apache

Commands - systemctl restart httpd

```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ sudo systemctl restart httpd
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$
```

## 18. Extract the Plugins Source File

Commands -

```
cd ~/downloads
tar zxvf nagios-plugins-2.3.3.tar.gz
cd nagios-plugins-2.3.3
```

```
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ sudo systemctl restart httpd
[ec2-user@ip-172-31-37-11 nagios-4.4.6]$ cd ~/downloads
tar zxvf nagios-plugins-2.3.3.tar.gz
cd nagios-plugins-2.3.3
nagios-plugins-2.3.3/
nagios-plugins-2.3.3/perlmods/
nagios-plugins-2.3.3/perlmods/Config-Tiny-2.14.tar.gz
nagios-plugins-2.3.3/perlmods/parent-0.226.tar.gz
nagios-plugins-2.3.3/perlmods/Test-Simple-0.98.tar.gz
nagios-plugins-2.3.3/perlmods/Makefile.in
nagios-plugins-2.3.3/perlmods/version-0.9903.tar.gz
nagios-plugins-2.3.3/perlmods/Makefile.am
nagios-plugins-2.3.3/perlmods/Module-Runtime-0.013.tar.gz
nagios-plugins-2.3.3/perlmods/Module-Metadata-1.000014.tar.gz
nagios-plugins-2.3.3/perlmods/Params-Validate-1.08.tar.gz
nagios-plugins-2.3.3/perlmods/Class-Accessor-0.34.tar.gz
nagios-plugins-2.3.3/perlmods/Try-Tiny-0.18.tar.gz
nagios-plugins-2.3.3/perlmods/Module-Implementation-0.07.tar.gz
nagios-plugins-2.3.3/Makefile
nagios-plugins-2.3.3/perlmods/Perl-OSType-1.003.tar.gz
nagios-plugins-2.3.3/perlmods/install_order
nagios-plugins-2.3.3/perlmods/Nagios-Plugin-0.36.tar.gz
nagios-plugins-2.3.3/perlmods/Math-Calc-Units-1.07.tar.gz
nagios-plugins-2.3.3/perlmods/Module-Build-0.4007.tar.gz
nagios-plugins-2.3.3/ABOUT-NLS
nagios-plugins-2.3.3/configure.ac
nagios-plugins-2.3.3/Makefile.in
nagios-plugins-2.3.3/config.h.in
nagios-plugins-2.3.3/ChangeLog
nagios-plugins-2.3.3/AUTHORS
nagios-plugins-2.3.3/lib/
nagios-plugins-2.3.3/lib/parse_ini.h
```

## 19. Compile and Install Plugins

Commands -

```
./configure --with-nagios-user=nagios --with-nagios-group=nagios
make
sudo make install
```

```
[ec2-user@ip-172-31-37-11 nagios-plugins-2.3.3]$./configure --with-nagios-user=nagios --with-nagios-group=nagios
./configure --with-nagios-user=nagios --with-nagios-group=nagios
make make install
sudo make install
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether to disable maintainer-specific portions of Makefiles... yes
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
checking for gcc... gcc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking for style of include used by make... GNU
checking dependency style of gcc... gcc3
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
```

## 20. Start Nagios

Commands -

```
sudo chkconfig --add nagios
sudo chkconfig nagios on
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
sudo systemctl start nagios
```

```
[ec2-user@ip-172-31-37-11 nagios-plugins-2.3.3]$ sudo chkconfig --add nagios
sudo chkconfig nagios on
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
sudo systemctl start nagios
error reading information on service nagios: No such file or directory
Note: Forwarding request to 'systemctl enable nagios.service'.
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /usr/lib/systemd/system/nagios.service.

Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPLv3

Website: https://www.nagios.org
Reading configuration data...
 Read main config file okay...
 Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
 Checked 8 services.
 Checked 1 hosts.
 Checked 1 host groups.
 Checked 0 service groups.
 Checked 1 contacts.
 Checked 1 contact groups.
 Checked 24 commands.
 Checked 5 time periods.
 Checked 0 host escalations.
 Checked 0 service escalations.
Checking for circular paths...
```

## 21. Check the Status of Nagios

Commands -

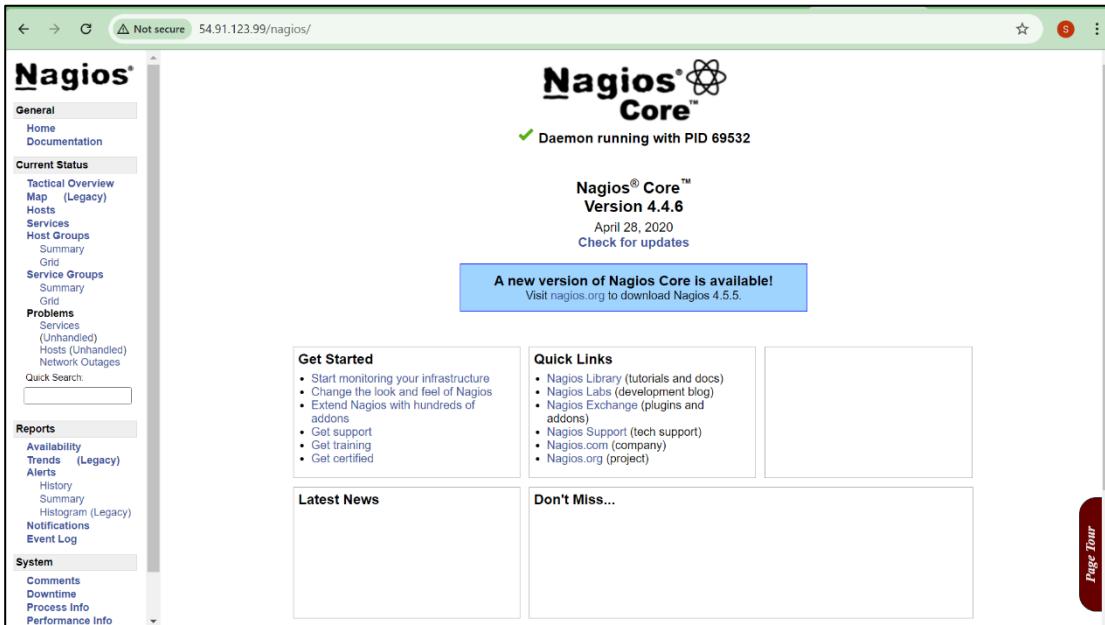
```
sudo systemctl status nagios
```

```
Things look okay - No serious problems were detected during the pre-flight check
[ec2-user@ip-172-31-37-11 nagios-plugins-2.3.3]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.6
 Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
 Active: active (running) since Tue 2024-10-08 14:14:32 UTC; 1min 33s ago
 Docs: https://www.nagios.org/documentation
 Process: 69530 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Process: 69531 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 69532 (nagios)
 Tasks: 6 (limit: 1112)
 Memory: 2.3M
 CPU: 30ms
 CGroup: /system.slice/nagios.service
 └─69532 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
 ├─69533 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
 ├─69534 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
 ├─69535 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
 ├─69536 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
 └─69537 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: qh: core query handler registered
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: qh: echo service query handler registered
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: qh: help for the query handler registered
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: Successfully registered manager as @wproc with query handler
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: Registry request: name=Core Worker 69536;pid=69536
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: Registry request: name=Core Worker 69535;pid=69535
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: Registry request: name=Core Worker 69534;pid=69534
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: Registry request: name=Core Worker 69533;pid=69533
Oct 08 14:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: Successfully launched command file worker with pid 69537
[ec2-user@ip-172-31-37-11 nagios-plugins-2.3.3]$
```

## 22. Access Nagios Web Interface

- Copy the Public IP address of your EC2 instance.
- Open your browser and navigate to [http://<your\\_public\\_ip\\_address>/nagios](http://<your_public_ip_address>/nagios).
- Enter the username nagiosadmin and the password you set in Step 16.



## Conclusion:

After installing and configuring Nagios Core, Plugins, and NRPE on a Linux machine, We have a robust continuous monitoring setup, ensuring proactive issue detection and optimal system performance.

# **EXPERIMENT NO. 10**

### Aim:

To perform Port, Service monitoring, Windows/Linux server monitoring using Nagios.

## **Implementation:**

## Prerequisites

- AWS Free Tier
  - Nagios Server running on an Amazon Linux Machine

## 1. Confirm Nagios is Running on the Server

## Commands -

```
sudo systemctl status nagios
```

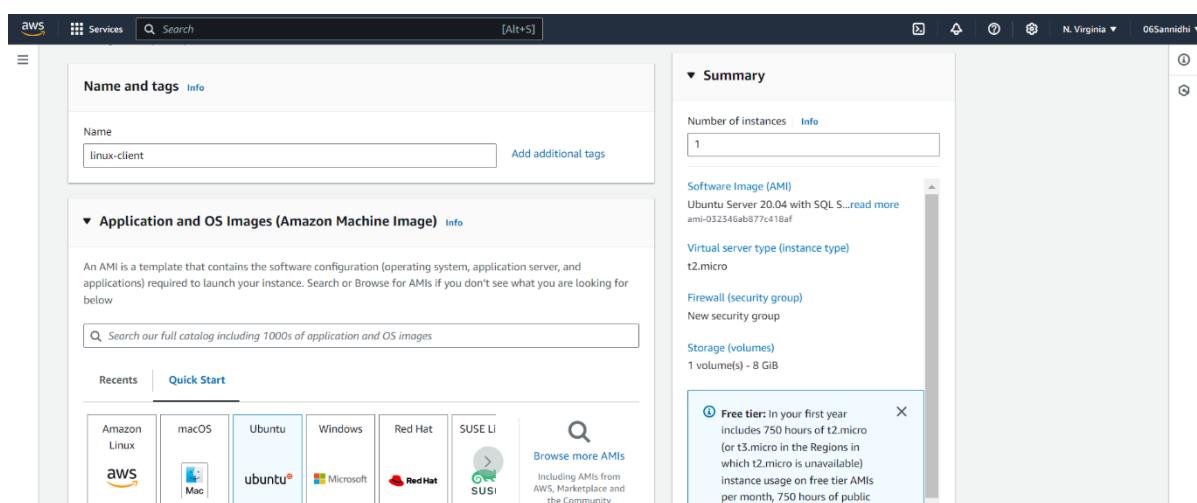
- Proceed if you see that Nagios is active and running.

```
[ec2-user@ip-172-31-37-11 ~]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
 Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
 Active: active (running) since Fri 2014-10-03 14:14:32 UTC; 1h 23min ago
 Docs: https://www.nagios.org/documentation
 Process: 69530 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Process: 69531 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 69532 (nagios)
 Tasks: 6 (limit: 1112)
 Memory: 3.2M
 CPU: 1.082s
 CGGroup: /system.slice/nagios.service
 └─69532 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
 ├─69533 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
 ├─69534 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
 ├─69535 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
 ├─69536 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
 └─69537 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 08 14:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: early_timeout=0; exited ok=1; wait_status=32512; error_code=0;
Oct 08 14:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: stderr line 01: /bin/mail: No such file or directory
Oct 08 14:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: stderr line 02: /usr/bin/printf: write error: Broken pipe
Oct 08 15:14:32 ip-172-31-37-11.ec2.internal nagios[69532]: Auto-save of retention data completed successfully.
Oct 08 15:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: SERVICE_NOTIFY: nagiosadmin;localhost;Swap Usage;CRITICAL;notify-service-by-email;SWAP CRITICAL - 0 free
Oct 08 15:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: NOTIFY job 32 from worker Core Worker 69536 is a non-check helper but exited with return code 127
Oct 08 15:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: host=localhost; service=Swap Usage; contact=nagiosadmin
Oct 08 15:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: early_timeout=0; exited ok=1; wait_status=32512; error_code=0;
Oct 08 15:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: stderr line 01: /bin/mail: No such file or directory
Oct 08 15:21:54 ip-172-31-37-11.ec2.internal nagios[69532]: wproc: stderr line 02: /usr/bin/printf: write error: Broken pipe
Lines: 17/25 (END)
```

## 2. Create an Ubuntu 20.04 Server EC2 Instance

- Name it linux-client.
  - Use the same security group as the Nagios Host.



### 3. Verify Nagios Process on the Server

Commands -

```
ps -ef | grep nagios
```

```
[ec2-user@ip-172-31-37-11 ~]$ ps -ef | grep nagios
nagios 69532 1 0 14:14 ? 00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios 69533 69532 0 14:14 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 69534 69532 0 14:14 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 69535 69532 0 14:14 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 69536 69532 0 14:14 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 69537 69532 0 14:14 ? 00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
[ec2-user 75736 74742 0 15:55 pts/0 00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-37-11 ~]$
```

i-05899dd5e5b12359c (nagios-host)

### 4. Become Root User and Create Directories

Commands -

```
sudo su
```

```
mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
```

```
[ec2-user@ip-172-31-37-11 ~]$ sudo su
[root@ip-172-31-37-11 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-37-11 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
sts
[root@ip-172-31-37-11 ec2-user]#
```

### 5. Copy Sample Configuration File

Commands -

```
cp /usr/local/nagios/etc/objects/localhost.cfg
```

```
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

```
[root@ip-172-31-37-11 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-37-11 ec2-user]# sudo nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

### 6. Edit the Configuration File

Commands -

```
sudo nano
```

```
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

- Change hostname to linuxserver everywhere in the file.
- Change address to the public IP address of your linux-client.

```
Define a host for the local machine
define host {
 use linux-server ; Name of host template to use
 ; This host definition will inherit all variables that are defined
 ; in (or inherited by) the linux-server host template definition.

 host_name linuxserver
 alias linuxserver
 address 54.88.143.206
}
```

- Change hostgroup\_name under hostgroup to linux-servers1.

```
#####
HOST GROUP DEFINITION
#
#####

Define an optional hostgroup for Linux machines

define hostgroup {
 hostgroup_name linux-servers ; The name of the hostgroup
 alias Linux Servers ; Long name of the group
 members linuxserver1 ; Comma separated list of hosts that belong to this group
}

#####
#
```

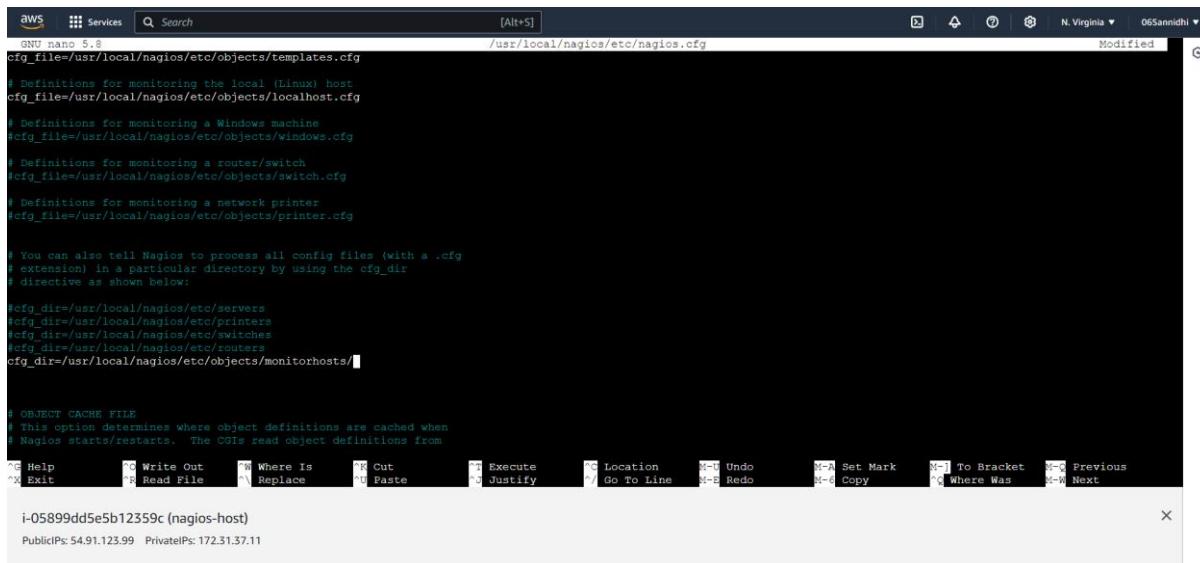
## 7. Update Nagios Configuration

Commands -

`sudo nano /usr/local/nagios/etc/nagios.cfg`

- Add the following line:

`cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/`



## 8. Verify Configuration Files

Commands -

`sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg`

- Ensure there are no errors.

```
[root@ip-172-31-37-11 ec2-user]# sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org
Reading configuration data...
 Read main config file okay...
Warning: Duplicate definition found for service 'HTTP' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg', starting on line 152)
Warning: Duplicate definition found for service 'SSH' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg', starting on line 138)
Warning: Duplicate definition found for service 'Swap Usage' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg', starting on line 125)
Warning: Duplicate definition found for service 'Current Load' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg', starting on line 112)
Warning: Duplicate definition found for service 'Total Processes' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg', starting on line 100)
Warning: Duplicate definition found for service 'Current Users' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg', starting on line 86)
Warning: Duplicate definition found for service 'Root Partition' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg', starting on line 72)
Warning: Duplicate definition found for service 'PING' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg', starting on line 58)
 Read object config files okay...

Running pre-flight check on configuration data...
```

```

Running pre-flight check on configuration data...
Checking objects...
 Checked 8 services.
 Checked 2 hosts.
 Checked 2 host groups.
 Checked 0 service groups.
 Checked 1 contact.
 Checked 1 contact groups.
 Checked 24 commands.
 Checked 5 time periods.
 Checked 0 host escalations.
 Checked 0 service escalations.
Checking for circular paths...
 Checked 2 hosts
 Checked 0 service dependencies
 Checked 0 host dependencies
 Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

total Warnings: 0
total Errors: 0

Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-37-11 ec2-user]#

```

## 9. Restart Nagios Service

Commands -  
 sudo systemctl restart nagios

## 10. SSH into the Client Machine

- Use SSH or EC2 Instance Connect to access the linux-client.

## 11. Update Package Index and Install Required Packages

Commands -  
 sudo apt update -y  
 sudo apt install gcc -y  
 sudo apt install -y nagios-nrpe-server nagios-plugins

```

ubuntu@ip-172-31-46-201:~$ sudo apt update -y
sudo apt install gcc -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
0% [3 InRelease 102 kB/126 kB 81%] [Connecting to security.ubuntu.com (185.125.190
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [382 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
57% [5 Packages store 0 B] [7 Translation-en 0 B/5982 kB 0%] [6 Packages 107 kB/38
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
75% [5 Packages store 0 B] [8 Components-amd64 140 kB/3871 kB 4%] [6 Packages 180
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Me
tadata [301 kB]
86% [5 Packages store 0 B] [9 Commands-amd64 152 kB/301 kB 50%] [6 Packages 205 kB
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packa
ges [269 kB]
87% [5 Packages store 0 B] [10 Packages 161 kB/269 kB 60%] [6 Packages 208 kB/382

```

## 12. Edit NRPE Configuration File

## Commands -

```
sudo nano /etc/nagios/nrpe.cfg
```

- Add your Nagios host IP address under allowed\_hosts:  
allowed\_hosts=<Nagios\_Host\_IP>

```
ubuntu@ip-172-31-46-201:~$ sudo apt install -y nagios-nrpe-server nagios-plugins

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'monitoring-plugins' instead of 'nagios-plugins'
The following additional packages will be installed:
libavahi-client3 libavahi-common-data libavahi-common3 libcups2t64 libdbiit64 libldb2 libmysqclient21 libnet-snmp-perl libpq5 libraddr14 libsmnclient0 libsnmp-base
libsmnp40t64 libtalloc2c libtbt1 libtevent0t64 liburiparser1 libwbcclient0 monitoring-plugins-basic monitoring-plugins-common monitoring-plugins-standard mysql-common
python3-gpg python3-ldb python3-markdown python3-samba python3-talloc python3-tdb rpcbind samba-common-bin samba-dsdb-modules samba-libs smbcclient snmp
Suggested packages:
 cups libcurl4-openssl-dev libdigest-hmac-perl libio-socket-inet6-perl snmp-mibs-downloader icinga2 nagios-plugins-contrib fping postfix | sendmail-bin
 exim4-daemon-heavy | exim4-daemon-light gstat xinetd | inetd python-markdown-doc helmdai-clients python3-dnsproxy cifs-utils
The following NEW packages will be installed:
libavahi-client3 libavahi-common-data libavahi-common3 libcups2t64 libdbiit64 libldb2 libmysqclient21 libnet-snmp-perl libpq5 libraddr14 libsmnclient0 libsnmp-base
libsmnp40t64 libtalloc2c libtbt1 libtevent0t64 liburiparser1 libwbcclient0 monitoring-plugins-basic monitoring-plugins-common monitoring-plugins-standard mysql-common nagios-nrpe-server python3-gpg python3-ldb python3-markdown python3-samba python3-talloc python3-tdb rpcbind samba-common-bin samba-dsdb-modules samba-libs smbcclient snmp
0 upgraded, 37 newly installed, 0 to remove and 6 not upgraded.
Need to get 16.1 MB of archives.
After this operation, 72.0 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 nagios-nrpe-server amd64 4.1.0-1ubuntu3 [356 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 rpchand amd64 1.2.6-7ubuntu2 [46.5 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libavahih-1client3 amd64 0.8-13ubuntu6 [29.7 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libavahih-common3 amd64 0.8-13ubuntu6 [23.3 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libavahih-client3 amd64 0.8-13ubuntu6 [26.8 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/upgrades/main amd64 libtalloc2t64 amd64 2.4.7-1.2.ubuntu7.3 [272 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libtbt1 libtbt1 amd64 0.9.6-1ubuntu1 [25.7 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libtalloc2c libtalloc2c amd64 2.4.2-1ubuntu2 [27.3 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libtbt1 libtbt1 amd64 1.4.10-1ubuntu1 [46.8 kB]
```

### 13. Restart NRPE Server Commands –

```
sudo systemctl restart nagios-nrpe-server
```

## 14. Check Nagios Dashboard

- Open your browser and navigate to `http://<Nagios_Host_IP>/nagios`.
  - Log in with `nagiosadmin` and the password you set earlier.
  - You should see the new host `linuxserver` added.
  - Click on `Hosts` to see the host details.
  - Click on `Services` to see all services and ports being monitored

Not secure 54.91.123.99/nagios/ ☆ ↴ S

# Nagios®

**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  
Process Info

**Current Network Status**  
Last Update: Fri Oct 19 16:33:41 UTC 2024  
Updated every 90 seconds  
Nagios® Core™ 4.4.6 - www.nagios.org  
Logged in as nagiosadmin

**Host Status Totals**  

|    |      |             |         |
|----|------|-------------|---------|
| Up | Down | Unreachable | Pending |
| 2  | 0    | 0           | 0       |

  
**All Problems All Types**  

|   |   |
|---|---|
| 0 | 2 |
|---|---|

**Service Status Totals**  

|    |         |         |          |         |
|----|---------|---------|----------|---------|
| Ok | Warning | Unknown | Critical | Pending |
| 6  | 1       | 0       | 1        | 0       |

  
**All Problems All Types**  

|   |   |
|---|---|
| 2 | 8 |
|---|---|

**Host Status Details For All Host Groups**

Limit Results:  Host Status \*♦ Last Check \*♦ Duration \*♦ Status Information

| Host *♦     | Status *♦ | Last Check *♦       | Duration *♦  | Status Information                        |
|-------------|-----------|---------------------|--------------|-------------------------------------------|
| linusserver | UP        | 10-08-2024 16:29:09 | 0d 0h 9m 32s | PING OK - Packet loss = 0%, RTA = 1.61 ms |
| localhost   | UP        | 10-08-2024 16:28:54 | 0d 2h 19m 9s | PING OK - Packet loss = 0%, RTA = 0.03 ms |

Results 1 - 2 of 2 Matching Hosts

**Host Information**

- Last Updated: Tue Oct 8 16:36:26 UTC 2024
- Updated every 90 seconds
- Nagios® Core™ 4.4.6 - www.nagios.org
- Logged in as nagiosadmin

**Host**  
linuxserver  
(linuxserver)

**Member of**  
linux-servers1

54.88.143.206

**Host State Information**

|                                     |                                                                |
|-------------------------------------|----------------------------------------------------------------|
| <b>Host Status:</b>                 | <b>UP</b> (for 0d 0h 12m 17s)                                  |
| <b>Status Information:</b>          | PING OK - Packet loss = 0%, RTA = 1.02 ms                      |
| <b>Performance Data:</b>            | rta=1.016000ms;3000.000000;5000.000000;0.000000 pl=0%;80;100;0 |
| <b>Current Attempt:</b>             | 1/10 (HARD state)                                              |
| <b>Last Check Time:</b>             | 10-08-2024 16:34:09                                            |
| <b>Check Type:</b>                  | ACTIVE                                                         |
| <b>Check Latency / Duration:</b>    | 0.000 / 4.11 seconds                                           |
| <b>Next Scheduled Active Check:</b> | 10-08-2024 16:39:09                                            |
| <b>Last State Change:</b>           | 10-08-2024 16:24:09                                            |
| <b>Last Notification:</b>           | N/A (notification 0)                                           |
| <b>Is This Host Flapping?</b>       | <b>NO</b> (0.0% state change)                                  |
| <b>In Scheduled Downtime?</b>       | <b>NO</b>                                                      |
| <b>Last Update:</b>                 | 10-08-2024 16:36:18 ( 0d 0h 0m 8s ago)                         |

|                        |                |
|------------------------|----------------|
| <b>Active Checks:</b>  | <b>ENABLED</b> |
| <b>Passive Checks:</b> | <b>ENABLED</b> |
| <b>Obsessing:</b>      | <b>ENABLED</b> |
| <b>Notifications:</b>  | <b>ENABLED</b> |
| <b>Event Handler:</b>  | <b>ENABLED</b> |
| <b>Flap Detection:</b> | <b>ENABLED</b> |

**Host Commands**

- Locate host on map
- Disable active checks of this host
- Re-schedule the next check of this host
- Submit passive check result for this host
- Stop accepting passive checks for this host
- Stop obsessing over this host
- Disable notifications for this host
- Send custom host notification
- Schedule downtime for this host
- Schedule downtime for all services on this host
- Disable notifications for all services on this host
- Enable notifications for all services on this host
- Schedule a check of all services on this host
- Disable checks of all services on this host
- Enable checks of all services on this host
- Disable event handler for this host
- Disable flap detection for this host
- Clear flapping state for this host

**Host Comments**

Add a new comment | Delete all comments

Entry Time Author Comment Comment ID Persistent Type Expires Actions

This host has no comments associated with it.

## Conclusion:

To perform port, service, and Windows/Linux server monitoring using Nagios, configure the necessary plugins and agents, define the monitoring parameters in the configuration files, and set up alerting mechanisms to ensure timely notifications of any issues. This comprehensive approach ensures robust monitoring and quick response to potential problems, maintaining the health and performance of your IT infrastructure.

# EXPERIMENT NO. 11

**Aim:** To understand AWS Lambda, its workflow, various functions and create your first Lambda functions using Python / Java / Nodejs.

## Implementation:

### Steps to create an AWS Lambda function

1. Open up the Lambda Console and click on the Create button. Be mindful of where you create your functions since Lambda is region-dependent.

The screenshot shows the AWS Lambda console homepage. The main heading is 'AWS Lambda' with the tagline 'lets you run code without thinking about servers.' Below this, there's a section titled 'How it works' containing a code snippet:

```

1 * exports.handler = async (event) => {
2 console.log('Event:', event);
3 return 'Hello from Lambda!';
4 };
5

```

At the bottom of the page, there's a 'Create a function' button.

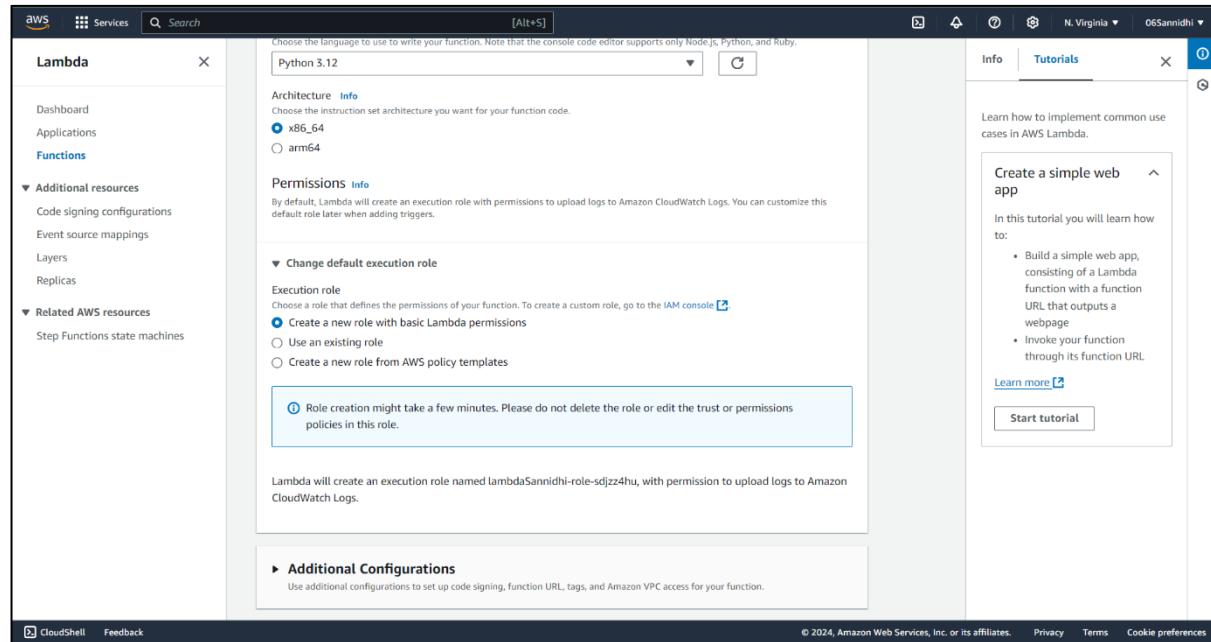
The screenshot shows the 'Create function' wizard in the AWS Lambda console. The left sidebar has a 'Functions' link under 'Additional resources'. The main form is titled 'Create function' and includes the following steps:

- Basic information**: Function name is set to 'LambdaSannidhi'.
- Runtime**: Set to 'Python 3.12'.
- Architecture**: Set to 'x86\_64'.
- Permissions**: A note states that Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs.

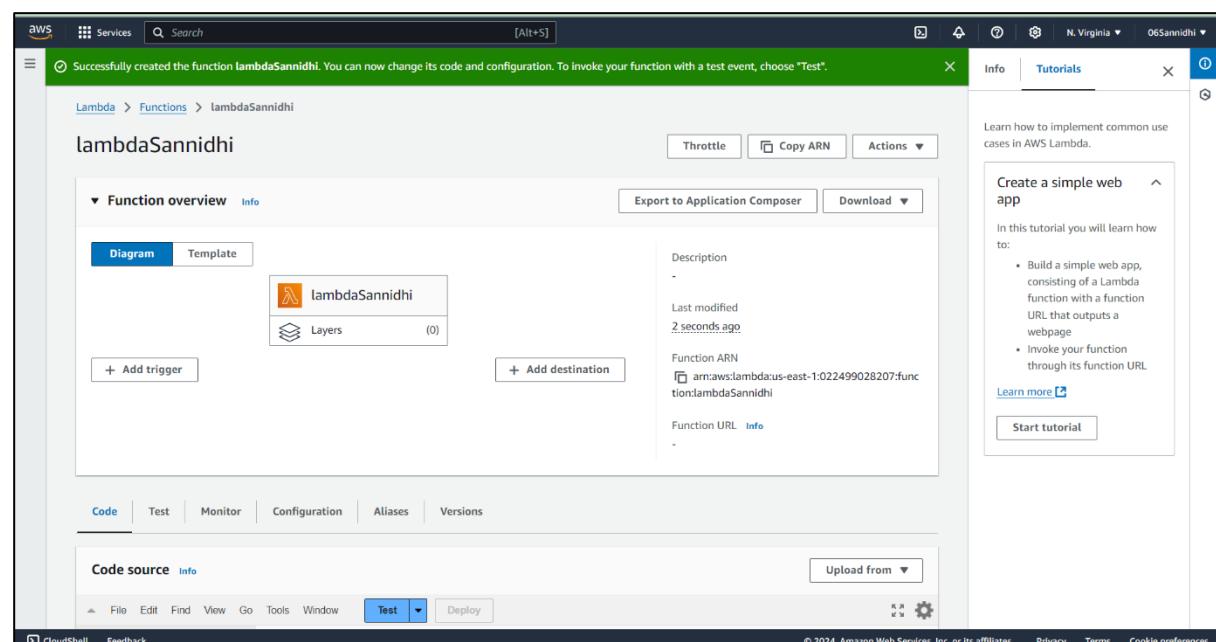
The right sidebar has a 'Tutorials' tab open, showing a preview of the 'Create a simple web app' tutorial.

2. Choose to create a function from scratch or use a blueprint, i.e templates defined by AWS for you with all configuration presets required for the most common use cases.

Then, choose a runtime env for your function, under the dropdown, you can see all the options AWS supports, Python, Nodejs, .NET and Java being the most popular ones. After that, choose to create a new role with basic Lambda permissions if you don't have an existing one.



3. This process will take a while to finish and after that, you'll get a message that your function was successfully created.



#### 4. Edit Basic Settings:

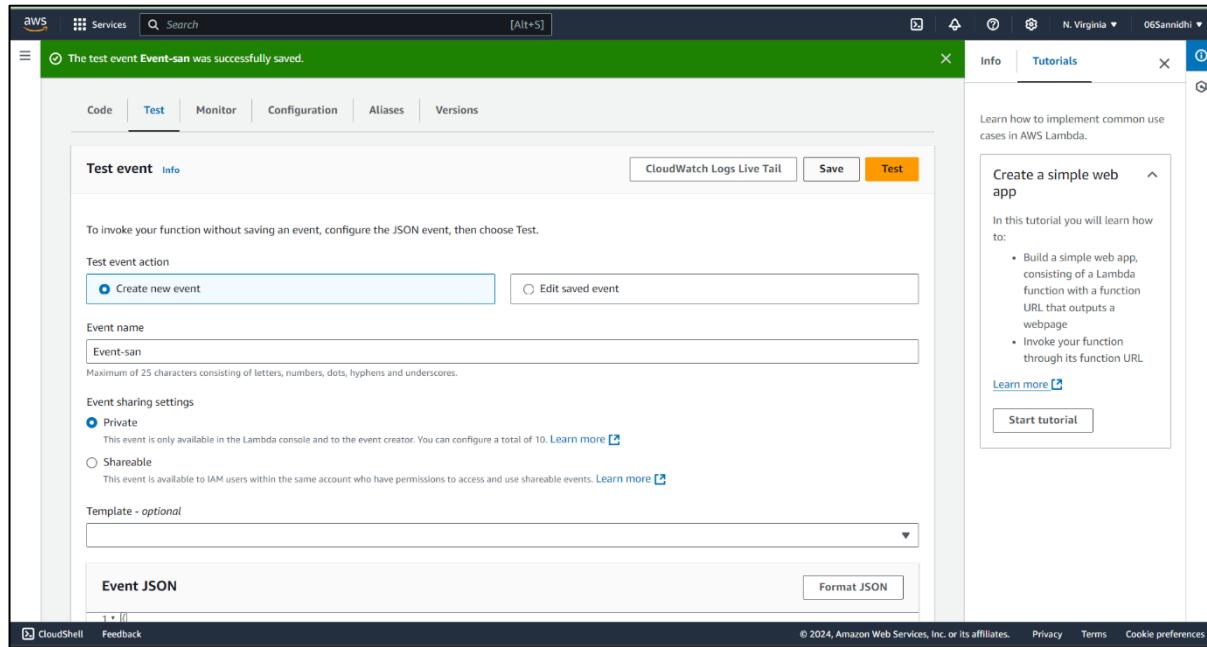
On the function's Configuration tab, locate the Basic settings section

The screenshot shows the AWS Lambda console. A green success message at the top says "Successfully created the function lambdaSannidhi. You can now change its code and configuration. To invoke your function with a test event, choose 'Test'." The left sidebar has a "General configuration" section with various tabs like Triggers, Permissions, Destinations, etc. The main area shows "General configuration" with fields: Description (empty), Memory (128 MB), Ephemeral storage (512 MB), and Timeout (0 min 3 sec). On the right, there's a "Tutorials" sidebar with a "Create a simple web app" section.

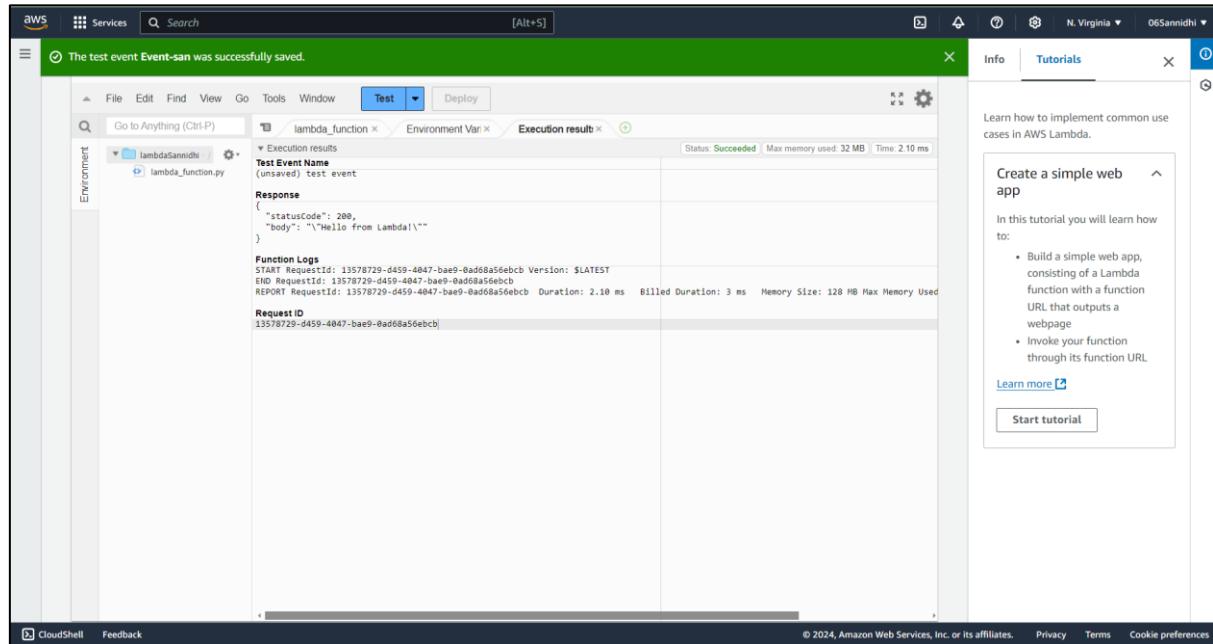
#### 5. Enter a description and change Memory and Timeout. I've changed the Timeout period to 1 sec since that is sufficient for now.

The screenshot shows the "General configuration" tab in the AWS Lambda function configuration. It includes fields for Memory (128 MB), Ephemeral storage (512 MB), SnapStart (None), and Timeout (0 min 1 sec). The "Execution role" section is also visible, showing "Use an existing role" selected with "service-role/lambdaSannidhi-role-sdjz4hu". A "Save" button is at the bottom right.

## 6. Configuring test event which triggers when the function is tested



## 7. Now click on Test and you should be able to see the results.



## Conclusion:

AWS Lambda is a serverless computing service that allows you to run code without managing servers, making it highly scalable, cost-effective, and easy to use. It automatically manages the compute resources, executes your code in response to specific events such as API calls, file uploads, or database updates, and scales based on the demand.

## EXPERIMENT NO. 12

**Aim:** To create a Lambda function which will log “An Image has been added” once you add an object to a specific bucket in S3.

### Implementation:

1. Create an S3 Bucket: First, create an S3 bucket that will store the objects. This bucket will act as the trigger source for the Lambda function.

**Create bucket Info**  
Buckets are containers for data stored in S3.

**General configuration**

AWS Region: US East (N. Virginia) us-east-1  
Bucket type: **General purpose**

Bucket name: **lambdaBucket1**  
Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming [?]

Copy settings from existing bucket - *optional*: Choose bucket

Format: s3://bucket/prefix

**Object Ownership Info**  
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.

**Successfully created bucket "lambdaBucket1"**  
To upload files and folders, or to configure additional bucket settings, choose **View details**.

**Amazon S3 > Buckets**

**Account snapshot - updated every 24 hours** All AWS Regions  
Storage lens provides visibility into storage usage and activity trends. Learn more [?]

**General purpose buckets (5) Info All AWS Regions**  
Buckets are containers for data stored in S3.

| Name                                                 | AWS Region                      | IAM Access Analyzer                          | Creation date                            |
|------------------------------------------------------|---------------------------------|----------------------------------------------|------------------------------------------|
| <a href="#">codepipeline-eu-north-1-954428616777</a> | Europe (Stockholm) eu-north-1   | <a href="#">View analyzer for eu-north-1</a> | August 12, 2024, 20:03:34 (UTC+05:30)    |
| <a href="#">codepipeline-us-east-1-262039424207</a>  | US East (N. Virginia) us-east-1 | <a href="#">View analyzer for us-east-1</a>  | September 17, 2024, 23:20:15 (UTC+05:30) |
| <a href="#">lambdaBucket1</a>                        | US East (N. Virginia) us-east-1 | <a href="#">View analyzer for us-east-1</a>  | October 19, 2024, 19:42:17 (UTC+05:30)   |
| <a href="#">my-pinterest-website</a>                 | Europe (Stockholm) eu-north-1   | <a href="#">View analyzer for eu-north-1</a> | August 6, 2024, 20:58:28 (UTC+05:30)     |
| <a href="#">www.skwebsite.com</a>                    | Europe (Stockholm) eu-north-1   | <a href="#">View analyzer for eu-north-1</a> | August 6, 2024, 20:09:00 (UTC+05:30)     |

2. Create the Lambda Function: Set up a new Lambda function using AWS Lambda's console. You can choose a runtime environment like Python, Node.js, or Java. Write code that logs a message like "An Image has been added" when triggered.

The screenshot shows the 'Create function' wizard in the AWS Lambda console. The 'Basic information' step is selected. The 'Function name' field contains 'lambdaimageloaders'. The 'Runtime' dropdown is set to 'Python 3.12'. The 'Architecture' dropdown is set to 'x86\_64'. The 'Permissions' section indicates that a default execution role will be created. On the right, a sidebar titled 'Create a simple web app' provides a tutorial summary and a 'Start tutorial' button.

The screenshot shows the 'Create function' wizard in the AWS Lambda console, with the 'Additional Configurations' step visible. It includes sections for 'Execution role', 'Existing role', and 'Additional Configurations'. The 'Execution role' dropdown shows 'service-role/lambdaSannidhi-role-sdjzz4hu'. The 'Create function' button is highlighted in orange at the bottom. A URL for the Lambda console is visible in the browser address bar.

```
import json
import logging

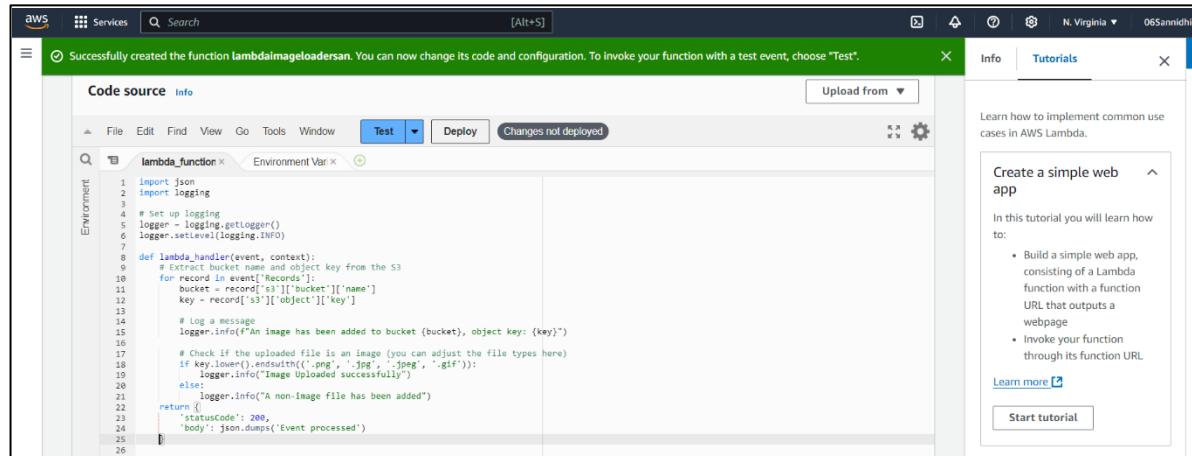
Set up logging
logger = logging.getLogger()
logger.setLevel(logging.INFO)

def lambda_handler(event, context):
 # Extract bucket name and object key
 from the S3 event for record in
 event['Records']:
 bucket =
 record['s3']['bucket']['name']
 key =
 record['s3']['object']['key']

 # Log a message
 logger.info(f"An image has been added to bucket {bucket}, object key: {key}")

 # Check if the uploaded file is an image (you can adjust the
 file types here) if key.lower().endswith('.png', '.jpg', '.jpeg',
 '.gif'):
 logger.info("Image
Uploaded successfully")
 else:
 logger.info("A non-image file has been added")

 return {
 'statusCode': 200,
 'body': json.dumps('Event processed')
 }
```



3. Configure S3 Trigger: Link the S3 bucket to the Lambda function by setting up a trigger. Specify that the function should be triggered when an object is created in the bucket (e.g., when an image is uploaded).

The screenshot shows the 'Add trigger' page for an AWS Lambda function. The 'Trigger configuration' section is active, showing the selection of an S3 event source. A dropdown menu is open, with 'S3' selected. Below it, a search bar contains the ARN 'arn:s3:lambdabucketsan'. The 'Event types' section shows 'All object create events' selected. The 'Prefix - optional' field contains 'e.g. images/' and the 'Suffix - optional' field contains 'e.g. jpg'. On the right side, there is a sidebar titled 'Create a simple web app' with a brief description and a 'Start tutorial' button.

The screenshot shows the 'Configuration' tab for an AWS Lambda function. In the left sidebar, under 'Triggers', the 'S3: lambdabucketsan' trigger is listed. The main panel displays the function's ARN and URL. The sidebar on the right contains a 'Create a simple web app' tutorial.

4. Upload an object (e.g., an image) to the S3 bucket to test the trigger

The screenshot shows the AWS S3 'Upload' interface. At the top, the navigation bar includes 'Amazon S3 > Buckets > lambdabucketsan > Upload'. Below this, the main area is titled 'Upload' with an 'Info' link. A note states: 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)'.

In the center, there's a large input field with the placeholder 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.' Below it, a table lists 'Files and folders (1 Total, 175.6 KB)'. The table has columns for 'Name', 'Folder', and 'Type'. One item is listed: 'hero-section-img.jpeg' (image/jpeg). Buttons for 'Remove', 'Add files', and 'Add folder' are at the top of the table.

On the right, a 'Destination' section shows the destination as 's3://lambdabucketsan'. It includes a 'Destination details' link and a 'Permissions' link. At the bottom of the interface, there are links for 'CloudShell', 'Feedback', and copyright information: '© 2024, Amazon Web Services, Inc. or its affiliates.' followed by 'Privacy', 'Terms', and 'Cookie preferences'.

The screenshot shows the AWS S3 'Upload: status' interface. At the top, a green banner displays 'Upload succeeded' with a link to 'View details below.' A 'Close' button is in the top right corner.

The main area is titled 'Upload: status' with a 'Close' button. A note says: 'The information below will no longer be available after you navigate away from this page.' Below this is a 'Summary' section with a table:

| Destination          | Succeeded                  | Failed            |
|----------------------|----------------------------|-------------------|
| s3://lambdabucketsan | 1 file, 175.6 KB (100.00%) | 0 files, 0 B (0%) |

Below the summary is a tabbed section with 'Files and folders' (selected) and 'Configuration'. The 'Files and folders' section shows a table of uploaded files:

| Files and folders (1 Total, 175.6 KB) |        |            |          |                                              |       |  |
|---------------------------------------|--------|------------|----------|----------------------------------------------|-------|--|
| <a href="#">Find by name</a>          |        |            |          |                                              |       |  |
| Name                                  | Folder | Type       | Size     | Status                                       | Error |  |
| hero-section...                       | -      | image/jpeg | 175.6 KB | <span style="color: green;">Succeeded</span> | -     |  |

At the bottom, there are links for 'CloudShell', 'Feedback', and copyright information: '© 2024, Amazon Web Services, Inc. or its affiliates.' followed by 'Privacy', 'Terms', and 'Cookie preferences'.

5. Test the Setup: Upload an object (e.g., an image) to the S3 bucket to test the trigger. The Lambda function should execute and log the message “An Image has been added” in AWS CloudWatch Logs

The screenshot shows the AWS CloudWatch Log Groups interface. On the left, there's a navigation sidebar with options like Favorites and recents, Dashboards, Alarms, Logs (selected), Metrics, X-Ray traces, Events, Application Signals, Network monitoring, and Insights. Under Logs, 'Log groups' is selected. The main area shows the log group details for '/aws/lambda/lambdaimageloadersan'. It includes fields for Log class (Info, Standard), ARN (arn:aws:logs:us-east-1:022499028207:log-group:/aws/lambda/lambdaimageloadersan:"), Creation time (Now), Retention (1 day), Stored bytes (~), Metric filters (0), Subscription filters (0), Contributor Insights rules (~), KMS key ID (~), Anomaly detection (Configure), Data protection (~), and Sensitive data count (~). Below this, there are tabs for Log streams, Tags, Anomaly detection, Metric filters, Subscription filters, Contributor Insights, and Data protection. The Log streams tab is selected, showing 0 log streams. At the bottom, there are buttons for Create log stream and Search all log streams, along with filter and search options.

The screenshot shows the AWS CloudWatch Log Events interface for the '/aws/lambda/lambdaimageloadersan' log group. It features a header with 'Actions', 'Start tailing', and 'Create metric filter' buttons. Below is a filter bar with a search input ('Filter events - press enter to search'), time range buttons (Clear, 1m, 30m, 1h, 12h, Custom, UTC timezone), and a 'Display' dropdown. The main area displays log events in a table with columns for 'Timestamp' and 'Message'. The first event is 'No older events at this moment. [Retry](#)'. Subsequent events show the Lambda function starting, logging an image upload, and completing the process. The last event is 'No newer events at this moment. Auto retry paused. [Resume](#)'.

## Conclusion:

Integrating AWS Lambda with S3 allows for real-time, automated processing of events such as file uploads. In this example, a Lambda function is configured to log a message whenever an image is added to a specific S3 bucket. This setup demonstrates the power and flexibility of serverless computing by automating tasks without requiring manual intervention or server management. By leveraging AWS Lambda, developers can efficiently handle event-driven workflows, reduce operational overhead, and quickly deploy scalable solutions that respond to specific actions within cloud environment.