

# EXPERIMENT NO. 1

NAME: MOHIT PATIL

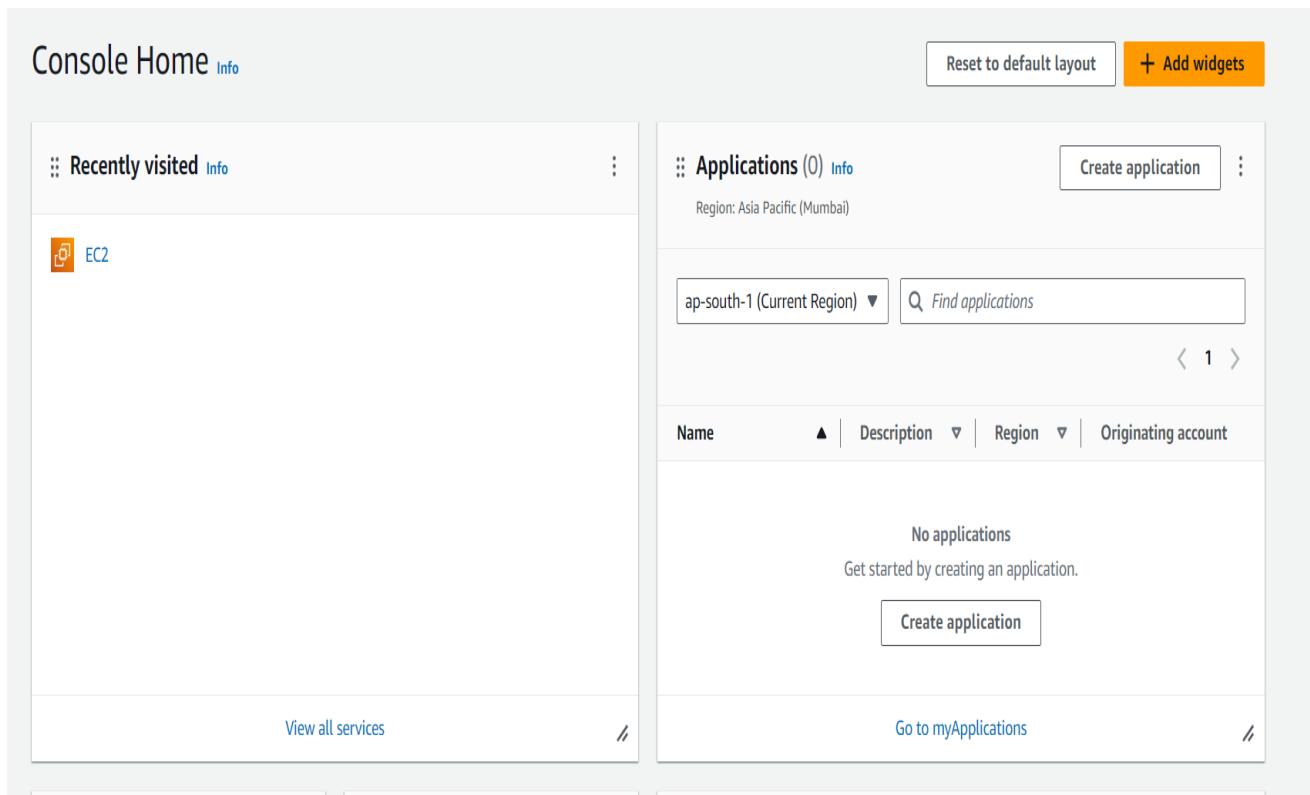
CLASS:D15A

ROLLNO.37

**Aim :** To understand the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration.

## EC2 Instance Creation and static site hosting

1. Login to your AWS account



## 2. Click on EC2 and then create an instance by clicking on instances

The screenshot shows the AWS EC2 Home page. At the top, there's a summary of resources used in the Asia Pacific (Mumbai) Region:

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

Below this, there are sections for "EC2 Free Tier" offers and "Launch instance".

**EC2 Free Tier** (Info): Offers for all AWS Regions. 0 EC2 free tier offers in use.

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

**Service health**: AWS Health Dashboard. Region: Asia Pacific (Mumbai). Status: This service is operating normally.

**Account attributes**: Default VPC: `vpc-0b8f24d7c64f9775f`. Settings: [Edit, View, Delete, Create new]

### 3. After an instance is created wait for it to come to Running state

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations (New), Images (selected), AMIs, and AMI Catalog. The main content area has a title 'Instances (1/1) Info' and a search bar. It displays one instance: 'aws 1' (Instance ID: i-0e6218a10f73b4de7), which is 'Running' (Status check: Initializing). The instance is of type t2.micro, located in the ap-south-1b availability zone, with a public IPv4 address ec2-52-66-25. There are buttons for 'Connect', 'Actions', and 'Launch instances'.

### 4. After doing that you will see this UI

The screenshot shows a terminal window with the AWS logo at the top. The terminal output is as follows:

```
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

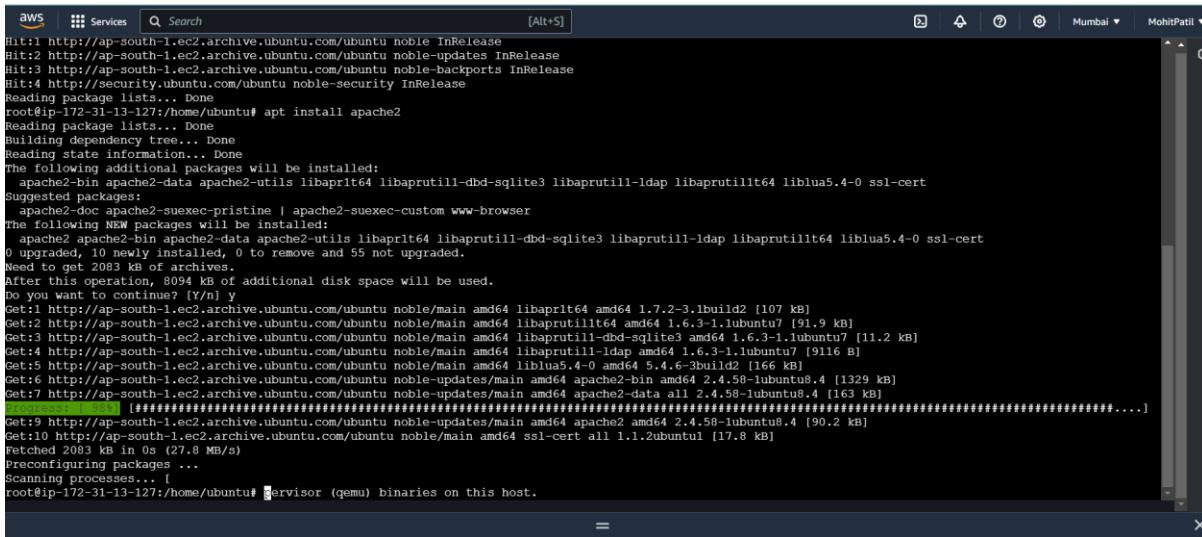
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

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-172-31-13-127:~$ sudo su
root@ip-172-31-13-127:/home/ubuntu# apt update
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [296 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:11 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:12 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
```

## 5. Follow these steps and then run these commands



```
aws Services Search [Alt+S] Mumbai MohitPatil
Bit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Bit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Bit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Bit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
root@ip-172-31-13-127:/home/ubuntu# apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 55 not upgraded.
Need to get 2083 kB of archives.
After this operation, 8094 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert [107 kB]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1l64 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert [91.9 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert [11.2 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-ldap libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert [9116 B]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblua5.4-0 liblua5.4-0 liblua5.4-0 liblua5.4-0 [166 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-bin libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert [1329 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-data all 2.4.58-lubuntu8.4 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert [163 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert [163 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 amd64 apache2 2.4.58-lubuntu8.4 [90.2 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 ssl-cert all 1.1.2ubuntu1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1l64 liblua5.4-0 ssl-cert [17.8 kB]
Fetched 2083 kB in 0s (27.8 MB/s)
Preconfiguring packages ...
Scanning processes... [root@ip-172-31-13-127:/home/ubuntu# ]ervisor (qemu) binaries on this host.
```

## 6. After that the ip-address which was given while running the instance, copy that and paste that on chrome, make sure that it is http and not https



## 7. Create a file using vi command and save it using : wq

```
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-hosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /usr/lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /usr/lib/systemd/system/apache-htcacheclean.service.
Processing triggers for ufw (0.36.2-6) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu0.2) ...
Scanning processes...
Scanning linux 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 guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-45-95:~# cd /var/www/html
root@ip-172-31-45-95:/var/www/html# ls
index.html
root@ip-172-31-45-95:/var/www/html# vi test.html
root@ip-172-31-45-95:/var/www/html# ls
index.html test.html
root@ip-172-31-45-95:/var/www/html# i-0e1ecb14b4c39a638 (mohit)
PublicIPs: 13.201.55.80 PrivateIPs: 172.31.45.95
```

8. After saving that file go that page where ubuntu is listed and then on the link add “/your\_file\_name.html” and then whatever you saved on that file will be displayed

Customized T-Shirts for All :)

Tagline on the Shirt:

Color:

Size:

Quantity:

Delivery Date:

Delivery Details:

Recipient's Name:

Address:

Email:

Phone Number:

Additional Comments:

Comments:

# Static Hosting using S3 bucket

## Step1: Create bucket

The screenshot shows the 'Create bucket' configuration page in the AWS S3 console. The 'General configuration' section is visible, showing the AWS Region set to 'Asia Pacific (Mumbai) ap-south-1'. The 'Bucket name' field contains 'mohitbucket'. A note below the field states: 'Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)'.

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

**Index document**  
Specify the home or default page of the website.  
`index.html`

## Step 2: Add resources

The screenshot shows the AWS CloudFront console. At the top, a green notification bar says "Upload succeeded" with a link to "View details below". Below this, there's a summary table with columns for Destination, Status, and Failed. Under "Destination", it shows "s3://mohitpatilbucket" with "Succeeded" status and "39 files, 176.4 KB (100.00%)". Under "Failed", it shows "0 files, 0 B (0%)". Below the table, there are two tabs: "Files and folders" (selected) and "Configuration". The main area displays a table titled "Files and folders (39 Total, 176.4 KB)". The table has columns for Name, Folder, Type, Size, Status, and Error. All files listed have a status of "Succeeded". Some file names are partially visible, such as "bloggingblo...", "download.jp...", "images.jpeg", "index.html", "style.css", "yin-yang-yin...", "COMMIT\_EDIT...", "config", "description", and "HEAD".

The screenshot shows the AWS S3 Bucket properties page for "mohitpatilbucket". At the top, a green notification bar says "Successfully edited static website hosting." Below this, the navigation path is "Amazon S3 > Buckets > mohitpatilbucket". The bucket name "mohitpatilbucket" is shown with a "info" link. The "Properties" tab is selected. The "Bucket overview" section shows basic information: AWS Region (Asia Pacific (Mumbai) ap-south-1), Amazon Resource Name (ARN) (arn:aws:s3:::mohitpatilbucket), and Creation date (August 21, 2024, 20:58:25 (UTC+05:30)). The "Bucket Versioning" section is expanded, showing that versioning is "Disabled". It includes a note about Multi-factor authentication (MFA) delete and a link to learn more. The "Tags (0)" section is at the bottom right.

The screenshot shows the AWS S3 console interface. At the top, there's a green header bar with the message "Successfully edited public access" and "View details below." Below this, the main title is "Make public: status". A blue info box states: "The information below will no longer be available after you navigate away from this page." Under the "Summary" section, there are two rows: "Source" (s3://mohitpatilbucket/p/a3 bootstrap/) and "Successfully edited public access" (39 objects, 176.4 KB). To the right, there's a row for "Failed to edit public access" (0 objects). Below the summary, there are tabs for "Failed to edit public access" and "Configuration". Under "Failed to edit public access", it says "0 Failed to edit public access (0)". A search bar and a table header ("Name", "Folder", "Type", "Last modified", "Size", "Error") are shown, followed by the message "No objects failed to edit".

### Step 3 : visit hosted website

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "Customized T-Shirts" and has the URL "13.201.55.80/test.html". The page content is a form for ordering customized t-shirts. It includes fields for "Tagline on the Shirt" (with placeholder "Enter your tagline"), "Color" (dropdown menu "Select a color"), "Size" (dropdown menu "Select a size"), "Quantity" (text input "Enter quantity"), "Delivery Date" (text input "dd-mm-yyyy"), and "Delivery Details" (fields for "Recipient's Name", "Address", "Email", and "Phone Number"). Below these, there are sections for "Additional Comments:-" (text area "Any additional comments or special instructions") and "Comments:" (text area "Comments:"). At the bottom, there is a "Submit Order" button.

# EC2 Dynamic Site Hosting

Step 1 : Open Console and clone the github repository

The screenshot shows a terminal session in an AWS CloudShell window. The terminal output is as follows:

```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-15-151:~$ pws
Command 'pws' not found, did you mean:
  command 'aws' from snap aws-cli (1.15.58)
  command 'pms' from deb pmx (0.42-1.1)
  command 'pts' from deb openfaas-client (1.8.10-2.1ubuntu3.1)
  command 'psw' from deb wise (2.4.1-23)
  command 'pps' from deb libpmix-bin (5.0.1-4)
  command 'aws' from deb awscli (2.14.6-1)
  command 'pcs' from deb pcs (0.11.6-1ubuntu1)
  command 'pwd' from deb coreutils (9.4-2ubuntu2)
  command 'pws' from deb ratpoison (1.4.9-1)
  command 'pws' from deb lwm2 (2.03.16-2ubuntu1)
  command 'own' from deb python3-pwntools (4.11.1-1)
  command 'ps' from deb procps (2:4.0.4-2ubuntu1)
See 'snap info <snapname>' for additional versions.
ubuntu@ip-172-31-15-151:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-15-151:~$ git clone https://github.com/Mohitpatil344/zapier_html.git
Cloning into 'zapier_html'...
remote: Enumerating objects: 18, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 18 (delta 6), reused 12 (delta 2), pack-reused 0 (from 0)
Receiving objects: 100% (18/18), 31.36 MiB | 17.13 MiB/s, done.
Resolving deltas: 100% (6/6), done.
ubuntu@ip-172-31-15-151:~$ ls
zapier_html
ubuntu@ip-172-31-15-151:~$ cd zapier_html
ubuntu@ip-172-31-15-151:~/zapier_html$ ls
1699438656630.jpeg          i2.webp
17221957322455bh4inh-wicemakar.in-speech.mp3 index.html      mobile-application-development-guidelines-riseuplabs.webp
'Screenshot 2024-07-29 005945.png' 'invideo-ai-1080 Zapier_ Your Ultimate Tech Partner! 2024-07-28.mp4'
ubuntu@ip-172-31-15-151:~/zapier_html$ 

i-0022d489c88af599c(mohit)
PublicIPs: 3.111.57.218 PrivateIPs: 172.31.15.151
```

26 updates can be applied immediately.
22 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See <https://ubuntu.com/esm> or run: sudo pro status

Last login: Wed Aug 21 17:55:18 2024 from 13.233.177.4
ubuntu@ip-172-31-15-151:~\$ git clone https://github.com/PranavPol-01/dyanamic\_site.git
fatal: destination path 'dyanamic\_site' already exists and is not an empty directory.
ubuntu@ip-172-31-15-151:~\$ cd dyanamic\_site
ubuntu@ip-172-31-15-151:~/dyanamic\_site\$ npm install

added 93 packages, and audited 94 packages in 2s

16 packages are looking for funding
 run `npm fund` for details

found 0 vulnerabilities
ubuntu@ip-172-31-15-151:~/dyanamic\_site\$ node index.js
Server is running on port 3000

i-0022d489c88af599c(mohit)
PublicIPs: 3.111.57.218 PrivateIPs: 172.31.15.151

## Step 2 : Install necessary Packages and run website on port 3000



# Cloud 9 IDE Site Hosting

The screenshot shows the AWS Cloud9 homepage. At the top right, there is a white call-to-action box with the text "New AWS Cloud9 environment" and a yellow "Create environment" button.

### Details

Name: Test123  
Limit of 60 characters, alphanumeric, and unique per user.

Description - optional  
Limit 200 characters.

Environment type: [Info](#)  
Determines what the Cloud9 IDE will run on.

New EC2 instance  
Cloud9 creates an EC2 instance in your account. The configuration of your EC2 instance cannot be changed by Cloud9 after creation.

Existing compute  
You have an existing instance or server that you'd like to use.

### New EC2 instance

Instance type: [Info](#)  
The memory and CPU of the EC2 instance that will be created for Cloud9 to run on.

t2.micro (1 GiB RAM + 1 vCPU)  
Free-tier eligible. Ideal for educational users and exploration.

t3.small (2 GiB RAM + 2 vCPU)  
Recommended for small web projects.

m5.large (8 GiB RAM + 2 vCPU)  
Recommended for production and most general-purpose development.

Additional instance types  
Explore additional instances to fit your need.

Platform: [Info](#)  
This will be installed on your EC2 instance. We recommend Amazon Linux 2023.

Amazon Linux 2023

Timeout  
How long Cloud9 can be inactive (no user input) before auto-hibernating. This helps prevent unnecessary charges.

30 minutes

① For capabilities similar to AWS Cloud9, explore AWS Toolkits in your own IDE and AWS CloudShell in the AWS Management Console. [Learn more](#)

AWS Cloud9 > Environments

Environments (1)

Delete View details Open in Cloud9 Create environment

My environments

| Name    | Cloud9 IDE           | Environment type | Connection         | Permission | Owner ARN  |
|---------|----------------------|------------------|--------------------|------------|--|
| Test123 | <a href="#">Open</a> | EC2 instance     | Secure Shell (SSH) | Owner      | arn:aws:sts::554378108602:assumed-role/voclabs/user3402848=PATANKAR_ARYAN_ANIL |

The screenshot shows the AWS Cloud9 IDE interface. At the top, there's a navigation bar with File, Edit, Find, View, Go, Run, Tools, Window, Support, Preview, and Run buttons. On the far right, there's a user icon and the word "Share". Below the navigation bar is a search bar with the placeholder "Go to Anything (Ctrl-P)". To the left of the main workspace is a sidebar titled "Test123 - homekit" containing a file tree with "c9" and "README.md". The main workspace has a dark background with white text. It features a large title "AWS Cloud9" and a subtitle "Welcome to your development environment". Below this, a paragraph explains the Cloud9 environment: "AWS Cloud9 allows you to write, run, and debug your code with just a browser. You can tour the IDE, write code for AWS Lambda and Amazon API Gateway, share your IDE with others in real time, and much more." To the right of this text is a "Getting started" section with three buttons: "Create File", "Upload Files...", and "Clone from GitHub". At the bottom of the screen, there's a terminal window titled "bash - [ip-172-31-11-129.x]" showing the command "voclabs:~/environment \$".



# EXPERIMENT 2

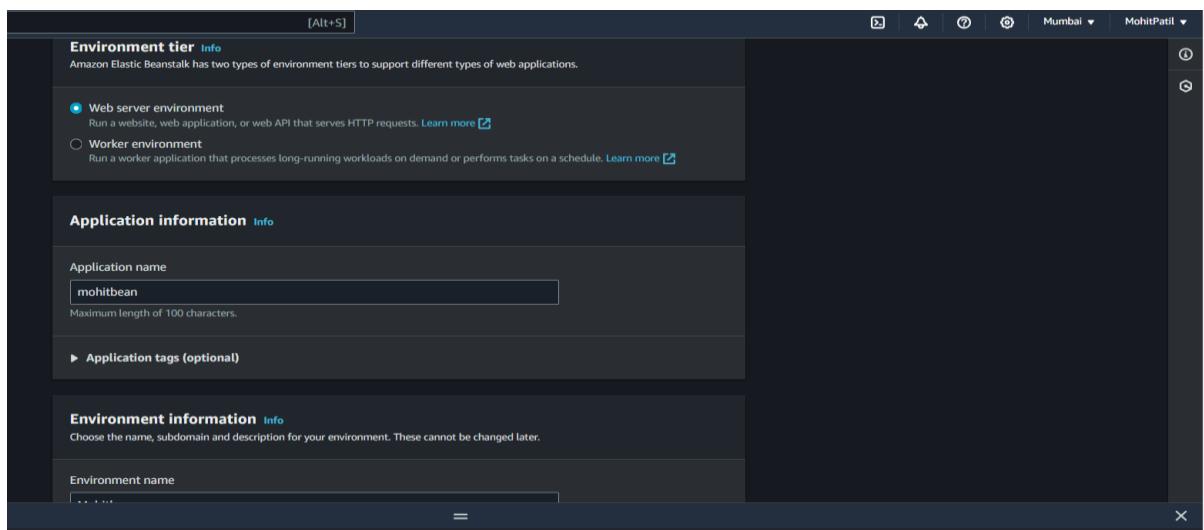
NAME:MOHIT PATIL CLASS:D15A

ROLL NO:37

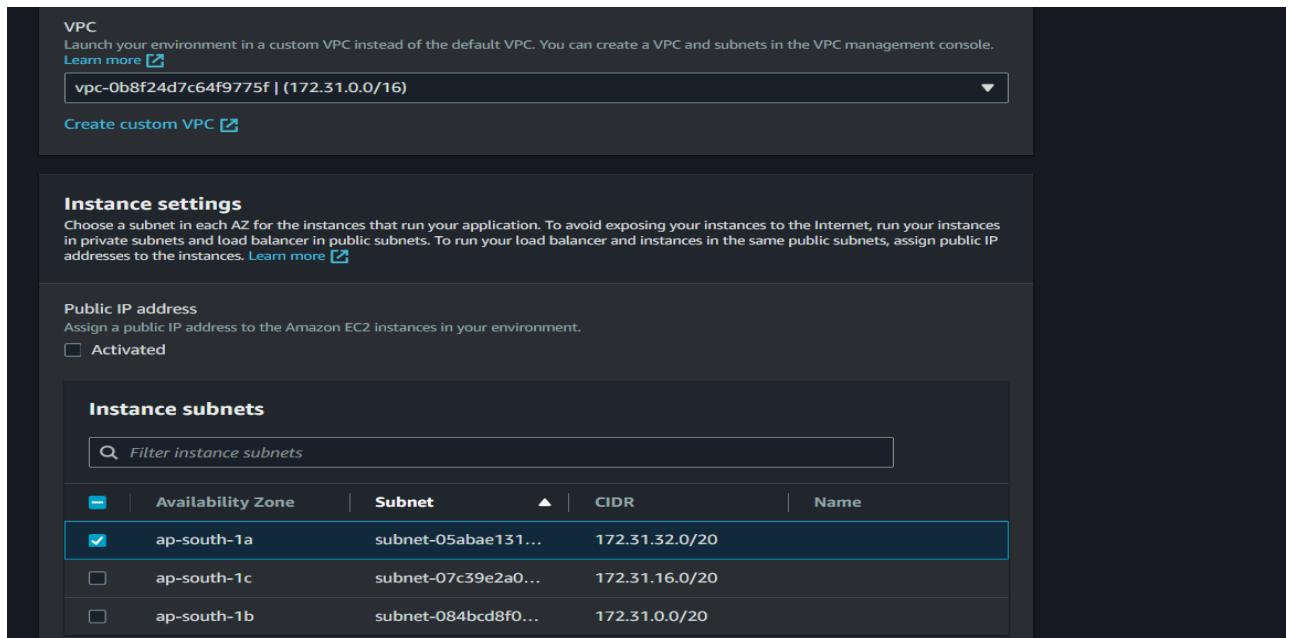
**Aim :**To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

## Elastic Beanstalk

### Step 1: create environment



### Step 2 : add your Ec2 key pair and instance profile



### Step 3 : add security config and review all settings

**Monitoring** Info

**Health reporting**

Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The EnvironmentHealth custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see [Amazon CloudWatch Pricing](#).

**System**

- Basic
- Enhanced

**CloudWatch Custom Metrics - Instance**

**CloudWatch Custom Metrics - Environment**

**Health event streaming to CloudWatch Logs**

Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

**Log streaming**

Activated (standard CloudWatch charges apply.)

**Retention**

7

Lifecycle

**Review** Info

**Step 1: Configure environment**

**Environment information**

|  |                    |
|--|--------------------|
| Environment tier                                     | Application name   |
| Web server environment                               | mohitbean          |
| Environment name                                     | Application code   |
| Mohitbean-env  | Sample application |
| Platform   |                    |
| arn:aws:elasticbeanstalk:ap-south-1:platform/PHP 8.3 |                    |
| running on 64bit Amazon Linux 2023/4.3.2             |                    |

**Step 2: Configure service access**

**Service access** Info

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

**Elastic Beanstalk**

**Environment successfully launched.**

**Mohitbean-env**

**Environment overview**

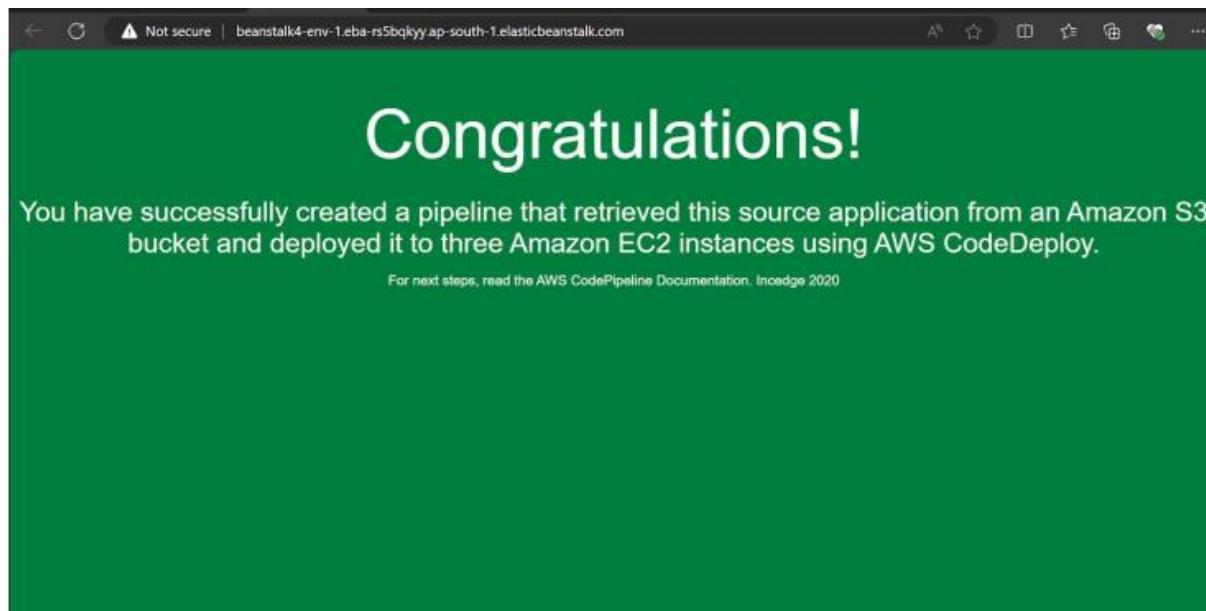
|  |                  |
|--|------------------|
| Health   | Environment ID   |
| Ok   | e-8jcp97pasv     |
| Domain   | Application name |
| Mohitbean-env.eba-qsim2f4q.ap-south-1.elasticbeanstalk.com | mohitbean        |

**Platform**

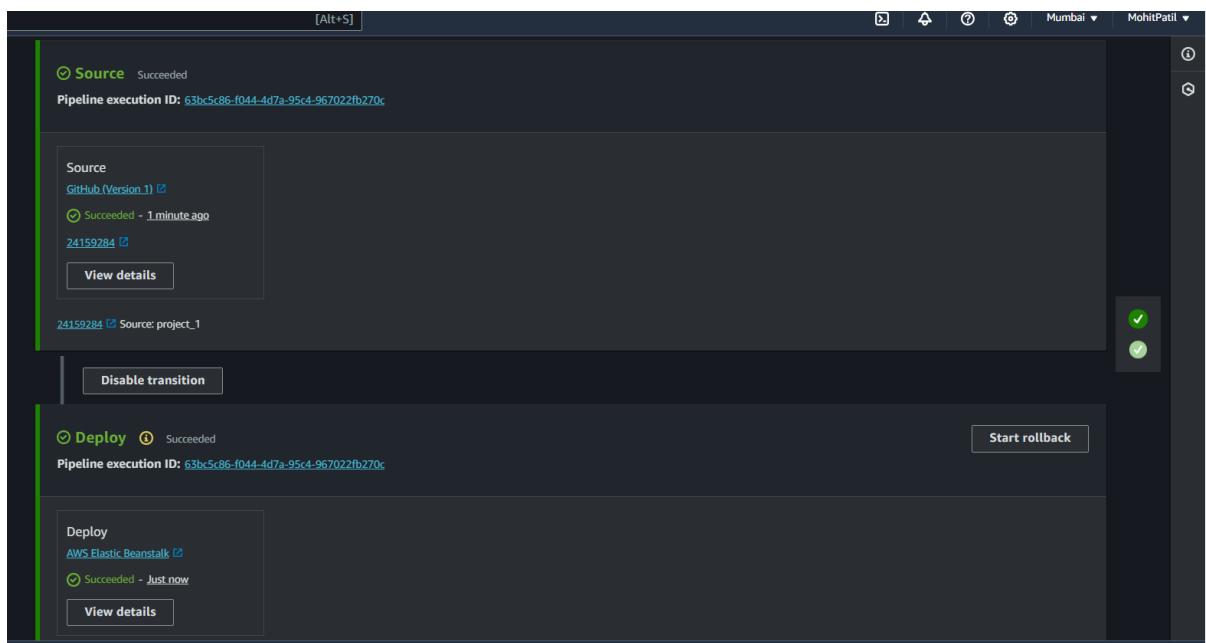
|  |                |
|--|----------------|
| Platform   | Change version |
| PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2 |                |
| Running version                                  | Platform state |
| -  | Supported      |

**Events** (12) Info

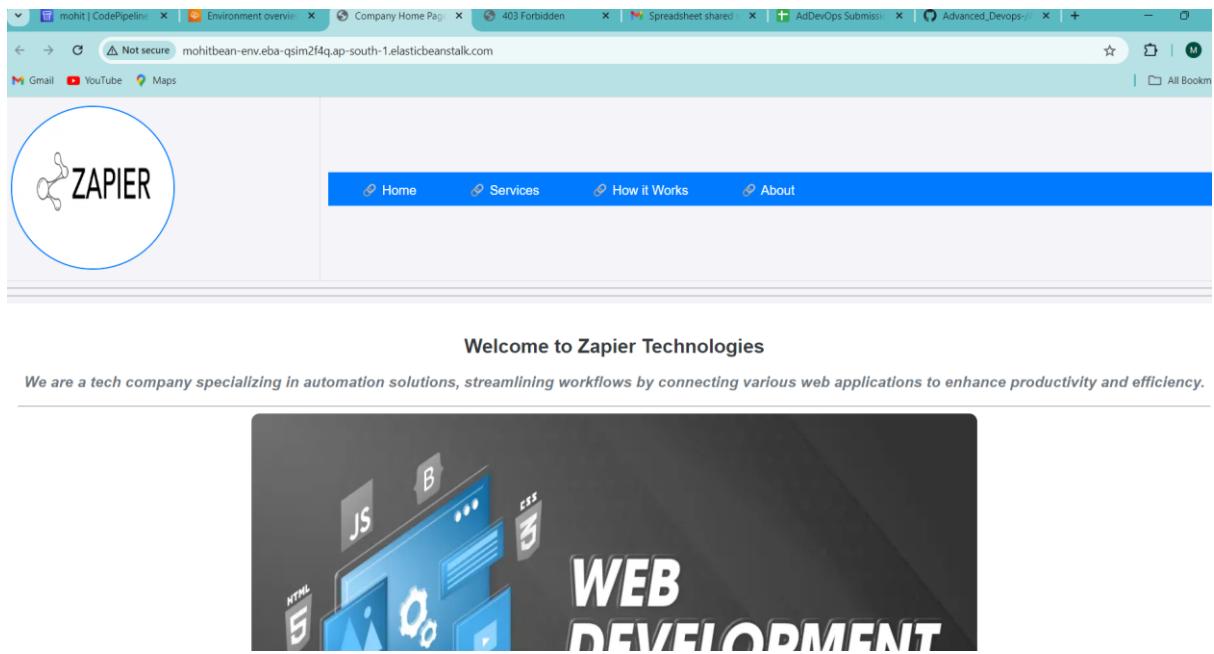
Step 4 : Beanstalk environment is created



view the pipeline build and deployment



Check the deployed website at beanstalk link



# Experiment:3

Name: Mohit Patil Class:D15A Roll no.37

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

1. Create 3 EC2 Ubuntu Instances on AWS.

|  | Monitor | ID                  | Status  | Type     | Health            | View alarms   | Last updated | Action     |
|--|---------|---------------------|---------|----------|-------------------|---------------|--------------|------------|
|  | Node-1  | i-0b7bce11cbbc6c6d0 | Running | t2.micro | 2/2 checks passed | View alarms + | ap-south-1b  | ec2-13-23- |
|  | Master  | i-0818f775837a32042 | Running | t2.micro | 2/2 checks passed | View alarms + | ap-south-1b  | ec2-52-66- |
|  | Node-2  | i-0f91747696236dd64 | Running | t2.micro | Initializing      | View alarms + | ap-south-1b  | ec2-3-7-25 |

1. Now click on connect to instance, then click on SSH client.
2. Now copy the ssh from the example and paste it on command prompt.(I used gitbash)

The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with the AWS logo, Services, search, and user information (Mumbai, MohitPatil). Below the bar, a welcome message for Amazon Linux 2023 is displayed, featuring a stylized tree graphic made of hashtags. The main area is a terminal window with the following content:

```
'#'
~\ _##_#
~~ \###\
~~ \##|
~~ \#/ __
~~ V~, .->
~~ .-
~~ ./
~/m/'[ec2-user@ip-172-31-10-187 ~]$ sudo su'
```

3. After this type on all 3 machines Yum install docker -y

```
~~~  
~~.~.  
~/m/  
[ec2-user@ip-172-31-10-245 ~]$ sudo su  
[root@ip-172-31-10-245 ec2-user]# yum install docker -y
```

4. To start the docker on master and slave perform this command: Systemctl start docker

```
containerd-1.20-1.amzn2023.0.1.x86_64           docker-25.0.6-1.amzn2023.0.2.x86_64  
.x86_64          iptables-libs-1.8.8-3.amzn2023.0.2.x86_64  
iptables-nft-1.8.8-3.amzn2023.0.2.x86_64        libcgroup-3.0-1.amzn2023.0.1.x86_64  
.x86_64          libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64  
libnfnetwork-1.0.1-19.amzn2023.0.2.x86_64       libnftnl-1.2.2-2.amzn2023.0.2.x86_64  
runc-1.1.13-1.amzn2023.0.1.x86_64  
pigz-2.5-1.amzn2023.0.3.x86_64
```

Complete!

```
[root@ip-172-31-10-245 ec2-user]#
```

i-0b7bce11cbcb6c6d0 (Node-1)

X

PublicIPs: 13.233.152.101 PrivateIPs: 172.31.10.245

```
Complete!  
[root@ip-172-31-10-187 ec2-user]# systemctl start docker
```

i-0818f775837a32042 (Master)

X

PublicIPs: 52.66.241.212 PrivateIPs: 172.31.10.187

## **EXTRA:**

To check if docker is installed or not

Docker -v

```
[root@ip-172-31-84-37 ec2-user]# systemctl start docker
[root@ip-172-31-84-37 ec2-user]# sudo su
[root@ip-172-31-84-37 ec2-user]# yum repolist
repo id                                repo name
amazonlinux                            Amazon Linux 2023 repository
kernel-livepatch                        Amazon Linux 2023 Kernel Livepatch repository
[root@ip-172-31-84-37 ec2-user]# docker --version
Docker version 25.0.5, build 5dc9bcc
```

## 5. Now to install kubeadm on master and Nodes :

Installing kubeadm: Go the official documentation

off kubeadm.

```
complete:
[ec2-user@ip-172-31-14-163 ~]$ sudo service docker start
Redirecting to /bin/systemctl start docker.service
[ec2-user@ip-172-31-14-163 ~]$ sudo systemctl enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[ec2-user@ip-172-31-14-163 ~]$ sudo systemctl start docker
[ec2-user@ip-172-31-14-163 ~]$ █
```

/ installing-kubeadm

 Edit  
 Create  
 Create  
 Print

# Installing kubeadm

This page shows how to install the `kubeadm` toolbox. For information on how to create a cluster with `kubeadm` once you have performed this installation process, see the [Creating a cluster with `kubeadm`](#) page.



This installation guide is for Kubernetes v1.31. If you want to use a different Kubernetes version, please refer to the following pages instead:

- [Installing kubeadm \(Kubernetes v1.30\)](#)
- [Installing kubeadm \(Kubernetes v1.29\)](#)
- [Installing kubeadm \(Kubernetes v1.28\)](#)
- [Installing kubeadm \(Kubernetes v1.27\)](#)

Before you begin  
Verify the system requirements  
unique IP addresses  
Check network connectivity  
Check root user permissions  
Swap configuration  
Installation steps  
Installation troubleshooting  
Configuration  
Troubleshooting  
What's new

## Before you begin

- A compatible Linux host. The Kubernetes project provides generic instructions for Linux distributions based on Debian and Red Hat, and

## 6. Scroll down and select Red Hat based distributions:

### Note:

There's a dedicated package repository for each Kubernetes minor version. If you want to install a minor version other than v1.31, please see the installation guide for your desired minor version.

[Edit](#) [Create](#) [Create](#) [Print](#)

[Debian-based distributions](#)

[Red Hat-based distributions](#)

[Without a package manager](#)

### 1. Set SELinux to `permissive` mode:

These instructions are for Kubernetes 1.31.

```
# Set SELinux in permissive mode (effectively disabling it)
sudo setenforce 0
sudo sed -i 's/^SELINUX=enforcing$/SELINUX=permissive/' /etc/selinux/config
```

### Caution:

- Setting SELinux in permissive mode by running `setenforce 0` and `sed ...` effectively disables it. This is required to allow containers to access the host filesystem; for example, some cluster network plugins require that. You have to do this until SELinux support is improved in the kubelet.
- You can leave SELinux enabled if you know how to configure it but it may require

## 7. Now copy the command on all 3 machines:

Before you start  
Verify the host  
unique identifier  
Check network  
Check requirements  
Swap configuration  
Installing Kubernetes  
Installing the kubelet  
Configuring the kubelet  
Troubleshooting  
What's next

### 1. Set SELinux to `permissive` mode:

These instructions are for Kubernetes 1.31.

```
# Set SELinux in permissive mode (effectively disabling it)
sudo setenforce 0
sudo sed -i 's/^SELINUX=enforcing$/SELINUX=permissive/' /etc/selinux/config
```

Reading the documentation for the version of Kubernetes that you plan to install.

```
# This overwrites any existing configuration in /etc/yum.repos.d/kubernetes.repo
cat <<EOF | sudo tee /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=Kubernetes
baseurl=https://pkgs.k8s.io/core:/stable:/v1.31/rpm/
enabled=1
gpgcheck=1
gpgkey=https://pkgs.k8s.io/core:/stable:/v1.31/rpm/repo/repodata/repomd.xml.key
exclude=kubelet kubeadm kubectl cri-tools kubernetes-cni
EOF
```

### 3. Install kubelet, kubeadm and kubectl:

```
sudo yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes
```

### 8.Run yum repolist command for checking the repositories

```
/m/
Last login: Wed Sep 18 14:58:05 2024 from 13.233.177.3
[ec2-user@ip-172-31-10-187 ~]$ sudo su
[root@ip-172-31-10-187 ec2-user]# yum repolist
repo id                                repo name
amazonlinux                             Amazon Linux
 2023 repository                         Amazon Linux
kernel-livepatch                         Amazon Linux
 2023 Kernel Livepatch repository
[root@ip-172-31-10-187 ec2-user]# []
```

```

8/9 Verifying : kubernetes-cni-1.5.1-150500.1.1.x86_64
9/9

Installed:
  conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64           cri-tools-1.31.1-150500
  .1.1.x86_64                                         kubeadm-1.31.1-150500.1.1.x86_64
  kubectl-1.31.1-150500.1.1.x86_64          kubelet-1.31.1-150500.1
  .1.x86_64                                         kubernetes-cni-1.5.1-150500.1.1.x86_64
  libnetfilter_cthelper-1.0.0-21.amzn2023.0.2.x86_64    libnetfilter_cttimeout-
  1.0.0-19.amzn2023.0.2.x86_64      libnetfilter_queue-1.0.5-2.amzn2023.0.2.x86_64

Complete!
[root@ip-172-31-10-187 ec2-user]# sudo systemctl enable --now kubelet
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /usr/lib/systemd/system/kubelet.service.
[root@ip-172-31-10-187 ec2-user]# █
i-0818f775837a32042 (Master)
PublicIPs: 52.66.241.212 PrivateIPs: 172.31.10.187

```

```

Complete!
[root@ip-172-31-10-187 ec2-user]# sudo systemctl enable --now kubelet
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /usr/lib/systemd/system/kubelet.service.
[root@ip-172-31-10-187 ec2-user]# yum repolist
repo id                                repo name
amazonlinux                             Amazon Linux 2023 repository
kernel-livepatch                         Amazon Linux 2023 Kernel Livepatch repository
kubernetes                             Kubernetes
[root@ip-172-31-10-187 ec2-user]# █
i-0818f775837a32042 (Master)
PublicIPs: 52.66.241.212 PrivateIPs: 172.31.10.187

```

## EXTRA :

Got error in initializing Kubernetes

```

[root@ip-172-31-31-240 ec2-user]# kubeadm init
[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
W0908 11:25:45.820964    2320 checks.go:1080] [preflight] WARNING: Couldn't create the interface used for talking to CRI runtime service: validate service connection: validate CRI v1 runtime API for endpoint "unix:///var/run/containerd/containerd.sock": connection error: desc = "transport: Error while dialing: dial unix /var/run/containerd/containerd.sock: [WARNING FileExisting-tc]: tc not found in system path"
error execution phase preflight: [preflight] Some fatal errors occurred:
  [ERROR FileContent--proc-sys-net-ipv4-ip_forward]: /proc/sys/net/ipv4/ip_forward contents are not set to 1
[preflight] If you know what you are doing, you can make a check non-fatal with `--ignore-preflight-errors='...` to see the stack trace of this error execute with --v=5 or higher

```

9.Copy paste the commands in all three instances

```
[root@ip-172-31-10-187 ec2-user]# kubeadm init
[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
  [WARNING FileExisting-socat]: socat not found in system path
  [WARNING FileExisting-tc]: tc 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'
W0918 15:44:45.531044    3429 checks.go:846] detected that the sandbox image "registry.k8s.io/pause:3.8" of the container runtime is inconsistent with that used by kubeadm. It is recommended to use "registry.k8s.io/pause:3.10" as the CRI sandbox image.
```

```
export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.10.187:6443 --token v5qt9c.8kc7lgb6xwill8co \
    --discovery-token-ca-cert-hash sha256:b9765a8ba4dc546e8263bd7b7531f78b4b8582c9757a9371e00ab90a71a38785
[root@ip-172-31-10-187 ec2-user]# mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
[root@ip-172-31-10-187 ec2-user]# export KUBECONFIG=/etc/kubernetes/admin.conf
[root@ip-172-31-10-187 ec2-user]# i-0818f775837a32042 (Master)
PublicIPs: 13.126.147.65 PrivateIPs: 172.31.10.187
```

10.After pasting the connection link in the nodes run the kubectl get nodes command to view the connected nodes successfully

```
ubuntu@ip-172-31-17-23:~$ kubectl get nodes
NAME           STATUS   ROLES      AGE   VERSION
ip-172-31-17-23 Ready    control-plane   3m56s   v1.29.0
ip-172-31-18-12 Ready    <none>        37s   v1.29.0
ip-172-31-26-153 Ready    <none>        24s   v1.29.0
ubuntu@ip-172-31-17-23:~$ kubectl get nodes
NAME           STATUS   ROLES      AGE   VERSION
ip-172-31-17-23 Ready    control-plane   9m34s   v1.29.0
ip-172-31-18-12 Ready    <none>        6m15s   v1.29.0
ip-172-31-26-153 Ready    <none>        6m2s   v1.29.0
ubuntu@ip-172-31-17-23:~$ |
```

# Experiment:4

Name: Mohit Patil Class:D15A Roll no.37

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

## What is Kubernetes?

Kubernetes, often referred to as K8s, is an open-source container orchestration platform that automates the deployment, scaling, and management of containerized applications. Originally developed by Google, it has become the industry standard for managing container workloads due to its flexibility and robust features.

- **Core Concepts of Kubernetes**
  - Containers: These are lightweight, portable packages that include everything needed to run an application, ensuring consistency across different environments.
  - 
  - Pods: The smallest deployable units in Kubernetes, pods can contain one or more containers that share storage and network resources.
  - 
  - Nodes: A node is a worker machine in the Kubernetes cluster that runs at least one pod. Nodes can be either physical or virtual machines.
  - 
  - Clusters: A cluster comprises multiple nodes that run containerized applications. The control plane manages the cluster's state.
  - 
  - Services: Services provide stable endpoints for accessing pods and facilitate load balancing and service discovery.
  - 
  - Deployments: A deployment manages the lifecycle of pods, allowing users to specify the number of replicas and facilitating rolling updates and rollbacks.

## Role of Kubernetes

### What is Kubectl?

Kubectl is the command-line interface used to interact with the Kubernetes API server. It enables users to manage resources within a Kubernetes cluster effectively.

### Configuration Files

Configuration files are essential for defining how resources should be created or modified within Kubernetes. Users can employ declarative configurations (using YAML/JSON files) or imperative commands directly in the terminal.

## Application Deployment on Kubernetes

- Define Application Requirements: Identify necessary resources such as CPU, memory, storage, etc.
- Create Deployment Configurations: Write deployment manifests specifying container images, replicas for scaling, health checks, etc.
- Deploying with Kubectl: Use kubectl commands like kubectl apply to deploy applications based on these configurations.
- Monitoring and Scaling Applications: Monitor performance metrics and adjust deployments based on traffic demands.
- Updating Applications: Modify deployment configurations for updates; Kubernetes supports rolling updates by default.
- Rollback Capabilities: If an update causes issues, kubectl allows easy rollback to previous versions using commands like kubectl rollout undo.

## Step 1. Creation of 2 EC2 Ubuntu Instances on AWS.

| Find Instance by attribute or tag (case-sensitive) |        |                     |                          |               |                                    |                            | All states        | <           | 1 | > | ⋮ |
|--|--------|---------------------|--------------------------|---------------|------------------------------------|----------------------------|-------------------|-------------|---|---|---|
|  | Name ↗ | Instance ID         | Instance state           | Instance type | Status check                       | Alarm status               | Availability Zone | Public IPv4 |   |   |   |
| <input checked="" type="checkbox"/>                | Master | i-0818f775837a32042 | <span>Running</span> ⓘ ⓘ | t2.large      | <span>2/2 checks passed</span> ⓘ ⓘ | <span>View alarms</span> + | ap-south-1b       | ec2-3-7-24  |   |   |   |
| <input type="checkbox"/>                           | mohit  | i-0022d489c88af599c | <span>Stopped</span> ⓘ ⓘ | t2.micro      | -                                  | <span>View alarms</span> + | ap-south-1b       | -           |   |   |   |
| <input checked="" type="checkbox"/>                | Node-1 | i-0b7bce11cbbc6c6d0 | <span>Running</span> ⓘ ⓘ | t2.micro      | <span>2/2 checks passed</span> ⓘ ⓘ | <span>View alarms</span> + | ap-south-1b       | ec2-13-23   |   |   |   |
| <input type="checkbox"/>                           | Node-2 | i-0f91747696236dd64 | <span>Stopped</span> ⓘ ⓘ | t2.micro      | <span>2/2 checks passed</span> ⓘ ⓘ | <span>View alarms</span> + | ap-south-1b       | -           |   |   |   |

## Step 2.Edit inbound rules of security group ‘launch-wizard-1’ and set ‘All Traffic’

The screenshot shows the AWS EC2 Security Groups console. A success message at the top says 'Inbound security group rules successfully modified on security group (sg-0820b863c8a52594f | launch-wizard-6)'. Below it, the security group details for 'sg-0820b863c8a52594f - launch-wizard-6' are shown. Under the 'Details' section, it lists the security group name as 'launch-wizard-6', owner as '445567072095', security group ID as 'sg-0820b863c8a52594f', and VPC ID as 'vpc-0b8f24d7c64f9775f'. It also shows an 'Inbound rules count' of 1 permission entry and an 'Outbound rules count' of 1 permission entry. The 'Inbound rules' tab is active, displaying one rule: 'All traffic' from 'Anywhere' to port 22. There is an 'Edit inbound rules' button at the bottom of the list.

## Step 3. Set master and worker as hostname on respective servers

```
ubuntu@ip-172-31-46-38:~$ sudo su
root@ip-172-31-46-38:/home/ubuntu# sudo hostnamectl set-hostname master
```

## Step 4.Installation of docker

```
root@ip-172-31-46-38:/home/ubuntu# sudo hostnamectl set-hostname master
root@ip-172-31-46-38:/home/ubuntu# sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:13 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [528 kB]
Get:14 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [127 kB]
Get:15 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [8352 B]
Get:16 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [368 kB]
Get:17 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [151 kB]
Get:18 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [45.0 kB]
```

```

root@ip-172-31-46-38:/home/ubuntu# 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 dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-buildx docker-compose-v2 docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 133 not upgraded.

```

```

sudo: systemctl: command not found
root@ip-172-31-46-38:/home/ubuntu# sudo systemctl enable docker
root@ip-172-31-46-38:/home/ubuntu# 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 Wed 2024-09-18 19:11:49 UTC; 2min 29s ago
TriggeredBy: • docker.socket
   Docs: https://docs.docker.com
 Main PID: 2364 (dockerd)
    Tasks: 9
   Memory: 25.7M (peak: 26.0M)
     CPU: 203ms
    CGroup: /system.slice/docker.service
           └─2364 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Sep 18 19:11:49 master systemd[1]: Starting docker.service - Docker Application Container Engine...
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.623361653Z" level=info msg="Starting up"
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.623933122Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-resolved, so using
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.717378513Z" level=info msg="Loading containers: start."
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.925520997Z" level=info msg="Loading containers: done."
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.943558249Z" level=info msg="Docker daemon" commit=24.0.7-0ubuntu4.1 graphdriver=overlay2 versio
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.943655279Z" level=info msg="Daemon has completed initialization"
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.989624164Z" level=info msg="API listen on /run/docker.sock"
Sep 18 19:11:49 master systemd[1]: Started docker.service - Docker Application Container Engine.
lines 1-21/21 (END)

i.0bd5556e5332ef8aa (master_1)

```

## Step 5. Installation of Kubernetes-

```

sudo: systemctl: command not found
root@ip-172-31-46-38:/home/ubuntu# sudo systemctl enable docker
root@ip-172-31-46-38:/home/ubuntu# 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 Wed 2024-09-18 19:11:49 UTC; 2min 29s ago
TriggeredBy: • docker.socket
   Docs: https://docs.docker.com
 Main PID: 2364 (dockerd)
    Tasks: 9
   Memory: 25.7M (peak: 26.0M)
     CPU: 203ms
    CGroup: /system.slice/docker.service
           └─2364 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Sep 18 19:11:49 master systemd[1]: Starting docker.service - Docker Application Container Engine...
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.623361653Z" level=info msg="Starting up"
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.623933122Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-resolved, so using
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.717378513Z" level=info msg="Loading containers: start."
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.925520997Z" level=info msg="Loading containers: done."
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.943558249Z" level=info msg="Docker daemon" commit=24.0.7-0ubuntu4.1 graphdriver=overlay2 versio
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.943655279Z" level=info msg="Daemon has completed initialization"
Sep 18 19:11:49 master dockerd[2364]: time="2024-09-18T19:11:49.989624164Z" level=info msg="API listen on /run/docker.sock"
Sep 18 19:11:49 master systemd[1]: Started docker.service - Docker Application Container Engine.
lines 1-21/21 (END)

i.0bd5556e5332ef8aa (master_1)

```

```

root@ip-172-31-46-38:/home/ubuntu#      install ca certificate
install: cannot stat 'ca': No such file or directory
root@ip-172-31-46-38:/home/ubuntu# curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo
apt-key add -
cat << EOF | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb https://apt.kubernetes.io/ kubernetes-xenial main

EOF
sudo apt-get update
sudo apt-get install -y kubelet kubeadm kubectl
usage: sudo -h | -K | -k | -v
usage: sudo -v [-ABkNnS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -u [-ABkNnS] [-g group] [-h host] [-p prompt] [-U user]
           [-u user] [command [arg ...]]
usage: sudo [-ABbEHkNnPS] [-r role] [-t type] [-C num] [-D directory]
           [-g group] [-h host] [-p prompt] [-R directory] [-T timeout]
           [-u user] [VAR=value] [-i | -s] [command [arg ...]]
usage: sudo -e [-ABkNnS] [-r role] [-t type] [-C num] [-D directory]
           [-g group] [-h host] [-p prompt] [-R directory] [-T timeout]
           [-u user] file ...
warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).

```

## Step.6 Kubernetes Deployment

```

E: Unable to locate package kubeadm
E: Unable to locate package kubectl
root@ip-172-31-46-38:/home/ubuntu# sudo apt-get install -y apt-transport-https ca-certificates curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
The following additional packages will be installed:
  libcurl3t64-gnutls libcurl4t64
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  curl libcurl3t64-gnutls libcurl4t64
3 upgraded, 1 newly installed, 0 to remove and 130 not upgraded.
Need to get 904 kB of archives.
After this operation, 38.9 kB of additional disk space will be used.
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 apt-transport-https all 2.7.14build2 [3974 B]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 curl amd64 8.5.0-2ubuntu10.4 [227 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl4t64 amd64 8.5.0-2ubuntu10.4 [341 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl3t64-gnutls amd64 8.5.0-2ubuntu10.4 [333 kB]
Fetched 904 kB in 0s (26.6 MB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 68108 files and directories currently installed.)
(Reading database ... 68108 files and directories currently installed.)

```

```

bash: https://packages.cloud.google.com/apt/doc/apt-key.gpg: No such file or directory
root@ip-172-31-46-38:/home/ubuntu# sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg https://packages.cloud.google.com/apt/doc/apt-key.gpg
root@ip-172-31-46-38:/home/ubuntu# 

```

```

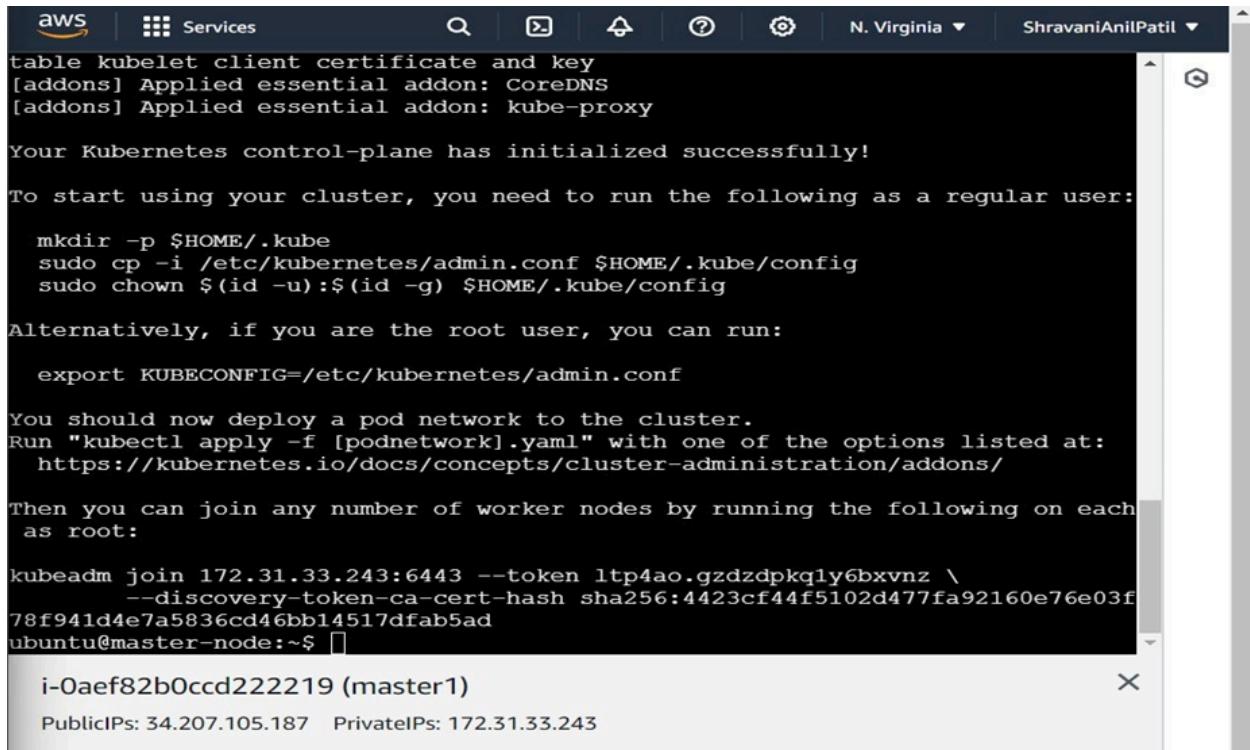
bash: /etc/apt/sources.list.d/kubernetes.list: Permission denied
root@ip-172-31-46-38:/home/ubuntu# echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-focal main" | sudo
do tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-focal main
root@ip-172-31-46-38:/home/ubuntu# sudo apt get update
E: Invalid operation get
root@ip-172-31-46-38:/home/ubuntu# sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Ign:5 https://packages.cloud.google.com/apt kubernetes-focal InRelease
Err:6 https://packages.cloud.google.com/apt kubernetes-focal Release
  404  Not Found [IP: 142.251.42.14 443]
Reading package lists... Done
E: The repository 'https://apt.kubernetes.io kubernetes-focal' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
root@ip-172-31-46-38:/home/ubuntu# 

```

```
E: Unable to locate package kubectl
root@ip-172-31-46-38:/home/ubuntu# sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Ign:5 https://packages.cloud.google.com/apt kubernetes-focal InRelease
Err:6 https://packages.cloud.google.com/apt kubernetes-focal Release
  404  Not Found [IP: 142.250.192.142 443]
Reading package lists... Done
E: The repository 'https://apt.kubernetes.io kubernetes-focal Release' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
root@ip-172-31-46-38:/home/ubuntu# sudo apt-get install -y kubectl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package kubectl
root@ip-172-31-46-38:/home/ubuntu#
```

## Extra:

```
kubectl 1.31.1 from Canonical✓ installed
root@ip-172-31-46-38:/home/ubuntu# kubectl version --client
Client Version: v1.31.1
Kustomize Version: v5.4.2
root@ip-172-31-46-38:/home/ubuntu#
```



The screenshot shows a terminal window within an AWS CloudShell interface. The terminal output is as follows:

```
aws | Services | Q | N. Virginia | ShravaniAnilPatil
table kubelet client certificate and key
[addons] Applied essential addon: CoreDNS
[addons] Applied essential addon: kube-proxy

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config

Alternatively, if you are the root user, you can run:

export KUBECONFIG=/etc/kubernetes/admin.conf

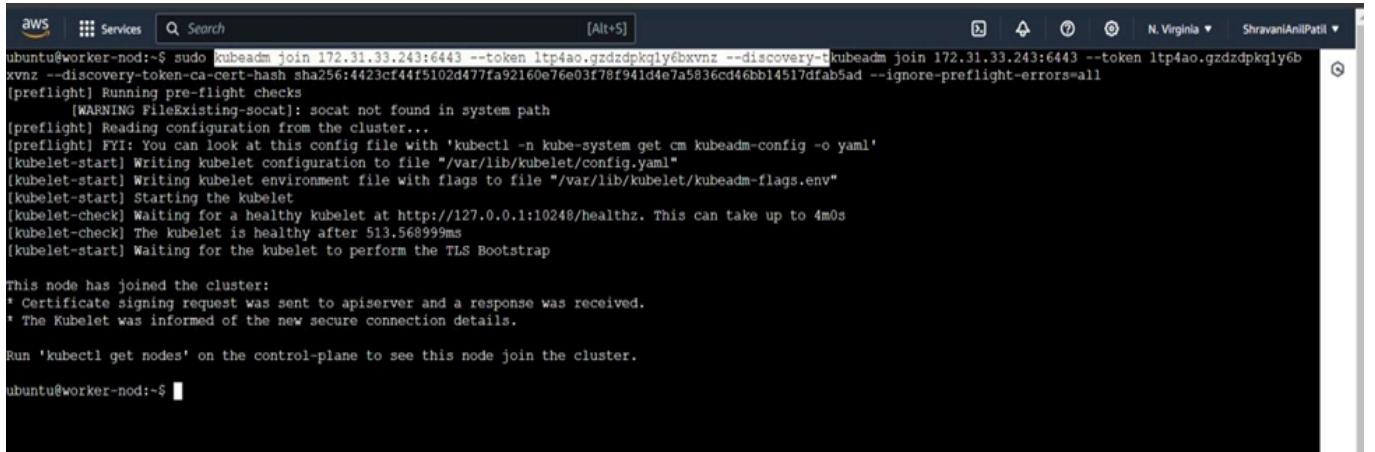
You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each
as root:

kubeadm join 172.31.33.243:6443 --token ltp4ao.gzdzdpkqly6bxvnz \
    --discovery-token-ca-cert-hash sha256:4423cf44f5102d477fa92160e76e03f
78f941d4e7a5836cd46bb14517dfab5ad
ubuntu@master-node:~$
```

The terminal prompt at the bottom shows the session identifier: i-Oaef82b0cccd222219 (master1).

## Step 7. Deploy Pod Network to Cluster and Join Worker Node to Cluster

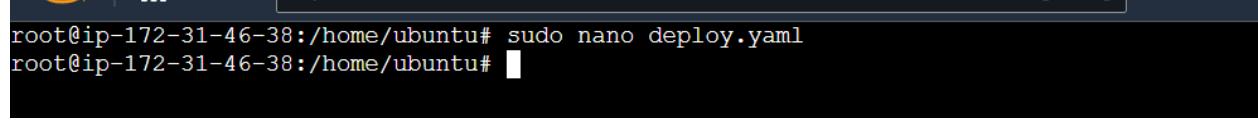


```
aws | Services | Search | [Alt+S] | N. Virginia | ShravaniAnilPatil | ⓘ
ubuntu@worker-nod:~$ sudo kubeadm join 172.31.33.243:6443 --token ltp4ao.gzdzdpkqly6bxvnz --discovery-token-ca-cert-hash sha256:4423cf44f5102d477fa92160e76e03f78f941d4e7a5836cd46bb14517dfab5ad --ignore-preflight-errors=all
[kubelet-start] Starting the kubelet
[kubelet-check] Waiting for a healthy kubelet at http://127.0.0.1:10248/healthz. This can take up to 4m0s
[kubelet-check] The kubelet is healthy after 513.56899ms
[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.
ubuntu@worker-nod:~$
```

## Step 8. Create one file deploy.yaml



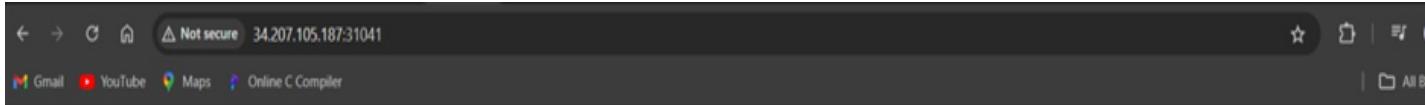
```
root@ip-172-31-46-38:/home/ubuntu# sudo nano deploy.yaml
root@ip-172-31-46-38:/home/ubuntu#
```

```
ubuntu@master-node:~$ cat deploy.yaml
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
ubuntu@master-node:~$ █
```

## Step 9 : Create Deployment

```
[root@ip-172-31-46-38 ~]# curl http://localhost:8080
curl: (7) Failed to connect to localhost port 8080: Connection refused
The connection to the server localhost:8080 was refused - did you specify the right host or port?
[root@ip-172-31-46-38 ~]# kubectl create -f deploy.yaml █
```

```
service/nginx-deployment exposed
ubuntu@master-node:~$ kubectl get svc
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)      AGE
kubernetes     ClusterIP  10.96.0.1    <none>       443/TCP     4h43m
nginx-deployment   LoadBalancer  10.101.59.94  <pending>   80:31041/TCP  4m34s
ubuntu@master-node:~$ █
```



# Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

## EXPERIMENT 5

NAME:MOHIT PATIL DIV:D15A ROLL NO:37

Aim:Installation and configuration of Terraform in windows

The screenshot shows the Terraform website's "Install" section. On the left sidebar, under "Operating Systems", "Windows" is selected. The main content area displays "Binary download" options for Windows:

- AMD64 Version: 1.9.5 [Download](#)
- ARM64 Version: 1.9.5 [Download](#)

Below this, there is a "Windows" section with its own "Binary download" options:

- 386 Version: 1.9.5 [Download](#)
- AMD64 Version: 1.9.5 [Download](#)

On the right side, there is a sidebar titled "About Terraform" with a brief description and links to "Featured docs" like Introduction to Terraform, Configuration Language, Terraform CLI, HCP Terraform, and Provider Use. At the bottom right, there is a "HCP Terraform" section with the tagline "Automate your infrastructure provisioning at any scale".

X

←  Extract Compressed (Zipped) Folders

### Select a Destination and Extract Files

Files will be extracted to this folder:

C:\Users\student.VESIT505-08\Downloads\terraform\_1.9.5\_windows\_386 (1)

[Browse...](#)

Show extracted files when complete

[Extract](#)

[Cancel](#)

| User variables for student |   |
|----------------------------|---|
| Variable                   | Value   |
| OneDrive                   | C:\Users\student.VESIT505-08\OneDrive           |
| path                       | F:\terraform                                    |
| TEMP                       | C:\Users\student.VESIT505-08\AppData\Local\Temp |
| TMP                        | C:\Users\student.VESIT505-08\AppData\Local\Temp |

|        |         |        |
|--------|---------|--------|
| New... | Edit... | Delete |
|--------|---------|--------|

| System variables     |   |
|----------------------|---|
| Variable             | Value   |
| ChocolateyInstall    | C:\ProgramData\chocolatey   |
| ComSpec              | C:\WINDOWS\system32\cmd.exe                                       |
| DriverData           | C:\Windows\System32\Drivers\DriverData                            |
| ETHEREUM_SOCKET      | \.\pipe\geth.ipc  |
| NUMBER_OF_PROCESSORS | 16  |
| OS                   | Windows_NT  |
| Path                 | C:\Program Files\Common Files\Oracle\Java\javapath;C:\Program ... |

|        |         |        |
|--------|---------|--------|
| New... | Edit... | Delete |
|--------|---------|--------|

|    |        |
|----|--------|
| OK | Cancel |
|----|--------|

| New User Variable                                  |   |
|--|---|
| Variable name:                                     | path  |
| Variable value:                                    | terraform                                     |
| <input type="button" value="Browse Directory..."/> | <input type="button" value="Browse File..."/> |
| <input type="button" value="OK"/>                  | <input type="button" value="Cancel"/>         |

```
Windows PowerShell + X - X

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\student.VESIT505-08> 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

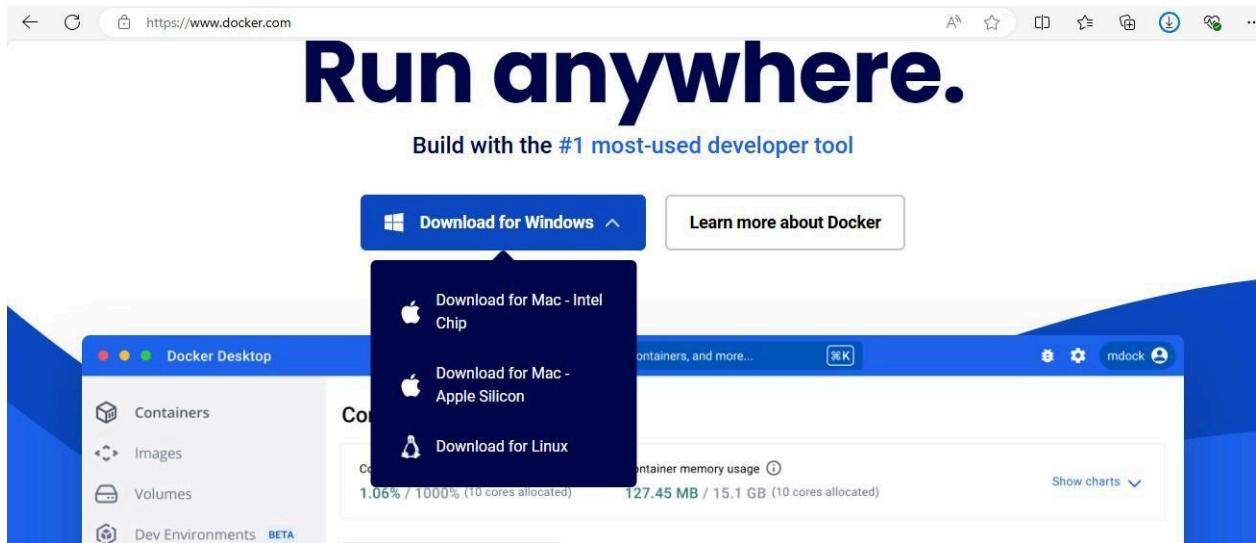
All other commands:
  console   Try Terraform expressions at an interactive command prompt
  fmt       Reformat your configuration in the standard style
  force-unlock Release a stuck lock on the current workspace
  get       Install or upgrade remote Terraform modules
  graph    Generate a Graphviz 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
```

## Experiment 6

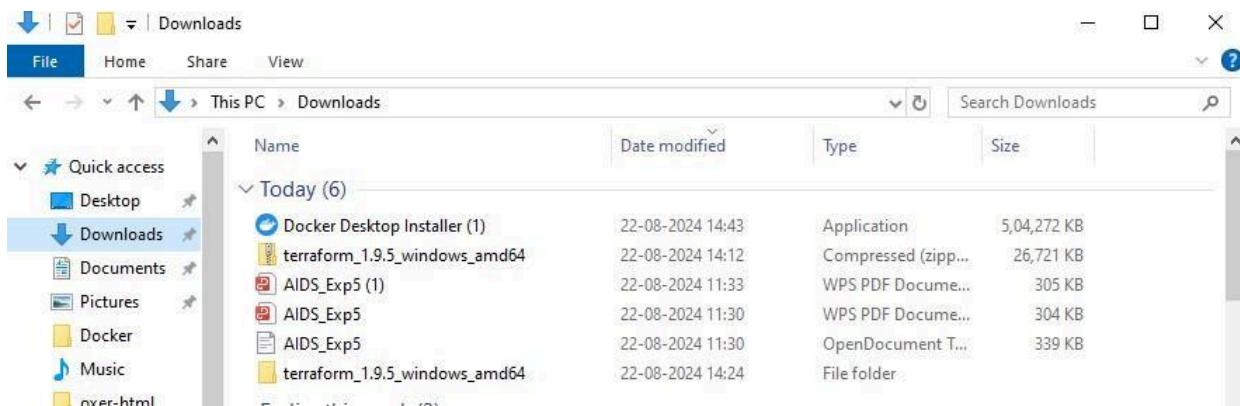
**Aim:**

To Build, change, and destroy AWS / GCP /Microsoft Azure/ DigitalOcean infrastructure Using Terraform.(S3 bucket or Docker)

Step 1: Download Docker from [www.docker.com](https://www.docker.com)



Step 2: The Docker is successfully downloaded. Now, run the docker installer and complete the installation.





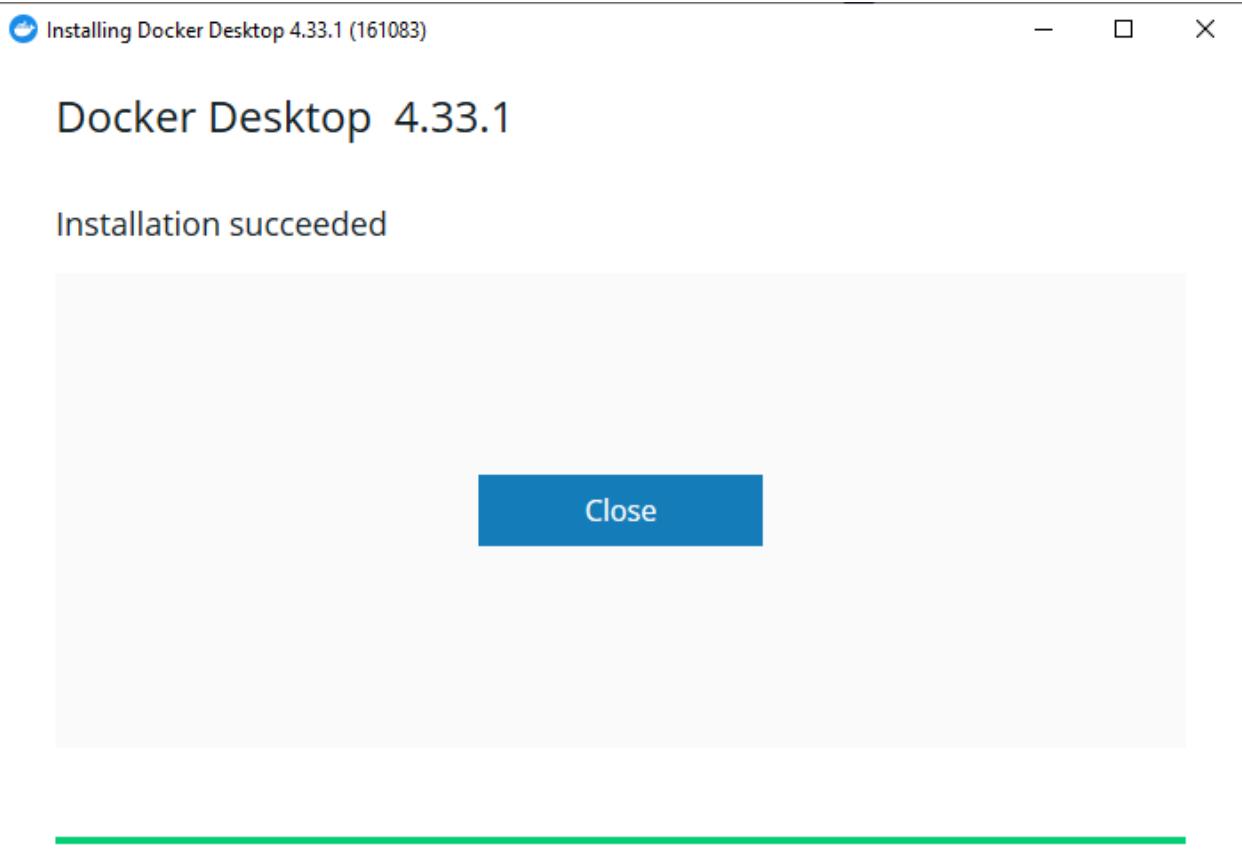
Installing Docker Desktop 4.33.1 (161083)



## Docker Desktop 4.33.1

Unpacking files...

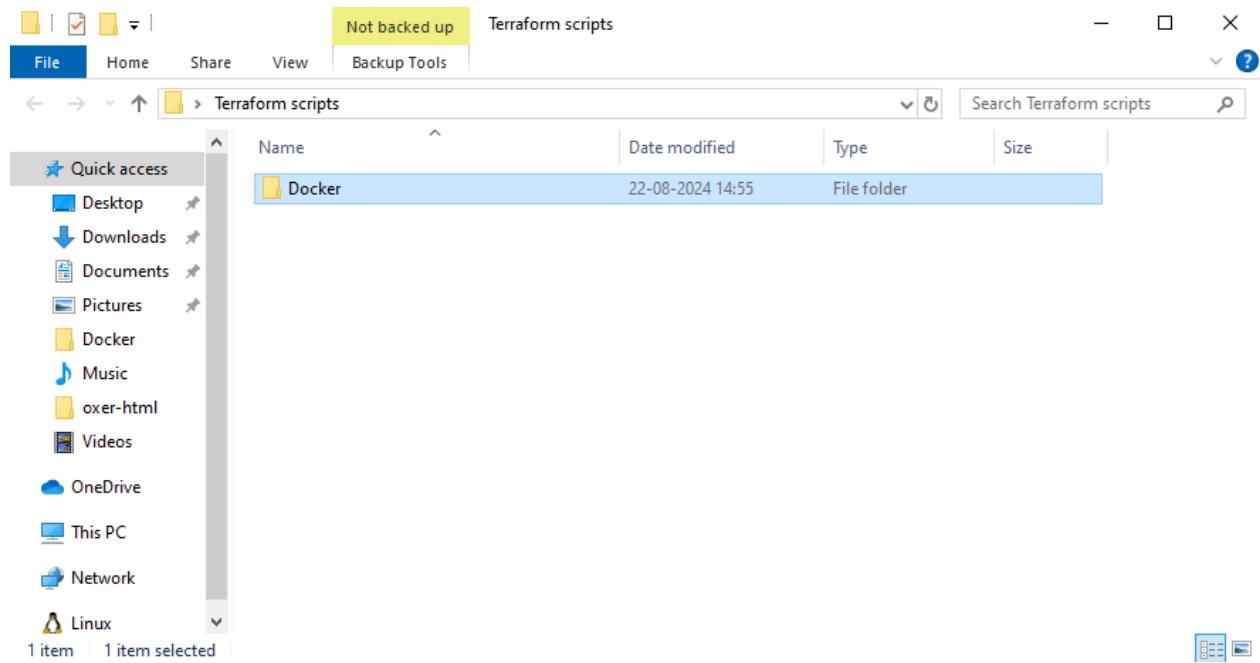
```
Unpacking file: resources/docker-desktop.iso
Unpacking file: resources/ddvp.ico
Unpacking file: resources/config-options.json
Unpacking file: resources/componentsVersion.json
Unpacking file: resources/bin/docker-compose
Unpacking file: resources/bin/docker
Unpacking file: resources/.gitignore
Unpacking file: InstallerCli.pdb
Unpacking file: InstallerCli.exe.config
Unpacking file: frontend/vk_swiftshader_icd.json
Unpacking file: frontend/v8_context_snapshot.bin
Unpacking file: frontend/snapshot_blob.bin
Unpacking file: frontend/resources/regedit/vbs/util.vbs
Unpacking file: frontend/resources/regedit/vbs/regUtil.vbs
```



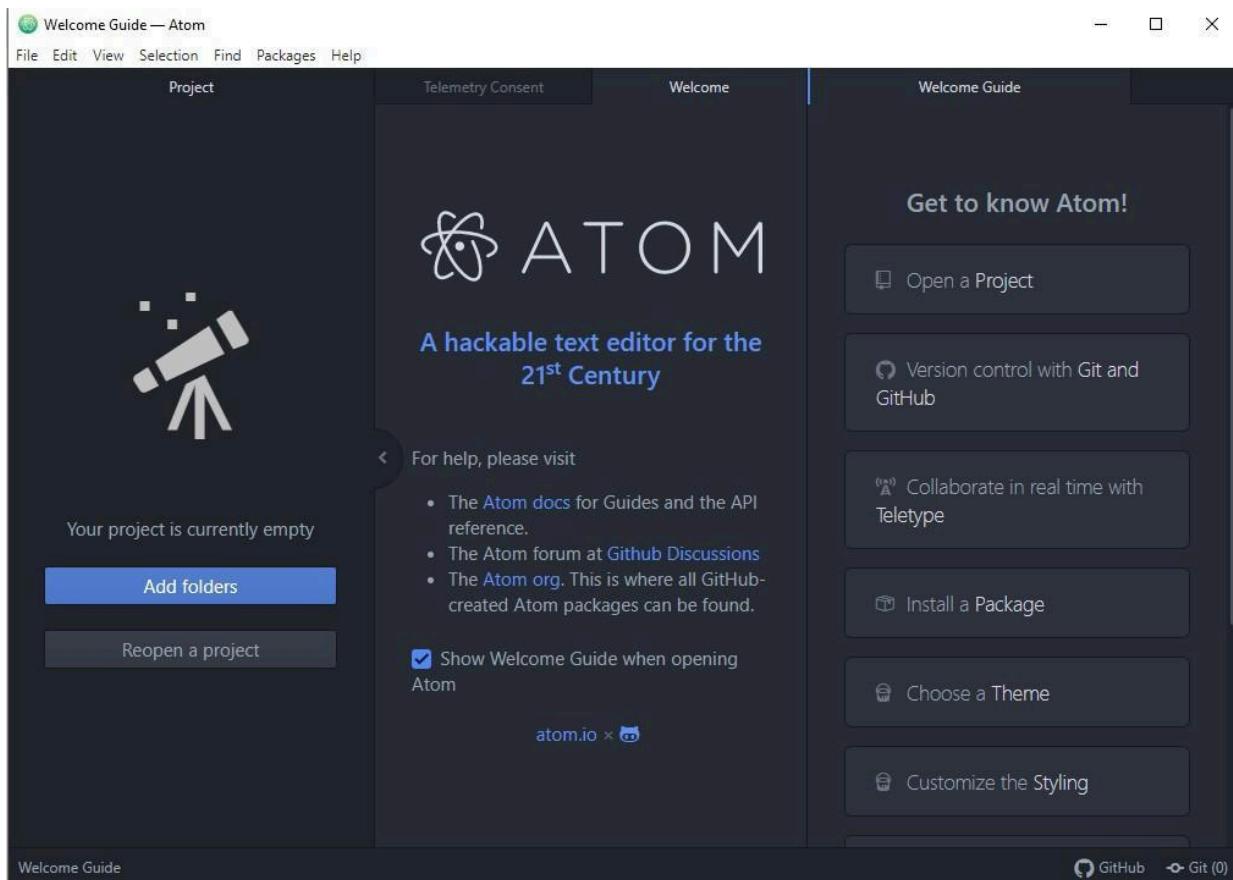
Step 3: Open Command Prompt and run as administrator. Enter the command docker --version, to check whether the docker is successfully installed.

A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the output of the "docker --version" command, which displays "Docker version 27.1.1, build 6312585". Below this, the Docker help text is displayed, listing various commands like run, exec, ps, build, pull, push, images, login, logout, search, version, and info, along with their descriptions. At the bottom, management commands like builder, buildx\*, compose\*, container, and context are listed.

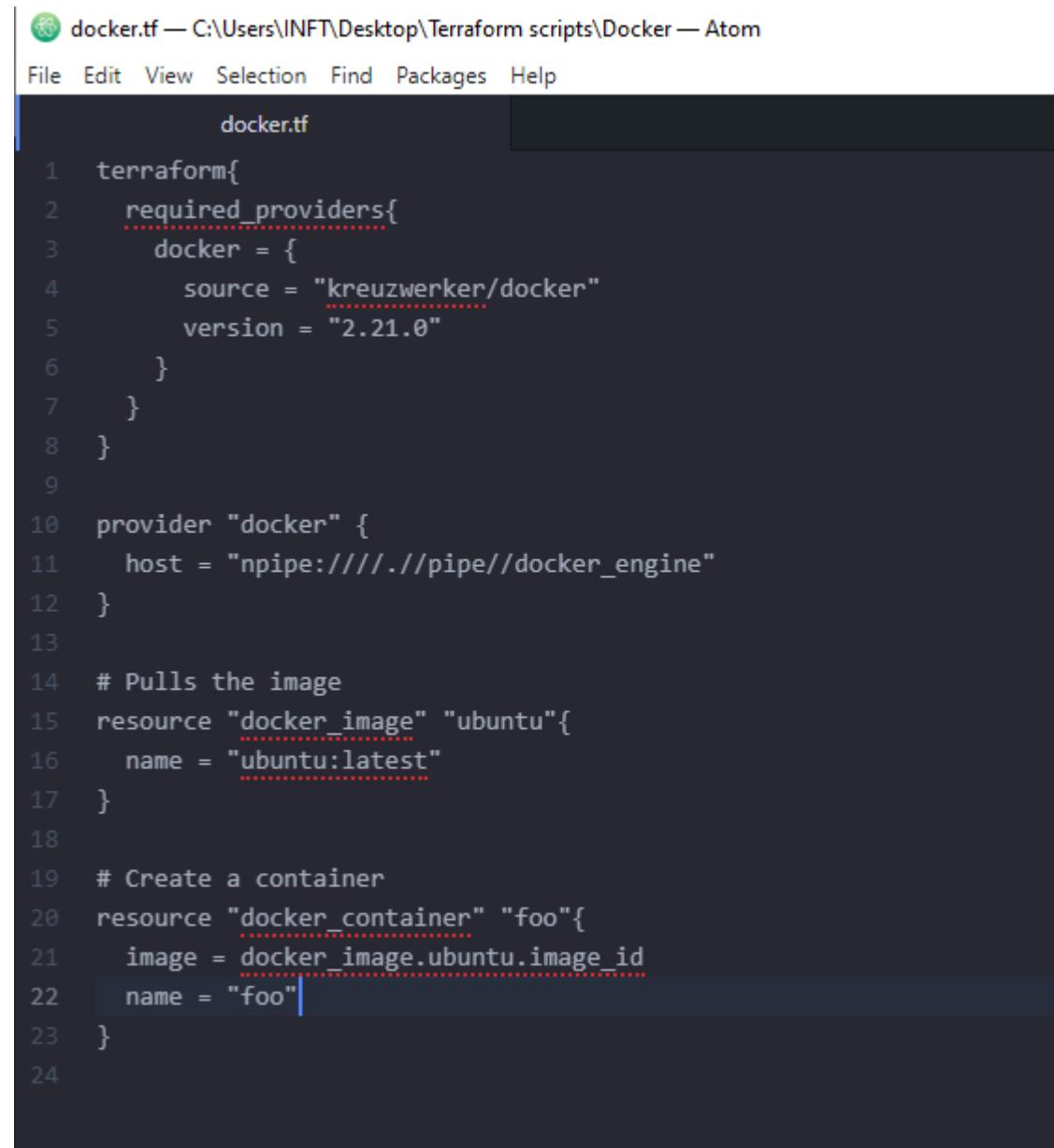
Step 4: Create a folder Terraform\_scripts and inside it create a folder named Docker.



## Step 5: Download Atom Editor.



Step 6: Run the following script in the Atom Editor



The screenshot shows the Atom code editor with a dark theme. The title bar says "docker.tf — C:\Users\INFT\Desktop\Terraform scripts\Docker — Atom". The menu bar includes File, Edit, View, Selection, Find, Packages, and Help. The main pane displays a Terraform configuration file with syntax highlighting. The code is as follows:

```
1 terraform{  
2     required_providers{  
3         docker = {  
4             source = "kreuzwerker/docker"  
5             version = "2.21.0"  
6         }  
7     }  
8 }  
9  
10 provider "docker" {  
11     host = "npipe://./pipe/docker_engine"  
12 }  
13  
14 # Pulls the image  
15 resource "docker_image" "ubuntu"{  
16     name = "ubuntu:latest"  
17 }  
18  
19 # Create a container  
20 resource "docker_container" "foo"{  
21     image = docker_image.ubuntu.image_id  
22     name = "foo"  
23 }  
24
```

Step 7: Open Windows Explorer and run the following command `terraform init`, `terraform plan`, `terraform apply`, `terraform destroy` and `docker images`.

```
Windows PowerShell
PS C:\Users\INFT\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.
PS C:\Users\INFT\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)
```

```
Windows PowerShell
Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
PS C:\Users\INFT\Desktop\Terraform_scripts\Docker> terraform apply

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
    + remove_volumes = true
    + restart         = "no"
    + rm              = false
    + runtime         = (known after apply)
    + security_opts  = (known after apply)
    + shm_size        = (known after apply)
    + start           = true
    + stdn_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.

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_image.ubuntu: Creating...
docker_image.ubuntu: Still creating... [10s elapsed]
docker_image.ubuntu: Creation complete after 11s [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_container.foo: Creating...

Error: container exited immediately

with docker_container.foo,
on docker.tf line 20, in resource "docker_container" "foo":
  20: resource "docker_container" "foo" {
```

```
Windows PowerShell

PS C:\Users\INFT\Desktop\Terraform_scripts\Docker> terraform destroy
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:
- destroy

Terraform will perform the following actions:

# 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, 1 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_image.ubuntu: Destroying... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_image.ubuntu: Destruction complete after 1s

Destroy complete! Resources: 1 destroyed.
PS C:\Users\INFT\Desktop\Terraform_scripts\Docker>
```

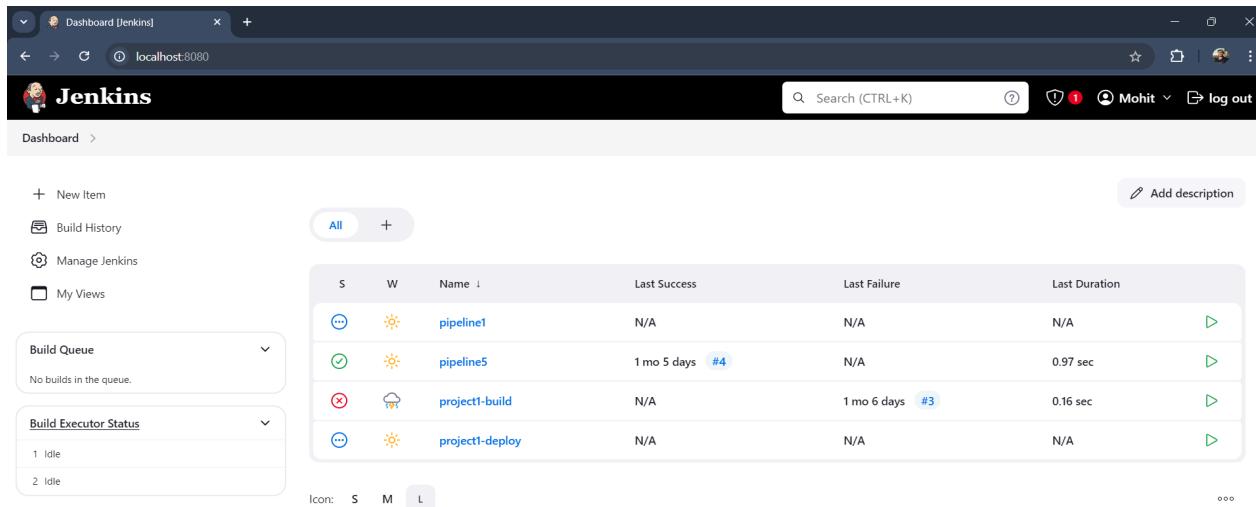
```
Windows PowerShell

PS C:\Users\INFT\Desktop\Terraform_scripts\Docker> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
PS C:\Users\INFT\Desktop\Terraform_scripts\Docker>
```

## EXPERIMENT 7

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

- Open up Jenkins Dashboard on localhost, port 8080 or whichever port it is at for you.

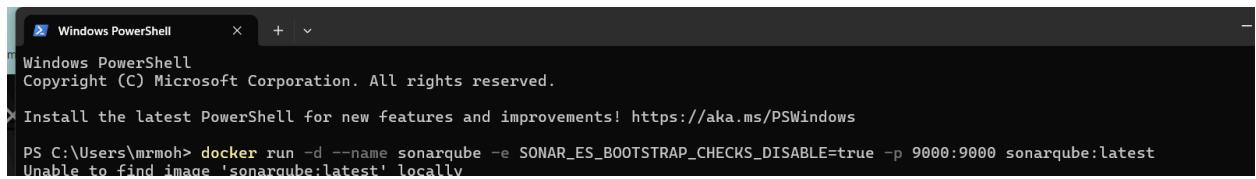


The screenshot shows the Jenkins dashboard at [localhost:8080](http://localhost:8080). The main area displays a table of build pipelines:

| S   | W  | Name                            | Last Success   | Last Failure   | Last Duration |
|-----|----|---------------------------------|----------------|----------------|---------------|
| ... | ☀️ | <a href="#">pipeline1</a>       | N/A            | N/A            | N/A           |
| ✓   | ☀️ | <a href="#">pipeline5</a>       | 1 mo 5 days #4 | N/A            | 0.97 sec      |
| ✗   | ☁️ | <a href="#">project1-build</a>  | N/A            | 1 mo 6 days #3 | 0.16 sec      |
| ... | ☀️ | <a href="#">project1-deploy</a> | N/A            | N/A            | N/A           |

On the left sidebar, there are links for 'New Item', 'Build History', 'Manage Jenkins', and 'My Views'. Below these are sections for 'Build Queue' (No builds in the queue) and 'Build Executor Status' (1 Idle, 2 Idle). A search bar and a 'Add description' button are also visible at the top right.

- Run SonarQube in a Docker container using this command

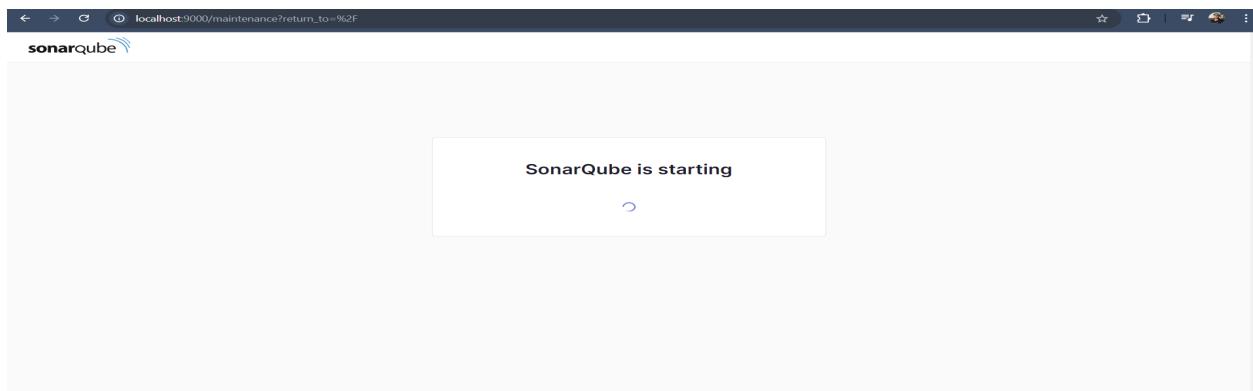


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\mrmmoh> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
Unable to find image 'sonarqube:latest' locally
```

- Once the container is up and running, you can check the status of SonarQube at localhost port 9000.



- Login to SonarQube using username admin and password admin.

A screenshot of the SonarQube login page. At the top is the Sonar logo. Below it is a light gray header with a blue gear icon. The main area has a white background with a blue decorative graphic. The title 'Log in to SonarQube' is centered. There are two input fields: 'Login \*' containing 'admin' and 'Password \*' containing '\*\*\*\*\*'. At the bottom are two buttons: 'Go back' and a blue 'Log in' button.

- Create a manual project in SonarQube with the name sonarqube

1 of 2

## Create a local project

Project display name \*

Project key \*

Main branch name \*

The name of your project's default branch [Learn More](#) 

CancelNext

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

Name ↓

Enabled

SonarQube Scanner for Jenkins 2.17.2

This plugin allows an easy integration of [SonarQube](#), the open source platform for Continuous Inspection of code quality.  
[Report an issue with this plugin](#)



- Under Jenkins ‘Configure System’, look for SonarQube Servers and enter the details.

SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

Environment variables

SonarQube installations

List of SonarQube installations

|                             |  |                                     |
|-----------------------------|--|-------------------------------------|
| Name                        | <input type="text" value="sonarqube"/>                                       | <span style="color: red;">X</span>  |
| Server URL                  | Default is <a href="http://localhost:9000">http://localhost:9000</a>         |                                     |
|                             | <input type="text" value="http://localhost:9000"/>                           |                                     |
| Server authentication token | SonarQube authentication token. Mandatory when anonymous access is disabled. |                                     |
|                             | <input type="text" value="- none -"/>  | <span style="color: blue;">▼</span> |
|                             | <a href="#">+ Add +</a>  |                                     |
|                             | <a href="#">Advanced</a> <span style="color: blue;">▼</span>                 |                                     |

[Add SonarQube](#)

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

SonarQube Scanner installations

[Add SonarQube Scanner](#)

|   |   |
|---|---|
| <input type="checkbox"/> <b>SonarQube Scanner</b>   | <span style="color: red;">X</span>                        |
| Name  | <input type="text" value="sonarqube"/>                    |
| <input checked="" type="checkbox"/> Install automatically <span style="color: blue;">?</span> |   |
| <b>Install from Maven Central</b>   |   |
| Version   | <input type="text" value="SonarQube Scanner 6.2.0.4584"/> |
| <span style="color: blue;">▼</span>   |   |
| <a href="#">Add Installer</a> <span style="color: blue;">▼</span>                             |   |

[Add SonarQube Scanner](#)

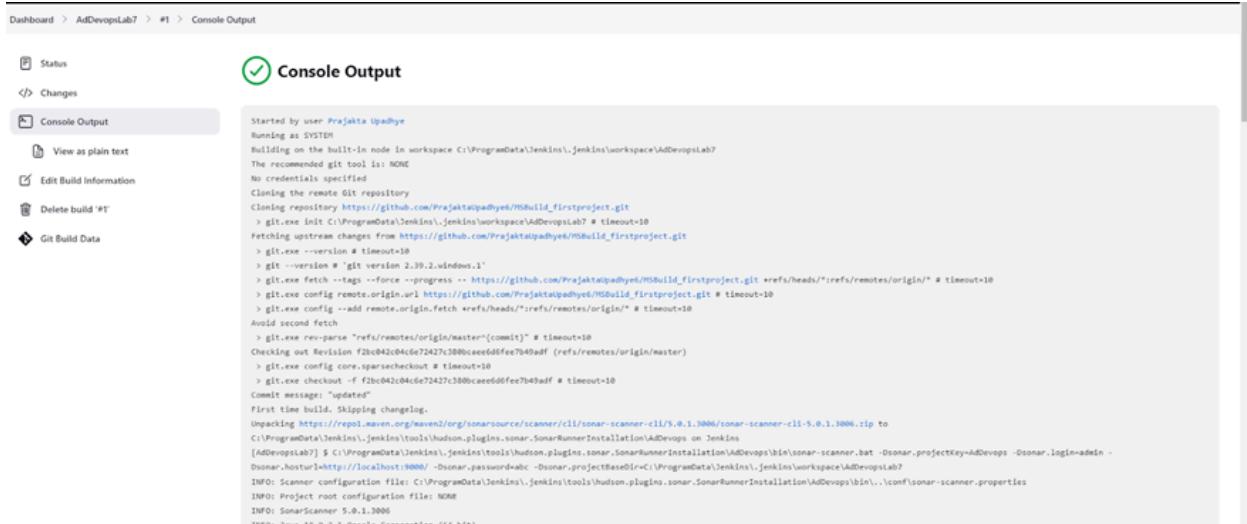
- After the configuration, create a New Item in Jenkins, choose a freestyle project.

- Choose this GitHub repository in Source Code Management.

The screenshot shows the Jenkins configuration interface for a Git repository. The 'Repository URL' field contains the URL [https://github.com/shazforiot/MSBuild\\_firstproject.git](https://github.com/shazforiot/MSBuild_firstproject.git). The 'Branch Specifier' field contains the branch specifier `*/master`.

The Jenkins dashboard for the 'sonarcube' project is shown. The left sidebar includes links for Status, Changes, Workspace, Build Now, Configure, Delete Project, SonarQube, and Rename. The main area displays the SonarQube logo and the text 'Permalinks'. At the bottom, there are links for Build History, Atom feed for all, and Atom feed for failures.

- Check the console output.

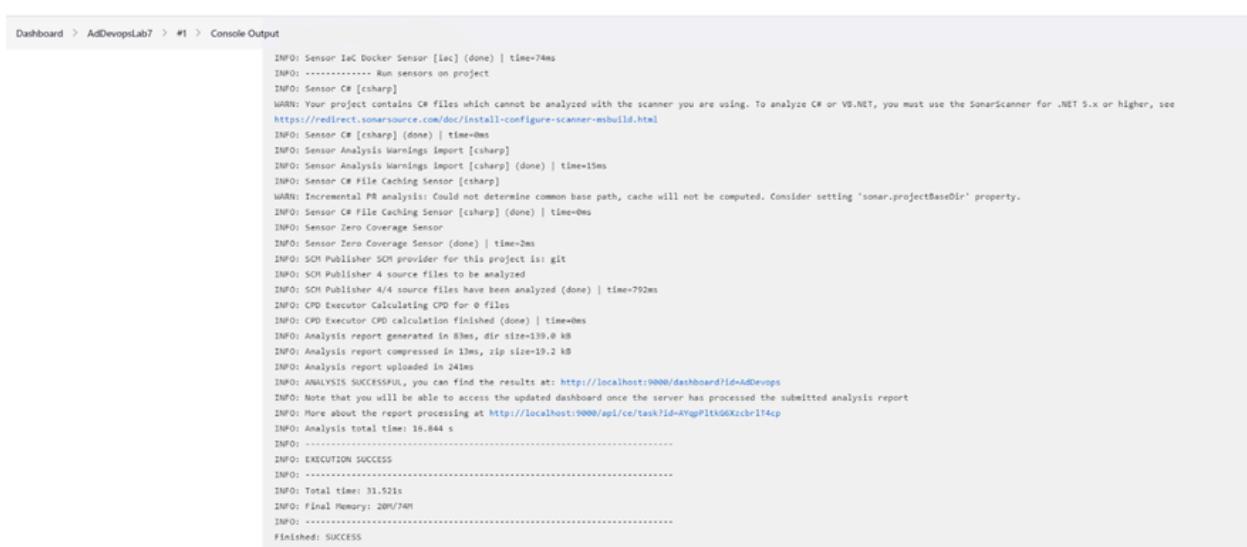


```

Started by user Prajatka Upadhye
Running as SYSTEM
Building on the built-in node in workspace C:\ProgramData\Jenkins\jenkins\workspace\AdDevopsLab7
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository https://github.com/PrajatkaUpadhye/MSBuild_firstproject.git
> git.exe init C:\ProgramData\Jenkins\jenkins\workspace\AdDevopsLab7 # timeout=10
Fetching upstream changes from https://github.com/PrajatkaUpadhye/MSBuild_firstproject.git
> git.exe --version # timeout=10
> git --version # git version 2.39.2.windows.1"
> git.exe fetch --tags --force --progress -- https://github.com/PrajatkaUpadhye/MSBuild_firstproject.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe config remote.origin.url https://github.com/PrajatkaUpadhye/MSBuild_firstproject.git # timeout=10
> git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision f2b042c04de72427c3080caed6dfee7049aff (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2b042c04de72427c3080caed6dfee7049aff # timeout=10
Commit message: "updated"
First time build. Stopping changelog.
Unpacking https://repo.maven.org/maven2/org/sonar/source/scanner/cil/sonar-scanner-cil/5.0.1.3006/sonar-scanner-cil-5.0.1.3006.zip to
C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\AdDevops on Jenkins
[AdDevopsLab7] $ C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\AdDevops\bin\sonar-scanner.bat -Dsonar.projectKey=AdDevops -Dsonar.login=admin -Dsonar.hostUrl=http://localhost:9000/ -Dsonar.password=abc -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\jenkins\workspace\AdDevopsLab7
INFO: Scanner configuration file: C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\AdDevops\bin..\conf\sonar-scanner.properties
INFO: Project root configuration file: NORM
INFO: SonarScanner 5.0.1.3006
INFO: Java 18.0.2.1 Oracle Corporation (64-bit)

Dashboard > AdDevopsLab7 > #1 > Console Output

```

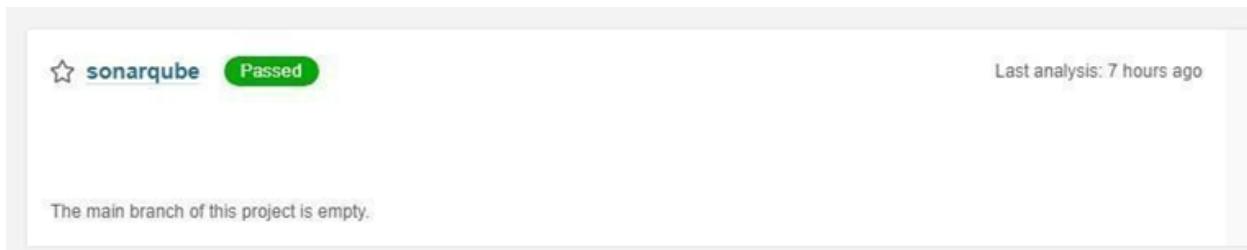


```

INFO: Sensor IaC Docker Sensor [iac] (done) | time=74ms
INFO: -----
INFO: Sensor CM [csharp]
WARN: Your project contains CM files which cannot be analyzed with the scanner you are using. To analyze CM or VB.NET, you must use the SonarScanner for .NET 5.x or higher, see https://redirect.sonarsource.com/doc/install/configure-scanner-msbuild.html
INFO: Sensor CM [csharp] (done) | time=0ms
INFO: Sensor Analysis Warnings Import [csharp]
INFO: Sensor Analysis Warnings Import [csharp] (done) | time=15ms
INFO: Sensor CM File Caching Sensor [csharp]
WARN: Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir' property.
INFO: Sensor CM File Caching Sensor [csharp] (done) | time=0ms
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=2ms
INFO: SON Publisher SON provider for this project is: git
INFO: SON Publisher 4 source files to be analyzed
INFO: SON Publisher 4/4 source files have been analyzed (done) | time=792ms
INFO: CPD Executor Calculating CPD for 0 files
INFO: CPD Executor CPD calculation finished (done) | timenodes
INFO: Analysis report generated in 3hrs, dir size=139.0 kB
INFO: Analysis report compressed in 1hrs, zip size=19.2 kB
INFO: Analysis report uploaded in 24hrs
INFO: ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=AdDevops
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
INFO: More about the report processing at http://localhost:9000/api/ce/task?id=AVgpltk06Kzcbrl4cp
INFO: Analysis total time: 16.844 s
INFO: -----
INFO: EXECUTION SUCCESS
INFO: -----
INFO: Total time: 31.521s
INFO: Final Memory: 209/748
INFO: -----
Finished: SUCCESS

```

- Once the build is complete, check the project in SonarQube.



localhost:9000/dashboard?id=sonarqube-test&codeScope=overall

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More Q

sonarqube-test / main ?

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

main Version not provided Set as homepage

Quality Gate \* Passed Last analysis 14 minutes ago

The last analysis has warnings. See details

New Code Overall Code

Security Reliability Maintainability

0 Open issues 0 Open issues 0 Open issues

0 H 0 M 0 L 0 H 0 M 0 L 0 H 0 M 0 L

A A A

Accepted issues Coverage Duplications

0 On 0 lines to cover. 0.0% On 86 lines.

Valid issues that were not fixed

Security Hotspots

Detailed description: This screenshot shows the SonarQube dashboard for the project 'sonarqube-test'. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. Below the navigation is a breadcrumb trail: sonarqube-test / main. The main content area is titled 'main' and shows the 'Quality Gate' status as 'Passed' with a green checkmark icon. A note indicates that the last analysis had warnings, with a link to 'See details'. The dashboard is divided into several sections: 'New Code' and 'Overall Code' (selected), 'Security' (0 open issues, A grade), 'Reliability' (0 open issues, A grade), 'Maintainability' (0 open issues, A grade), 'Accepted issues' (0), 'Coverage' (On 0 lines to cover), and 'Duplications' (0.0%, On 86 lines). A 'Security Hotspots' section is partially visible at the bottom. The overall layout is clean and modern, using a light color scheme and large, legible fonts.

## 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.

Step 1 : Visit the following link to download the SonarScanner CLI -

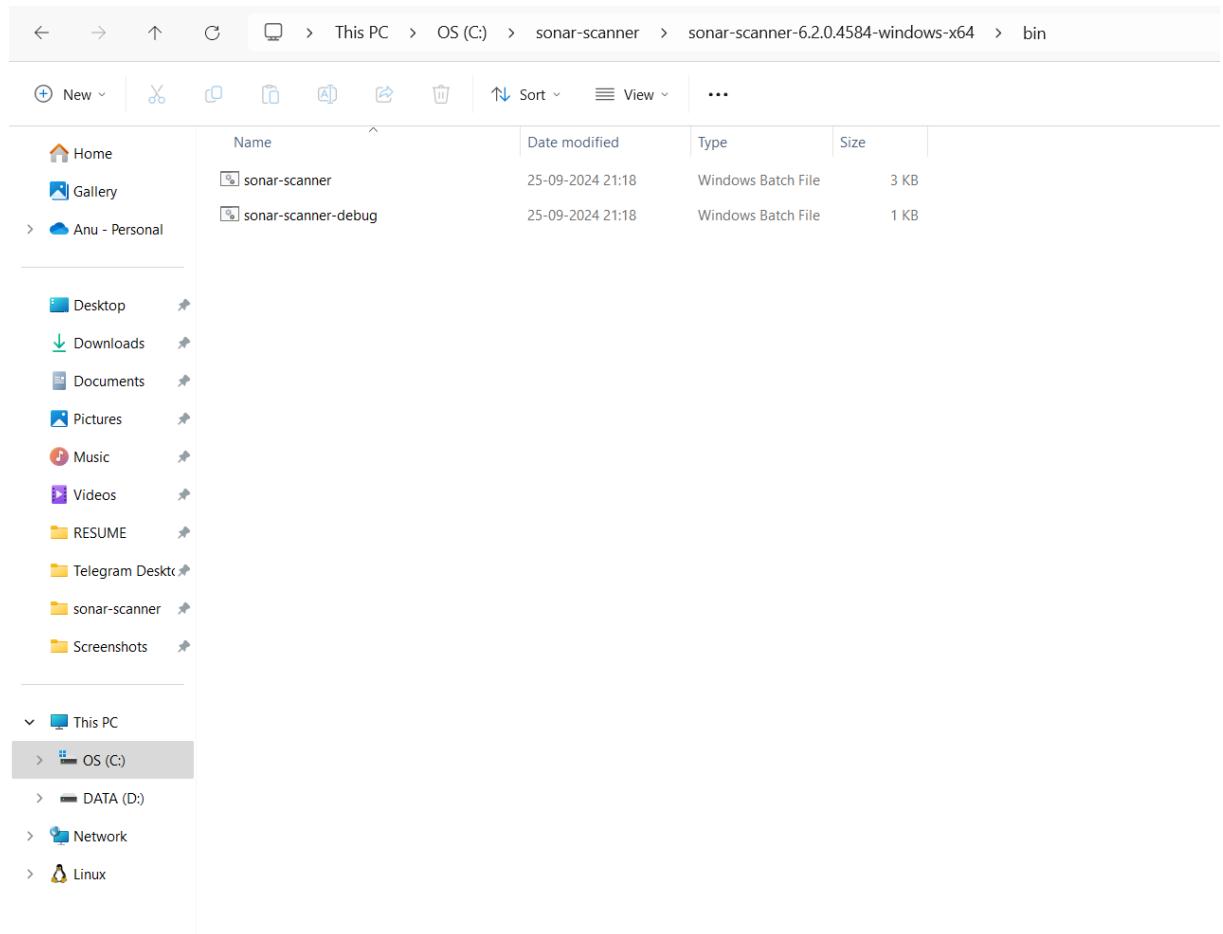
<https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscanner/> and then click on Windows x-64 to download the zip file.

The screenshot shows the SonarScanner CLI page on the SonarQube documentation site. The left sidebar contains navigation links for SonarQube, Docs 10.6, and various scanner-related topics like Analyzing source code, Scanners, and SonarScanner CLI. The main content area features a heading 'SonarScanner CLI' and a section for version 6.2, released on 2024-09-17. It highlights support for PKCS12 truststore generated with OpenSSL and provides download links for Linux x64, Linux AArch64, Windows x64, macOS x64, macOS AArch64, Docker Any (Requires a pre-installed JVM), and Release notes. A note states that SonarScanners run on code checked out. To the right, there's a 'On this page' sidebar with links to configuring projects, running the CLI from a zip file, and other documentation sections.

Step 2: Extract the content in C drive and name the folder sonar-scanner

The screenshot shows a Windows File Explorer window with the path 'This PC > OS (C:)'. The 'sonar-scanner' folder is highlighted. The details pane on the right shows the following information for the 'sonar-scanner' folder:

| Details       | File location    | Date modified    |
|---------------|------------------|------------------|
| Type          | C:\              | 25-09-2024 21:16 |
| File location | C:\              |                  |
| Date modified | 25-09-2024 21:16 |                  |



Step 3: Open Command Prompt and run as administrator and run the following commands –

```
cd C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin
```

```
dir
```

```
sonar-scanner.bat
```

```
Administrator: Command Prompt
C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>dir
Volume in drive C is OS
Volume Serial Number is E83B-22BB

Directory of C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin

25-09-2024 21:18    <DIR>        .
25-09-2024 21:18    <DIR>        ..
25-09-2024 21:18            805 sonar-scanner-debug.bat
25-09-2024 21:18            2,553 sonar-scanner.bat
25-09-2024 21:18            3,358 bytes
25-09-2024 21:18           2 File(s)   3,358 bytes
25-09-2024 21:18          2 Dir(s)  8,509,411,328 bytes free

C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>sonar-scanner.bat
22:44:22.348 INFO Scanner configuration file: C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin\..\conf\sonar-scanner.properties
22:44:22.353 INFO Project root configuration file: NONE
22:44:22.369 INFO SonarScanner CLI 6.2.0.4584
22:44:22.370 INFO Java 17.0.12 Eclipse Adoptium (64-bit)
22:44:22.371 INFO Windows 11 10.0 amd64
22:44:22.389 INFO User cache: C:\Users\User\.sonar\cache
22:44:22.827 INFO JRE provisioning: os[windows], arch[amd64]
22:44:23.921 INFO EXECUTION FAILURE
22:44:23.923 INFO Total time: 1.577s
22:44:23.923 ERROR Error during SonarScanner CLI execution
java.lang.IllegalStateException: Error status returned by url [https://api.sonarcloud.io/analysis/jres?os=windows&arch=amd64]: 401
    at org.sonarsource.scanner.lib.internal.http.ServerConnection.callUrl(ServerConnection.java:182)
    at org.sonarsource.scanner.lib.internal.http.ServerConnection.callApi(ServerConnection.java:145)
    at org.sonarsource.scanner.lib.internal.http.ServerConnection.callRestApi(ServerConnection.java:123)
    at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreMetadata(JavaRunnerFactory.java:159)
    at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreFromServer(JavaRunnerFactory.java:138)
    at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.createRunner(JavaRunnerFactory.java:85)
    at org.sonarsource.scanner.lib.internal.ScannerEngineLauncherFactory.createLauncher(ScannerEngineLauncherFactory.java:53)
    at org.sonarsource.scanner.lib.ScannerEngineBootstrapper.bootstrap(ScannerEngineBootstrapper.java:118)
    at org.sonarsource.scanner.cli.Main.analyze(Main.java:75)
    at org.sonarsource.scanner.cli.Main.main(Main.java:63)
22:44:23.925 ERROR
22:44:23.926 ERROR Re-run SonarScanner CLI using the -X switch to enable full debug logging.

C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>
```

Step 4: Open Jenkins and create a pipeline and name the pipeline SonarQube Pipeline and then click on okay.

## Enter an item name

SonarQube Pipeline

&gt; Required field



## Freestyle project

Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.



## Maven project

Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.



## Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



## Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.



## Folder

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.



## Branch Pipeline

OK Create a set of Pipeline projects according to detected branches in one SCM repository.

Step 5: In the configuration, under the Pipeline Section write the following Pipeline Script -  
node {

```
stage('Cloning the GitHub Repo') {
    git 'https://github.com/shazforiot/MSBuild_firstproject.git'
}

stage('SonarQube analysis') {
    withSonarQubeEnv('sonarqube') {
        bat "C:/sonar-scanner/sonar-scanner-6.2.0.4584-windows-x64/bin/sonar-scanner.bat \
            -D sonar.login=admin \
            -D sonar.password=sonarqube \
            -D sonar.projectKey=sonarqube-test \
            -D sonar.exclusions=vendor/**,resources/**,*/*.java \
            -D sonar.host.url=http://127.0.0.1:9000/"
    }
}
```

Then click on the save button.

**Configure**

General

Advanced Project Options

Pipeline

**Pipeline****Definition**

Pipeline script

**Script** ?

```

1+ node {
2+   stage('Cloning the GitHub Repo') {
3     git 'https://github.com/shafioriot/MSBuild_firstproject.git'
4   }
5+
6+   stage('SonarQube analysis') {
7     withSonarQubeEnv('sonarqube') {
8       bat "C:/sonar-scanner/sonar-scanner-6.2.0.4584-windows-x64/bin/sonar-scanner.bat" \
9           -D sonar.login=admin \
10          -D sonar.password=sonarqube \
11          -D sonar.projectKey=sonarqube-test \
12          -D sonar.host.url=http://127.0.0.1:9000 \
13          -D sonar.host.url=http://127.0.0.1:9000"
14     }
15   }
16 }
```

 Use Groovy Sandbox ?

Pipeline Syntax

**Save****Apply**

REST API Jenkins 2.452.3

**Step 6:** Now, click on Build Now and the build is successful.

**Jenkins**

Dashboard > SonarQube Pipeline >

**SonarQube Pipeline**

Status Changes Build Now Configure Delete Pipeline Full Stage View SonarQube Stages Rename Pipeline Syntax Add description Disable Project

**Stage View**

Average stage times: (Average full run time: ~22s)

|                  | Cloning the GitHub Repo | SonarQube analysis |
|------------------|-------------------------|--------------------|
| #10 Sep 25 21:55 | No Changes              | 1s 20s             |
| #9 Sep 25 21:52  | No Changes              | 19s 549ms failed   |
| #8 Sep 25 21:43  | No Changes              | 1s 1s failed       |
| #7 Sep 25 21:33  | No Changes              | 1s 957ms failed    |

Build History trend ▾ Filter... #10 Sep 25, 2024, 9:55 PM #9 Sep 25, 2024, 9:52 PM #8

```

21:56:16.244 INFO ----- Run sensors on project
21:56:16.428 INFO Sensor C# [csharp]
21:56:16.429 WARN Your project contains C# files which cannot be analyzed with the scanner you are using. To analyze C# or VB.NET, you must use the SonarScanner for .NET 5.x or higher, see https://redirect.sonarsource.com/doc/install-configure-scanner-msbuild.html
21:56:16.429 INFO Sensor C# [csharp] (done) | time=1ms
21:56:16.430 INFO Sensor Analysis Warnings import [csharp]
21:56:16.432 INFO Sensor Analysis Warnings import [csharp] (done) | time=2ms
21:56:16.432 INFO Sensor C# File Caching Sensor [csharp]
21:56:16.432 WARN Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir' property.
21:56:16.432 INFO Sensor C# File Caching Sensor [csharp] (done) | time=1ms
21:56:16.433 INFO Sensor Zero Coverage Sensor
21:56:16.450 INFO Sensor Zero Coverage Sensor (done) | time=16ms
21:56:16.494 INFO CPU Executor Calculating CPD for 0 files
21:56:16.494 INFO CPU Executor CPD calculation finished (done) | time=0ms
21:56:16.530 INFO SCM revision ID 'f2bc042c04c6e72427c380bcaeedd6fee7b49adf'
21:56:16.704 INFO Analysis report generated in 178ms, dir size=200.5 kB
21:56:16.773 INFO Analysis report compressed in 68ms, zip size=21.9 kB
21:56:16.930 INFO Analysis report uploaded in 155ms
21:56:16.931 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://127.0.0.1:9000/dashboard?id=sonarqube-test
21:56:16.931 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
21:56:16.932 INFO More about the report processing at http://127.0.0.1:9000/api/ce/task?id=af67fb15-719a-4b23-8f38-5edc7a765dae
21:56:16.940 INFO Analysis total time: 16.723 s
21:56:16.943 INFO SonarScanner Engine completed successfully
21:56:17.042 INFO EXECUTION SUCCESS
21:56:17.044 INFO Total time: 19.574s
[Pipeline]
[Pipeline] // withSonarQubeEnv
[Pipeline]
[Pipeline] // stage
[Pipeline]
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

Step 7: Now, visit <http://127.0.0.1:9000/dashboard?id=sonarqube-test> to see the result.

The screenshot shows the SonarQube dashboard for the 'sonarqube-test' project. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. The main content area features a large green 'Passed' status indicator with a checkmark icon. Below this, there are several cards providing detailed analysis metrics:

- Quality Gate**: Status is Passed.
- New Code**: 0 Open issues (A).
- Overall Code**: 0 Open issues (A).
- Security**: 0 Open issues (A). Sub-sections show 0 H, 0 M, 0 L.
- Reliability**: 0 Open issues (A). Sub-sections show 0 H, 0 M, 0 L.
- Maintainability**: 0 Open issues (A). Sub-sections show 0 H, 0 M, 0 L.
- Accepted issues**: 0 (Valid issues that were not fixed).
- Coverage**: 0.0% (On 0 lines to cover).
- Duplications**: 0.0% (On 86 lines).
- Security Hotspots**: 0 (A).

At the bottom right of the dashboard, there are links for Project Settings and Project Information.

127.0.0.1:9000/dashboard?id=sonarqube-test&codeScope=overall

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More Q

sonarqube-test / main ?

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

Learn how to improve your code base by cleaning only new code.

Take the Tour Not now

Quality Gate Passed Last analysis 45 minutes ago

The last analysis has warnings. See details

New Code Overall Code

| Security   | Reliability                        | Maintainability                         |
|--|------------------------------------|---|
| 0 Open issues<br>0 H 0 M 0 L                             | 68k Open issues<br>0 H 47k M 21k L | 164k Open issues<br>7 H 143k M 21k L    |
| Accepted issues<br>0<br>Valid issues that were not fixed | Coverage<br>On 0 lines to cover.   | Duplications<br>50.6%<br>On 759k lines. |

Security Hotspots

## **EXPERIMENT NO. 9**

**NAME :MOHIT PATIL CLASS : D15A ROLL NO. : 37**

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

Theory:

What is Nagios?

Nagios is an open-souíce monitoíng tool designed to monitoí systems, netwoíks, and infíastíuctuíe. It helps oíganizations identify and íesolve II' infíastíuctuíe issues befoíe they affect cítical business píocesses. Nagios píovides monitoring and aleítng services for service, switches, applications, and seívices.

Key features of Nagios

1. Monitoring: Nagios can monitoí a wide íange of netwoík seívices (HTTP, SMTP, POP3, etc.), host íesouíces (processor load, disk usage, system logs, etc.), and enviíonmental factoís (tempeíatuíe, humidity, etc.).
2. Aleítng: When an issue is detected, Nagios can send aleíts via email, SMS, or custom scíipts to notify administíatoís.
3. Repoíting: Nagios píovides detailed íepoíts and logs of outages, events, notifications, and aleít íesponses, helping in histoíical analysis and SLA compliance.
4. Scalability: Nagios is designed to scale and can monitoí laíge, complex enviíonments.
5. Ílexibility: With a wide íange of plugins and add-ons, Nagios can be customized to meet specific monitoíng needs.

How Nagios Woíks

1. Configuáation: Administíatoís configuíe Nagios to monitoí specific seívices and hosts. This involves defining what to monitoí, how to monitoí it, and what actions to take when issues aíe detected.
2. Plugins: Nagios uses plugins to gatheí infoíimation about the status of vaíious seívices and hosts. These plugins can be custom scíipts or píe-built ones available in the Nagios community.
3. Scheduling: Nagios schedules íegulaí checks of the defined seívices and hosts using the configuíed plugins.
4. Alerting: If a check indicates a píoblem, Nagios tíiggeí an aleít. Aleíts can be configuíed to escalate if not acknowledged within a ceítain timefíame.
5. Web Inteíface: Nagios píovides a web inteíface foí viewing the status of monitoíed seívices and hosts, acknowledging aleíts, and geneíating íepoíts.

## Continuous Monitoring

Continuous monitoring is a process that involves constantly tracking and analyzing the performance and security of IT systems. This practice is crucial for identifying and responding to issues in real-time, ensuring system reliability, and maintaining security.

Key benefits include:

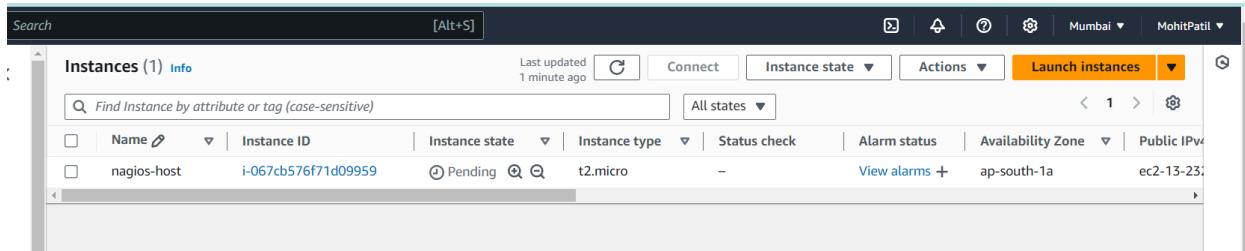
- Real-time insights into system performance.
- Early detection of issues to prevent downtime.
- Enhanced security through continuous threat detection.
- Improved compliance with regulatory standards.

## Setting Up Nagios

1. Installation: Install Nagios on a server, typically a Linux-based system.
2. Configuration files: Edit configuration files to define what to monitor and how to monitor it. This includes defining hosts, services, contacts, and notification methods.
3. Plugins: Install and configure necessary plugins to monitor specific services and hosts.
4. Web Interface: Set up the web interface to allow easy access to monitoring data and alert management.
5. Testing: Test the configuration to ensure that Nagios is correctly monitoring the defined services and hosts and that alerts are being sent as expected.

## Implementation :

1. Create an Amazon Linux EC2 Instance
  - Name it nagios-host.



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 AWS CloudFormation console with the 'Security group rule ID' table. The table has columns for Security group rule ID, Type, Protocol, Port range, Source, and Description - optional. A row for port 5666 is selected, showing the details: Type: SSH, Protocol: TCP, Port range: 5666, Source: Anywhere (0.0.0.0/0). The 'Delete' button is visible for each row.

### 3. Connect to Your EC2 Instance

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

The screenshot shows an Amazon Linux 2023 terminal window. The URL <https://aws.amazon.com/linux/amazon-linux-2023> is displayed in the terminal window.

### 4. Update Package Indices and Install Required Packages

Commands -

`sudo yum update`

`sudo yum install http 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
```

```
[ec2-user@ip-172-31-80-22 ~]$ sudo useradd nagios
[ec2-user@ip-172-31-80-22 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

## 6. Create a New User Group

Commands -

```
sudo groupadd nagcmd
```

```
[ec2-user@ip-172-31-80-22 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-80-22 ~]$
```

## 7. Add Users to the Group

Commands -

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

```
[ec2-user@ip-172-31-80-22 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
```

## 8. Create a Directory for Nagios Downloads

Commands -

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

```
[ec2-user@ip-172-31-80-22 ~]$ mkdir ~/downloads
cd ~/downloads
[ec2-user@ip-172-31-80-22 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
```

## 9. Download Nagios and Plugins Source Files

Commands -

```
Wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz wget
https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
```

```
[ec2-user@ip-172-31-80-22 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
--2024-09-26 08:56:36-- https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00::f03c:92ff:fe7:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11333414 (11M) [application/x-gzip]
Saving to: 'nagios-4.4.6.tar.gz'

nagios-4.4.6.tar.gz          100%[=====]   10.81M  12.6MB/s    in 0.9s

2024-09-26 08:56:37 (12.6 MB/s) - 'nagios-4.4.6.tar.gz' saved [11333414/11333414]

--2024-09-26 08:56:37-- https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2782610 (2.7M) [application/x-gzip]
Saving to: 'nagios-plugins-2.3.3.tar.gz'
```

## 10. Extract the Nagios Source file Commands -

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

```
[ec2-user@ip-172-31-80-22 downloads]$ tar xzvf 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
```

## 11. Run the Configuration Script Commands -

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

```
[ec2-user@ip-172-31-80-22 nagios-4.4.6]$ ./configure --with-command-group=nagcmd
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
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 whether make sets $(MAKE)... yes
checking whether ln -s works... yes
checking for strip... /usr/bin/strip
checking how to run the C preprocessor... gcc -E
```

## 12. Compile the Souice Code Commands - `make all`

```
[ec2-user@ip-172-31-80-22 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 -DSCORE -c -o nagios.o nagios.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o broker.o broker.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o nebmods.o nebmods.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o ../common/shared.o ../common/shared.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o query-handier.o query-handier.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -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 -DSCORE -c -o checks.o checks.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o config.o config.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o commands.o commands.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o events.o events.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o flapping.o flapping.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o logging.o logging.c
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DSCORE -c -o macros-base.o ../common/macros.c
```

## 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
```

```
*** 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>

```
*****
```

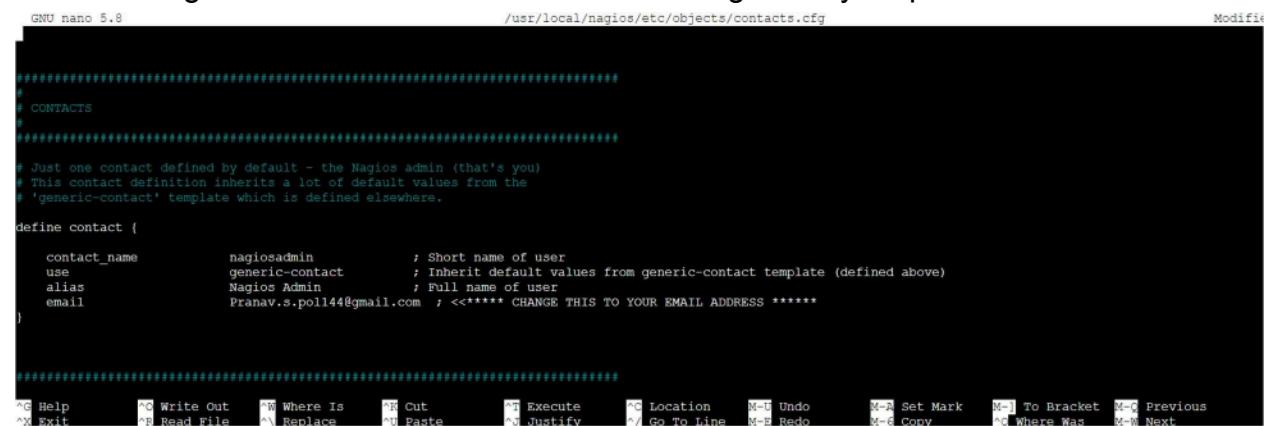
Enjoy.

## 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.



```
GNU nano 5.8          /usr/local/nagios/etc/objects/contacts.cfg      Modified

#####
# CONTACTS
#
#####

# 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 generic-contact template (defined above)
    alias             Nagios Admin          ; Full name of user
    email             Pranav.s.poli44@gmail.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****

}

#####

^G Help      ^C Write Out     ^A Where Is      ^X Cut      ^T Execute      ^F Location      M-U Undo      M-C Set Mark      M-B To Bracket      M-Q Previous
^X Exit      ^R Read File     ^W Replace      ^V Paste      ^J Justify      ^G Go To Line      M-E Redo      M-S Copy      M-D Where Was      M-N Next
```

## 15. Configure the Web Interface

Commands -

```
sudo make install-webconf
```

```
[ec2-user@ip-172-31-80-22 nagios-4.4.6]$ sudo nano /usr/local/nagios/etc/objects/contacts.cfg
[ec2-user@ip-172-31-80-22 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-172-31-80-22 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-172-31-80-22 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
```

## 17. Restart Apache

Commands -

```
sudo systemctl restart httpd
```

```
[ec2-user@ip-172-31-80-22 nagios-4.4.6]$ sudo systemctl restart httpd
[ec2-user@ip-172-31-80-22 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-172-31-80-22 nagios-4.4.6]$ sudo systemctl restart httpd  
[ec2-user@ip-172-31-80-22 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/perlmods/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
```

## 19. Compile and Install Plugins

Commands -

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

```
[ec2-user@ip-172-31-80-22 nagios-plugins-2.3.3]$ ./configure --with-nagios-user=nagios --with-nagios-group=nagios  
make  
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  
checking for egrep... /usr/bin/grep -E
```

## 20. Staít 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-172-31-80-22 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: GPL

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.
```

## 21. Check the Status of Nagios

Commands -

```
sudo systemctl status nagios
```

```
[ec2-user@ip-172-31-80-22 nagios-plugins-2.3.3]$ 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 Thu 2024-09-26 09:51:34 UTC; 1min 34s ago
    Docs: https://www.Nagios.org/documentation
  Process: 68229 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
  Process: 68230 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 68231 (nagios)
   Tasks: 6 (limit: 1112)
  Memory: 2.3M
    CPU: 33ms
   CGroup: /system.slice/nagios.service
           ├─68231 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           ├─68232 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─68233 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─68234 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─68235 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           └─68236 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 26 09:51:31 ip-172-31-80-22.ec2.internal nagios[68231]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Sep 26 09:51:31 ip-172-31-80-22.ec2.internal nagios[68231]: qh: core query handler registered
Sep 26 09:51:31 ip-172-31-80-22.ec2.internal nagios[68231]: qh: echo service query handler registered
Sep 26 09:51:31 ip-172-31-80-22.ec2.internal nagios[68231]: qh: help for the query handler registered
Sep 26 09:51:31 ip-172-31-80-22.ec2.internal nagios[68231]: wproc: Successfully registered manager as @wproc with query handler
Sep 26 09:51:31 ip-172-31-80-22.ec2.internal nagios[68231]: wproc: Registry request: name=Core Worker 68234;pid=68234
Sep 26 09:51:31 ip-172-31-80-22.ec2.internal nagios[68231]: wproc: Registry request: name=Core Worker 68235;pid=68235
Sep 26 09:51:31 ip-172-31-80-22.ec2.internal nagios[68231]: wproc: Registry request: name=Core Worker 68233;pid=68233
```

## 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.

The screenshot shows the Nagios Core 4.4.6 web interface. At the top, a browser window displays a 'Sign in to access this site' dialog box. The URL in the address bar is 54.147.245.126/nagios. The dialog box contains fields for 'Username' and 'Password', and buttons for 'Sign in' and 'Cancel'. Below the dialog, the main Nagios dashboard is visible. The dashboard features a header with the Nagios Core logo and a message indicating the daemon is running with PID 68231. On the left, a sidebar menu includes sections for General (Home, Documentation), Current Status (Tactical Overview, Map (Legacy), Hosts, Services, Host Groups, Service Groups, Problems), Reports (Availability, Trends (Legacy), Alerts, History, Histogram (Legacy), Notifications, Event Log), and System (Comments, Downtime, Process Info, Performance Info, Scheduling Queue, Configuration). The main content area includes a 'Get Started' section with a bulleted list of steps, a 'Quick Links' section with links to various Nagios resources, and two empty boxes for 'Latest News' and 'Don't Miss...'. A copyright notice at the bottom states: 'Copyright © 2010-2020 Nagios Core Development Team and Community Contributors. Copyright © 1999-2009 Ethan Galstad. See the THANKS file for more information on contributors.'

## Conclusion:

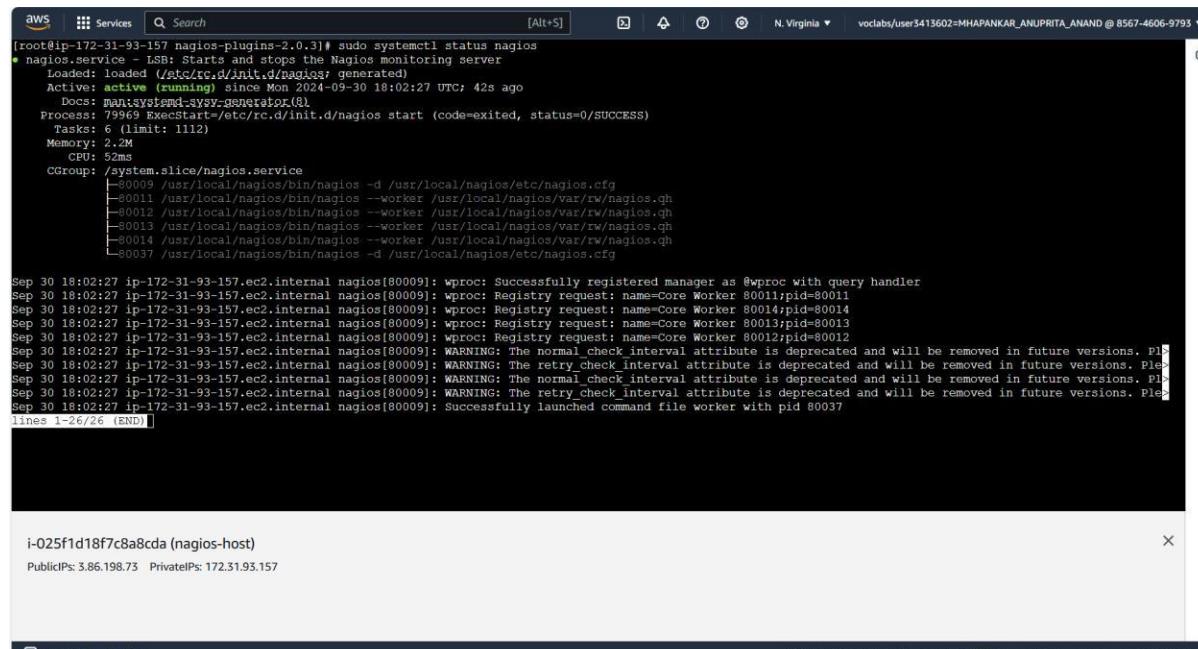
After installing and configuring Nagios Core, Plugins, and NRPE on a Linux machine, we have a robust continuous monitoring setup, ensuring proactive issue detection and optimal system performance.

## Experiment 10

MOHIT PATIL D15 37

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

Step 1: Initially confirm that Nagios is running on the server side. For this run the following command -  
sudo systemctl status nagios  
on the nagios-host instance.

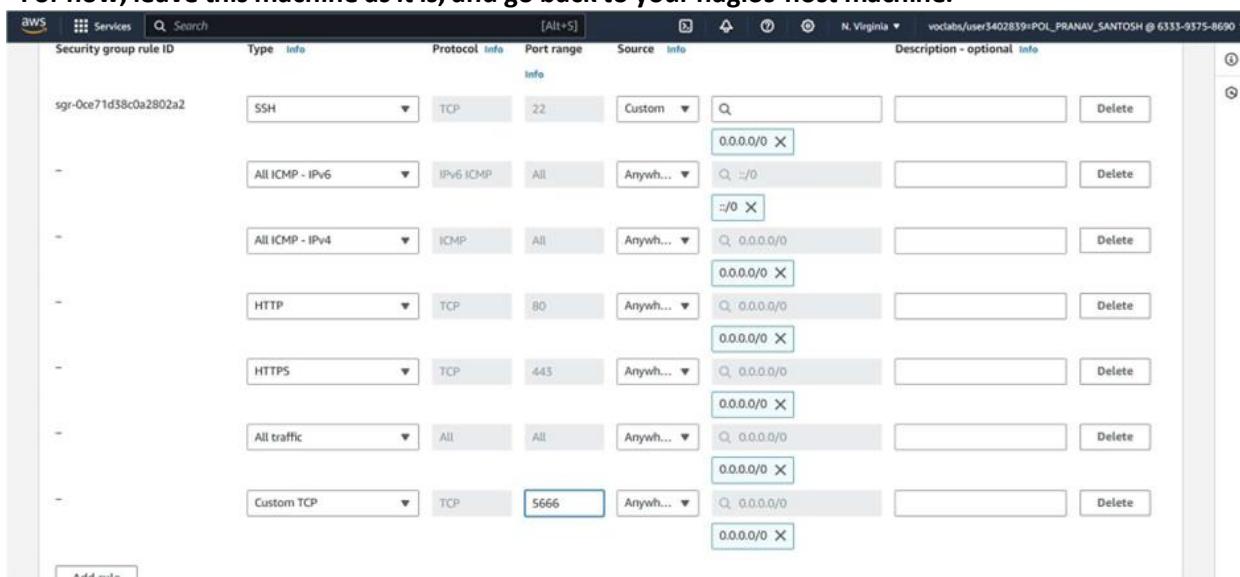


```
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# sudo systemctl status nagios
● nagios.service - LSB: Starts and stops the Nagios monitoring server
   Loaded: loaded (/etc/rc.d/init.d/nagios; generated)
   Active: active (running) since Mon 2024-09-30 18:02:27 UTC; 42s ago
     Docs: man:lspsvc(8) man:sysvrcv(8)
   Process: 79969 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
   Tasks: 6 (limit: 1112)
   Memory: 2.2M
      CPU: 52ms
     Cpuio: 0.00%
    Cgroup: /system.slice/nagios.service
            └─80009 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
              ├─80011 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              ├─80012 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              ├─80013 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              ├─80014 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              └─80037 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Successfully registered manager as @wproc with query handler
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80011;pid=80011
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80014;pid=80014
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80013;pid=80013
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80012;pid=80012
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please note that the default value is now 5 minutes.
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please note that the default value is now 1 minute.
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please note that the default value is now 5 minutes.
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: Successfully launched command file worker with pid 80037
lines 1-26/26 (END)
```

i-025f1d18f7c8a8cda (nagios-host)  
Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

Step 2: Once confirmed, make another instance with the same security group as that of nagios-host.  
For now, leave this machine as it is, and go back to your nagios-host machine.



| Security group rule ID | Type            | Protocol  | Port range | Source   | Description - optional |
|------------------------|-----------------|-----------|------------|----------|------------------------|
| sgr-Oce71d38c0a2802a2  | SSH             | TCP       | 22         | Custom   | 0.0.0.0/0              |
| -                      | All ICMP - IPv6 | IPv6 ICMP | All        | Anywh... | /:/0                   |
| -                      | All ICMP - IPv4 | ICMP      | All        | Anywh... | 0.0.0.0/0              |
| -                      | HTTP            | TCP       | 80         | Anywh... | 0.0.0.0/0              |
| -                      | HTTPS           | TCP       | 443        | Anywh... | 0.0.0.0/0              |
| -                      | All traffic     | All       | All        | Anywh... | 0.0.0.0/0              |
| -                      | Custom TCP      | TCP       | 5666       | Anywh... | 0.0.0.0/0              |

Step 3: Now run the following command -  
ps -ef | grep nagios

```

aws Services Search [Alt+S] N. Virginia v vclabs/user3413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9793 ▾
Active: active (running) since Mon 2024-09-30 18:02:27 UTC; 42s ago
  Docs: man:systemd-sysv-generator(8)
Process: 79969 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
Tasks: 6 (limit: 1112)
Memory: 2.2M
CPU: 52ms
CGroup: /system.slice/nagios.service
└─80009 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
   ├─80011 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
   ├─80012 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
   ├─80013 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
   ├─80014 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
   └─80037 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Successfully registered manager as #wproc with query handler
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80011;pid=80011
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80014;pid=80014
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80013;pid=80013
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: wproc: Registry request: name=Core Worker 80012;pid=80012
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The normal_check_interval attribute is deprecated and will be removed in future versions. Please see https://nagios-plugins.org/doc/troubleshooting.html#normal_check_interval_deprecated
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please see https://nagios-plugins.org/doc/troubleshooting.html#retry_check_interval_deprecated
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please see https://nagios-plugins.org/doc/troubleshooting.html#retry_check_interval_deprecated
Sep 30 18:02:27 ip-172-31-93-157.ec2.internal nagios[80009]: Successfully launched command file worker with pid 80037

[root@ip-172-31-93-157 nagios-plugins-2.0.3]# ps -ef | grep nagios
nagios 80009 1 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios 80011 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 80012 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 80013 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 80014 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 80037 80009 0 18:02 ? 0:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
root 81960 3110 0 18:35 pts/1 0:00:00 grep --color=auto nagios
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# 
```

Step 4: Now, run the following commands -

`sudo su`

`mkdir /usr/local/nagios/etc/objects/monitorhosts`

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

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

```

aws Services Search [Alt+S] N. Virginia v vclabs/user3413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9793 ▾
root@ip-172-31-93-157 nagios-plugins-2.0.3]# sudo su
root@ip-172-31-93-157 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts
root@ip-172-31-93-157 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
root@ip-172-31-93-157 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/localhost.cfg
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg: No such file or directory
cp: missing destination file operand after '/usr/local/nagios/etc/objects/localhost.cfg'
try 'cp --help' for more information.
root@ip-172-31-93-157 nagios-plugins-2.0.3]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
root@ip-172-31-93-157 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg: No such file or directory
root@ip-172-31-93-157 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
cp: missing destination file operand after '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg'
try 'cp --help' for more information.
root@ip-172-31-93-157 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
root@ip-172-31-93-157 nagios-plugins-2.0.3]# 
```

i-025f1d18f7c8a8cda (nagios-host)

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

Step 5: Open `linuxserver.cfg` using the the following command -

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

**Change the hostname to linuxserver (EVERYWHERE ON THE FILE)**

**Change address to the public IP address of your LINUX CLIENT.**

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

GNU nano 5.8 /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg Modified

```

# Example of how you can create configuration entries to monitor
# the local (Linux) machine.

#####
#####

HOST DEFINITION

#####
#####

# 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      linux-server
    alias          linux-server
    address        3.95.202.23
}

#####

^G Help      ^O Write Out   ^W Where Is   ^F Cut       ^I Execute   ^C Location   M-U Undo   M-A Set Mark   M-J To Bracket   M-K Previous
^X Exit      ^R Read File    ^\ Replace    ^U Paste     ^J Justify    ^Y Go To Line  M-E Redo   M-G Copy      ^Q Where Was    M-W Next

```

i-025f1d18f7c8a8cda (nagios-host)

Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

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

GNU nano 5.8 /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg Modified

```

check_command  check_local_swap!20!10
}

#####

# Define a service to check SSH on the local machine.
# Disable notifications for this service by default, as not all users may have SSH enabled.

define service{
    use            local-service          ; Name of service template to use
                                ; This service definition will inherit all variables that are defined
                                ; in (or inherited by) the local-service service template definition.

    host_name      linuxserver
    service_description  SSH
    check_command   check_ssh
    notifications_enabled  0
}

#####

# Define a service to check HTTP on the local machine.
# Disable notifications for this service by default, as not all users may have HTTP enabled.

define service{
    use            local-service          ; Name of service template to use
                                ; This service definition will inherit all variables that are defined
                                ; in (or inherited by) the local-service service template definition.

    host_name      linuxserver
    service_description  HTTP
    check_command   check_http
    notifications_enabled  0
}

#####

^G Help      ^O Write Out   ^W Where Is   ^F Cut       ^I Execute   ^C Location   M-U Undo   M-A Set Mark   M-J To Bracket   M-K Previous
^X Exit      ^R Read File    ^\ Replace    ^U Paste     ^J Justify    ^Y Go To Line  M-E Redo   M-G Copy      ^Q Where Was    M-W Next

```

i-025f1d18f7c8a8cda (nagios-host)

Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

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

Step 6: Open Nagios config file and add the following line -  
nano /usr/local/nagios/etc/nagios.cfg

Then add this line -  
cfg\_dir=/usr/local/nagios/etc/objects/monitorhosts/

```

AWS Services Search [Alt+S] N. Virginia vocabs/user3413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9793 ▾ Modified ⓘ
GNU nano 5.8
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg

# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
cfg_file=/usr/local/nagios/etc/objects/windows.cfg

# Definitions for monitoring a router/switch
cfg_file=/usr/local/nagios/etc/objects/switch.cfg

# Definitions for monitoring a network printer
cfg_file=/usr/local/nagios/etc/objects/printer.cfg

# You can also tell Nagios to process all config files (with a .cfg
# extension) in a particular directory by using the cfg_dir
# directive as shown below:

cfg_dir=/usr/local/nagios/etc/servers
cfg_dir=/usr/local/nagios/etc/printers
cfg_dir=/usr/local/nagios/etc/switches
cfg_dir=/usr/local/nagios/etc/routers

cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/[]

^G Help      ^O Write Out   ^W Where Is   ^R Cut        ^T Execute    ^C Location   M-U Undo    M-A Set Mark  M-J To Bracket M-V Previous
^X Exit      ^P Read File   ^\ Replace    ^U Paste      ^J Justify    ^Y Go To Line  M-E Redo    M-C Copy     M-Q Where Was  M-W Next

```

i-025f1d18f7c8a8cda (nagios-host)  
Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

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

## Step 8: Verify configuration files using the following command -

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

If there are no errors, run the following command -

`sudo service nagios start`

```

AWS Services Search [Alt+S] N. Virginia vocabs/user3413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9793 ▾ Modified ⓘ
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# nano /usr/local/nagios/etc/nagios.cfg
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# nano /usr/local/nagios/etc/nagios.cfg
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# /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: The normal check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
WARNING: The normal check_interval attribute is deprecated and will be removed in future versions. Please use check_interval instead.
WARNING: The retry_check_interval attribute is deprecated and will be removed in future versions. Please use retry_interval instead.
error: could not find any host matching 'linuxserver' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 45)
error: could not expand members specified in hostgroup (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 45)
Error processing object config files!

>>> One or more problems was encountered while processing the config files...

Check your configuration file(s) to ensure that they contain valid
directives and data definitions. If you are upgrading from a previous
version of Nagios, you should be aware that some variables/definitions
may have been removed or modified in this version. Make sure to read
the HTML documentation regarding the config files, as well as the
"What's New" section to find out what has changed.

[root@ip-172-31-93-157 nagios-plugins-2.0.3]# []

```

i-025f1d18f7c8a8cda (nagios-host)  
Public IPs: 3.86.198.73 Private IPs: 172.31.93.157

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

aws Services Search [Alt+S] N. Virginia v voclabs/user3413602=MHAPANKAR\_ANUPRITA\_ANAND @ 8567-4606-9793 ▾

```
[root@ip-172-31-93-157 nagios-plugins-2.0.3]# service nagios restart
Restarting nagios (via systemctl):
[ OK ]
```

i-025f1d18f7c8a8cda (nagios-host) X

PublicIPs: 3.86.198.73 PrivateIPs: 172.31.93.157

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

Step 9: After entering the correct credentials, you will see this page.

| Instances (1) <a href="#">Info</a> |                        | Last updated <span style="border: 1px solid #ccc; padding: 2px;">C</span> | Connect  | Instance state ▾ | Actions ▾    | Launch instances ▾            | Mumbai ▾          | MohitPatil ▾ |
|------------------------------------|------------------------|---|--|------------------|--------------|-------------------------------|-------------------|--------------|
|                                    |                        | <input type="text"/> Find Instance by attribute or tag (case-sensitive)   |  | All states ▾     |              |                               |                   |              |
|                                    | Name <a href="#">P</a> | Instance ID   | Instance state   | Instance type    | Status check | Alarm status                  | Availability Zone | Public IPv4  |
| <input type="checkbox"/>           | nagios-host            | i-067cb576f71d09959   | <span>Pending</span> <a href="#">Q</a> <a href="#">Q</a> | t2.micro         | -            | <a href="#">View alarms</a> + | ap-south-1a       | ec2-13-23-   |

Not secure 3.86.198.73/nagios/

# Nagios\*

**Current Network Status**  
Last Updated: Sep 30, 19 13:49 UTC 2024  
Updated every 90 seconds  
Nagios® Core™ 4.6.6 - www.nagios.org  
Logged in as: nagosadmin

**General**  
[Home](#)  
[Document](#)

**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](#)  
[Items \(Legacy\)](#)  
[Alerts](#)  
[History](#)  
[Summary](#)  
[Histogram \(Legacy\)](#)  
[Notifications](#)  
[Event Log](#)

**System**  
[Comments](#)  
[Downtime](#)  
[Process Info](#)  
[Performance Info](#)  
[Scheduling Queue](#)  
[Configuration](#)

**Host Status Totals**  
Up: 2 Down: 0 Unreachable: 0 Pending: 0  
All Problems: 0 All Types: 2

**Service Status Totals**  
Ok: 6 Warning: 1 Unknown: 0 Critical: 1 Pending: 8  
All Problems: 2 All Types: 16

**Host Status Details For All Host Groups**

| Host        | Status | Last Check          | Duration      | Status Information                        |
|-------------|--------|---------------------|---------------|---|
| linuxserver | UP     | 09-30-2024 19:13:16 | 0d 0h 0m 33s+ | PING OK - Packet loss = 0%, RTA = 1.82 ms |
| localhost   | UP     | 09-30-2024 19:01:49 | 0d 1h 11m 22s | PING OK - Packet loss = 0%, RTA = 0.04 ms |

Results 1 - 2 of 2 Matching Hosts

Page **1** / 1

# Experiment No 11

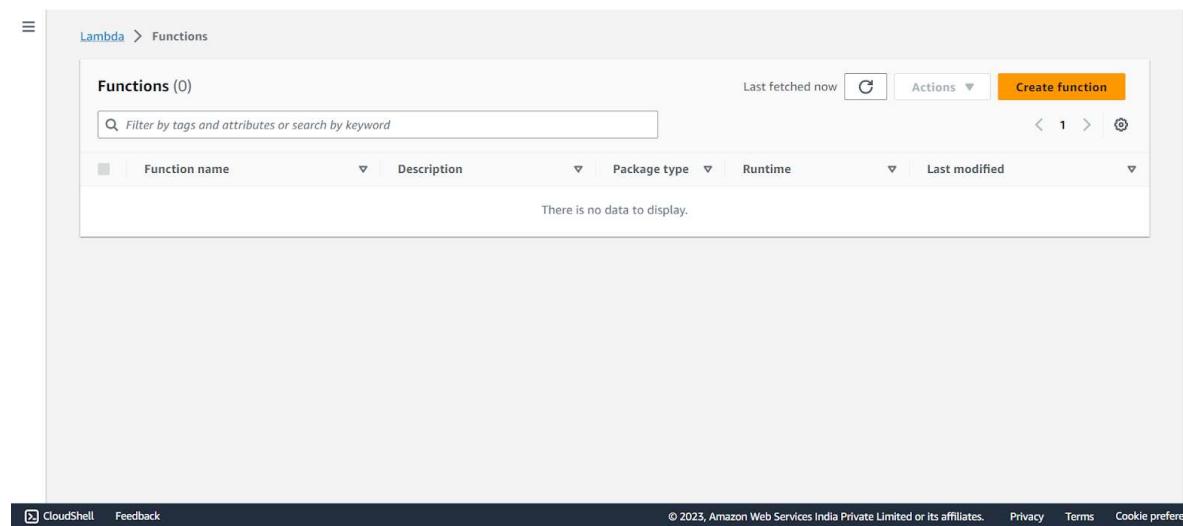
**Mohit Patil**  
**D15A 37**

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

## **Steps to create an AWS Lambda function**

Step 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 Functions page. At the top, there is a breadcrumb navigation: Lambda > Functions. Below the breadcrumb, there is a search bar labeled "Filter by tags and attributes or search by keyword". To the right of the search bar are buttons for "Last fetched now", "Actions", and a prominent orange "Create function" button. Below the search bar is a table header with columns: "Function name", "Description", "Package type", "Runtime", and "Last modified". A message "There is no data to display." is centered below the table. At the bottom of the page, there are links for "CloudShell", "Feedback", and copyright information: "© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences".

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.

Lambda > Functions > Create function

### Create function Info

AWS Serverless Application Repository applications have moved to [Create application](#).

Author from scratch  
Start with a simple Hello World example.

Use a blueprint  
Build a Lambda application from sample code and configuration presets for common use cases.

Container image  
Select a container image to deploy for your function.

#### Basic information

Function name  
Enter a name that describes the purpose of your function.  
`myFunctionName`

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info  
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.  
`Node.js 18.x`

Architecture Info  
Choose the instruction set architecture you want for your function code.  
 x86\_64

CloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Lambda > Functions > Create function Info

AWS Serverless Application Repository applications have moved to [Create application](#).

Author from scratch  
Start with a simple Hello World example.

Use a blueprint  
Build a Lambda application from sample code and configuration presets for common use cases.

Container image  
Select a container image to deploy for your function.

#### Basic information

Function name  
Enter a name that describes the purpose of your function.  
`myPythonLambdaFunction`

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info  
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.  
`Python 3.11`

Architecture Info  
Choose the instruction set architecture you want for your function code.  
 x86\_64

arm64

#### Permissions Info

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

https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/create/app... © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Lambda > Functions > Create function

### Create function Info

AWS Serverless Application Repository applications have moved to [Create application](#).

Author from scratch  
Start with a simple Hello World example.

Use a blueprint  
Build a Lambda application from sample code and configuration presets for common use cases.

Container image  
Select a container image to deploy for your function.

#### Basic information

Function name  
Enter a name that describes the purpose of your function.  
`myPythonLambdaFunction`

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info  
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.  
`Python 3.11`

Architecture Info  
Choose the instruction set architecture you want for your function code.  
 x86\_64

arm64

#### Permissions Info

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

Change default execution role

Advanced settings

Create function Cancel

Click on the Create button.

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

The screenshot shows the AWS Lambda Function Overview page for a function named "myPythonLambdaFunction". The top navigation bar includes "Lambda > Functions > myPythonLambdaFunction". Below the title, there's a "Function overview" section with a thumbnail of the function icon, a "Layers" section showing "(0)", and buttons for "+ Add trigger" and "+ Add destination". To the right, there are sections for "Description", "Last modified" (15 seconds ago), "Function ARN" (arn:aws:lambda:ap-south-1:447953971928:function:myPythonLambdaFunction), and "Function URL" (info). Below this, tabs for "Code", "Test", "Monitor", "Configuration", "Aliases", and "Versions" are visible. The "Code source" tab is selected, showing a code editor with the following Python code:

```
1 import json
2
3 def lambda_handler(event, context):
4     # TODO Implement
5     return {
6         'statusCode': 200,
7         'body': json.dumps('Hello from Lambda!')
8     }
```

This screenshot shows the same Lambda function overview page after changes have been made. The "Code source" tab is still selected, and the code editor now contains the following updated Python code:

```
1 import json
2
3 def lambda_handler(event, context):
4     # TODO Implement
5     return {
6         'statusCode': 200,
7         'body': json.dumps('Hello from Lambda!')
8     }
```

4. To change the configuration, open up the Configuration tab and under General Configuration, choose Edit.

Here, you can enter a description and change Memory and Timeout. I've changed the Timeout period to 1 sec since that is sufficient for now.

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

|                                 |  |                             |
|---------------------------------|--|-----------------------------|
| <b>General configuration</b>    | <b>General configuration</b> <small>Info</small> | <small>Edit</small>         |
| Triggers                        | Description<br>-                                 | Memory<br>128 MB            |
| Permissions                     | Timeout<br>0 min 3 sec                           | Ephemeral storage<br>512 MB |
| Destinations                    | SnapStart <small>Info</small><br>None            |                             |
| Function URL                    |  |                             |
| Environment variables           |  |                             |
| Tags                            |  |                             |
| VPC                             |  |                             |
| Monitoring and operations tools |  |                             |
| Concurrency                     |  |                             |
| Asynchronous invocation         |  |                             |
| Code signing                    |  |                             |
| Database proxies                |  |                             |
| File systems                    |  |                             |
| State machines                  |  |                             |

cloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

**Services**  [Alt+S]

Lambda > Functions > [myPythonLambdaFunction](#) > Edit basic settings

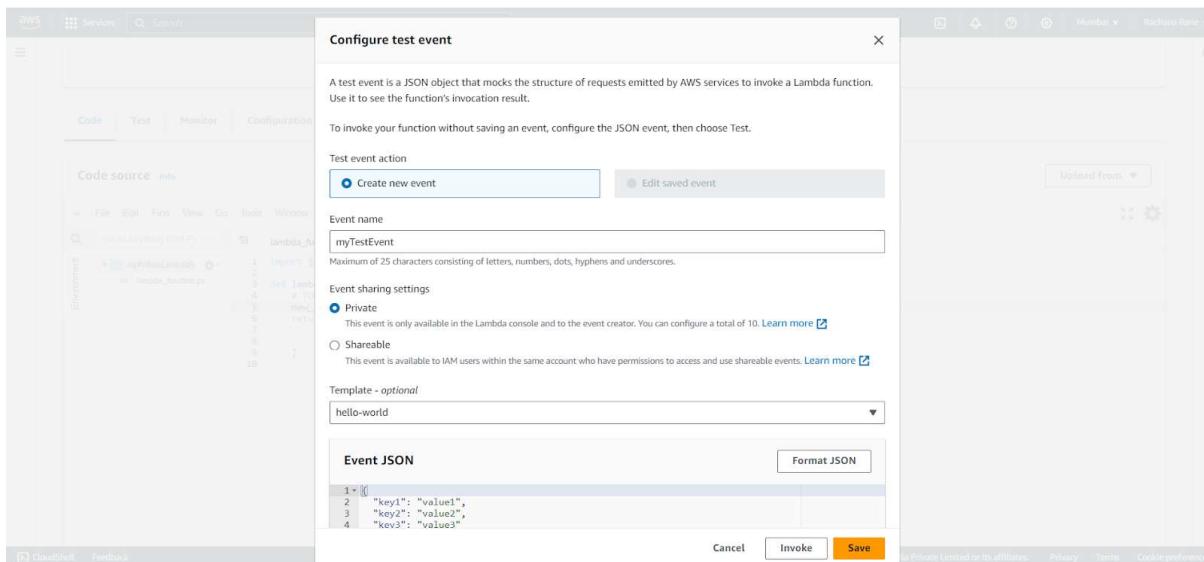
## Edit basic settings

|  |
|--|
| <b>Basic settings</b> <small>Info</small>  |
| Description – optional   |
| <input type="text"/>   |
| <b>Memory</b> <small>Info</small>  |
| Your function is allocated CPU proportional to the memory configured.  |
| <input type="text" value="128"/> MB  |
| Set memory to between 128 MB and 10240 MB  |
| <b>Ephemeral storage</b> <small>Info</small>   |
| You can configure up to 10 GB of ephemeral storage (/tmp) for your function. <a href="#">View pricing</a>  |
| <input type="text" value="512"/> MB  |
| Set ephemeral storage (/tmp) to between 512 MB and 10240 MB.   |
| <b>SnapStart</b> <small>Info</small>   |
| Reduce startup time by having Lambda cache a snapshot of your function after the function has initialized. To evaluate whether your function code is resilient to snapshot operations, review the <a href="#">SnapStart compatibility considerations</a> . |
| <input type="text" value="None"/>  |
| Supported runtimes: Java 11, Java 17.  |
| <b>Timeout</b>   |
| <input type="text" value="0"/> min <input type="text" value="1"/> sec  |
| <b>Execution role</b>  |

5. You can make changes to your function inside the code editor. You can also upload a zip file of your function or upload one from an S3 bucket if needed.  
Press Ctrl + S to save the file and click Deploy to deploy the changes.

```
import json
def lambda_handler(event, context):
    # TODO implement
    new_string="Hello! how are you?"
    return {
        'statusCode': 200,
        'body': json.dumps('Hello from Lambda!')
    }
```

6. Click on Test and you can change the configuration, like so. If you do not have anything in the request body, it is important to specify two curly braces as valid JSON, so make sure they are there.



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

The screenshot displays two side-by-side interfaces for testing and managing AWS Lambda functions.

**Top Interface (Test Results):**

- Header:** The test event myTestEvent was successfully saved.
- Navigation:** File, Edit, Find, View, Go, Tools, Window, Test (selected), Deploy, Changes not deployed.
- Environment:** myPythonLambdaFunction, lambda\_function.
- Execution result:** Status: Succeeded, Max memory used: 40 MB, Time: 1.66 ms.
- Test Event Name:** myTestEvent.
- Response:** { "statusCode": 200, "body": "\Hello from Lambda!" }
- Function Logs:** START RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc Version: \$LATEST  
END RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc  
REPORT RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc Duration: 1.66 ms Billed Duration: 2 ms Memory Size: 128 MB Max Memory Used: 40 MB Init Duration: 110.05 ms
- Request ID:** 7d26f404-f1da-4435-9faf-8dbb2a2733cc

**Bottom Interface (Code Source):**

- Header:** The test event myTestEvent was successfully saved.
- Navigation:** File, Edit, Find, View, Go, Tools, Window, Test (selected), Deploy, Changes not deployed.
- Environment:** myPythonLambdaFunction, lambda\_function.
- Execution results:** Status: Succeeded, Max memory used: 40 MB, Time: 1.66 ms.
- Test Event Name:** myTestEvent.
- Response:** { "statusCode": 200, "body": "\Hello from Lambda!" }
- Function Logs:** START RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc Version: \$LATEST  
END RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc  
REPORT RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc Duration: 1.66 ms Billed Duration: 2 ms Memory Size: 128 MB Max
- Request ID:** 7d26f404-f1da-4435-9faf-8dbb2a2733cc

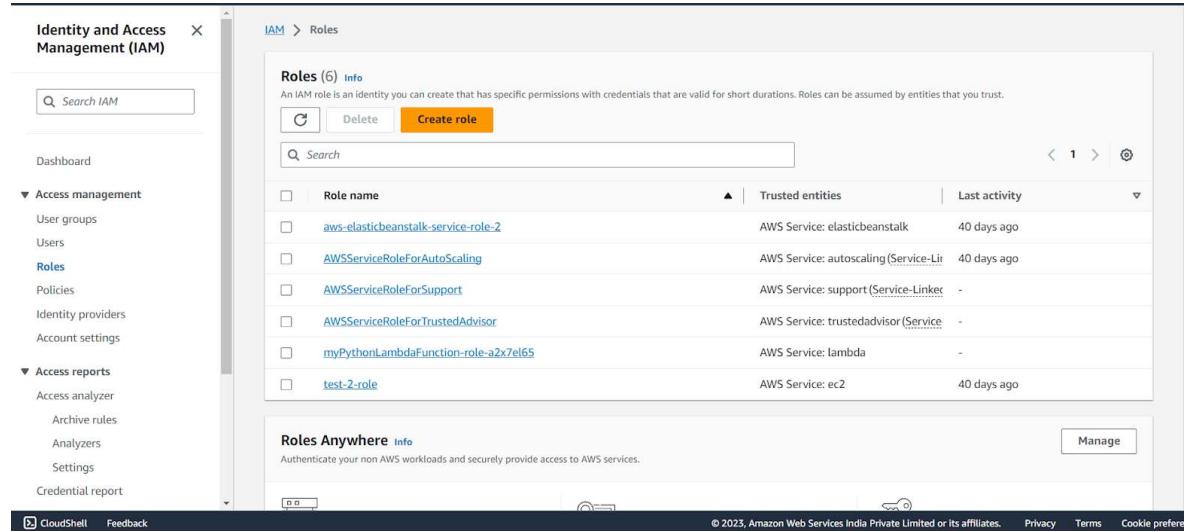
**Common Footer:** © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

**Conclusion:** Thus, we understood AWS Lambda, its workflow, various functions and created our first Lambda functions using Python / Java / Nodejs.

## Experiment No 12

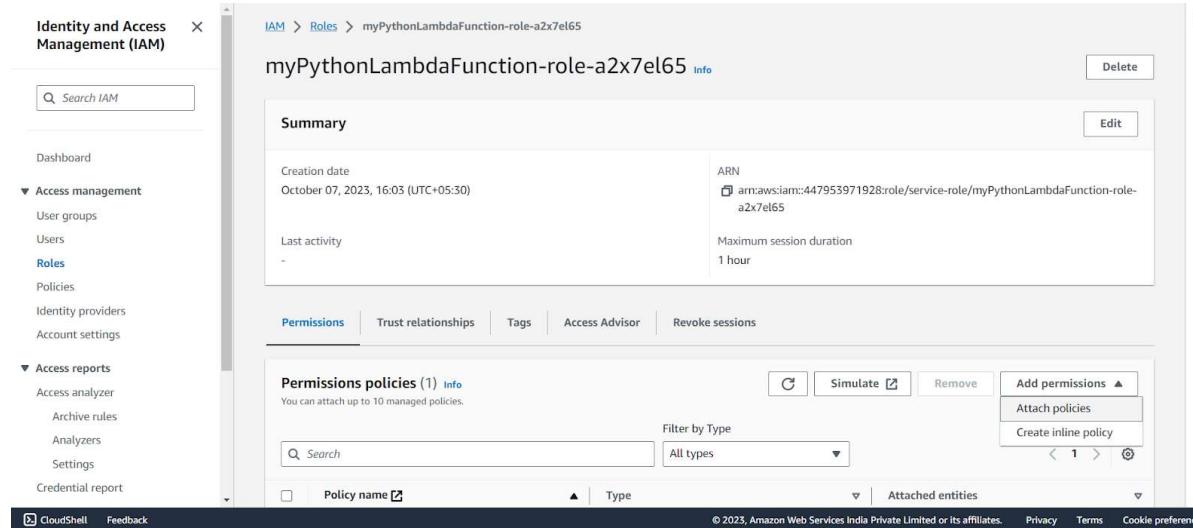
**Mohit Patil**  
**D15A 37**

Step 1: Open up the IAM Console and under Roles, choose the Role we previously created for the Python Lambda Function (You can find your role name configuration of your Lambda function).



The screenshot shows the AWS IAM Roles page. On the left, there's a navigation sidebar with options like Dashboard, Access management (User groups, Users, Roles, Policies, Identity providers, Account settings), Access reports (Access analyzer, Archive rules, Analyzers, Settings, Credential report), and CloudShell/Feedback. The main content area is titled 'Roles (6) Info' and contains a table with six rows. The columns are 'Role name', 'Trusted entities', and 'Last activity'. The roles listed are: 'aws-elasticbeanstalk-service-role-2' (AWS Service: elasticbeanstalk, 40 days ago), 'AWSServiceRoleForAutoScaling' (AWS Service: autoscaling (Service-Linker), 40 days ago), 'AWSServiceRoleForSupport' (AWS Service: support (Service-Linker), -), 'AWSServiceRoleForTrustedAdvisor' (AWS Service: trustedadvisor (Service-Linker), -), 'myPythonLambdaFunction-role-a2x7el65' (AWS Service: Lambda, -), and 'test-2-role' (AWS Service: ec2, 40 days ago). Below the table, there's a section titled 'Roles Anywhere' with a 'Manage' button.

Step 2: Under Attach Policies, add S3-ReadOnly and CloudWatchFull permissions to this role.



The screenshot shows the detailed view of the 'myPythonLambdaFunction-role-a2x7el65' role. The left sidebar is identical to the previous screenshot. The main page title is 'myPythonLambdaFunction-role-a2x7el65 Info'. It has tabs for Summary (Edit button), Permissions (selected), Trust relationships, Tags, Access Advisor, and Revoke sessions. In the 'Permissions' tab, there's a sub-section 'Permissions policies (1) Info' with a note 'You can attach up to 10 managed policies.' A search bar and filter dropdown ('All types') are present. A modal window is open over the table, titled 'Add permissions ▲', with two options: 'Attach policies' and 'Create inline policy'. The table below shows one policy attached: 'Policy name' (S3-ReadOnly), 'Type' (Managed policy), and 'Attached entities' (myPythonLambdaFunction-role-a2x7el65). The bottom of the page includes standard AWS footer links: © 2023, Amazon Web Services India Private Limited or its affiliates., Privacy, Terms, and Cookie preferences.

**S3-ReadOnly**

The screenshot shows the 'Add permissions' dialog in the AWS IAM console. The path is IAM > Roles > myPythonLambdaFunction-role-a2x7el65 > Add permissions. The title is 'Attach policy to myPythonLambdaFunction-role-a2x7el65'. The 'Current permissions policies (1)' section is expanded, showing one policy: 'AmazonS3ReadOnlyAccess'. The 'Other permissions policies (882)' section is expanded, showing two policies: 'CloudWatchFullAccess' and 'CloudWatchFullAccessV2'. Both are AWS managed policies. A search bar at the top filters results by type ('All types'). Buttons for 'Cancel' and 'Add permissions' are at the bottom right.

## CloudWatchFull

This screenshot is identical to the previous one, but it includes a success message at the top: 'Policy was successfully attached to role.' The rest of the interface is the same, showing the 'CloudWatchFullAccess' policy attached to the role.

After successful attachment of policy you will see something like this you will be able to see the updated policies.

The screenshot shows the 'Permissions' tab in the AWS IAM console for the 'myPythonLambdaFunction-role-a2x7el65' role. The success message 'Policy was successfully attached to role.' is still present. The 'Permissions policies (3)' section shows three policies: 'AmazonS3ReadOnlyAccess', 'AWSLambdaBasicExecutionRole-c4946a...', and 'CloudWatchFullAccess'. The 'Attached entities' column shows that each policy is attached to 1 entity. A 'Permissions boundary (not set)' section is also visible.

### Step 3: Open up AWS Lambda and create a new Python function.

Lambda > Functions > Create function

Create function [Info](#)  
AWS Serverless Application Repository applications have moved to [Create application](#).

Author from scratch  
Start with a simple Hello World example.

Use a blueprint  
Build a Lambda application from sample code and configuration presets for common use cases.

Container image  
Select a container image to deploy for your function.

**Basic information**

Function name  
Enter a name that describes the purpose of your function.

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime [Info](#)  
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.  
 [C](#)

Architecture [Info](#)  
Choose the instruction set architecture you want for your function code.  
 x86\_64  
 arm64

Permissions [Info](#)  
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

[CloudShell](#) [Feedback](#)

© 2023, Amazon Web Services India Private Limited or its affiliates. [Privacy](#) [Terms](#) [Cookie preference](#)

Under Execution Role, choose the existing role, then select the one which was previously created and to which we just added permissions.

Architecture [Info](#)  
Choose the instruction set architecture you want for your function code.  
 x86\_64  
 arm64

Permissions [Info](#)  
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▼ Change default execution role

Execution role  
Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).  
 Create a new role with basic Lambda permissions  
 Use an existing role  
 Create a new role from AWS policy templates

Existing role  
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.  
 [C](#)

View the [myPythonLambdaFunction-role-a2x7el65 role](#) on the IAM console.

► Advanced settings

[Cancel](#) [Create function](#)

[CloudShell](#) [Feedback](#)

© 2023, Amazon Web Services India Private Limited or its affiliates. [Privacy](#) [Terms](#) [Cookie preference](#)

Step 4: The function is up and running.

The screenshot shows the AWS Lambda Function Overview page for a function named "AdvDevops-ex12". The top bar indicates success: "Successfully created the function AdvDevops-ex12. You can now change its code and configuration. To invoke your function with a test event, choose 'Test'." Below this, the function name "AdvDevops-ex12" is displayed along with a "Layers" section showing "(0)". There are buttons for "Add trigger" and "Add destination". On the right, there's a "Description" field with a minus sign, "Last modified 6 seconds ago", "Function ARN" (arn:aws:lambda:ap-south-1:447953971928:function:AdvDevops-ex12), and a "Function URL" link. At the bottom, tabs for "Code", "Test", "Monitor", "Configuration", "Aliases", and "Versions" are visible, along with CloudShell and Feedback links.

Step 5: Make the following changes to the function and click on the deploy button. This code basically logs a message and logs the contents of a JSON file which is uploaded to an S3 Bucket and then deploy the code.

The screenshot shows the AWS Lambda Code Editor for the "lambda\_function" file of the "AdvDevops-ex12" function. The editor displays the following Python code:

```
1 import json
2 import boto3
3 import urllib
4
5 def lambda_handler(event, context):
6
7     s3_client = boto3.client('s3')
8     bucket_name = event['Records'][0]['s3']['bucket']['name']
9     key = event['Records'][0]['s3']['object']['key']
10    key_urlib.parse.unquote_plus(key, encoding='utf-8')
11    message = 'A file has been added with key ' + key + ' to the bucket ' + bucket_name
12    print(message)
13    response = s3_client.get_object(Bucket=bucket_name, Key=key)
14    contents = response['Body'].read().decode()
15    contents = json.loads(contents)
16
17    print("These are the Contents of the File: \n", contents)
18
```

The interface includes tabs for "Environment", "lambda\_function", and "Environment Var". At the bottom, it shows "CloudShell", "Feedback", and copyright information: "© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences".

Step 6: Click on Test and choose the 'S3 Put' Template.

Screenshot of the AWS Lambda console showing the creation of a new function named "AdvDevops-ex12".

The "Code" tab is selected, displaying the code source in a code editor:

```
1 import json
2 import boto3
3 import urllib
4
5 def lambda_handler(event, context):
```

A modal window titled "Configure test event" is open, showing the configuration for a test event:

- Test event action:** Create new event (selected)
- Event name:** test
- Event sharing settings:** Private (selected)
- Template - optional:** s3-put
- Event JSON:** (Empty field)

Buttons at the bottom of the modal include: Cancel, Invoke, and Save.

And Save it.

Step 7: Open up the S3 Console and create a new bucket.

Amazon S3

▶ Account snapshot  
Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

Buckets (3) [Info](#)  
Buckets are containers for data stored in S3. [Learn more](#)

Find buckets by name

| Name                                     | AWS Region                       | Access                | Creation date                        |
|--|----------------------------------|-----------------------|--------------------------------------|
| elasticbeanstalk-ap-south-1-447953971928 | Asia Pacific (Mumbai) ap-south-1 | Objects can be public | August 7, 2023, 14:24:02 (UTC+05:30) |
| www.hellorachana.com                     | Asia Pacific (Mumbai) ap-south-1 | ⚠️ Public             | July 30, 2023, 15:05:34 (UTC+05:30)  |
| www.htmlwebsite.com                      | Asia Pacific (Mumbai) ap-south-1 | ⚠️ Public             | July 30, 2023, 15:49:06 (UTC+05:30)  |

CloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Step 8: With all general settings, create the bucket in the same region as the function.

Amazon S3 > Buckets > Create bucket

Create bucket [Info](#)  
Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name: AdvDevopsexp12  
Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

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

Copy settings from existing bucket - optional  
Only the bucket settings in the following configuration are copied.  
[Choose bucket](#)

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.

CloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Step 9: Click on the created bucket and under properties, look for events.

Event notifications (0)  
Send a notification when specific events occur in your bucket. [Learn more](#)

No event notifications  
Choose [Create event notification](#) to be notified when a specific event occurs.  
[Create event notification](#)

Amazon EventBridge  
For additional capabilities, use Amazon EventBridge to build event-driven applications at scale using S3 event notifications. [Learn more](#) or see [EventBridge pricing](#)

Send notifications to Amazon EventBridge for all events in this bucket  
Off

Transfer acceleration  
Use an accelerated endpoint for faster data transfers. [Learn more](#)

Transfer acceleration  
Disabled

CloudShell Feedback © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Click on Create Event Notification.

Step 10: Mention an event name and check Put under event types.

The screenshot shows the 'General configuration' section with an 'Event name' field containing 'S3putrequest'. Below it, there are optional fields for 'Prefix' (containing 'images/') and 'Suffix' (containing '.jpg'). The 'Event types' section shows 'Put' checked under 'Object creation'. At the bottom, there are links for 'CloudShell', 'Feedback', and copyright information: '© 2023, Amazon Web Services India Private Limited'.

Event name  
S3putrequest

Event name can contain up to 255 characters.

Prefix - optional  
Limit the notifications to objects with key starting with specified characters.  
images/

Suffix - optional  
Limit the notifications to objects with key ending with specified characters.  
.jpg

**Event types**  
Specify at least one event for which you want to receive notifications. For each group, you can choose an event type for all events, or you can choose one or more individual events.

**Object creation**

All object create events  
s3:ObjectCreated:  
 Put  
s3:ObjectCreated:Put

Post  
s3:ObjectCreated:Post

CloudShell Feedback © 2023, Amazon Web Services India Private Limited

Choose Lambda function as destination and choose your lambda function and save the changes.

The screenshot shows the 'Destination' section. A note says: 'Before Amazon S3 can publish messages to a destination, you must grant the Amazon S3 principal the necessary permissions to call the relevant API to publish messages to an SNS topic, an SQS queue, or a Lambda function.' Below, 'Lambda function' is selected as the destination. Under 'Specify Lambda function', 'Choose from your Lambda functions' is selected. The 'Lambda function' dropdown contains 'AdvDevops-ex12'. At the bottom are 'Cancel' and 'Save changes' buttons.

Before Amazon S3 can publish messages to a destination, you must grant the Amazon S3 principal the necessary permissions to call the relevant API to publish messages to an SNS topic, an SQS queue, or a Lambda function. [Learn more](#)

**Destination**  
Choose a destination to publish the event. [Learn more](#)

**Lambda function**  
Run a Lambda function script based on S3 events.

**SNS topic**  
Fanout messages to systems for parallel processing or directly to people.

**SQS queue**  
Send notifications to an SQS queue to be read by a server.

**Specify Lambda function**

**Choose from your Lambda functions**

Enter Lambda function ARN

**Lambda function**

AdvDevops-ex12

Cancel **Save changes**

Step 11: Refresh the Lambda function console and you should be able to see an S3 Trigger in the overview.

The screenshot shows the AWS Lambda Functions overview for the function 'AdvDevops-ex12'. In the 'Triggers' section, there is one entry for 'S3'. Below the triggers, there are buttons for 'Add destination' and 'Add trigger'. To the right, there are sections for 'Description', 'Last modified' (1 minute ago), 'Function ARN' (arn:aws:lambda:ap-south-1:447953971928:function:AdvDevops-ex12), and 'Function URL' (Info). At the bottom, there are tabs for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. A footer bar includes 'CloudShell', 'Feedback', 'Upload from', 'Privacy', 'Terms', and 'Cookie preference'.

Step 12: Now, create a dummy JSON file locally.

```
{ } dummy.json X
{ } dummy.json > ...
1   {
2     "firstname" : "Shashwat",
3     "lastname" : "Tripathi",
4     "gender" : "Male",
5     "age": 19
6 }
```

Step 13: Go back to your S3 Bucket and click on Add Files to upload a new file.

Step 14: Select the dummy data file from your computer and click Upload.

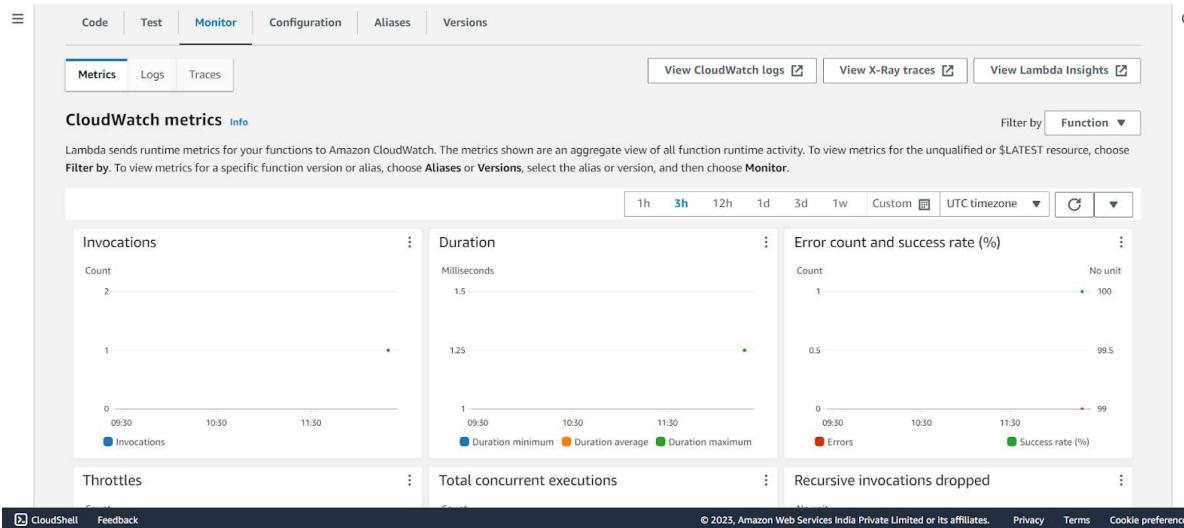
The screenshot shows the AWS S3 'Upload' interface. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, a search bar, and keyboard shortcut '[Alt+S]'. Below the navigation is a breadcrumb trail: 'Amazon S3 > Buckets > advopssexp12 > Upload'. The main title is 'Upload' with an 'Info' link. A note below says: '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](#)'.

A large dashed box area is labeled 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.' Below this is a table titled 'Files and folders (1 Total, 89.0 B)'. It contains one item: 'dummy.json' (application/json, 89.0 B). There are 'Remove', 'Add files', and 'Add folder' buttons above the table. A search bar 'Find by name' is also present. The 'Destination' section shows 'Destination s3://advopssexp12'. At the bottom, there are links for 'CloudShell' and 'Feedback', and a copyright notice: '© 2023, Amazon Web Services India Private Limited or its affiliates'.

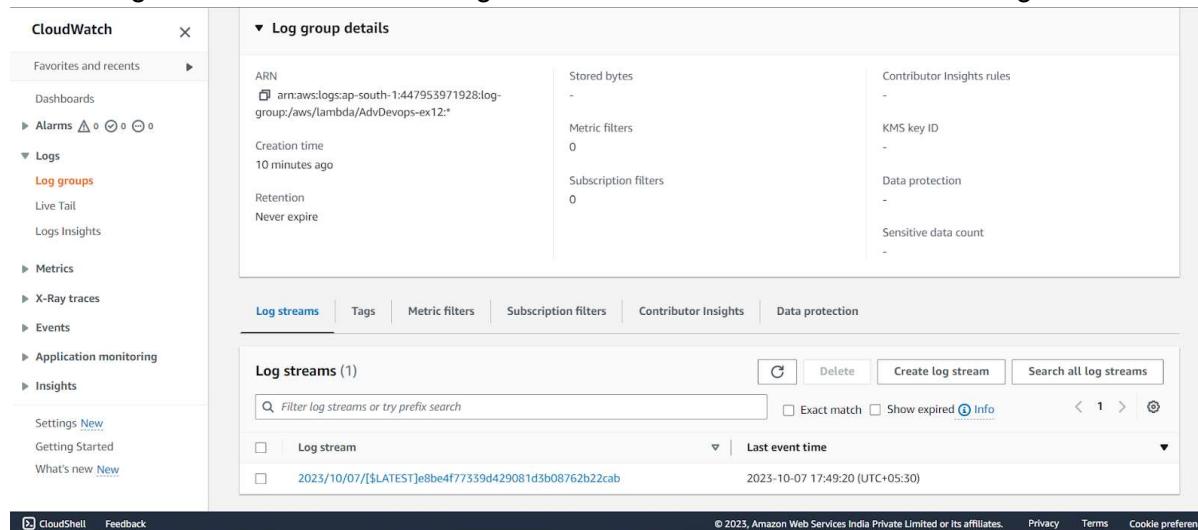
**Step 15:** After this make the necessary changes in the Test configuration file which we created it previously by replacing the Bucket Name and the ARN of Bucket.

The screenshot shows a JSON editor with the title 'Event JSON' and a 'Format JSON' button. The JSON code is a test event for a Lambda function. It includes line numbers from 10 to 38. The code defines a principalId ('EXAMPLE'), request parameters ('sourceIPAddress: "127.0.0.1"'), response elements ('x-amz-request-id: "EXAMPLE123456789", x-amz-id-2: "EXAMPLE123/5678abcdefghijklambdaisawesome/mnopqrstuvwxyzABCDEFGHIJ"'), and an S3 object ('s3: { s3SchemaVersion: "1.0", configurationId: "testConfigRule", bucket: { name: "advopssexp12", ownerIdentity: { principalId: "EXAMPLE" }, arn: "arn:aws:s3:::advopssexp12" }, object: { key: "test%2Fkey", size: 1024, eTag: "0123456789abcdef0123456789abcdef", sequencer: "0A1B2C3D4E5F678901" } } }').

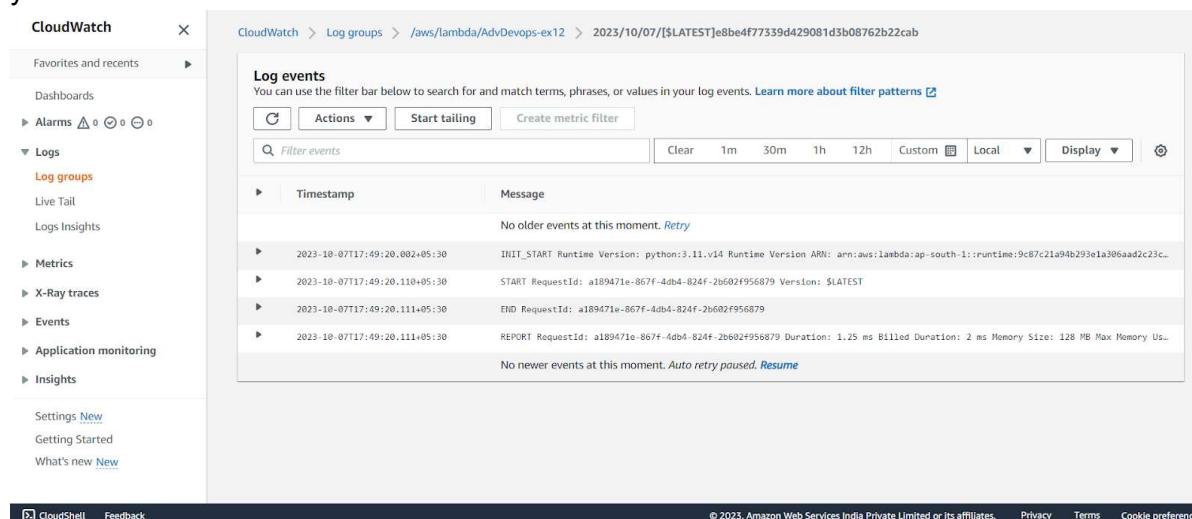
**Step 16:** Go back to your Lambda function , Refresh it and check the Monitor tab.



Under Log streams, click on View logs in Cloudwatch to check the Function logs.



Step 17: Click on this log Stream that was created to view what was logged by your function.



**Conclusion:** Thus, we have created a Lambda function which logs “An Image has been added” once you add an object to a specific bucket in S3.