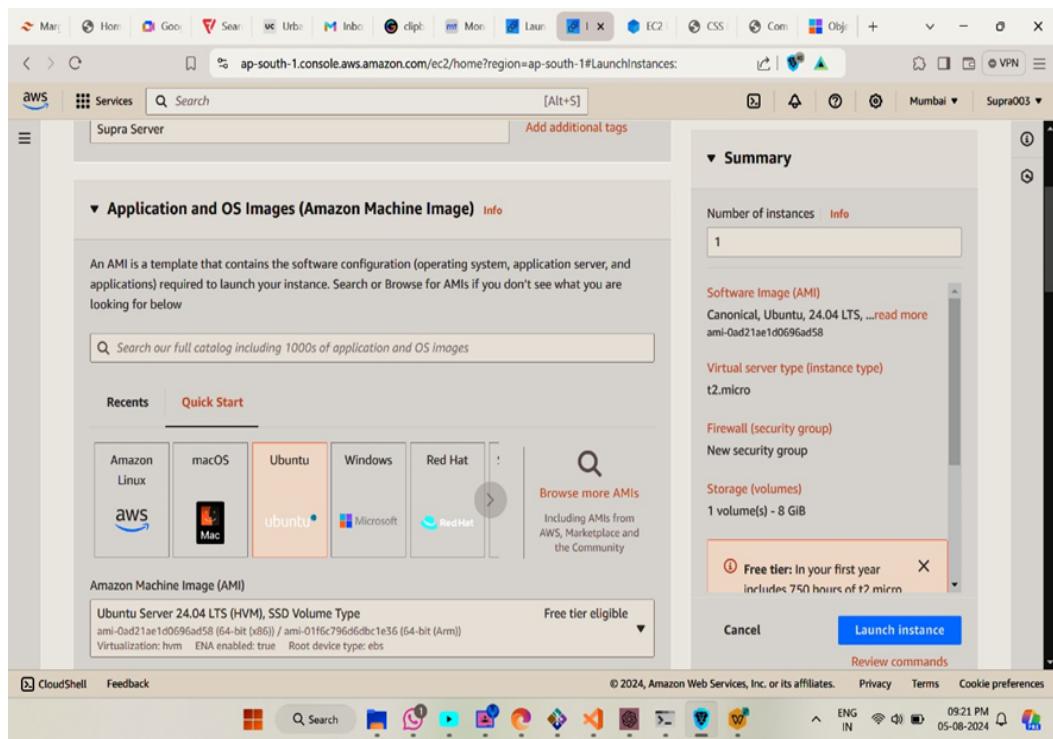


NAME-Laksh.V.Sodhai
CLASS-D-15A
ROLL.NO-59

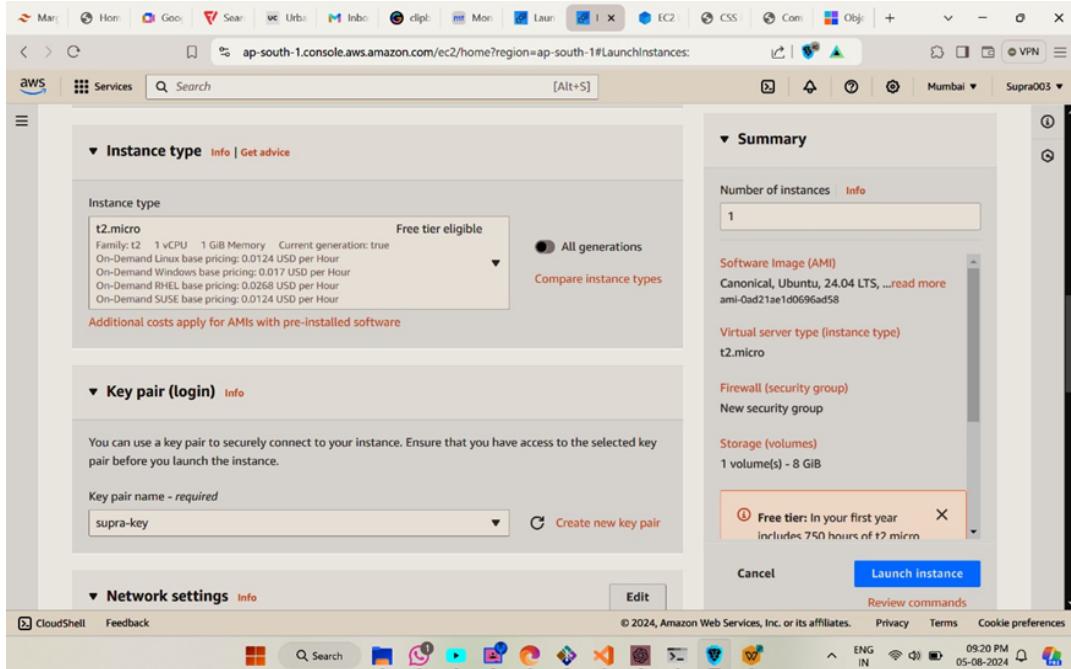
Experiment 01

- a)** To develop a website and host it on your local machine on a VM

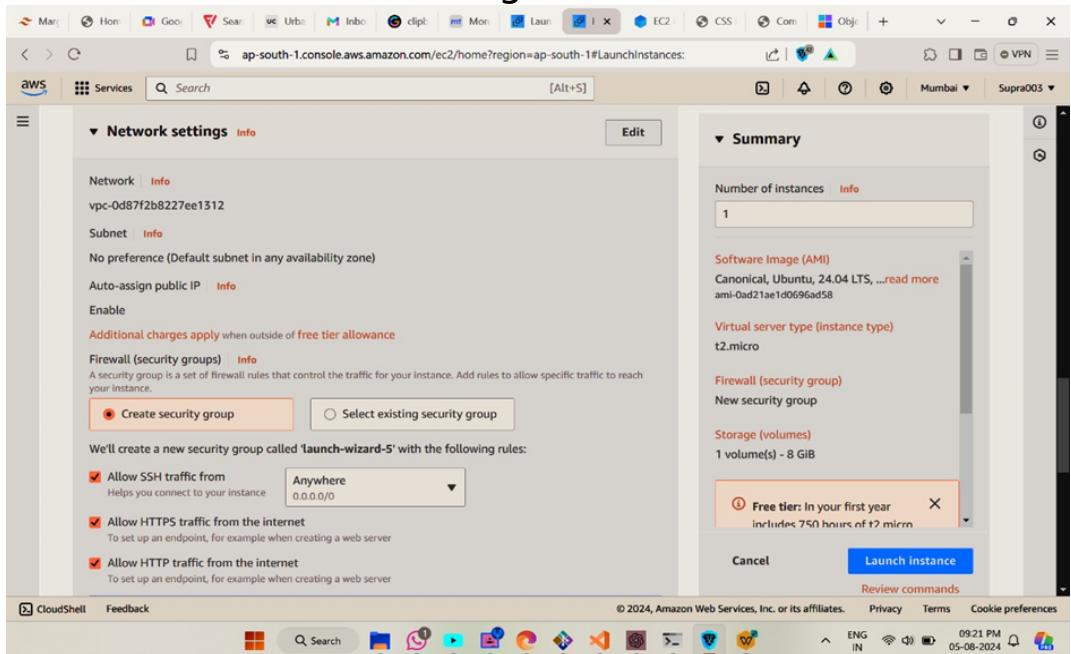
1. Give name to EC2 instance



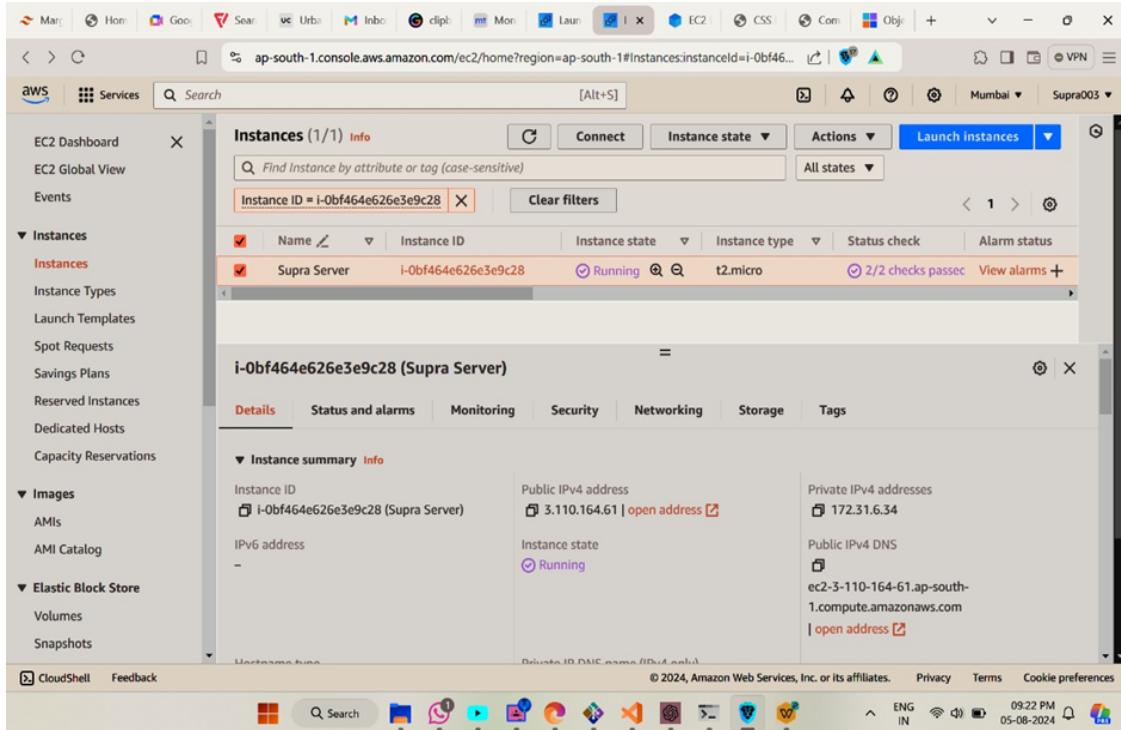
2. Create a key-pair login.



3. Edit network settings. And Launch Instance.



4. Instance successfully launched.



5. Create a virtual environment with key pair permission.

```
supra>ssh -i "supra-key.pem" ubuntu@3.110.164.61
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1009-aws x86_64)

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

System information as of Mon Aug  5 14:53:57 UTC 2024

System load: 0.01      Processes:          105
Usage of /: 22.7% of 6.71GB  Users logged in:   0
Memory usage: 20%        IPv4 address for enX0: 172.31.6.34
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

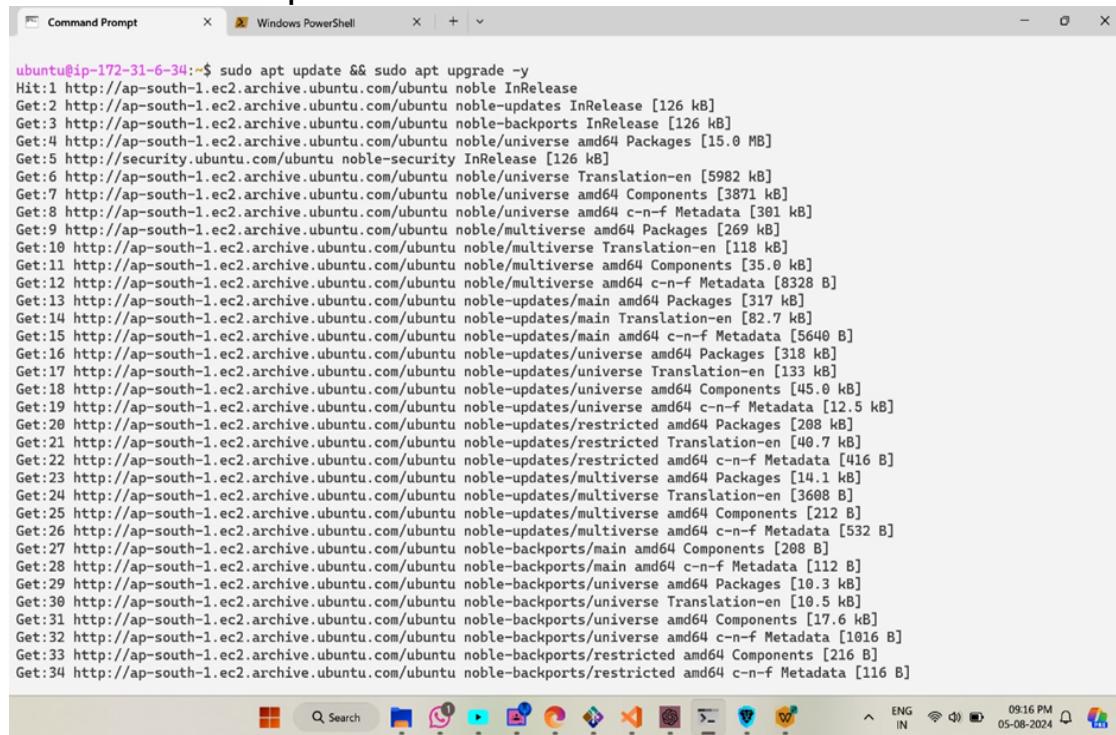
The list of available updates is more than a week old.
To check for new updates run: sudo apt update

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

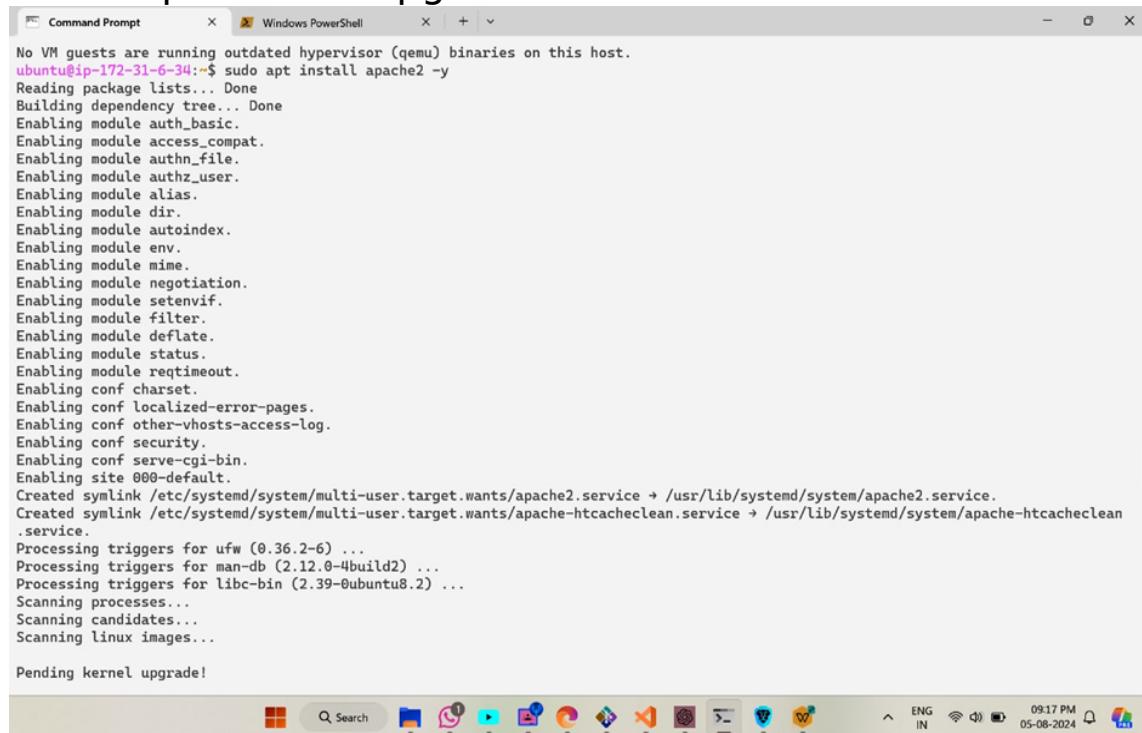
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

6. Install apache on virtual machine.



```
ubuntu@ip-172-31-6-34:~$ sudo apt update && sudo apt upgrade -y
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 kB]
Get:13 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/main amd64 Packages [317 kB]
Get:14 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/main Translation-en [82.7 kB]
Get:15 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/main amd64 c-n-f Metadata [5640 kB]
Get:16 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe amd64 Packages [318 kB]
Get:17 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe Translation-en [133 kB]
Get:18 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe amd64 Components [45.0 kB]
Get:19 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe amd64 c-n-f Metadata [12.5 kB]
Get:20 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/restricted amd64 Packages [208 kB]
Get:21 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/restricted Translation-en [40.7 kB]
Get:22 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/restricted amd64 c-n-f Metadata [416 kB]
Get:23 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/universe amd64 Packages [14.1 kB]
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/multiverse Translation-en [3608 kB]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/multiverse amd64 Components [212 kB]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/multiverse amd64 c-n-f Metadata [532 kB]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/main amd64 Components [208 kB]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/main amd64 c-n-f Metadata [112 kB]
Get:29 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/universe amd64 Packages [10.3 kB]
Get:30 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/universe Translation-en [10.5 kB]
Get:31 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/universe amd64 Components [17.6 kB]
Get:32 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/universe amd64 c-n-f Metadata [1016 kB]
Get:33 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/restricted amd64 Components [216 kB]
Get:34 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/restricted amd64 c-n-f Metadata [116 kB]
```

7. Update and upgrade it.



```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-6-34:~$ sudo apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Enabling module auth_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-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 (2.39-0ubuntu8.2) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Pending kernel upgrade!
```

8. Start the apache server.



```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-6-34:~$ sudo systemctl start apache2
ubuntu@ip-172-31-6-34:~$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2
ubuntu@ip-172-31-6-34:~$ client_loop: send disconnect: Connection reset

supra>scp -i "supra-key.pem" "C:\Users\91952\Downloads\index.html" ubuntu@3.110.164.61:/var/www/html/
scp: /var/www/html//index.html: Permission denied

supra>scp -i "supra-key.pem" "C:\Users\91952\Downloads\index.html" ubuntu@3.110.164.61:/var/www/html
scp: /var/www/html/index.html: Permission denied

supra>scp -i "supra-key.pem" "C:\Users\91952\Downloads\index.html" ubuntu@3.110.164.61:/home/ubuntu/
index.html      100% 2955   36.8KB/s  00:00

supra>ssh -i "supra-key.pem" ubuntu@3.110.164.61
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1009-aws x86_64)

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

System information as of Mon Aug  5 15:08:46 UTC 2024

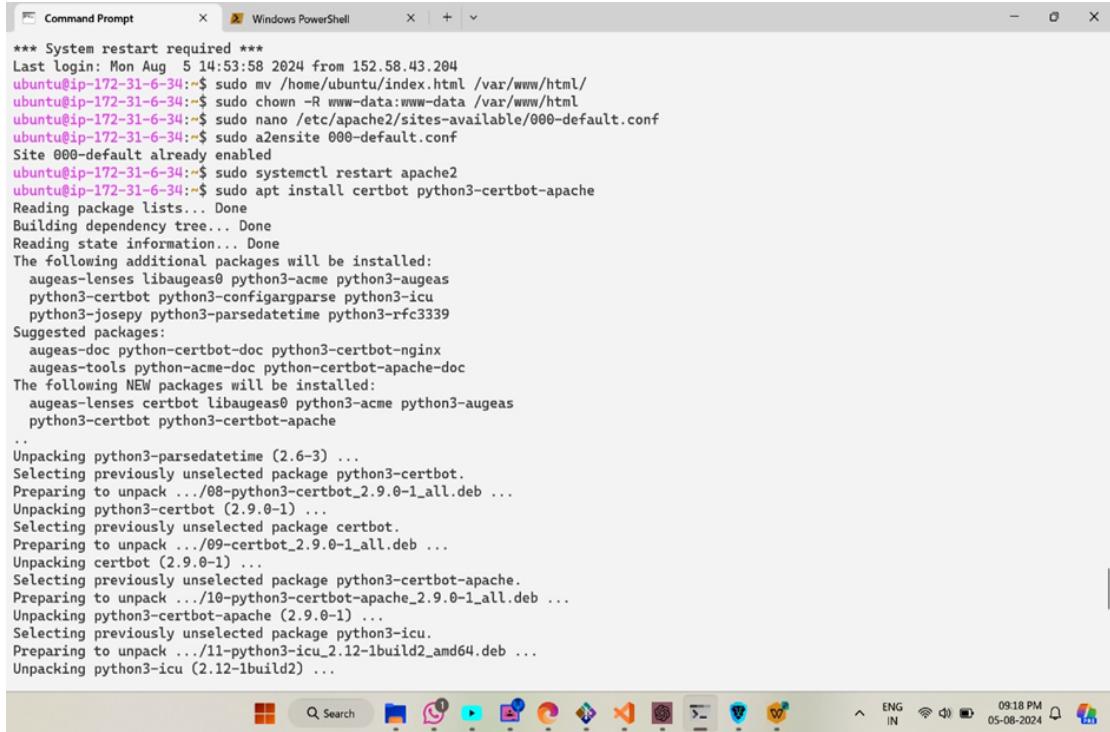
System load:  0.0          Processes:           110
Usage of /:   29.1% of 6.71GB  Users logged in:     0
Memory usage: 25%
Swap usage:   0%
IPv4 address for enX0: 172.31.6.34

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
```

9. Give file location to be executed.



```
*** System restart required ***
Last login: Mon Aug  5 14:53:58 2024 from 152.58.43.204
ubuntu@ip-172-31-6-34:~$ sudo mv /home/ubuntu/index.html /var/www/html/
ubuntu@ip-172-31-6-34:~$ sudo chown -R www-data:www-data /var/www/html/
ubuntu@ip-172-31-6-34:~$ sudo nano /etc/apache2/sites-available/000-default.conf
ubuntu@ip-172-31-6-34:~$ sudo a2ensite 000-default.conf
Site 000-default already enabled
ubuntu@ip-172-31-6-34:~$ sudo systemctl restart apache2
ubuntu@ip-172-31-6-34:~$ sudo apt install certbot python3-certbot-apache
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  augaeas-lenses libaugeas0 python3-acme python3-augeas
  python3-certbot python3-configargparse python3-icu
  python3-josepy python3-parsedatetime python3-rfc3339
Suggested packages:
  augaeas-doc python3-certbot-doc python3-certbot-nginx
  augaeas-tools python3-acme-doc python3-certbot-apache-doc
The following NEW packages will be installed:
  augaeas-lenses certbot libaugeas0 python3-acme python3-augeas
  python3-certbot python3-certbot-apache
...
Unpacking python3-parsedatetime (2.6-3) ...
Selecting previously unselected package python3-certbot.
Preparing to unpack .../08-python3-certbot_2.9.0-1_all.deb ...
Unpacking python3-certbot (2.9.0-1) ...
Selecting previously unselected package certbot.
Preparing to unpack .../09-certbot_2.9.0-1_all.deb ...
Unpacking certbot (2.9.0-1) ...
Selecting previously unselected package python3-certbot-apache.
Preparing to unpack .../10-python3-certbot-apache_2.9.0-1_all.deb ...
Unpacking python3-certbot-apache (2.9.0-1) ...
Selecting previously unselected package python3-icu.
Preparing to unpack .../11-python3-icu_2.12-1build2_amd64.deb ...
Unpacking python3-icu (2.12-1build2) ...

```

10. Output of the code on local machine.



Amazon

amazon

- [Services](#)
- [About Us](#)
- [Contact](#)

About Us

0:00 / 0:00

Amazon is a global leader in e-commerce and cloud computing, committed to providing customers with a vast selection of products and services.

Company Details
Address 410 Terry Ave N, Seattle, WA 98109, USA
Contact (206) 266-1000
Email info@amazon.com

Our Services

- Online Retail
- Amazon Prime Membership
- Amazon Web Services (AWS)
- Kindle eBooks and Devices
- Amazon Music and Video Streaming

Experiment 02

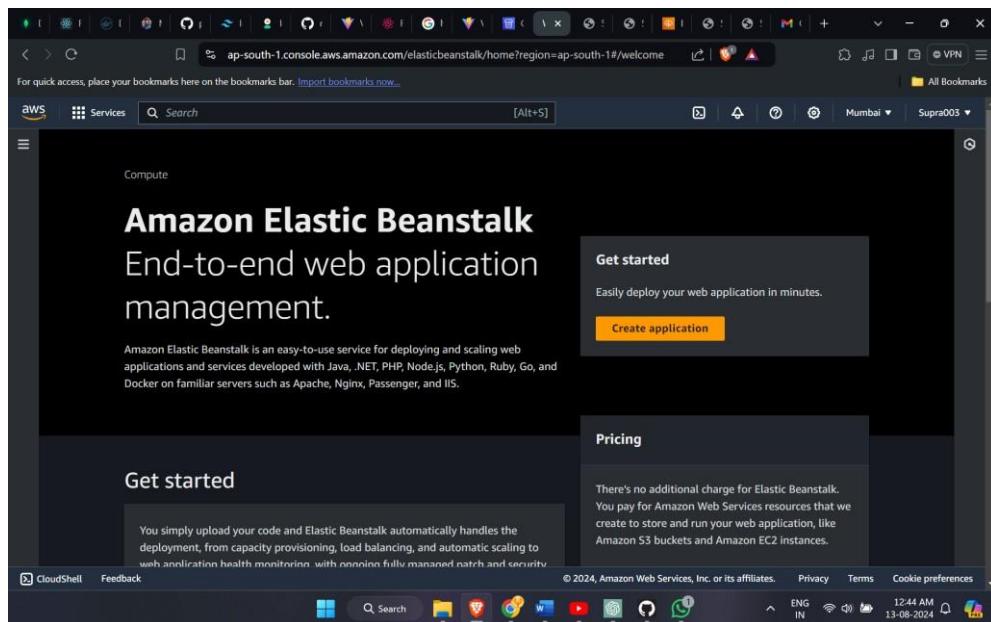
Name:Laksh Sodhai

Roll no: 59

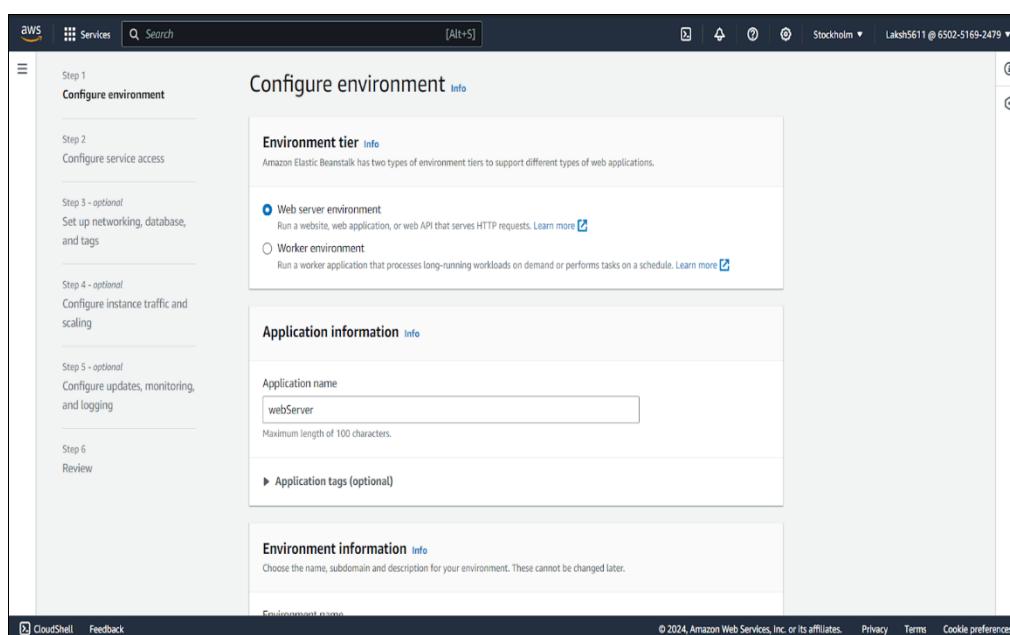
Aim: To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

Step-1: Beanstalk Deployment

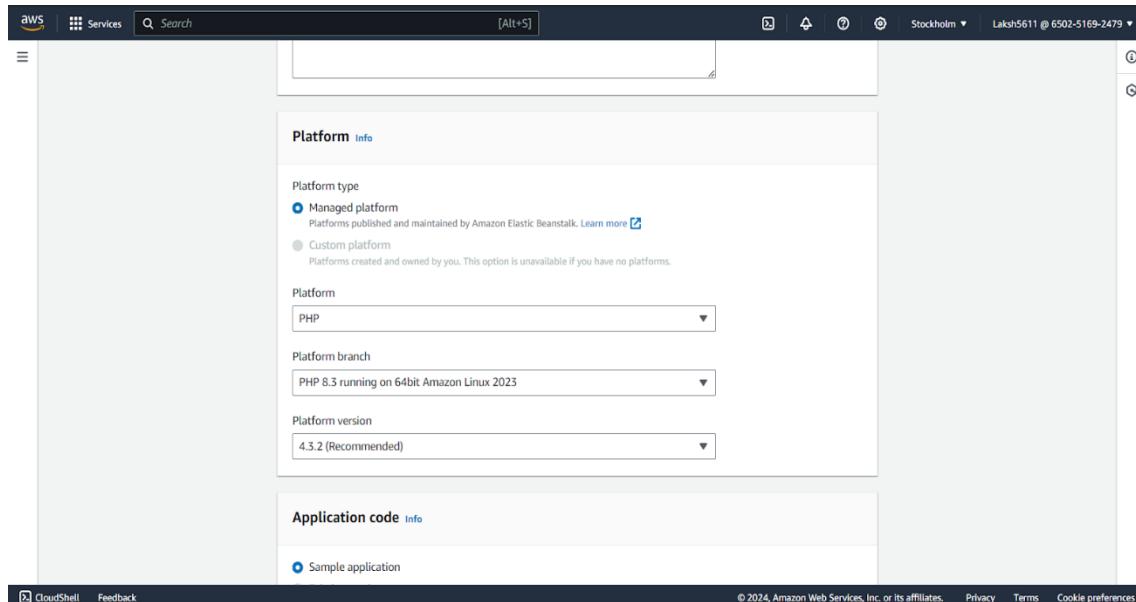
1. Login to your AWS account and search for Elastic Beanstalk in the search box



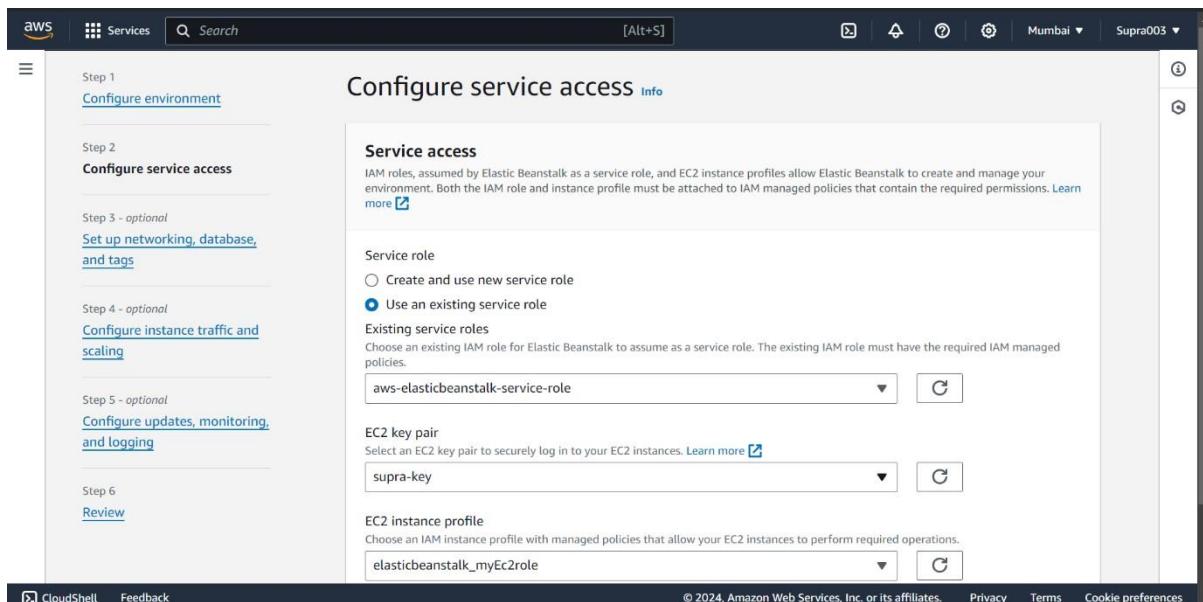
2. Open up Elastic Beanstalk and name your web app.



3. Choose Python from the drop-down menu and then click Create Application.



4. Choose the proper options



Virtual Private Cloud (VPC)

VPC
Launch your environment in a custom VPC instead of the default VPC. You can create a VPC and subnets in the VPC management console.
[Learn more](#)

vpc-0d87f2b8227ee1312 | (172.31.0.0/16)

[Create custom VPC](#)

Instance settings
Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. [Learn more](#)

Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.

Activated

Instance subnets

Filter instance subnets

Availability Zone	Subnet	CIDR	Name
us-east-1a	sg-0d87f2b8227ee1312	172.31.0.0/16	SupraApp-env-1

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

With the current setting, the environment enables only IMDSv2.

Deactivated

EC2 security groups
Select security groups to control traffic.

EC2 security groups (7)

Filter security groups

Group name	Group ID	Name
awseb-e-d9kh68ppqn-stack-1	sg-0632048b6f88d4dd	SupraApp-env-1
awseb-e-eczswppc8z-stack-1	sg-0de68df64bdffa855	WebApp02-env
default	sg-0a472ec49e7bb8a21	
launch-wizard-1	sg-01711621a355ca6a4	
launch-wizard-2	sg-048d55c094d22c63a	
launch-wizard-3	sg-0b5b28982c66cf258	
launch-wizard-4	sg-0d43e01a6548a50da	

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

Review [Info](#)

Step 1: Configure environment

[Edit](#)

Environment information

Environment tier	Application name
Web server environment	WebServer
Environment name	Application code
WebServer-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

[Edit](#)

Service access [Info](#)

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

Instance log streaming to CloudWatch logs

Configure the instances in your environment to stream logs to CloudWatch logs. You can set the retention to up to 10 years and configure Elastic Beanstalk to delete the logs when you terminate your environment. [Learn more](#)

Log streaming
(standard CloudWatch charges apply.)

Activated

Retention

7

Lifecycle

Keep logs after terminating envir...

Environment properties

The following properties are passed in the application as environment properties. [Learn more](#)

No environment properties have been configured.

[Add environment property](#)

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

Configure updates, monitoring, and logging - optional

Step 1
[Configure environment](#)

Step 2
[Configure service access](#)

Step 3 - optional
[Set up networking, database, and tags](#)

Step 4 - optional
[Configure instance traffic and scaling](#)

Step 5 - optional
[Configure updates, monitoring, and logging](#)

Step 6
Review

▼ 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

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

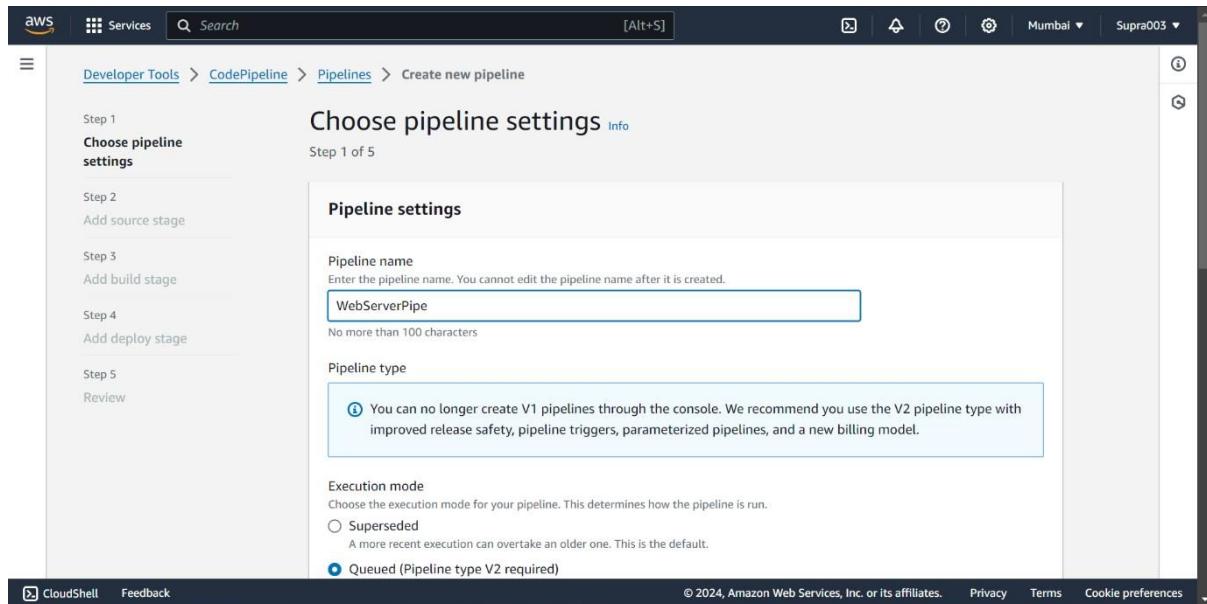
[CloudShell](#) [Feedback](#)

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

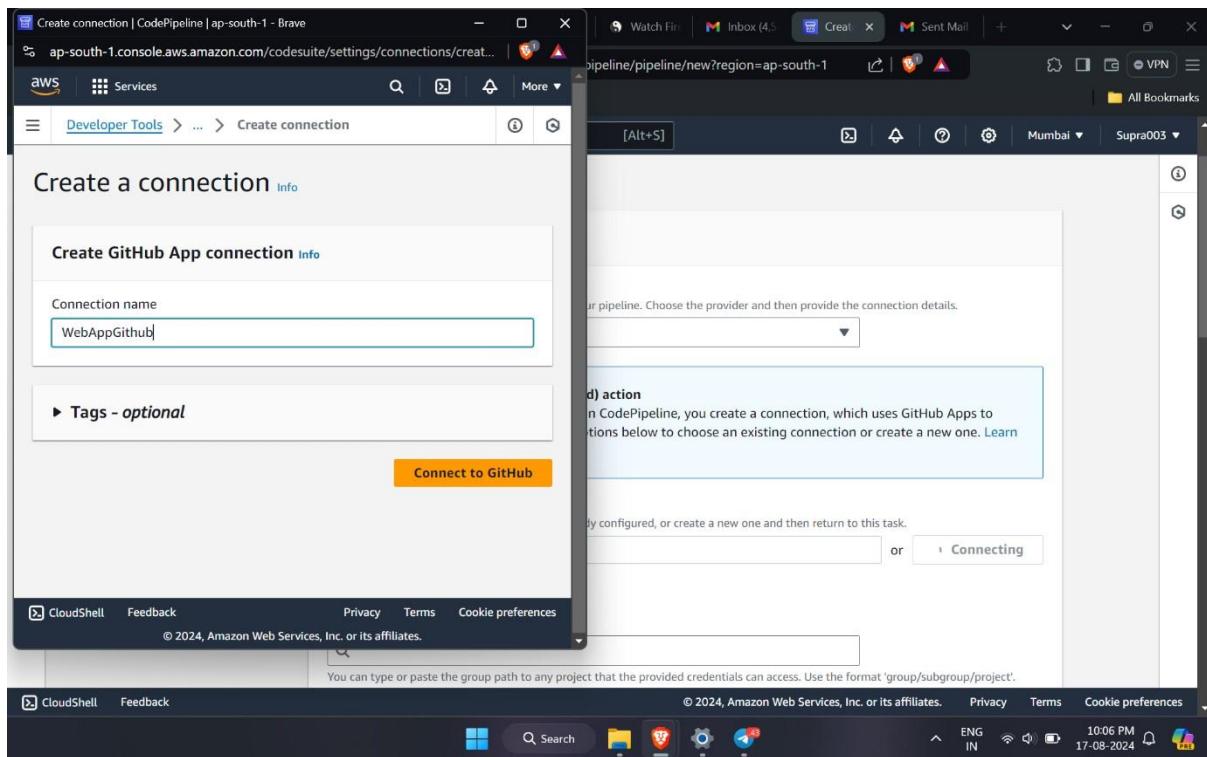
Step-2: Creating CodePipeline

Beanstalk creates a sample environment for you to deploy your application. By default, it creates an EC2 instance, a security group, an Auto Scaling group, an Amazon S3 Bucket, Amazon CloudWatch alarms and a domain name for your Application.

Create a CodePipeline and give pipeline name



Create a Github connection



Installed GitHub App - AWS Connector for GitHub - Brave

github.com/settings/installations/53728143

Repository access

All repositories
This applies to all current *and* future repositories owned by the resource owner.
Also includes public repositories (read-only).

Only select repositories
Select at least one repository.
Also includes public repositories (read-only).

Select repositories ▾

Selected 1 repository.

prajyots60/aws-codedepipeline-s3-codedeploy-linux-2.0

Save Cancel

Danger zone

You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:08 PM 17-08-2024

Create connection | CodePipeline | ap-south-1 - Brave

ap-south-1.console.aws.amazon.com/codesuite/settings/connections/creat...

Developer Tools > ... > Create connection

Beginning July 1, 2024, the console will create connections with codeconnections in the resource ARN. Resources with both service prefixes will continue to display in the console. [Learn more](#)

Connect to GitHub

GitHub connection settings [Info](#)

Connection name: WebAppGitHub

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

Q 53728143 or [Install a new app](#)

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

You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:08 PM 17-08-2024

Screenshot of the AWS CodePipeline 'Add source stage' configuration screen.

The left sidebar shows steps: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review).

The main panel is titled 'Source' under 'Source provider'. It shows 'GitHub (Version 2)' selected. A note about 'New GitHub version 2 (app-based) action' is present, along with a 'Connect to GitHub' button.

The 'Connection' section shows a connection named 'arn:aws:codeconnections:ap-south-1:010928222160:connection/38b1df1b-a' with a delete icon. A message says 'Ready to connect'.

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

Screenshot of the AWS CodePipeline 'Repository name' configuration screen.

The left sidebar shows steps: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review).

The main panel is titled 'Repository name'. It shows 'pjayots60/aws-codepipeline-s3-codedeploy-linux-2.0' selected in the search bar. A note says 'You can type or paste the group path to any project that the provided credentials can access. Use the format "group/subgroup/project".'

The 'Default branch' section shows 'master' selected.

The 'Output artifact format' section shows 'CodePipeline default' selected. A note says 'AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.'

The 'Trigger' section shows 'No filter' selected. A note says 'Starts your pipeline on any push and clones the HEAD.'

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

For deploy choose elastic beanstalk

The screenshot shows the 'Add deploy stage' step of a pipeline configuration. The left sidebar lists steps from 1 to 5: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review). The main area is titled 'Deploy' and contains the following fields:

- Deploy provider:** AWS Elastic Beanstalk
- Region:** Asia Pacific (Mumbai)
- Input artifacts:** SourceArtifact
- Application name:** WebServer
- Environment name:** WebServer-env
- Configure automatic rollback on stage failure:** Enabled (checkbox checked)

A message at the top states: "You cannot skip this stage. Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage."

The screenshot shows the 'Step 4: Add deploy stage' step of a pipeline configuration. The left sidebar lists steps from 1 to 5: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review). The main area is titled 'Step 4: Add deploy stage' and contains the following details:

Deploy action provider: AWS Elastic Beanstalk

Configuration:

- ApplicationName: WebServer
- EnvironmentName: WebServer-env
- Configure automatic rollback on stage failure: Enabled

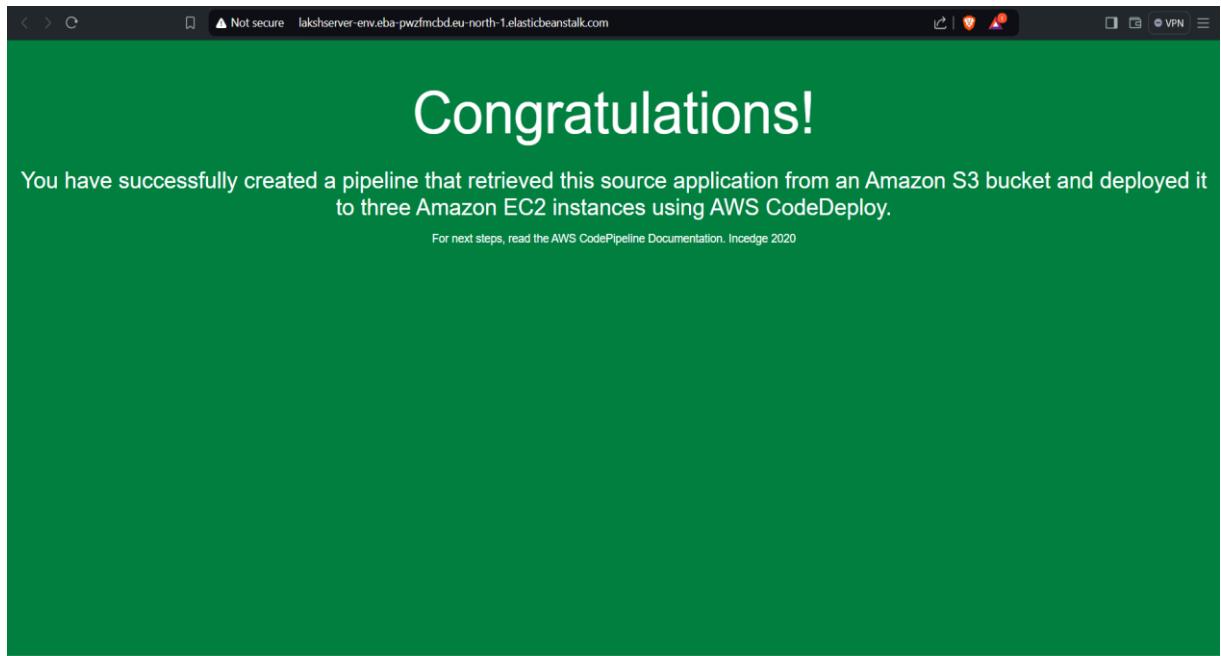
At the bottom right are buttons for **Create pipeline**, **Previous**, and **Cancel**.

Make connection between code pipeline and beanstalk. And Green means successfully connected.

The screenshot shows the AWS CodePipeline console for the 'WebServerPipe' pipeline. The pipeline type is V2 and the execution mode is QUEUED. The pipeline has two stages: Source and Deploy. The Source stage is successful, showing a GitHub commit from 'Version 2' with a status of 'Succeeded - 1 minute ago'. The Deploy stage is also successful, showing an AWS Elastic Beanstalk deployment with a status of 'Succeeded - Just now'. The pipeline execution ID is 8e246baaf-df87-463f-925d-7d5d2ee71640. There are green checkmarks in the status bar on the right.

Click on domain to view your hosting site

The screenshot shows the AWS Elastic Beanstalk console for the 'WebServer' application. It displays the 'Application WebServer environments' section, which contains one environment named 'WebServer-env'. The environment is in a 'Green' state, indicating it is healthy. The domain assigned to this environment is 'WebServer-env.eba-227p9xyx...'. The environment was created on August 17, 2024, at 22:... and is running version 'code-pipeline'. The URL of the page is https://ap-south-1.console.aws.amazon.com/elasticbeanstalk/home?region=ap-south-1#.



Hosting a static website on Amazon S3

The screenshot shows the 'Create bucket' page in the AWS Management Console. Under 'General configuration', the 'Bucket name' is set to 'Laksh-S3'. The 'Bucket type' dropdown is open, showing 'General purpose' (selected) and 'Directory - New'. Other configuration options like 'Copy settings from existing bucket - optional' and 'Advanced settings' are also visible.

The screenshot shows the 'Default encryption' page. It indicates that server-side encryption is automatically applied to new objects stored in the bucket. The 'Encryption type' dropdown is set to 'Server-side encryption with Amazon S3 managed keys (SSE-S3)'. Other options like 'AWS KMS' and 'Dual-layer server-side encryption' are listed. The 'Bucket Key' section shows 'Enable' selected. A note at the bottom states: 'After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.'

The screenshot shows the 'Edit static website hosting' page for the 'laksh-s3' bucket. It has sections for 'Static website hosting' (disabled), 'Hosting type' (hosting a static website), and 'Index document' (set to 'index.html'). A note at the bottom explains that content must be publicly readable. The page includes standard AWS navigation and footer links.

Screenshot of the AWS IAM Policy Editor showing a policy for the bucket ARN arn:aws:s3:::laksh-s3. The policy grants full access (GetObject) to the bucket to all users.

```
1 ▼ {
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Effect": "Allow",
6             "Principal": "*",
7             "Action": "s3:GetObject",
8             "Resource": "arn:aws:s3:::laksh-s3/*"
9         }
10    ]
11 }
12
```

The right sidebar shows the policy editor interface with options for editing, removing statements, adding actions, choosing services, and viewing included and available services.

Buckets Overview:

- Successfully created bucket "laksh-s3".** To upload files and folders, or to configure additional bucket settings, choose [View details](#).
- Amazon S3 > Buckets**
- Account snapshot - updated every 24 hours** (All AWS Regions)
Storage lens provides visibility into storage usage and activity trends. [Learn more](#)
- General purpose buckets** (3) [Info](#) [All AWS Regions](#)
Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
codepipeline-eu-north-1-959538756346	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 19, 2024, 18:48:15 (UTC+05:30)
elasticbeanstalk-eu-north-1-140023404138	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 19, 2024, 18:34:28 (UTC+05:30)
laksh-s3	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 19, 2024, 18:53:10 (UTC+05:30)

aws Services Search [Alt+S] Stockholm Laksh005

Amazon S3 > Buckets > laksh-s3 > Upload

Upload Info

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

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

Files and folders (1 Total, 2.8 KB)

All files and folders in this table will be uploaded.

Name	Type
index.html	text/html

Destination Info

Destination
s3://laksh-s3

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

aws Services Search [Alt+S] Stockholm Laksh005

Upload succeeded View details below.

Upload: status

The information below will no longer be available after you navigate away from this page.

Summary

Destination	Succeeded	Failed
s3://laksh-s3	1 file, 2.8 KB (100.00%)	0 files, 0 B (0%)

Files and folders Configuration

Files and folders (1 Total, 2.8 KB)

Name	Folder	Type	Size	Status	Error
index.html	-	text/html	2.8 KB	Succeeded	-

Screenshot of the AWS S3 console showing bucket configuration settings.

Buckets

- Access Grants
- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- IAM Access Analyzer for S3

Storage Lens

- Dashboards
- Storage Lens groups
- AWS Organizations settings

Feature spotlight

AWS Marketplace for S3

Amazon S3

Object Lock
Disabled

Requester pays

When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled. [Learn more](#)

Requester pays
Disabled

Static website hosting

Use this bucket to host a website or redirect requests. [Learn more](#)

Static website hosting
Enabled

Hosting type
Bucket hosting

Bucket website endpoint

When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more](#)

<http://laksh-s3-website.eu-north-1.amazonaws.com>

CloudShell Feedback

Not secure laksh-s3-website.eu-north-1.amazonaws.com

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

Amazon

amazon

Services About Us Contact

About Us

0:00 / 0:00

Amazon is a global leader in e-commerce and cloud computing, committed to providing customers with a vast selection of products and services.

Company Details

Address 410 Terry Ave N, Seattle, WA 98109, USA

Contact (206) 266-1000

Email info@amazon.com

Our Services

- Online Retail
- Amazon Prime Membership
- Amazon Web Services (AWS)
- Kindle eBooks and Devices
- Amazon Music and Video Streaming

Promotional Video

[Video Player Placeholder]

PRACTICAL NO: - 3

Case Study: Understanding Kubernetes Cluster Architecture

Introduction: -

Kubernetes is an open-source platform used to deploy and maintain a group of containers in a virtualized environment. In practice, Kubernetes is most commonly used alongside Docker for better control and implementation of containerized applications. Containerized applications “bundle” applications together with all its files, libraries, and packages required for it to run reliably and efficiently on different platforms. However, it operates at the container level rather than at the hardware level.

The name Kubernetes is derived from a Greek term meaning ‘helmsman’ or ‘pilot.’ True to this word, Kubernetes provides the guiding force for developer platforms to transition from virtual machines (VMs) to containers and the statically scheduled to the dynamically scheduled. This means no more manual integration and configuration when you move from a testing environment to an actual production environment or from on-premise to the cloud! The Kubernetes logical compute environment offers common services to all the applications in the cluster as part of the ecosystem for the software to run consistently.

Features of Kubernetes

- Automates various manual processes and controls server hosting and launching
- Manages containers offer security, and networking and storage services
- Monitors and continuously checks the health of nodes and containers
- Automates rollback for changes that go wrong
- Mounts and adds a storage system to run apps

Purpose of Kubernetes

The primary purpose of Kubernetes is to enable developers to write and deploy applications that can run seamlessly across multiple operating environments. Traditionally, application performance and deployment were tightly coupled to specific infrastructures, often requiring adherence to cloud provider-specific constructs and back-end storage systems. This dependency resulted in infrastructure lock-in, limiting flexibility and scalability.

Kubernetes addresses this challenge by abstracting the underlying infrastructure, allowing developers to deploy cloud-native applications in containers without restrictions. This means applications can be managed and scaled consistently across different environments—be it in the cloud, on-premises, or in hybrid setups—providing true infrastructure independence and operational flexibility.

Working of Kubernetes

Before exploring how Kubernetes operates, it's essential to grasp the concept of containers and their significance in modern application development. A container is a small, lightweight virtual machine (VM) that does not have device drivers and shares its operating system among the applications. It is a good way to bundle and run applications in a production environment. However, you need to manage these containers in a proper way so that there is no downtime. This is where Kubernetes comes to the rescue.

Kubernetes works as a "container orchestration system" that manages the lifecycle of containerized applications and automates the deployment of several containers. Containers running the same applications are usually grouped together into Pods. There can be one or multiple containers in a single Pod and each of them shares the same IP address and resources such as memory and storage. By grouping the containers in this manner, Kubernetes eliminates the need to cram multiple functionalities in one single container. There is a dedicated container orchestrator which supervises these groups and ensures that they operate correctly.

Kubernetes in DevOps

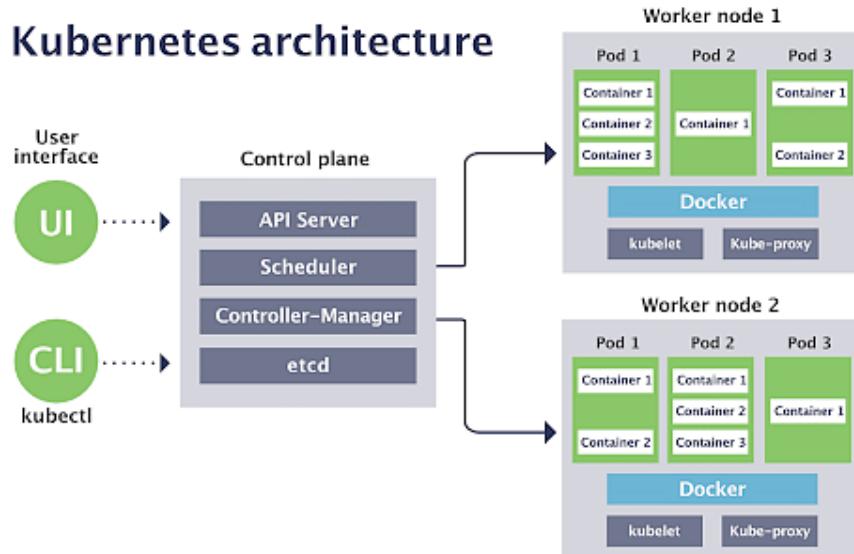
Kubernetes is more than just a container orchestration tool; it's a powerful enabler of DevOps practices. By bridging the gap between IT operations and development, Kubernetes fosters a collaborative DevOps environment, ensuring that software and its dependencies are shared seamlessly across different environments.

Kubernetes facilitates various stages of the software lifecycle, enhancing the build-test-deploy timeline:

- **Developer Environment:** Helps run software consistently in any setting, ensuring that applications behave the same across different environments.
- **QA/Testing Process:** Coordinates pipelines between test and production environments, streamlining the testing process.
- **Sys-Admin:** Once configured, Kubernetes runs anything, simplifying system administration tasks.
- **Operations:** Provides a comprehensive solution for building, shipping, and scaling software, making operations smoother and more efficient.

Kubernetes Cluster Architecture

A Kubernetes cluster consists of a set of worker machines, called nodes, that run containerized applications. The cluster is managed by a control plane, which is responsible for maintaining the desired state of the cluster, such as which applications are running and where they are running.



Control Plane: The control plane is the brain of the Kubernetes cluster, responsible for managing the desired state of the cluster, making decisions on scheduling, and responding to cluster events.

- **API Server:** The API server acts as the front-end for the Kubernetes control plane. It exposes the Kubernetes API, which is used by both internal components and external users (via CLI or UI) to communicate with the cluster.
- **Scheduler:** The scheduler is responsible for assigning newly created pods to nodes in the cluster. It evaluates the resource requirements of the pods against the available resources on the nodes, ensuring optimal placement.
- **Controller Manager:** This component runs various controllers that manage the state of the cluster. For example, it ensures that the number of pod replicas matches the desired configuration and handles node failures.
- **etcd:** A key-value store that holds all the configuration data for the Kubernetes cluster, including the current state and the desired state of the objects in the cluster. It is essential for maintaining cluster consistency and recovery.

Worker Nodes: Worker nodes are the machines where the application workloads run. Each worker node contains the services necessary to run pods and communicate with the control plane.

- **Kubelet:** The kubelet is an agent that runs on each worker node. It ensures that containers are running in a pod as expected by the control plane. It communicates with the API server to receive instructions and report back the status of the node and its workloads.
- **Kube-proxy:** Kube-proxy is a network proxy that runs on each worker node. It manages the networking for the pods, ensuring that each pod can communicate with others, both within and outside the cluster.
- **Docker (or other container runtimes):** Docker is the container runtime that runs and manages the containers on the worker nodes. It pulls container images from a registry, starts and stops containers, and manages container storage and networking.

User Interfaces: Users interact with the Kubernetes cluster through two primary interfaces:

- **UI (User Interface):** A graphical interface that provides an easy way to manage and monitor the cluster.
- **CLI (Command-Line Interface, e.g., kubectl):** A more powerful tool for managing the cluster, allowing users to interact with the API server directly via command-line commands.

This architecture allows Kubernetes to abstract away the underlying infrastructure, providing a consistent and scalable environment for deploying and managing containerized applications across different environments.

Conclusion:-

In conclusion, Kubernetes is a powerful, open-source platform that automates the deployment and management of containerized applications. Its architecture ensures seamless operation across diverse environments, promoting flexibility and scalability. By abstracting the underlying infrastructure and supporting DevOps practices, Kubernetes enhances the software development lifecycle. This results in streamlined operations, reduced downtime, and consistent application performance.

Install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms.

- Create 3 EC2 instances, one for the master node and two for the worker nodes

Summary

Number of instances: 3

Software Image (AMI): Amazon Linux 2023 AMI 2023.5.2...
ami-0182f373e6f6fb9c85

Virtual server type (instance type): t2.medium

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance

Launch instance

Summary

Number of instances: 3

Software Image (AMI): Amazon Linux 2023 AMI 2023.5.2...
ami-0182f373e6f6fb9c85

Virtual server type (instance type): t2.medium

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance

Launch instance

Instances (3) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IP
master	i-04e60dd4f6abefca6	Running	t2.medium	Initializing	View alarms +	us-east-1b	ec2-44-222-207.co...	44.222.2
worker-1	i-04e6f01c6f2e6cd68	Running	t2.medium	Initializing	View alarms +	us-east-1b	ec2-54-146-243-168.co...	54.146.2
worker-2	i-0825e9af4a2bfa3cb	Running	t2.medium	Initializing	View alarms +	us-east-1b	ec2-3-80-70-36.comput...	3.80.70.3

Select an instance

- After the instances have been created, copy the text given in the example part of each of the three instances into git bash.

EC2 > Instances > i-0e65e6b3d1a332a44 > Connect to instance

Connect to instance Info

Connect to your instance i-0e65e6b3d1a332a44 (master) using any of these options

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

Instance ID
i-0e65e6b3d1a332a44 (master)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is kubernetes_cluster.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 "kubernetes_cluster.pem"
4. Connect to your instance using its Public DNS:
ec2-54-211-197-39.compute-1.amazonaws.com

Example:
ssh -i "kubernetes_cluster.pem" ubuntu@ec2-54-211-197-39.compute-1.amazonaws.com

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

```
ubuntu@ip-172-31-29-63: ~
Microsoft Windows [Version 10.0.22631.4112]
(c) Microsoft Corporation. All rights reserved.

C:\Users\TEJAS\Downloads>ssh -i "kubernetes_cluster.pem" ubuntu@ec2-54-211-197-39.compute-1.amazonaws.com
The authenticity of host 'ec2-54-211-197-39.compute-1.amazonaws.com (54.211.197.39)' can't be established.
ED25519 key fingerprint is SHA256:E7RRKG6LxQFHsNbWLb8zgsskyIdj0YXpInW0b08/5vnA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-211-197-39.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1022-aws x86_64)

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

System information as of Sun Sep 15 17:48:58 UTC 2024

 System load:  0.28      Processes:          115
 Usage of /:   20.7% of  7.57GB   Users logged in:     0
 Memory usage: 5%           IPv4 address for eth0: 172.31.29.63
 Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.
```

- Update the package manager on all nodes:

```
ubuntu@ip-172-31-29-63: ~
$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1806 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2023 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [295 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [13.3 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2377 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [352 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [17.8 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [2437 kB]
Get:19 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [409 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [584 B]
Get:21 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [902 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [419 kB]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [616 B]
```

➤ Installing Required Packages for HTTPS and Certificate Transport on Ubuntu

```
ubuntu@ip-172-31-29-63:~$ 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 (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  curl libcurl4
2 upgraded, 1 newly installed, 0 to remove and 67 not upgraded.
Need to get 485 kB of archives.
After this operation, 170 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.13 [1510 B]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.17 [194 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libcurl4 amd64 7.81.0-1ubuntu1.17 [290 kB]
Fetched 485 kB in 0s (17.3 MB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 65320 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.13_all.deb ...
Unpacking apt-transport-https (2.4.13) ...
Preparing to unpack .../curl_7.81.0-1ubuntu1.17_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.17) over (7.81.0-1ubuntu1.16) ...
```

➤ Installing Docker on Ubuntu

```
ubuntu@ip-172-31-29-63:~$ sudo apt install docker.io -y
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-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 67 not upgraded.
Need to get 75.5 MB of archives.
After this operation, 284 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6-1 [63.6 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 bridge-utils amd64 1.7-1ubuntu3 [34.4 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 runc amd64 1.1.12-0ubuntu2~22.04.1 [8405 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 containerd amd64 1.7.12-0ubuntu2~22.04.1 [37.8 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 dns-root-data all 2023112702~ubuntu0.22.04.1 [5136 B]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 dnsmasq-base amd64 2.90-0ubuntu0.22.04.1 [374 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 docker.io amd64 24.0.7-0ubuntu2~22.04.
```

➤ Enabling Docker and Disabling Swap on Ubuntu

```
ubuntu@ip-172-31-29-63:~$ sudo systemctl enable --now docker
ubuntu@ip-172-31-29-63:~$ sudo swapoff -a
```

➤ Load necessary kernel modules for networking and iptables:

```
ubuntu@ip-172-31-29-63:~$ cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
overlay
br_netfilter
EOF
overlay
br_netfilter
ubuntu@ip-172-31-29-63:~$ sudo modprobe overlay
sudo modprobe br_netfilter
```

- Configure sysctl settings for Kubernetes networking:

```
ubuntu@ip-172-31-29-63:~$ cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-iptables = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.ipv4.ip_forward = 1
EOF
net.bridge.bridge-nf-call-iptables = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.ipv4.ip_forward = 1

ubuntu@ip-172-31-29-63:~$ sudo sysctl --system
* Applying /etc/sysctl.d/10-console-messages.conf ...
kernel.printk = 4 4 1 7
* Applying /etc/sysctl.d/10-ipv6-privacy.conf ...
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.default.use_tempaddr = 2
* Applying /etc/sysctl.d/10-kernel-hardening.conf ...
kernel.kptr_restrict = 1
* Applying /etc/sysctl.d/10-magic-sysrq.conf ...
kernel.sysrq = 176
* Applying /etc/sysctl.d/10-network-security.conf ...
net.ipv4.conf.default.rp_filter = 2
net.ipv4.conf.all.rp_filter = 2
* Applying /etc/sysctl.d/10-ptrace.conf ...
kernel.yama.ptrace_scope = 1
* Applying /etc/sysctl.d/10-zero-page.conf ...
vm.mmap_min_addr = 65536
* Applying /etc/sysctl.d/50-cloudimg-settings.conf ...
net.ipv4.neigh.default.gc_thresh2 = 15360
net.ipv4.neigh.default.gc_thresh3 = 16384
net.netfilter.nf_conntrack_max = 1048576
* Applying /usr/lib/sysctl.d/50-default.conf ...
kernel.core_uses_pid = 1
net.ipv4.conf.default.rp_filter = 2
```

- Install Kubernetes tools on all nodes.

```
ubuntu@ip-172-31-29-63:~$ curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.29/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg
echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /' | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /

ubuntu@ip-172-31-29-63:~$ sudo apt-get update -y
sudo apt-get install -y kubelet="1.29.0-*" kubectl="1.29.0-*" kubeadm="1.29.0-*"
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:5 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.29/deb InRelease [1189 B]
Get:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.29/deb Packages [14.0 kB]
Fetched 15.1 kB in 0s (31.9 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Selected version '1.29.0-1.1' (isv:kubernetes:core:stable:v1.29:pkgs.k8s.io [amd64]) for 'kubelet'
Selected version '1.29.0-1.1' (isv:kubernetes:core:stable:v1.29:pkgs.k8s.io [amd64]) for 'kubectl'
Selected version '1.29.0-1.1' (isv:kubernetes:core:stable:v1.29:pkgs.k8s.io [amd64]) for 'kubeadm'
The following additional packages will be installed:
  conntrack cri-tools ebtables kubernetes-cni socat
The following NEW packages will be installed:

ubuntu@ip-172-31-29-63:~$ sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
```

Only on master node

➤ Initialize the Kubernetes Cluster on Master Node

```
ubuntu@ip-172-31-29-63:~$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16
I0915 17:58:19.113429    3356 version.go:256] remote version is much newer: v1.31.0; falling back to: stable-1.29
[init] Using Kubernetes version: v1.29.8
[preflight] Running pre-flight checks
[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 in beforehand using 'kubeadm config images pull'
W0915 17:58:28.696218    3356 checks.go:835] 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 that using "registry.k8s.io/pause:3.9" as the CRI sandbox image.
[certs] Using certificateDir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [ip-172-31-29-63 kubernetes kubernetes.default kubernetes.default.svc kubernetes.default.svc.cluster.local] and IPs [10.96.0.1 172.31.29.63]
[certs] Generating "apiserver-kubelet-client" certificate and key
[certs] Generating "front-proxy-ca" certificate and key
[certs] Generating "front-proxy-client" certificate and key
[certs] Generating "etcd/ca" certificate and key
[certs] Generating "etcd/server" certificate and key
[certs] etcd/server serving cert is signed for DNS names [ip-172-31-29-63 localhost] and IPs [172.31.29.63 127.0.0.1 ::1]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [ip-172-31-29-63 localhost] and IPs [172.31.29.63 127.0.0.1 ::1]
```

```
ubuntu@ip-172-31-29-63:~$ 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 172.31.29.63:6443 --token 05rvqu.66v5246nmoe81d5e \
--discovery-token-ca-cert-hash sha256:ff6e3056ea0d41a598919b1be9dbe765739c40a5aa6d37b5e4cbc45b1256c1c7
```

➤ Set up kubectl on the master node

```
ubuntu@ip-172-31-29-63:~$ mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
ubuntu@ip-172-31-29-63:~$ kubectl get nodes
NAME           STATUS      ROLES   AGE     VERSION
ip-172-31-29-63   NotReady   control-plane   3m      v1.29.0
ubuntu@ip-172-31-29-63:~$ |
```

➤ To enable communication between pods, install a pod network plugin like Flannel or Calico

```
ubuntu@ip-172-31-29-63:~$ sudo systemctl restart kubelet
ubuntu@ip-172-31-29-63:~$ mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
cp: overwrite '/home/ubuntu/.kube/config'? y
ubuntu@ip-172-31-29-63:~$ kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml
namespace/kube-flannel created
clusterrole.rbac.authorization.k8s.io/flannel created
clusterrolebinding.rbac.authorization.k8s.io/flannel created
serviceaccount/flannel created
configmap/kube-flannel-cfg created
daemonset.apps/kube-flannel-ds created
ubuntu@ip-172-31-29-63:~$ |

ubuntu@ip-172-31-29-63:~$ kubeadm token create --print-join-command
kubeadm join 172.31.29.63:6443 --token avprdu.7xw3ox1x6l9w6zf9 --discovery-token-ca-cert-hash sha256:ff6e3056ea0d41a598919b1be9dbe765739c40a5aa6d37b5e4cbc45b1256c1c7
ubuntu@ip-172-31-29-63:~$ |
```

Only on Worker nodes

➤ Join Worker Nodes to the Cluster

```
ubuntu@ip-172-31-20-115:~$ sudo kubeadm reset pre-flight checks
W0915 18:24:16.207647    3366 preflight.go:56] [reset] WARNING: Changes made to this host by 'kubeadm init' or 'kubeadm
join' will be reverted.
[reset] Are you sure you want to proceed? [y/N]: y
[preflight] Running pre-flight checks
W0915 18:24:24.608981    3366 removeetcdmember.go:106] [reset] No kubeadm config, using etcd pod spec to get data direct
ory
[reset] Deleted contents of the etcd data directory: /var/lib/etcd
[reset] Stopping the kubelet service
[reset] Unmounting mounted directories in "/var/lib/kubelet"
[reset] Deleting contents of directories: [/etc/kubernetes/manifests /var/lib/kubelet /etc/kubernetes/pki]
[reset] Deleting files: [/etc/kubernetes/admin.conf /etc/kubernetes/super-admin.conf /etc/kubernetes/kubelet.conf /etc/k
ubernetes/bootstrap-kubelet.conf /etc/kubernetes/controller-manager.conf /etc/kubernetes/scheduler.conf]

The reset process does not clean CNI configuration. To do so, you must remove /etc/cni/net.d

The reset process does not reset or clean up iptables rules or IPVS tables.
If you wish to reset iptables, you must do so manually by using the "iptables" command.

If your cluster was setup to utilize IPVS, run ipvsadm --clear (or similar)
to reset your system's IPVS tables.
```

- On the worker nodes, run the command provided by the master node during initialization .
It looks something like this: sudo kubeadm join :6443 --token --discovery-token-ca-cert-
hash sha256:

```
ubuntu@ip-172-31-20-115:~$ sudo kubeadm join 172.31.29.63:6443 --token avprdu.7xw3ox1x6l9w6zfw --discovery-token-ca-cer
t-hash sha256:ff6e3056ea0d41a598919b1be9dbe765739c40a5aa6d37b5e4cbc45b1256c1c7 --v=5
I0915 18:26:15.816542    3382 join.go:413] [preflight] found NodeName empty; using OS hostname as NodeName
I0915 18:26:15.816650    3382 initconfiguration.go:122] detected and using CRI socket: unix:///var/run/containerd/contai
nerd.sock
[preflight] Running pre-flight checks
I0915 18:26:15.816686    3382 preflight.go:93] [preflight] Running general checks
I0915 18:26:15.816765    3382 checks.go:280] validating the existence of file /etc/kubernetes/kubelet.conf
I0915 18:26:15.816774    3382 checks.go:280] validating the existence of file /etc/kubernetes/bootstrap-kubelet.conf
I0915 18:26:15.816784    3382 checks.go:104] validating the container runtime
I0915 18:26:15.833043    3382 checks.go:639] validating whether swap is enabled or not
I0915 18:26:15.833129    3382 checks.go:370] validating the presence of executable crictl
I0915 18:26:15.833152    3382 checks.go:370] validating the presence of executable conctrack
I0915 18:26:15.833171    3382 checks.go:370] validating the presence of executable ip
I0915 18:26:15.833203    3382 checks.go:370] validating the presence of executable iptables
I0915 18:26:15.833227    3382 checks.go:370] validating the presence of executable mount
I0915 18:26:15.833242    3382 checks.go:370] validating the presence of executable nsenter
I0915 18:26:15.833276    3382 checks.go:370] validating the presence of executable ebtables
I0915 18:26:15.833292    3382 checks.go:370] validating the presence of executable ethtool
I0915 18:26:15.833320    3382 checks.go:370] validating the presence of executable socat
I0915 18:26:15.833340    3382 checks.go:370] validating the presence of executable tc

bootstrap-kubelet.conf
I0915 18:26:15.933343    3382 kubelet.go:136] [kubelet-start] writing CA certificate at /etc/kubernetes/pki/ca.crt
I0915 18:26:15.933723    3382 kubelet.go:157] [kubelet-start] Checking for an existing Node in the cluster with name "ip
-172-31-20-115" and status "Ready"
I0915 18:26:15.936219    3382 kubelet.go:172] [kubelet-start] Stopping the kubelet
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...
I0915 18:26:17.103583    3382 cert_rotation.go:137] Starting client certificate rotation controller
I0915 18:26:17.104245    3382 kubelet.go:220] [kubelet-start] preserving the crisocket information for the node
I0915 18:26:17.104261    3382 patchnode.go:31] [patchnode] Uploading the CRI Socket information "unix:///var/run/contain
erd/containerd.sock" to the Node API object "ip-172-31-20-115" as an annotation

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@ip-172-31-20-115:~$ |
```

➤ Verify the Cluster

Once the worker node joins, check the status on the master node

```
ubuntu@ip-172-31-29-63:~$ kubectl get nodes
NAME        STATUS   ROLES     AGE      VERSION
ip-172-31-20-115 Ready    <none>   27m     v1.29.0
ip-172-31-20-200 Ready    <none>   24m     v1.29.0
ip-172-31-29-63  Ready    control-plane  55m     v1.29.0
ubuntu@ip-172-31-29-63:~$ |
```

PRACTICAL NO: - 4

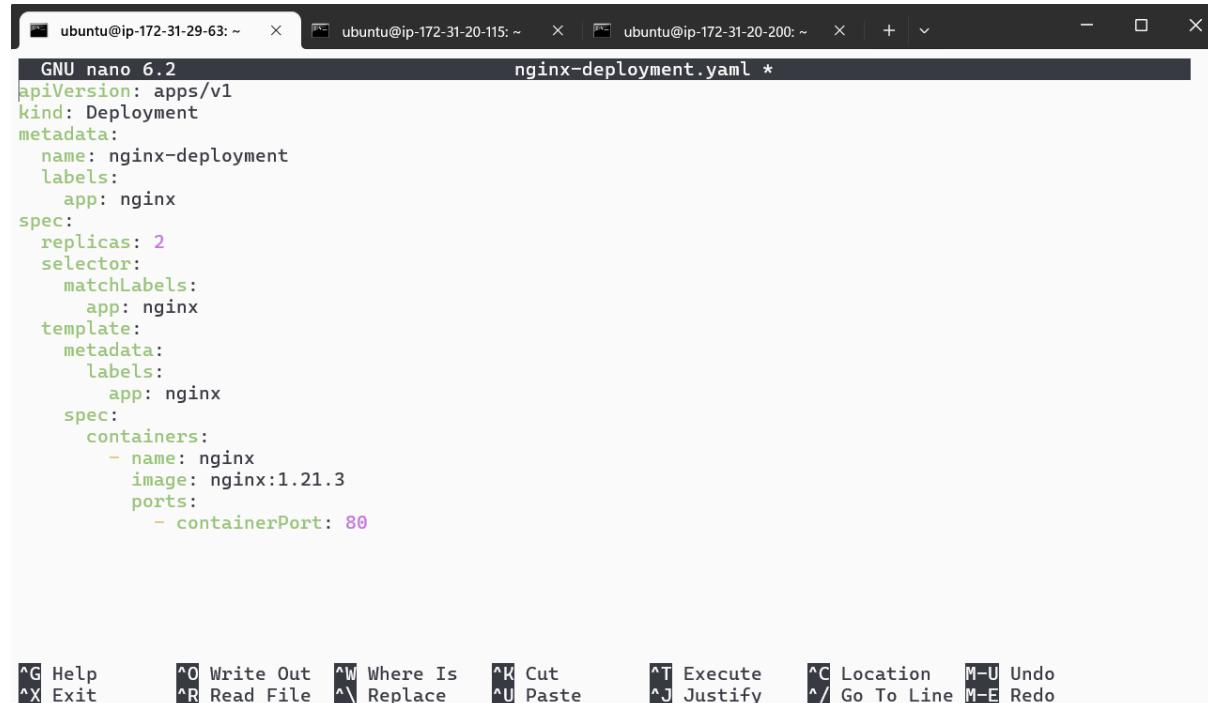
In previous practical, we have setup the Kubernetes cluster

➤ Deploying Your Application on Kubernetes

Create the YAML file: Use a text editor to create a file named nginx-deployment.yaml

```
ubuntu@ip-172-31-29-63:~$ nano nginx-deployment.yaml
ubuntu@ip-172-31-29-63:~$ |
```

Add the Deployment Configuration: Copy and paste the following YAML content into the file. Save and exit the editor (Press Ctrl+X, then Y, and Enter).



```
ubuntu@ip-172-31-29-63:~$ nano nginx-deployment.yaml
ubuntu@ip-172-31-29-63:~$ |
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.21.3
          ports:
            - containerPort: 80
```

GNU nano 6.2

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo
 ^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo

Create the YAML File: Create another file named nginx-service.yaml

```
ubuntu@ip-172-31-29-63:~$ nano nginx-service.yaml
ubuntu@ip-172-31-29-63:~$ |
```

Add the Service Configuration: Copy and paste the following YAML content into the file given below.

```
GNU nano 6.2                               nginx-service.yaml
apiVersion: v1
kind: Service
metadata:
  name: nginx-service
spec:
  selector:
    app: nginx
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
  type: LoadBalancer
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
```

Deploy the Application: Use kubectl to create the Deployment and Service from the YAML files.

```
ubuntu@ip-172-31-29-63:~$ kubectl apply -f nginx-deployment.yaml --validate=false
deployment.apps/nginx-deployment created

ubuntu@ip-172-31-29-63:~$ kubectl apply -f nginx-service.yaml --validate=false
service/nginx-service created
ubuntu@ip-172-31-29-63:~$ |
```

Verify the Deployment: Check the status of your Deployment, Pods and Services

```
ubuntu@ip-172-31-29-63:~$ kubectl get deployments
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   1/2     2           1          9m25s
ubuntu@ip-172-31-29-63:~$ kubectl get pods
NAME                           READY   STATUS    RESTARTS   AGE
nginx-deployment-6b4d6fdbf-6w4bm   1/1     Running   4 (98s ago)  9m18s
nginx-deployment-6b4d6fdbf-bhcwm   1/1     Running   4 (70s ago)  9m18s
ubuntu@ip-172-31-29-63:~$ kubectl get services
NAME            TYPE        CLUSTER-IP       EXTERNAL-IP      PORT(S)        AGE
kubernetes      ClusterIP   10.96.0.1      <none>          443/TCP       111m
nginx-service   LoadBalancer 10.110.88.111  <pending>       80:30132/TCP  110s
ubuntu@ip-172-31-29-63:~$ |
```

Describe the deployment(Extra)

```
ubuntu@ip-172-31-29-63:~$ kubectl get deployments
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   2/2     2           2          11m
ubuntu@ip-172-31-29-63:~$ kubectl describe deployment
Name:           nginx-deployment
Namespace:      default
CreationTimestamp: Sun, 15 Sep 2024 19:39:41 +0000
Labels:          app=nginx
Annotations:    deployment.kubernetes.io/revision: 1
Selector:        app=nginx
Replicas:       2 desired | 2 updated | 2 total | 1 available | 1 unavailable
StrategyType:   RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
    nginx:
      Image:      nginx:1.21.3
      Port:       80/TCP
      Host Port:  0/TCP
      Environment: <none>
      Mounts:    <none>
      Volumes:   <none>
  Conditions:
    Type      Status  Reason
    ----      ----   -----
    Progressing True   NewReplicaSetAvailable
    Available  False  MinimumReplicasUnavailable
  OldReplicaSets: <none>
  NewReplicaSet:  nginx-deployment-6b4d6fdbf (2/2 replicas created)
Events:
  Type      Reason     Age   From           Message
  ----      ----     --   --    -----
  Normal   ScalingReplicaSet 11m  deployment-controller  Scaled up replica set nginx-deployment-6b4d6fdbf to 2
ubuntu@ip-172-31-29-63:~$ |
```

Verify Service: Run the following command to check the services running in your cluster:

```
ubuntu@ip-172-31-29-63:~$ kubectl get service
NAME           TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)        AGE
kubernetes     ClusterIP  10.96.0.1    <none>       443/TCP       114m
nginx-service  LoadBalancer 10.110.88.111  <pending>    80:30132/TCP  4m59s
ubuntu@ip-172-31-29-63:~$ |
```

Forward the Service Port to Your Local Machine

```
ubuntu@ip-172-31-45-227:~$ kubectl port-forward service/nginx-service 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80

ubuntu@ip-172-31-45-227:~$ kubectl port-forward service/nginx-service 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
^Cubuntu@ip-172-31-45-227:~$ kubectl port-forward service/nginx-service 8081:8080
Forwarding from 127.0.0.1:8081 -> 80
Forwarding from [::1]:8081 -> 80
^Cubuntu@ip-172-31-45-227:~$ kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-deployment-776b8fd845-k9cx4  1/1     Running   0          113m
ubuntu@ip-172-31-45-227:~$ kubectl logs nginx-deployment-776b8fd845-k9cx4
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/09/12 06:35:51 [notice] 1#1: using the "epoll" event method
2024/09/12 06:35:51 [notice] 1#1: nginx/1.27.1
2024/09/12 06:35:51 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2024/09/12 06:35:51 [notice] 1#1: OS: Linux 6.5.0-1022-aws
2024/09/12 06:35:51 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2024/09/12 06:35:51 [notice] 1#1: start worker processes
2024/09/12 06:35:51 [notice] 1#1: start worker process 24
2024/09/12 06:35:51 [notice] 1#1: start worker process 25
```

Access the Application Locally

Open a Web Browser: Now open your web browser and go to the following URL:
<http://localhost:8080> You should see the application (in this case, Nginx) that you have deployed running in the Kubernetes cluster, served locally via port 8080.



ADVANCE DEVOPS EXP 5

Name:Laksh Vijay Sodhai

Class:D15A

Roll No:59

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

Installation for Windows:

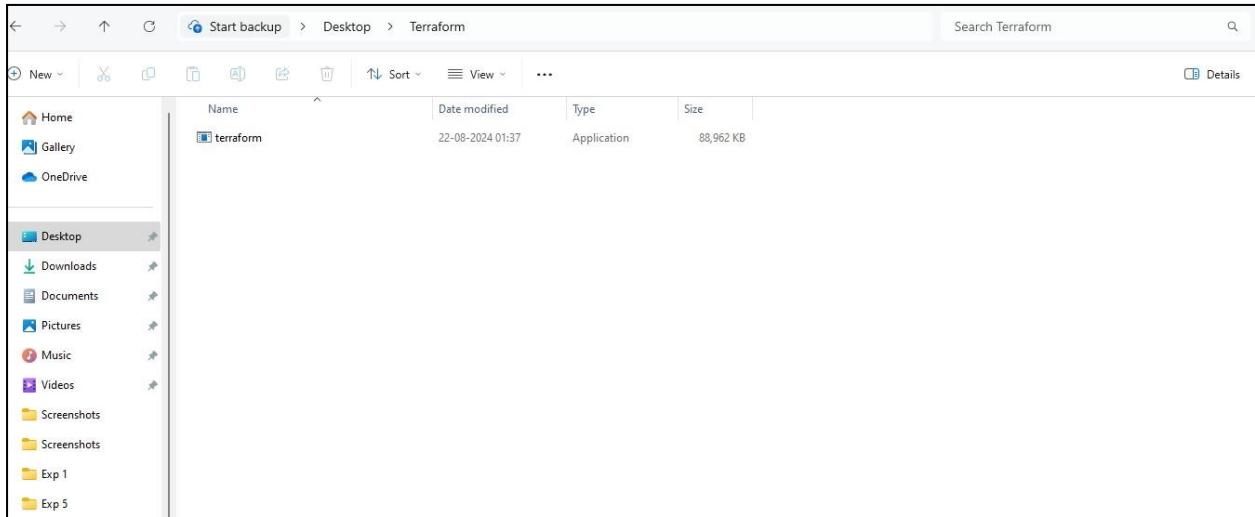
Step 1:Go to the website [terraform.io](https://www.terraform.io) and install Terraform from there..Select the AMD64 option for Windows and download Terraform.

The screenshot shows the Terraform website's 'Install Terraform' page for macOS. On the left, a sidebar lists 'Operating Systems' including macOS, Windows, Linux, FreeBSD, OpenBSD, and Solaris. The 'macOS' section is highlighted. The main content area has a title 'Install Terraform' with a version '1.9.5 (latest)'. Below this, the 'macOS' section is expanded, showing 'Package manager' with the command 'brew tap hashicorp/tap' and 'Binary download' options for 'AMD64' and 'ARM64'. To the right, a sidebar titled 'About Terraform' defines it as a tool for defining cloud and on-prem resources in human-readable configuration files. A 'Featured docs' sidebar includes links to 'Introduction to Terraform', 'Configuration Language', 'Terraform CLI', 'HCP Terraform', and 'Provider Use'. At the bottom right is a 'HCP Terraform' button.

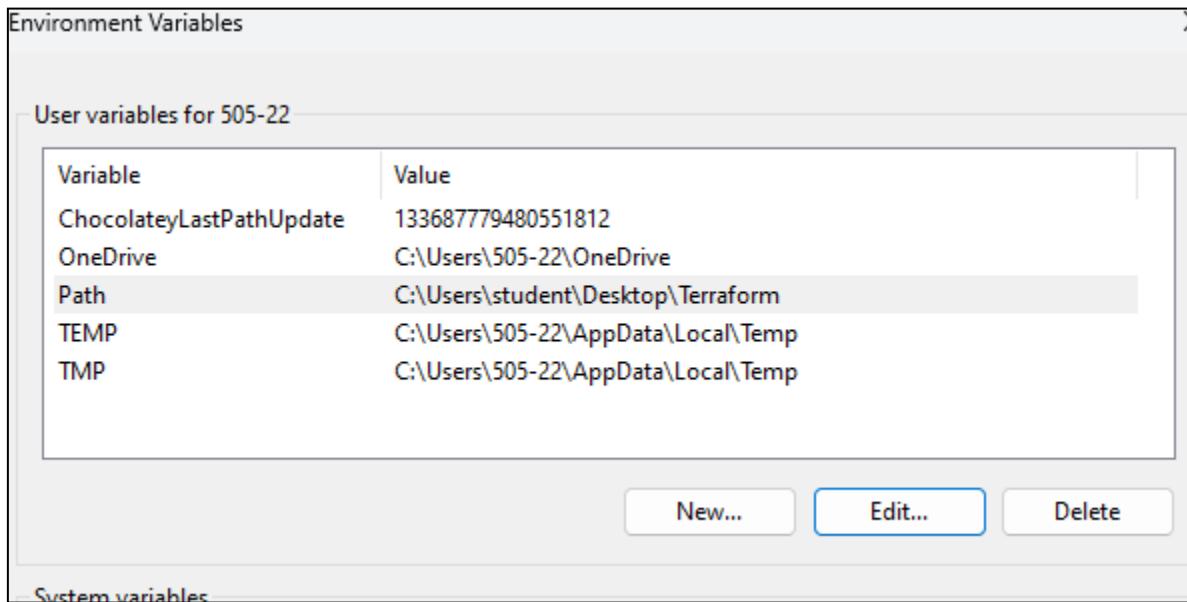
Step 2:Go to the zip file where Terraform is installed.

Name	Type	Compressed size	Password ...	Size	Ratio	Date modified
LICENSE	Text Document	2 KB	No	5 KB	63%	20-08-2024 17:35
terraform	Application	26,719 KB	No	88,962 KB	70%	20-08-2024 17:35

Step 3: Since the installed file is a zip file, create a new folder on desktop and copy the installed terraform application there.



Step 4: Now go to search bar, select edit environment variables option, then go to the path option. Now add the file path of the directory wherein we have installed the terraform application.



Step 5: Now go to the folder where we have installed terraform and open Powershell inside it. After this type ‘terraform’ to make sure that terraform has been installed on the system.

The command ‘terraform –version’ simply checks the current version of terraform that has been installed.

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

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

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

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

```
Global options (use these before the subcommand, if any):
  -chdir=DIR  Switch to a different working directory before executing the
              given subcommand.
  -help       Show this help output, or the help for a specified subcommand.
  -version    An alias for the "version" subcommand.

PS C:\Users\student\Desktop\Terraform> terraform --version
Terraform v1.9.4
on windows_amd64

Your version of Terraform is out of date! The latest version
is 1.9.5. You can update by downloading from https://www.terraform.io/downloads.html
PS C:\Users\student\Desktop\Terraform>
```

ADVANCE DEVOPS EXP 6

Name:Laksh Vijay Sodhai

Class:D15A

Roll No:59

Aim:To Build, change, and destroy AWS / GCP /Microsoft Azure/ DigitalOcean infrastructure Using Terraform.

(S3 bucket or Docker) fdp.

Part A:Creating docker image using terraform

Prerequisite:

- 1) Download and Install Docker Desktop from <https://www.docker.com/>

Step 1:Check Docker functionality

```
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\Users\student>docker

Usage: docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Common Commands:
  run      Create and run a new container from an image
  exec    Execute a command in a running container
  ps       List containers
  build   Build an image from a Dockerfile
  pull    Download an image from a registry
  push    Upload an image to a registry
  images  List images
  login   Log in to a registry
  logout  Log out from a registry
  search  Search Docker Hub for images
  version Show the Docker version information
  info    Display system-wide information

Management Commands:
  builder  Manage builds
  buildx*  Docker Buildx
  checkpoint  Manage checkpoints
  compose*  Docker Compose
  container  Manage containers
  context    Manage contexts
  debug*    Get a shell into any image or container
  desktop*  Docker Desktop commands (Alpha)
  dev*     Docker Dev Environments
  extension* Manages Docker extensions
  feedback* Provide feedback, right in your terminal!
```

Check for the docker version with the following command.

```
C:\Users\student>docker --version  
Docker version 27.1.1, build 6312585  
  
C:\Users\student>
```

Now, create a folder named ‘Terraform Scripts’ in which we save our different types of scripts which will be further used in this experiment.

Step 2: Firstly create a new folder named ‘Docker’ in the ‘TerraformScripts’ folder. Then create a new docker.tf file using Atom editor and write the following contents into it to create a Ubuntu Linux container.

Script:

```
terraform {  
    required_providers {  
        docker = {  
            source = "kreuzwerker/docker"  
            version = "2.21.0"  
        }  
    }  
}  
  
provider "docker" {  
    host = "npipe://./pipe/docker_engine"  
}  
  
# Pull the image  
resource "docker_image" "ubuntu" {  
    name = "ubuntu:latest"  
}  
  
# Create a container  
resource "docker_container" "foo" {  
    image = docker_image.ubuntu.image_id  
    name = "foo"  
    command = ["sleep", "3600"]
```

```
}
```

```
"` docker.tf  x
` docker.tf
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 # Pull 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   command = ["sleep", "3600"]
24 }
25 |
```

Step 3: Execute Terraform Init command to initialize the resources

```
● PS C:\Users\Admin\TerraformScripts> cd Docker
● PS C:\Users\Admin\TerraformScripts\Docke> 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.

Step 4: Execute Terraform plan to see the available resources

```
PS C:\Users\Admin\TerraformScripts\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         = [
        + "sleep",
        + "3600",
    ]
    + 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
+ stdin_open       = false
+ stop_signal      = (known after apply)
+ stop_timeout     = (known after apply)
+ tty               = false

+ healthcheck (known after apply)

+ labels (known after apply)
}
```

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

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

Step 5: Execute Terraform apply to apply the configuration, which will automatically create and run the Ubuntu Linux container based on our configuration. Using command : “**terraform apply**”

```
● PS C:\Users\Admin\TerraformScripts\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         = [
        + "sleep",
        + "3600",
    ]
    + 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
+ stdin_open      = false
+ stop_signal     = (known after apply)
+ stop_timeout    = (known after apply)
+ tty              = false

+ healthcheck (known after apply)

+ labels (known after apply)
}
```

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

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

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: Creation complete after 9s [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_container.foo: Creating...
docker_container.foo: Creation complete after 2s [id=01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

```

Docker images, Before Executing Apply step:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
------------	-----	----------	---------	------

Docker images, After Executing Apply step:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
ubuntu	latest	edbfe74c41f8	3 weeks ago	78.1MB

Step 6: Execute Terraform destroy to delete the configuration, which will automatically delete the Ubuntu Container.

```

● PS C:\Users\Admin\TerraformScripts\Docker> terraform destroy
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_container.foo: Refreshing state... [id=01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24]

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

Terraform will perform the following actions:

# docker_container.foo will be destroyed
- resource "docker_container" "foo" {
    - attach          = false -> null
    - command        = [
        - "sleep",
        - "3600",
    ] -> null
    - cpu_shares     = 0 -> null
    - dns            = [] -> null
    - dns_opts       = [] -> null
    - dns_search     = [] -> null
    - entrypoint     = [] -> null
    - env            = [] -> null
    - gateway        = "172.17.0.1" -> null
    - group_add      = [] -> null
    - hostname       = "01adf07e5918" -> null
    - id             = "01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24" -> null
    - image          = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
    - init           = false -> null
    - ip_address     = "172.17.0.2" -> null
    - ip_prefix_length = 16 -> null
    - ipc_mode       = "private" -> null
    - links          = [] -> null
    - log_driver     = "json-file" -> null
    - log_opts        = {} -> null
    - logs           = false -> null
    - max_retry_count = 0 -> null
}

```

```

- memory          = 0 -> null
- memory_swap    = 0 -> null
- must_run       = true -> null
- name           = "foo" -> null
- network_data   = [
  - {
    - gateway        = "172.17.0.1"
    - global_ipv6_prefix_length = 0
    - ip_address     = "172.17.0.2"
    - ip_prefix_length = 16
    - network_name   = "bridge"
    # (2 unchanged attributes hidden)
  },
  ] -> null
- network_mode   = "default" -> null
- privileged      = false -> null
- publish_all_ports = false -> null
- read_only       = false -> null
- remove_volumes = true -> null
- restart         = "no" -> null
- rm              = false -> null
- runtime         = "runc" -> null
- security_opts  = [] -> null
- shm_size        = 64 -> null
- start           = true -> null
- stdin_open      = false -> null
- stop_timeout    = 0 -> null
- storage_opts   = {} -> null
- sysctls         = {} -> null
- tmpfs           = {} -> null
- tty              = false -> null
# (8 unchanged attributes hidden)
}

```

```

# docker_image.ubuntu will be destroyed
resource "docker_image" "ubuntu" {
  - id      = "sha256:edbf74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest" -> null
  - image_id = "sha256:edbf74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
  - latest   = "sha256:edbf74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a" -> null
  - name     = "ubuntu:latest" -> null
  - repo_digest = "ubuntu@sha256:8a37d68f4f73ebf3d4efafbcf66379bf3728902a8038616808f04e34a9ab63ee" -> null
}

```

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

Do you really want to destroy all resources?

Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

```

docker_container.foo: Destroying... [id=01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24]
docker_container.foo: Destruction complete after 0s
docker_image.ubuntu: Destroying... [id=sha256:edbf74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_image.ubuntu: Destruction complete after 1s

```

Destroy complete! Resources: 2 destroyed.

Docker images After Executing Destroy step

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE

ADVANCE DEVOPS EXP 7

Name:Laksh Sodhai
Class:D15A
Roll No:59

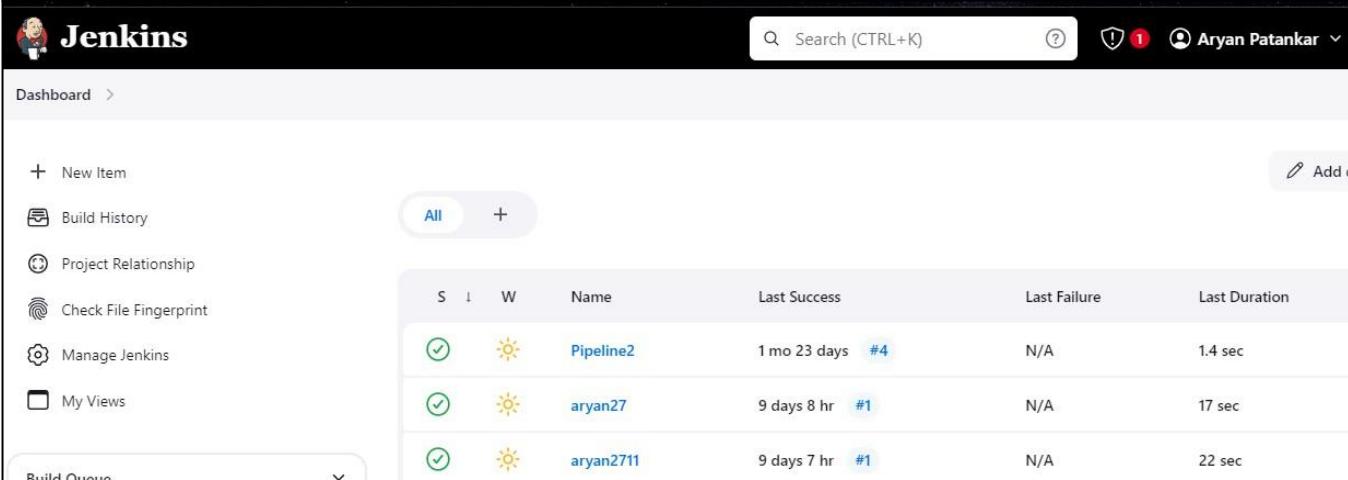
Aim: To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

Integrating Jenkins with SonarQube:

- Jenkins installed
- Docker Installed (for SonarQube)
- SonarQube Docker Image

Steps to integrate Jenkins with SonarQube

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



The screenshot shows the Jenkins dashboard with the following interface elements:

- Header:** Jenkins logo, Search bar (CTRL+K), Notifications (1), User Aryan Patankar.
- Left Sidebar:** Dashboard, New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, My Views, Build Queue.
- Center Content:** A table listing three Jenkins pipelines:

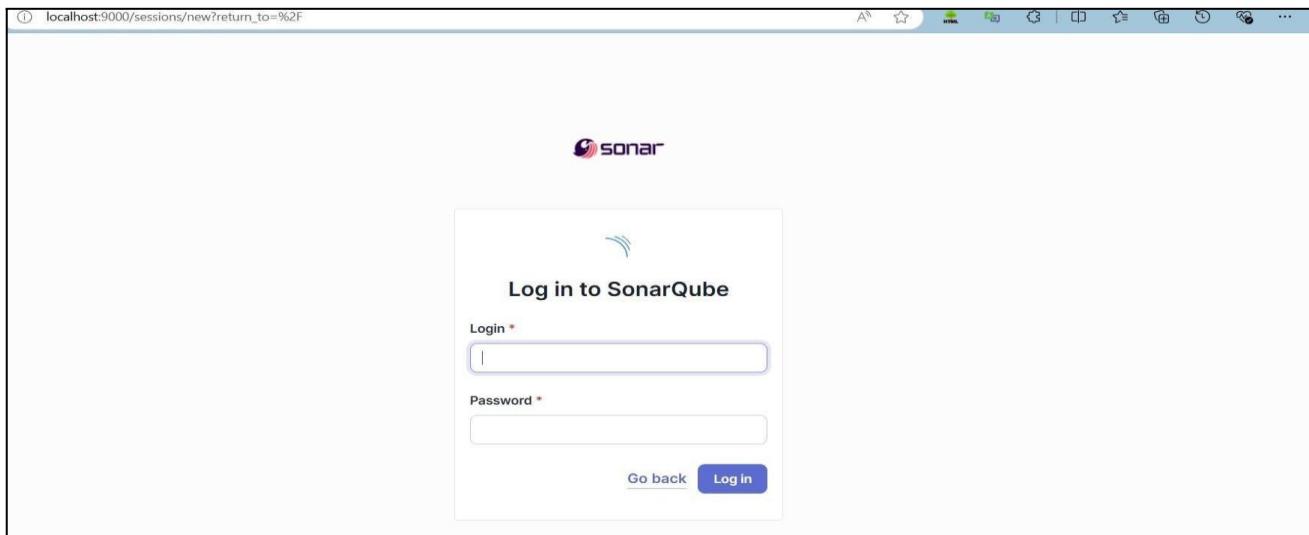
S	I	W	Name	Last Success	Last Failure	Last Duration
✓	Pipeline2	1 mo 23 days #4	N/A	1.4 sec
✓	aryan27	9 days 8 hr #1	N/A	17 sec
✓	aryan2711	9 days 7 hr #1	N/A	22 sec

2. Run SonarQube in a Docker container using this command -

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

```
PS C:\Windows\system32> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest  
Unable to find image 'sonarqube:latest' locally  
latest: Pulling from library/sonarqube  
7478e0ac0f23: Pull complete  
90a925ab929a: Pull complete  
7d9a34308537: Pull complete  
80338217a4ab: Pull complete  
1a5fd5c7e184: Pull complete  
7b87d6fa783d: Pull complete  
bd819c9b5ead: Pull complete  
4f4fb700ef54: Pull complete  
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde  
Status: Downloaded newer image for sonarqube:latest  
5ab3928e5e27607e3661d129731e4e600a9019574c7dc2767aa9b3bfdaa941be
```

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



4. Login to SonarQube using username admin and password admin.

sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More Q

How do you want to create your project?

Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform.

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

Import from Azure DevOps Import from Bitbucket Cloud Import from Bitbucket Server
Import from GitHub Import from GitLab

Are you just testing or have an advanced use-case? Create a local project.

Create a local project

5. 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](#)

Set up project for Clean as You Code

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

Choose the baseline for new code for this project

Use the global setting

Previous version
Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

Define a specific setting for this project

Previous version
Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

Number of days
Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will be closed.
Recommended for projects following continuous delivery.

Setup the project and come back to Jenkins Dashboard.

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

The screenshot shows the Jenkins 'Manage Jenkins' interface under the 'Plugins' section. A search bar at the top contains the text 'sonarq'. Below the search bar, there are tabs for 'Updates' (25), 'Available plugins' (selected), 'Installed plugins', and 'Advanced settings'. The main area displays a table for the 'SonarQube Scanner' plugin, version 2.17.2. The table includes columns for 'Install' (checkbox), 'Name ↓' (SonarQube Scanner), 'Type' (External Site/Tool Integrations), 'Build Reports', 'Status' (Released), and 'Last Published' (6 mo 29 days ago). A note below the table states: 'This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.'

The screenshot shows the Jenkins 'Manage Jenkins' interface under the 'Plugins' section. A sidebar on the left lists 'Updates' (25), 'Available plugins' (selected), 'Installed plugins', and 'Advanced settings'. The main area is titled 'Download progress' and shows the status of the SonarQube Scanner plugin. It indicates 'Preparation' with three steps: 'Checking internet connectivity', 'Checking update center connectivity', and 'Success'. It also shows 'SonarQube Scanner' with a green checkmark and 'Success', and 'Loading plugin extensions' with another green checkmark and 'Success'. At the bottom, there are links to 'Go back to the top page' (with a note '(you can start using the installed plugins right away)') and a checkbox for 'Restart Jenkins when installation is complete and no jobs are running'.

6. Under Jenkins ‘Manage Jenkins’ then go to ‘system’, scroll and look for **SonarQube Servers**

and enter the details.

Enter the Server Authentication token if needed.

In SonarQube installations: Under **Name** add <project name of sonarqube>, here we have named it as **adv_devops_7_sonarqube**

In **Server URL** Default is <http://localhost:9000>

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

adv_devops_7_sonarqube

Server URL

Default is http://localhost:9000

https://localhost:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

+ Add +

Advanced ▾

7. Search for SonarQube Scanner under Global Tool Configuration.

Choose the latest configuration and choose Install automatically.

Dashboard > Manage Jenkins > Tools

The screenshot shows the Jenkins 'Tools' configuration page. It includes sections for 'Gradle installations', 'SonarScanner for MSBuild installations', 'SonarQube Scanner installations', and 'Ant installations'. Each section has a 'Add [Tool]' button. The 'SonarQube Scanner installations' section is currently selected, indicated by a blue border around its header.

Check the “Install automatically” option. → Under name any name as identifier → Check the “Install automatically” option.

The screenshot shows the 'SonarQube Scanner installations' configuration screen. It displays a list of existing configurations. One configuration, named 'sonarqube_exp7', has the 'Install automatically' checkbox checked. The 'Install from Maven Central' section is expanded, showing the selected version 'SonarQube Scanner 6.1.0.4477'. There is also an 'Add Installer' dropdown menu.

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

adv_devops_exp7
» 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
Creates a set of Pipeline projects according to detected branches in one SCM repository.

OK

9. Choose this GitHub repository in Source Code Management.

https://github.com/shazforiot/MSBuild_firstproject.git

It is a sample hello-world project with no vulnerabilities and issues, just to test the integration.

The screenshot shows the 'Source Code Management' configuration page. The 'Git' option is selected. In the 'Repositories' section, a repository URL is entered: https://github.com/shazforiot/MSBuild_firstproject.git. The 'Credentials' dropdown is set to '- none -'. There is an 'Advanced' button and a 'Add Repository' button at the bottom.

10. Under **Select project → Configuration → Build steps → Execute SonarQube Scanner**, enter these Analysis properties. Mention the SonarQube Project Key, Login, Password, Source path and Host URL.

The screenshot shows the 'Configure' screen with the 'Build Environment' section selected. A dropdown menu is open, listing various build steps: Execute SonarQube Scanner, Execute Windows batch command, Execute shell, Invoke Ant, Invoke Gradle script, Invoke top-level Maven targets, Run with timeout, Set build status to "pending" on GitHub commit, SonarScanner for MSBuild - Begin Analysis, and SonarScanner for MSBuild - End Analysis. At the bottom of the dropdown is a 'Add build step ^' button. Below the dropdown, there is a 'Post-build Actions' section.

Execute SonarQube Scanner

JDK ?
JDK to be used for this SonarQube analysis
(Inherit From Job)

Path to project properties ?
[Empty input field]

Analysis properties ?

```
sonar.projectKey=adv_devops_7_sonarqube
sonar.host.url=http://localhost:9000
sonar.login=admin
sonar.sources=.
```

Additional arguments ?
[Empty input field]

JVM Options ?
[Empty input field]

Then save

Status **adv_devops_exp7**

- </> Changes
- Workspace
- Build Now
- Configure
- Delete Project
- SonarQube
- Rename

Add description Disable Project

SonarQube

Permalinks

- Last build (#2), 1 day 20 hr ago
- Last stable build (#2), 1 day 20 hr ago
- Last successful build (#2), 1 day 20 hr ago
- Last completed build (#2), 1 day 20 hr ago

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

Configuration Security Projects System Marketplace

	Administer System	Administer	Execute Analysis	Create
sonar-administrators System administrators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Quality Gates <input checked="" type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input checked="" type="checkbox"/> Projects
sonar-users Every authenticated user automatically belongs to this group	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Projects
Anyone DEPRECATED Anybody who browses the application belongs to this group. If authentication is not enforced, assigned permissions also apply to non-authenticated users.	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input type="checkbox"/> Projects
Administrator admin	<input checked="" type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input type="checkbox"/> Projects

IF CONSOLE OUTPUT FAILED:

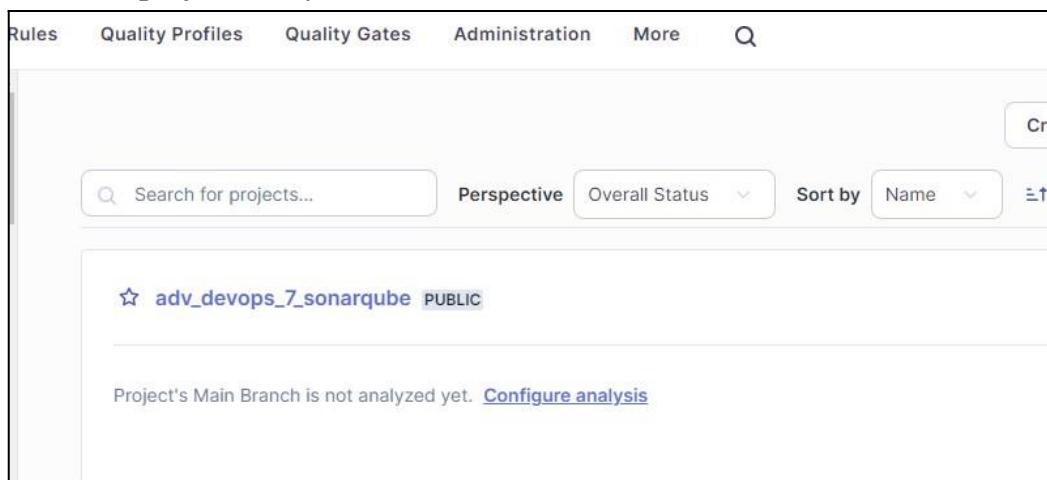
Step 1: Generate a New Authentication Token in SonarQube

1. Login to SonarQube:

- Open your browser and go to **http://localhost:9000**.
- Log in with your admin credentials (default username is **admin**, and the password is either **admin** or your custom password if it was changed).

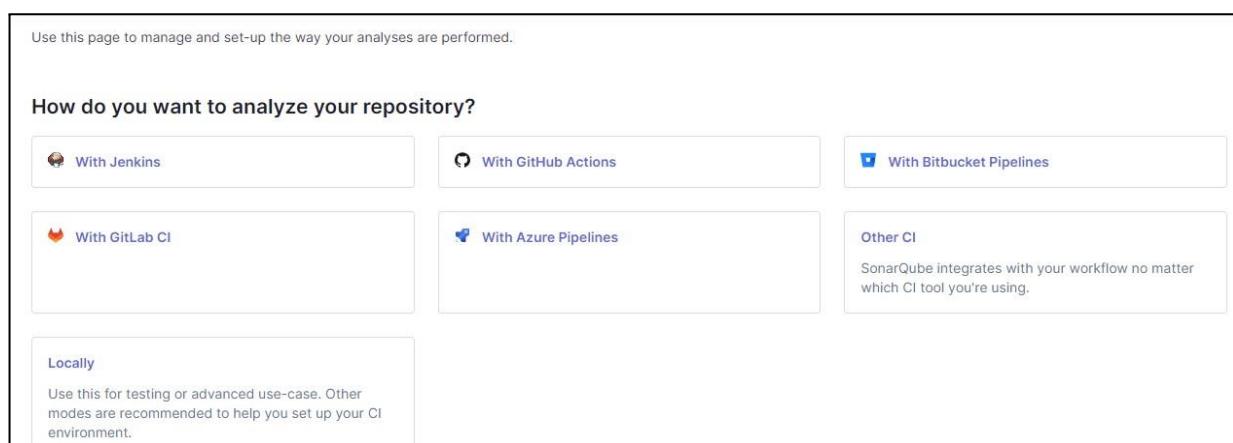
2. Generate a New Token:

- Go to the project that you have created on SonarQube.



The screenshot shows the SonarQube web interface. At the top, there is a navigation bar with links for Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. Below the navigation bar is a search bar labeled "Search for projects..." and a dropdown menu for "Perspective". Further down are dropdown menus for "Overall Status" and "Sort by", and a dropdown menu for "Name". The main content area displays a single project entry: "adv_devops_7_sonarqube PUBLIC". Below the project name, a message states: "Project's Main Branch is not analyzed yet. [Configure analysis](#)".

- Click on **Locally**



The screenshot shows the SonarQube CI setup interface. At the top, a message says: "Use this page to manage and set-up the way your analyses are performed." Below this, a section titled "How do you want to analyze your repository?" contains several options: "With Jenkins", "With GitHub Actions", "With Bitbucket Pipelines", "With GitLab CI", "With Azure Pipelines", and "Other CI". The "Locally" option is highlighted with a box around it. A note next to "Locally" says: "Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI environment." To the right of the "Other CI" button, there is a note: "SonarQube integrates with your workflow no matter which CI tool you're using."

- Further, Generate a Project token with the following details and click on generate.

1 Provide a token

[Generate a project token](#) [Use existing token](#)

Token name ?	Expires in
"adv_devops_7.sonarqube"	1 year ▼
Generate	

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

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

- Copy the token you get here and save it securely as we would need it in Jenkins.

1 Provide a token

"adv_devops_7.sonarqube": sqp_bfa5258ea4fd254f00c3d1d4e64205ebefcdd027 [✖](#)

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

[Continue](#)

Step 2: Update the Token in Jenkins

1. Go to Jenkins Dashboard:

- Open Jenkins and log in with your credentials.

The screenshot shows the Jenkins dashboard with the following details:

- Dashboard:** Shows a summary of recent builds and pipeline status.
- Left sidebar:**
 - + New Item
 - Build History
 - All
 - +
 - Project Relationship
 - Check File Fingerprint
 - Manage Jenkins
 - My Views
- Build Queue:** A dropdown menu showing the current build queue.
- Table:** A table listing recent builds for the Pipeline2 project.

S	I	W	Name	Last Success	Last Failure	Last Duration
✓	☀️	Pipeline2	1 mo 23 days	#4	N/A	1.4 sec
✓	☀️	aryan27	9 days 8 hr	#1	N/A	17 sec
✓	☀️	aryan2711	9 days 7 hr	#1	N/A	22 sec
- Header:** Includes the Jenkins logo, search bar, and user profile of Aryan Patankar.

2. Go to Dashboard—>Manage Jenkins—>Credentials

The screenshot shows the Jenkins 'Credentials' page under 'Manage Jenkins'. A single credential is listed:

T	P	Store ↓	Domain	ID	Name
		System	(global)	sonarqube_token	/*****

Below this, there's a section titled 'Stores scoped to Jenkins' with one entry:

P	Store ↓	Domains
	System	(global)

At the bottom, there are icons for 'Icon:', 'S', 'M', and 'L'.

3. Click on **global** under the domains part of Stores scoped to Jenkins section.Further click on add credentials.Proceed with the following details.Make sure to copy the token generated earlier in sonarqube and give any suitable name as the ID.

The screenshot shows the 'Add Credential' form in Jenkins:

- Kind:** Secret text
- Scope:** Global (Jenkins, nodes, items, all child items, etc)
- Secret:** (redacted)
- ID:** sonarqube-exp7
- Description:** advance devops exp7

A blue 'Create' button is at the bottom left.

4. After clicking on create we see that the given token has been added in Jenkins credentials.

The screenshot shows the 'Global credentials (unrestricted)' page in Jenkins:

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description
sonarqube-exp	advance devops exp7	Secret text	advance devops exp7

A blue '+ Add Credentials' button is at the top right.

5. Now go to **Manage Jenkins**—>**System**—>**SonarQube servers** and proceed with the following details. Reference the authentication token generated in the previous step.

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	adv_devops_7_sonarqube
Server URL	Default is http://localhost:9000 http://localhost:9000
Server authentication token	SonarQube authentication token. Mandatory when anonymous access is disabled. advance devops exp7
+ Add ▾	

6. Check the SonarQube Scanner Environment and add the server authentication token

Build Environment

Delete workspace before build starts

Use secret text(s) or file(s) ?

Add timestamps to the Console Output

Inspect build log for published build scans

Prepare SonarQube Scanner environment ?

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled. Will default to the one defined in the SonarQube installation.
advance devops exp7

+ Add ▾

Execute SonarQube Scanner

JDK ?
JDK to be used for this SonarQube analysis
(Inherit From Job)

Path to project properties ?

Analysis properties ?
sonar.projectKey=adv_devops_7_sonarqube
sonar.host.url=http://localhost:9000
-Dsonar.login=sqp_568834b7b5e77a92843e4b3072e044643ce921c1
sonar.sources=.

Additional arguments ?

JVM Options ?

12. Run the Jenkins build.

Dashboard > adv_devops_exp7 >

Status  **adv_devops_exp7**

 **SonarQube Quality Gate**

adv_devops_7_sonarqube  **Passed**
server-side processing:  **Success**

Permalinks

- Last build (#6), 1 min 55 sec ago
- Last stable build (#6), 1 min 55 sec ago
- Last successful build (#6), 1 min 55 sec ago
- Last failed build (#5), 17 min ago
- Last unsuccessful build (#5), 17 min ago
- Last completed build (#6), 1 min 55 sec ago

Build History 

 Filter... /

 **#6**
Sep 25, 2024, 10:04 PM 

Check the console Output

The screenshot shows the Jenkins 'Console Output' page for build #6 of the 'adv_devops_exp7' project. The left sidebar includes links for Status, Changes, Console Output (which is selected), Edit Build Information, Delete build '#6', and Timings. The main content area displays the following log output:

```
Started by user Aryan Patankar
Running as SYSTEM
Building on the built-in node in workspace C:\ProgramData\Jenkins\.jenkins\workspace\adv_devops_exp7
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\adv_devops_exp7\.git # timeout=10
Fetching changes from the remote Git repository
```

13. Once the build is complete, check project on SonarQube

The screenshot shows the SonarQube main dashboard for the 'adv_devops_7.sonarqube' project. The top navigation bar includes links for Overview, Issues, Security Hotspots, Measures, Code, Activity, Project Settings, and Project Information. The main content area shows the 'main' branch with a green 'Passed' status for the Quality Gate. It also displays the following code metrics:

Category	Value	Status
New Code	0 H, 0 M, 0 L	A
Overall Code	0 H, 0 M, 0 L	A
Security	0 Open issues	A
Reliability	0 Open issues	A
Maintainability	1 Open issue	A

In this way, we have integrated Jenkins with SonarQube for SAST.

ADVANCE DEVOPS EXP 8

Name:Laksh Sodhai

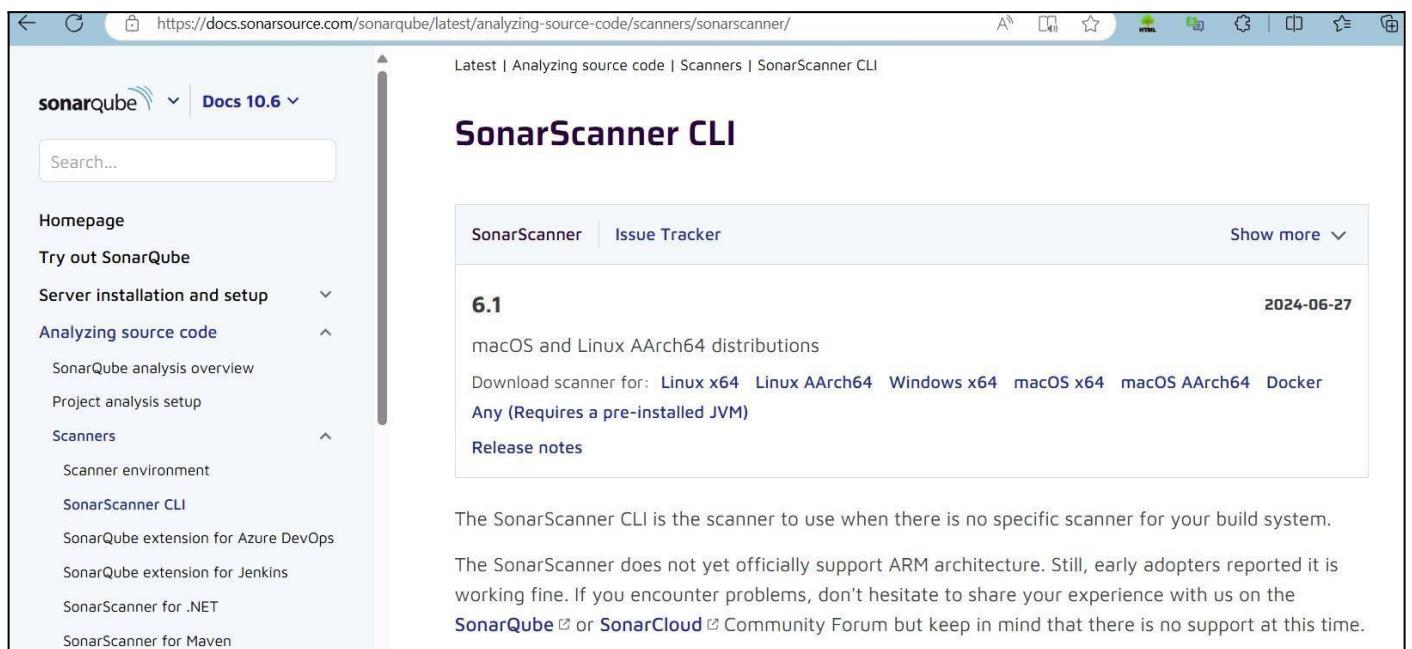
Class:D15A

Roll No:59

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: Download sonar scanner

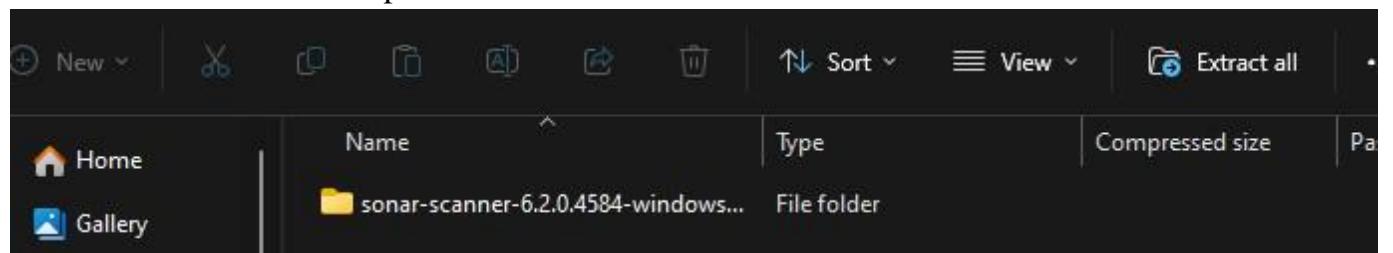
<https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscan>



The screenshot shows a web browser displaying the SonarScanner CLI documentation. The URL in the address bar is <https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscan>. The page title is "SonarScanner CLI". On the left, there is a sidebar with navigation links for "Homepage", "Try out SonarQube", "Server installation and setup", "Analyzing source code" (which is expanded to show "SonarQube analysis overview", "Project analysis setup", "Scanners" (expanded to show "Scanner environment", "SonarScanner CLI", "SonarQube extension for Azure DevOps", "SonarQube extension for Jenkins", "SonarScanner for .NET", "SonarScanner for Maven")), and "SonarScanner CLI". The main content area contains a section for "SonarScanner" and "Issue Tracker", a release note for version 6.1 (published 2024-06-27) which supports macOS and Linux AArch64 distributions, download links for various platforms, and a link to "Release notes". Below this, there is a note about the SonarScanner CLI being used when no specific scanner is available, and another note about the SonarScanner not supporting ARM architecture yet.

ner/ Visit this link and download the sonarqube scanner CLI.

Extract the downloaded zip file in a folder.



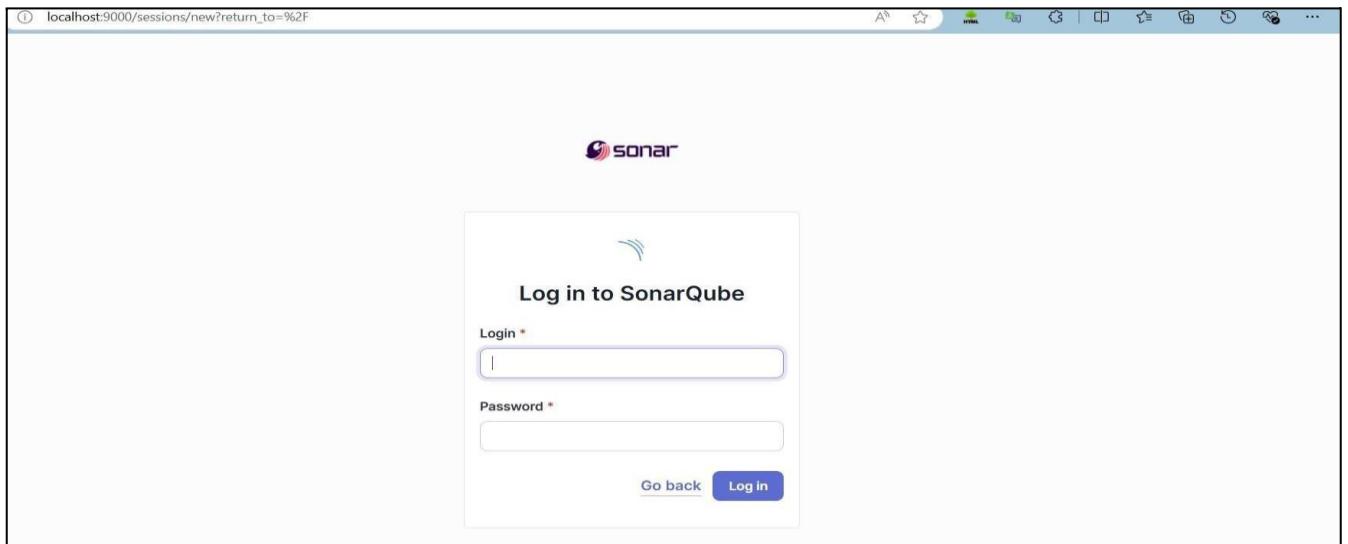
1. Install sonarqube image

Command: **docker pull**

sonarqube

```
C:\Windows\System32>docker pull sonarqube
Using default tag: latest
latest: Pulling from library/sonarqube
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Image is up to date for sonarqube:latest
docker.io/library/sonarqube:latest
```

- Once the container is up and running, you can check the status of



SonarQube at localhost port 9000.

3. Login to SonarQube using username admin and password admin.

A screenshot of the SonarQube interface. The top navigation bar includes links for 'Projects', 'Issues', 'Rules', 'Quality Profiles', 'Quality Gates', 'Administration', 'More', and a search icon. The main content area is titled 'How do you want to create your project?'. It asks if the user wants to benefit from SonarQube's features like repository import and Pull Request decoration, and suggests creating a project from a favorite DevOps platform. It then provides five options for importing projects: 'Import from Azure DevOps' (Setup), 'Import from Bitbucket Cloud' (Setup), 'Import from Bitbucket Server' (Setup), 'Import from GitHub' (Setup), and 'Import from GitLab' (Setup). At the bottom, there is a note about testing or advanced use-cases, followed by a large button labeled 'Create a local project'.

4. 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](#) 

[Cancel](#)

[Next](#)

2 of 2

Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code. This helps you focus on the Clean as You Code methodology. Learn more: [Defining New Code](#) 

Choose the baseline for new code for this project

Use the global setting

[Previous version](#)

Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

Define a specific setting for this project

[Previous version](#)

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

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

The screenshot shows the Jenkins dashboard with the following details:

- Left sidebar:** Includes links for "New Item", "Build History", "Project Relationship", "Check File Fingerprint", and "Manage Jenkins".
- Build Queue:** Shows "No builds in the queue."
- Build Executor Status:** Shows 1 Idle and 2 Idle nodes, with one node labeled "(offline)".
- Main Content:** A table listing build jobs:

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀️	Devops Pipeline	1 mo 13 days #4	N/A	0.61 sec
✓	☀️	devops_exp6_pipeline	24 days #1	N/A	2.2 sec
✓	☁️	maven_exp_6	17 days #13	17 days #12	9.2 sec
✗	☁️	maven_project	1 mo 13 days #3	1 mo 7 days #10	12 sec
✓	☀️	myNewJob	24 days #1	N/A	0.49 sec

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

The screenshot shows the Jenkins Manage Jenkins > Plugins page with the following details:

- Left sidebar:** Includes links for "Updates", "Available plugins" (selected), "Installed plugins", and "Advanced settings".
- Search bar:** Contains the text "sonarq".
- Plugin list:** Shows the "SonarQube Scanner" plugin version 2.17.2, released 6 months 29 days ago. It includes categories "External Site/Fool Integrations" and "Build Reports". A note states: "This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality." An "Install" button is present.

The screenshot shows the Jenkins Manage Jenkins > Plugins > Download progress page with the following details:

- Left sidebar:** Includes links for "Updates", "Available plugins" (selected), "Installed plugins", and "Advanced settings".
- Right sidebar:** Shows "Download progress" with sections for "Preparation" and "SonarQube Scanner".
 - Preparation:** Lists steps: "Checking internet connectivity", "Checking update center connectivity", and "Success".
 - SonarQube Scanner:** Shows "Success" for "Loading plugin extensions".
- Bottom:** Buttons for "Go back to the top page" and "Restart Jenkins when installation is complete and no jobs are running".

7. Under Jenkins ‘Manage Jenkins’ then go to ‘system’, scroll and look for **SonarQube Servers**

and enter the details.

Enter the Server Authentication token if needed.

In SonarQube installations: Under **Name** add <project name of sonarqube> for me
adv_devops_7_sonarqube

In **Server URL** Default is <http://localhost:9000>



The screenshot shows the Jenkins 'Manage Jenkins' interface, specifically the 'System' configuration page. It displays the configuration for a SonarQube server. The fields are as follows:

- Name:** sonarqube
- Server URL:** http://localhost:9000 (Default is http://localhost:9000)
- Server authentication token:** - none - (dropdown menu)
+ Add ▾ (button)
- Advanced ▾** (button)

8. Search for SonarQube Scanner under Global Tool Configuration.

Choose the latest configuration and choose Install automatically.

Dashboard > Manage Jenkins > Tools

The screenshot shows the Jenkins 'Tools' configuration page. At the top, there is a breadcrumb navigation: Dashboard > Manage Jenkins > Tools. Below the navigation, there are several sections for different build tools:

- Gradle installations**: Contains a button labeled "Add Gradle".
- SonarScanner for MSBuild installations**: Contains a button labeled "Add SonarScanner for MSBuild".
- SonarQube Scanner installations**: Contains a button labeled "Add SonarQube Scanner".
- Ant installations**: This section is currently selected, indicated by a blue border around its header.

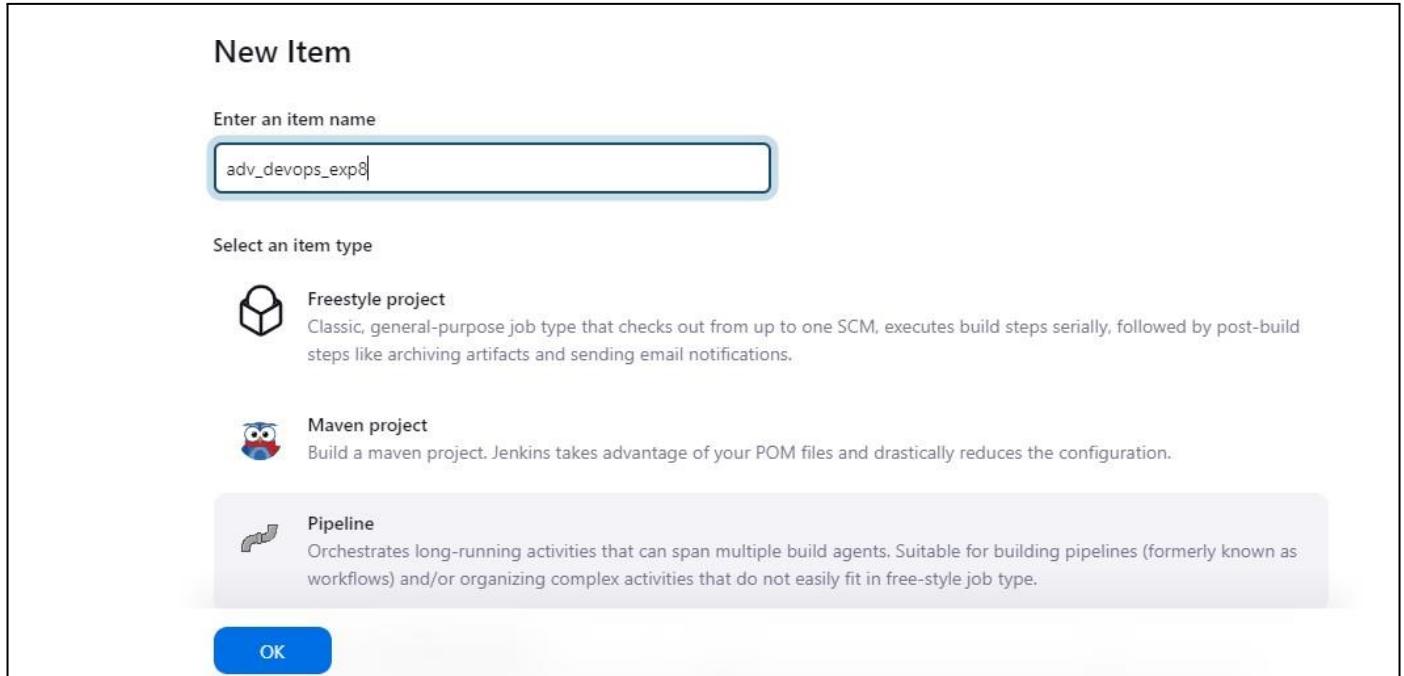
Check the “Install automatically” option. → Under name any name as identifier → Check

The screenshot shows the configuration dialog for the SonarQube Scanner tool. The title bar says "SonarQube Scanner". The form fields are as follows:

- Name**: A text input field containing "sonarqube_exp8".
- Install automatically**: A checkbox that is checked.
- Install from Maven Central**: A collapsed section header.
- Version**: A dropdown menu showing "SonarQube Scanner 6.2.0.4584".
- Add Installer**: A button to add more installers.

the “Install automatically” option.

9. After configuration, create a New Item → choose a pipeline project.



10. Under Pipeline script, enter the following:

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

  stage('SonarQube analysis') {
    withSonarQubeEnv('<Name_of_SonarQube_environment_on_Jenki
ns>') {
      sh """
        <PATH_TO SONARQUBE SCANNER FOLDER>/bin/sonar-scanner \
        -D sonar.login=<SonarQube_USERNAME> \
        -D sonar.password=<SonarQube_PASSWORD> \
        -D sonar.projectKey=<Project_KEY> \
        -D sonar.exclusions=vendor/**,resources/**,*/*.java \
        -D sonar.host.url=<SonarQube_URL>(default: http://localhost:9000/
      """
    }
  }
}
```

}

It is a java sample project which has a lot of repetitions and issues that will be detected by SonarQube.

Definition

Pipeline script

Script ?

```
1 > node {  
2 >   stage('Cloning the GitHub Repo') {  
3 >     git 'https://github.com/shazforiot/GOL.git'  
4 >   }  
5  
6 >   stage('SonarQube analysis') { withSonarQubeEnv('<Name_of_SonarQube_environment_on_Jenkins>') {  
7 >     sh """  
8 >       <PATH_TO SONARQUBE_SCANNER_FOLDER>/bin/sonar-scanner \  
9 >       -D sonar.login=admin \  
10 >      -D sonar.password=admin> \  
11 >      -D sonar.projectKey=sonarqube \  
12 >      -D sonar.exclusions=vendor/**,resources/**,**/*.java \  
13 >      -D sonar.host.url=http://localhost:9000  
14 >    """  
15 >  }  
16 > }  
17 > }  
18 >
```

Use Groovy Sandbox ?

[Pipeline Syntax](#)

11. Build project

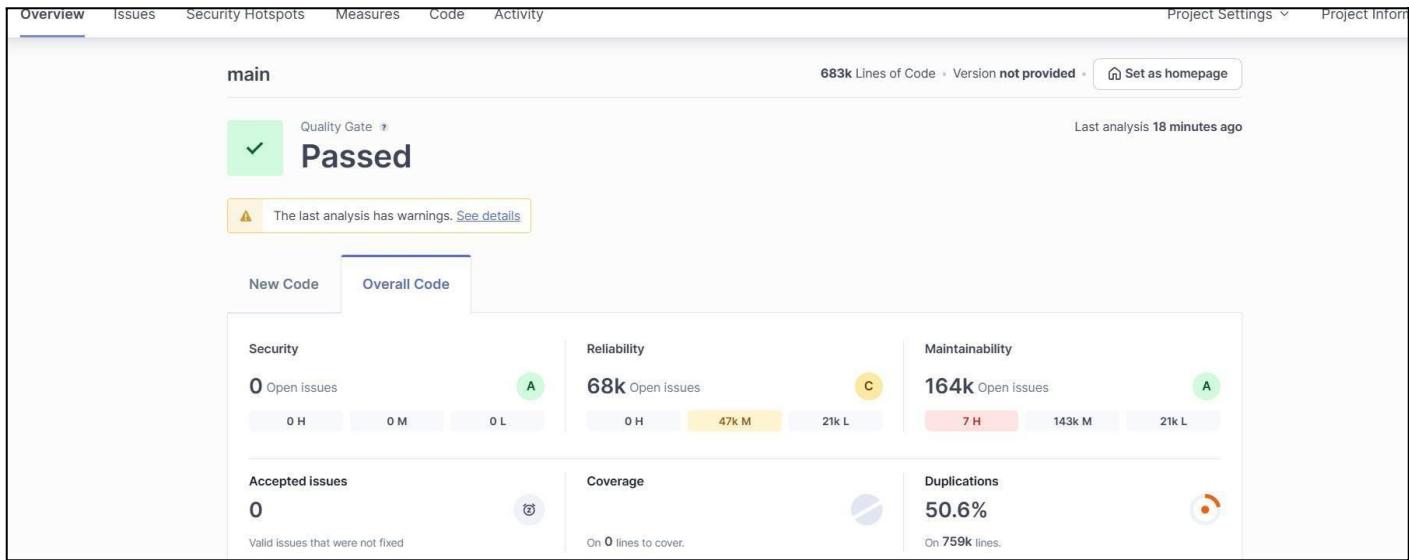
The screenshot shows the Jenkins Pipeline interface for the project 'adv_devops_exp8'. On the left, there's a sidebar with various options like Status, Changes, Build Now, Configure, Delete Pipeline, Full Stage View, SonarQube, Stages, Rename, and Pipeline Syntax. Below that is a 'Build History' section with a dropdown for 'trend' and a 'Filter...' input. A recent entry for build #9 is shown with a timestamp of Sep 18, 16:14, and a note 'No Changes'. The main area is titled 'Stage View' and displays three stages: 'Cloning the GitHub Repo' (3s), 'SonarQube analysis' (40s), and 'Deployment' (6min 2s). The 'Deployment' stage is currently running and has a progress bar indicating it's at 1s. A tooltip for the 'Deployment' stage says 'Deployment failed'. The overall average stage time is 3s, and the average full run time is approximately 6 minutes and 4 seconds.

12. Check console

The screenshot shows the Jenkins Pipeline interface for the project 'adv_devops_exp8'. The sidebar on the left includes options like Status, Changes, Console Output (which is selected), View as plain text, Edit Build Information, Delete build '#9', Timings, Git Build Data, Pipeline Overview, Pipeline Console, Replay, Pipeline Steps, Workspaces, and Previous Build. The main area is titled 'Console Output' and shows the log for build #9. The log starts with a note about skipping 4,246 KB of full log. It then lists several warning messages from JMeter's PropertyControlGui.html file, specifically regarding too many duplication references. The log continues with similar warnings for other files like 'gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html'. The log ends with a final warning message at the bottom.

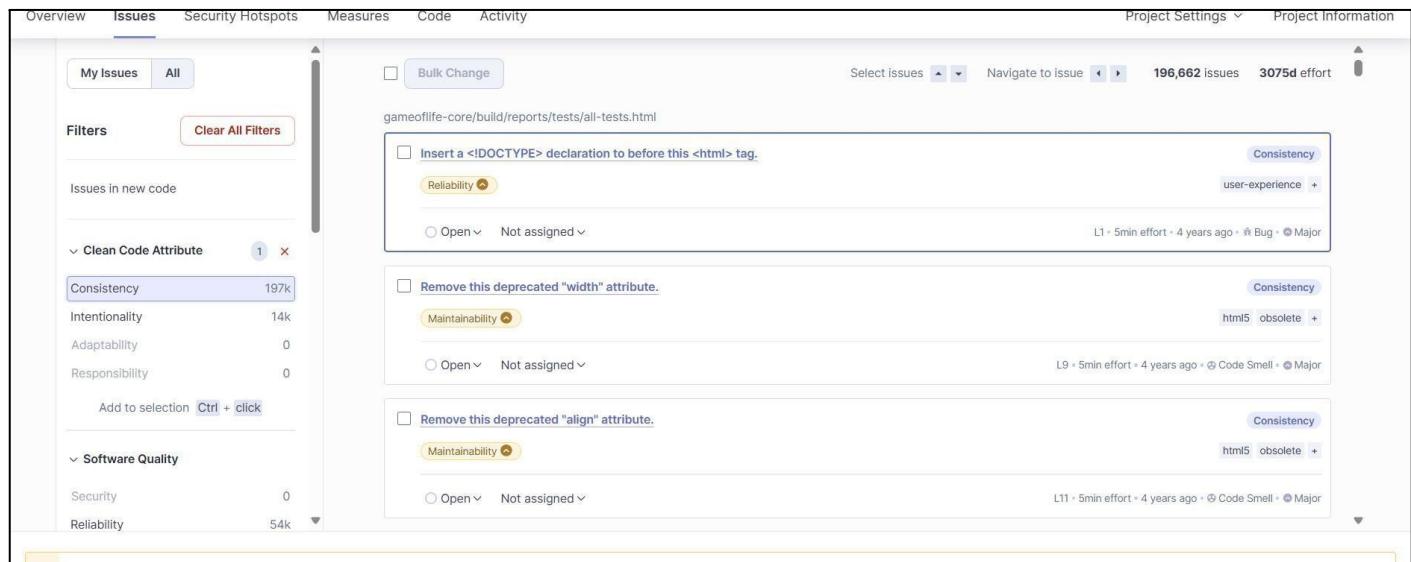
```
Skipping 4,246 KB.. Full Log
16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 512. Keep only the first 100 references.
16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 248. Keep only the first 100 references.
16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 886. Keep only the first 100 references.
16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 249. Keep only the first 100 references.
16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 662. Keep only the first 100 references.
16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 615. Keep only the first 100 references.
16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 664. Keep only the first 100 references.
16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 913. Keep only the first 100 references.
16:19:49.752 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 810. Keep only the first 100 references.
16:19:49.752 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 668. Keep only the first 100 references.
16:19:49.752 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 548. Keep only the first 100 references.
16:19:49.752 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 543. Keep only the first 100 references.
16:19:49.752 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 152. Keep only the first 100 references.
16:19:49.752 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line
```

13. Now, check the project in SonarQube



14. Code Problems

• Consistency



Intentionality

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

My Issues All

Bulk Change

Select issues ▾ Navigate to issue ▾ 13,887 issues 59d effort

Filters Clear All Filters

Issues in new code

✓ Clean Code Attribute 1 1 X

Consistency 197k

Intentionality 14k

Adaptability 0

Responsibility 0

Add to selection Ctrl + click

✓ Software Quality

Security 0

Reliability 14k

gameoflife-acceptance-tests/Dockerfile

Use a specific version tag for the image. Intentionality

Maintainability No tags +

Open Not assigned L1 - 5min effort 4 years ago ⚡ Code Smell ⚡ Major

Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. Intentionality

Maintainability No tags +

Open Not assigned L12 - 5min effort 4 years ago ⚡ Code Smell ⚡ Major

Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. Intentionality

Maintainability No tags +

Open Not assigned L12 - 5min effort 4 years ago ⚡ Code Smell ⚡ Major

This screenshot shows a software interface for code review and quality assurance. The top navigation bar includes links for Overview, Issues, Security Hotspots, Measures, Code, Activity, Project Settings, and Project Information. The Issues tab is selected. On the left, there's a sidebar with 'My Issues' and 'All' buttons, followed by 'Filters' and a 'Clear All Filters' button. Below that are sections for 'Issues in new code', 'Clean Code Attribute' (with sub-options for Consistency, Intentionality, Adaptability, and Responsibility), and 'Software Quality' (with sub-options for Security and Reliability). A note says 'Add to selection Ctrl + click'. The main content area displays a list of issues under the heading 'gameoflife-acceptance-tests/Dockerfile'. Each issue has a checkbox, a title, a severity level (L1 or L12), an effort estimate (e.g., '5min effort'), a timestamp ('4 years ago'), and a tag list ('⚡ Code Smell ⚡ Major'). The first issue is 'Use a specific version tag for the image.' and is categorized under 'Intentionality' and 'Maintainability'. The second and third issues are variations of 'Surround this variable with double quotes; otherwise, it can lead to unexpected behavior.' and are also categorized under 'Intentionality' and 'Maintainability'.

Bugs

gameoflife-core/build/reports/tests/all-tests.html

Add "lang" and/or "xml:lang" attributes to this "<html>" element Intentionality
Reliability accessibility wcag2-a +

Open ▾ Not assigned ▾ L1 × 2min effort × 4 years ago × Bug × Major

Insert a <!DOCTYPE> declaration to before this <html> tag. Consistency
Reliability user-experience +

Open ▾ Not assigned ▾ L1 × 5min effort × 4 years ago × Bug × Major

Add "<th>" headers to this "<table>". Intentionality
Reliability accessibility wcag2-a +

Open ▾ Not assigned ▾ L9 × 2min effort × 4 years ago × Bug × Major

Code Smells

gameoflife-acceptance-tests/Dockerfile

Use a specific version tag for the image. Intentionality
Maintainability No tags +

Open ▾ Not assigned ▾ L1 × 5min effort × 4 years ago × Code Smell × Major

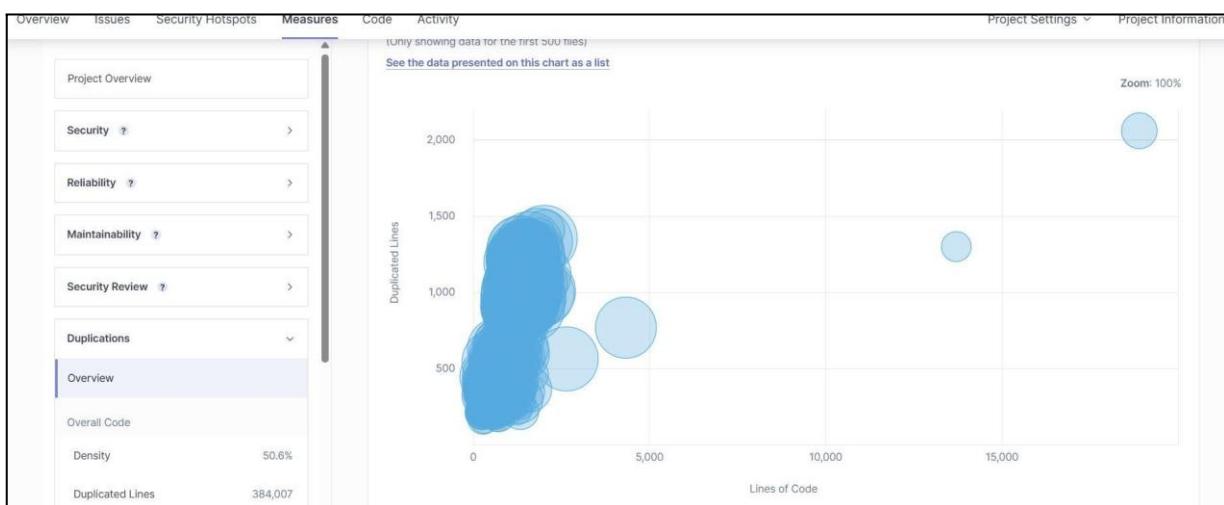
Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. Intentionality
Maintainability No tags +

Open ▾ Not assigned ▾ L12 × 5min effort × 4 years ago × Code Smell × Major

Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. Intentionality
Maintainability No tags +

Open ▾ Not assigned ▾ L12 × 5min effort × 4 years ago × Code Smell × Major

Duplications



Cyclomatic Complexities

The screenshot shows the SonarQube interface for the 'gameoflife' project. The top navigation bar includes 'Overview', 'Issues', 'Security Hotspots', 'Measures' (selected), 'Code', and 'Activity'. On the right, there are 'Project Settings' and 'Project Information' dropdowns. The main content area is titled 'Cyclomatic Complexity 1,112 See history'. It lists six components with their respective complexity counts: 'gameoflife-acceptance-tests' (18), 'gameoflife-build' (18), 'gameoflife-core' (18), 'gameoflife-deploy' (18), 'gameoflife-web' (1,094), and 'pom.xml' (18). A note at the bottom indicates '6 of 6 shown'. On the left, a sidebar lists various measures: Security, Reliability, Maintainability, Security Review, Duplications, Size, Complexity (selected), and Cyclomatic Complexity (1,112).

In this way, we have integrated Jenkins with SonarQube for SAST.

ADVANCE DEVOPS EXPERIMENT 9

Name:Laksh .V. Sodhai

Class:D15A

Roll No:59

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

Step 1: Create an Amazon Linux EC2 instance and name it as nagios-host

Instances (1) Info		Last updated 1 minute ago	Connect	Instance state ▾	Actions ▾	Launch instances ▾	Edit filters
		<input type="text"/> Find Instance by attribute or tag (case-sensitive)		All states ▾			
<input type="checkbox"/>	Name ↗	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability zone
<input type="checkbox"/>	nagios-host	i-08373a53cb8045f0a	Running Details Logs	t2.micro	Initializing	View alarms +	ap-south-1

Step 2:Edit the following inbound rules of the specified security groups and ensure HTTP,HTTPS,SSH,ICMP are accessible from anywhere

Inbound rules (7)						C	Manage tags	Edit inbound rules
						Search	< 1 >	Edit
▼	Security group rule...	IP version	Type	Protocol	Port range			
	sgr-0842dcf237958c987	IPv4	HTTPS	TCP	443			
	sgr-0e3b5fe756fe77f0a	IPv4	All traffic	All	All			
	sgr-07c7572562bdb3...	IPv4	Custom TCP	TCP	0			
	sgr-07882e9275b39c4...	IPv4	HTTP	TCP	80			
	sgr-08540b31df42cc513	IPv4	All ICMP - IPv4	ICMP	All			
	sgr-0dcbe24f99412dcfb	IPv6	Custom TCP	TCP	0			
	sgr-09ccae5af38c85345	IPv6	All ICMP - IPv6	IPv6 ICMP	All			

Step 3: Connect to your EC2 instance via the connect option available in EC2 instances menu

```
[ec2-user@ip-172-31-33-14 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:19:23 ago on Thu Sep 26 08:42:17 2024.
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
Installing:				
httpd	x86_64	2.4.62-1.amzn2023	amazonlinux	48 k
php0.3	x86_64	8.3.10-1.amzn2023.0.1	amazonlinux	10 k
Installing dependencies:				
apr	x86_64	1.7.2-2.amzn2023.0.2	amazonlinux	129 k
apr-util	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	98 k
generic-logos-httdp	noarch	18.0.0-12.amzn2023.0.3	amazonlinux	19 k
httpd-core	x86_64	2.4.62-1.amzn2023	amazonlinux	1.4 M
httpd-filesystem	noarch	2.4.62-1.amzn2023	amazonlinux	14 k
httpd-tools	x86_64	2.4.62-1.amzn2023	amazonlinux	81 k
libbrotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	315 k
libsodium	x86_64	1.0.19-4.amzn2023	amazonlinux	176 k
libxslt	x86_64	1.1.34-5.amzn2023.0.2	amazonlinux	241 k
mailman	noarch	2.1.49-2.amzn2023.0.3	amazonlinux	33 k

Step 4: Update and install the required packages

Use the following commands:

sudo yum update

sudo yum install httpd php

sudo yum install gcc glibc glibc-common

sudo yum install gd gd-devel

```
[ec2-user@ip-172-31-33-14 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:20:32 ago on Thu Sep 26 08:42:17 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	Repository	Size
Installing:				
gcc	x86_64	11.4.1-2.amzn2023.0.2	amazonlinux	32 M
Installing dependencies:				
annobin-docs	noarch	10.93-1.amzn2023.0.1	amazonlinux	92 k
annobin-plugin-gcc	x86_64	10.93-1.amzn2023.0.1	amazonlinux	887 k
cpp	x86_64	11.4.1-2.amzn2023.0.2	amazonlinux	10 M
gc	x86_64	8.0.4-5.amzn2023.0.2	amazonlinux	105 k
glibc-devel	x86_64	2.34-52.amzn2023.0.11	amazonlinux	27 k
glibc-headers-x86	noarch	2.34-52.amzn2023.0.11	amazonlinux	427 k
guile22	x86_64	2.2.7-2.amzn2023.0.3	amazonlinux	6.4 M
kernel-headers	x86_64	6.1.109-118.189.amzn2023	amazonlinux	1.4 M
libmpc	x86_64	1.2.1-2.amzn2023.0.2	amazonlinux	62 k
libtool-ltdl	x86_64	2.4.7-1.amzn2023.0.3	amazonlinux	38 k
libxml-crypt-devel	x86_64	4.4.33-7.amzn2023	amazonlinux	32 k

```
[ec2-user@ip-172-31-33-14 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:21:27 ago on Thu Sep 26 08:42:17 2024.
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
Installing:				
gd	x86_64	2.3.3-5.amzn2023.0.3	amazonlinux	139 k
gd-devel	x86_64	2.3.3-5.amzn2023.0.3	amazonlinux	38 k
Installing dependencies:				
brotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	314 k
brotli-devel	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	31 k
bzip2-devel	x86_64	1.0.8-6.amzn2023.0.2	amazonlinux	214 k
cairo	x86_64	1.17.6-2.amzn2023.0.1	amazonlinux	684 k
cmake-filesystem	x86_64	3.22.2-1.amzn2023.0.4	amazonlinux	16 k
fontconfig	x86_64	2.13.94-2.amzn2023.0.2	amazonlinux	273 k
fontconfig-devel	x86_64	2.13.94-2.amzn2023.0.2	amazonlinux	128 k
fonts-filesystem	noarch	1:2.0.5-12.amzn2023.0.2	amazonlinux	9.5 k
fontstmp	x86_64	2.12.2-5.amzn2022.0.1	amazonlinux	422 k

Step 5: Create a new nagios user by writing the following commands

```
sudo adduser -m nagios
```

```
sudo passwd nagios
```

```
Complete!
[ec2-user@ip-172-31-33-14 ~]$ sudo adduser -m nagios
[ec2-user@ip-172-31-33-14 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-33-14 ~]$ █
```

Step 6: Create a new user group using **sudo groupadd nagcmd** and

Add users to the group using the following commands:

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

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

```
Complete!
[ec2-user@ip-172-31-33-14 ~]$ sudo adduser -m nagios
[ec2-user@ip-172-31-33-14 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-33-14 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-33-14 ~]$ sudo usermod -a -G nagcmd nagios
[ec2-user@ip-172-31-33-14 ~]$ sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-33-14 ~]$ mkdir downloads
[ec2-user@ip-172-31-33-14 ~]$ cd downloads
[ec2-user@ip-172-31-33-14 downloads]$ wget https://sourceforge.net/projects/nagios/files/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz/download?use_mirror=excellmedia
--2024-09-26 09:15:54-- https://sourceforge.net/projects/nagios/files/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz/download?use_mirror=excellmedia
Resolving sourceforge.net (sourceforge.net)... 172.64.150.145, 104.18.37.111, 2606:4700:4400::6012:256f, ...
Connecting to sourceforge.net (sourceforge.net)|172.64.150.145|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz?ts=gAAAAABm9SzKFW7LwD1QAJ2jNzqmSJwAPAlmQ-eAJYK8z5Nmrv ifVkhbsV-qOfPsLUyICC6yvdHu6UeeIyvNzsVGUtr9BeQ%3D&use_mirror=excellmedia&r= [following]
```

Step 7: Create a directory for Nagios downloads using the following commands-

Commands -

```
mkdir ~/downloads
```

```
cd ~/downloads
```

Also download Nagios and plugin 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
```

```

Connecting to prdownloads.sourceforge.net (prdownloads.sourceforge.net)|204.68.111.105|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz [following]
--2024-09-26 09:38:43-- https://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz
Resolving downloads.sourceforge.net (downloads.sourceforge.net)... 204.68.111.105
Connecting to downloads.sourceforge.net (downloads.sourceforge.net)|204.68.111.105|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://excellmedia.dl.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz?viasf=1 [following]
--2024-09-26 09:38:45-- https://excellmedia.dl.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz?viasf=1
Resolving excellmedia.dl.sourceforge.net (excellmedia.dl.sourceforge.net)... 202.153.32.19, 2401:fb00:0:1fe:8000::5
Connecting to excellmedia.dl.sourceforge.net (excellmedia.dl.sourceforge.net)|202.153.32.19|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1805059 (1.7M) [application/x-gzip]
Saving to: 'nagios-4.0.8.tar.gz'

nagios-4.0.8.tar.gz          100%[=====]   1.72M  8.14MB/s    in 0.2s

2024-09-26 09:38:45 (8.14 MB/s) - 'nagios-4.0.8.tar.gz' saved [1805059/1805059]

[ec2-user@ip-172-31-33-14 downloads]$ ls
'download?use_mirror=excellmedia'  nagios-4.0.8.tar.gz
[ec2-user@ip-172-31-33-14 downloads]$ tar -xzf nagios-4.0.8.tar.gz
[ec2-user@ip-172-31-33-14 downloads]$ []

```

Step 8-Extract the nagios source file with the following commands

tar zxvf nagios-4.4.6.tar.gz

cd nagios-4.4.6

Then run the configuration script with the following command

/configure --with-command-group=nagcmd

```

Nagios user/group: nagios,nagios
Command user/group: nagios,nagcmd
Event Broker: yes
Install ${prefix}: /usr/local/nagios
Install ${includedir}: /usr/local/nagios/include/nagios
Lock file: ${prefix}/var/nagios.lock
Check result directory: ${prefix}/var/spool/checkresults
Init directory: /etc/rc.d/init.d
Apache conf.d directory: /etc/httpd/conf.d
Mail program: /bin/mail
Host OS: linux-gnu
IOBroker Method: epoll

```

Web Interface Options:

```

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

```

Review the options above for accuracy. If they look okay,
type 'make all' to compile the main program and CGIs.

```
[ec2-user@ip-172-31-33-14 nagios-4.0.8]$ ]
```

Step 9-Compile the source code with the following commands
make all

```
[ec2-user@ip-172-31-33-14 nagios-4.0.8]$ make all
cd ./base && make
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/base'
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o broker.o broker.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nebmods.o nebmods.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ../common/shared.o ../common/shared.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nerd.o nerd.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o query-handler.o query-handler.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
In function 'get_wproc_list',
  inlined from 'get_worker' at workers.c:224:12:
workers.c:209:17: warning: '%s' directive argument is null [-Wformat-overflow=]
  209 |         log_debug_info(DEBUGL_CHECKS, 1, "Found specialized worker(s) for '%s'", (slash && *slash != '/') ? slash : cmd_name);
               ^~~~~~
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o checks.o checks.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o config.o config.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o commands.o commands.c
commands.c: In function 'process_passive_service_check':
commands.c:2247:19: warning: assignment discards 'const' qualifier from pointer target type [-Wdiscarded-qualifiers]
```

Step 10-Install binaries,init script and sample config files

Commands -

./sudo make install

sudo make install-init

sudo make install-config

sudo make install-commandmode

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

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

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

*** External command directory configured ***

[ec2-user@ip-172-31-33-14 nagios-4.0.8]$ ]
```

Step 11-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

Step 12-Configure the Web Interface

Commands -

sudo make install-webconf

```

GNU nano 5.8                               [Alt+S]                               /usr/local/nagios/etc/objects/contacts.cfg
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                nagios@localhost ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
}

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

[] We only have one contact in this simple configuration file, so there is

^G Help      ^C Write Out    ^W Where Is      ^K Cut        ^T Execute      ^C Location      M-U Undo      M-A Set Mark    M-[ To Bracket
^X Exit      ^R Read File    ^\ Replace       ^U Paste      ^J Justify      ^/ Go To Line    M-E Redo      M-G Copy       ^C Where Was

```

Step 13-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

```

GNU nano 5.8                               [Alt+S]                               /usr/local/nagios/etc/objects/contacts.cfg
Modified

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                vaishnal16305@gmail.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
}

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

# We only have one contact in this simple configuration file, so there is

^G Help      ^C Write Out    ^W Where Is      ^K Cut        ^T Execute      ^C Location      M-U Undo      M-A Set Mark    M-[ To Bracket
^X Exit      ^R Read File    ^\ Replace       ^U Paste      ^J Justify      ^/ Go To Line    M-E Redo      M-G Copy       ^C Where Was

```

Step 14-. Extract the Plugins Source File

Commands -

cd ~/downloads

tar zxvf nagios-plugins-2.3.3.tar.gz

cd nagios-plugins-2.3.3

```

*** External command directory configured ***

[ec2-user@ip-172-31-33-14 nagios-4.0.8]$ sudo nano /usr/local/nagios/etc/objects/contacts.cfg
[ec2-user@ip-172-31-33-14 nagios-4.0.8]$ sudo make install-webconf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/httpd/conf.d/nagios.conf

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

```

Step 15-19. Compile and Install Plugins

Commands -

```
./configure --with-nagios-user=nagios --with-nagios-group=nagios
```

```
make
```

```
sudo make install
```

```
[ec2-user@ip-172-31-33-14 nagios-4.0.8]$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
```

Step 16-Start Nagios

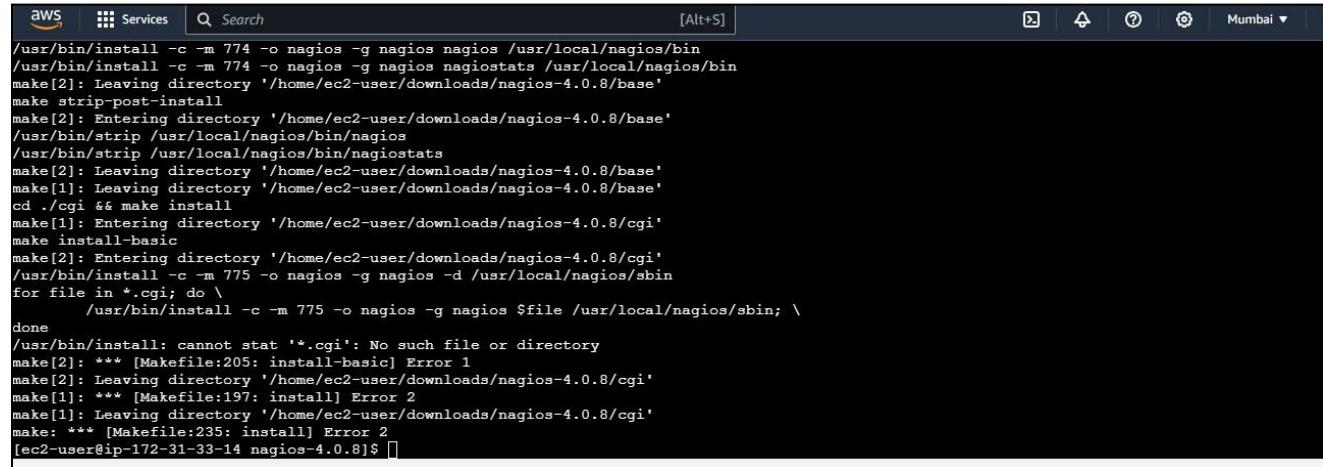
Commands -

```
sudo chkconfig --add nagios
```

```
sudo chkconfig nagios on
```

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

```
sudo systemctl start nagios
```



The screenshot shows a terminal window within the AWS Cloud9 IDE. The title bar includes the AWS logo, Services, Search, and [Alt+S]. The main area displays the command-line output of the nagios installation process. The output shows the compilation of the source code, the creation of binary files (nagios, nagiosstats, etc.), and the installation of these files into the /usr/local/nagios/bin directory. It also shows the creation of CGI scripts in the /usr/local/nagios/cgi directory. The process ends with an error message indicating that a file named *.cgi does not exist, which prevents the final step of the installation. The command entered was 'sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg'.

Step 17-Access Nagios Web Interface

- Copy the Public IP address of your EC2 instance.
- Open your browser and navigate to <http://nagios>.
- Enter the username nagiosadmin and the password you set in Step 16.

Nagios®

General

- Home
- Documentation

Current Status

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

Quick Search:

Reports

- Availability
- Trends (Legacy)
- Alerts
 - History
 - Summary
 - Histogram (Legacy)

Nagios® Core™

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



A new version of Nagios Core is available!
[Visit nagios.org to download Nagios 4.5.5.](#)

Get Started <ul style="list-style-type: none">Start monitoring your infrastructureChange the look and feel of NagiosExtend Nagios with hundreds of addonsGet supportGet trainingGet certified	Quick Links <ul style="list-style-type: none">Nagios Library (tutorials and docs)Nagios Labs (development blog)Nagios Exchange (plugins and addons)Nagios Support (tech support)Nagios.com (company)Nagios.org (project)	
Latest News	Don't Miss...	

ADVANCE DEVOPS EXPERIMENT 10

Name:Laksh .V. Sodhai
Class;D15A
Roll No:59

1) Launch an instance

Launch an ec2 instance.

Select Ubuntu as the os give a meaningful name of the instance.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The process is at the 'Name and tags' step. A single instance named 'exp10client' is selected. The 'Summary' panel on the right provides details: 1 instance, Canonical, Ubuntu, 24.04, ami-0e86e20dae9224db8, t2.micro instance type, launch-wizard-5 security group, and 1 volume(s) - 8 GiB storage. A note about the free tier is also present.

EC2 > Instances > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name Add additional tags

Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

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

Recent AMIs: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux Enterprise Server

Quick Start: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux Enterprise Server

Browse more AMIs Including AMIs from AWS, Marketplace and the Community

Summary

Number of instances Info 1

Software Image (AMI) Canonical, Ubuntu, 24.04, ami-0e86e20dae9224db8

Virtual server type (instance type) t2.micro

Firewall (security group) launch-wizard-5

Storage (volumes) 1 volume(s) - 8 GiB

Free tier: In your first 750 hours of t2.micro usage in the Regions in which it's available, you can launch up to 10 (unavailable) instance AMIs per month, receive a public IPv4 address upon launch, 30 GB of EBS storage per month, 30 million I/Os, 1 GB of traffic per month, and 100 GB of bandwidth per month to the internet.

Cancel

Select the same security group as given in exp9.

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

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

Recents Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux

Browse more AMIs Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-0e86e20dae9224db8 (64-bit (x86)) / ami-096ea6a12ea24a797 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible ▾

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

Architecture AMI ID Username | ⓘ Verified provider

64-bit (x86) ami-0e86e20dae9224db8 ubuntu

Cancel

▼ Summary

Number of instances 1

Software Images Canonical, Ubuntu, ami-0e86e20dae9224db8

Virtual server type t2.micro

Firewall (security group) launch-wizard-1

Storage (volumes) 1 volume(s) - 80 GB

Free tier eligible 750 hours per month, the Reg public IP per month, million IOPS, 100 GB internet

Make sure to select the same key-pair login used in the exp9 machine.

Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

nagios_exp_9 [Create new key pair](#)

Network settings [Info](#)

Network [Info](#)
vpc-07b6966cbfba88ee3

Subnet [Info](#)
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)
Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group Select existing security group

Common security groups [Info](#)

Select security groups [▼](#)

click on launch instance.

Now connect with this client machine using the ssh through your terminal(open a new terminal in your local machine and we will need both of the terminals open)

Instances (1/5) Info							
Find Instance by attribute or tag (case-sensitive)		All states ▼					
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
Master	i-0ab175e9c60cc3a23	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-3-82-156-160.com...
node-1	i-08ad30b7114767ca2	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-3-85-110-80.com...
node-2	i-03c70d364fb762af5	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-54-226-209-38.co...
nagios_host_e...	i-0820376be204a7fcf	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-54-224-175-95.co...
exp10client	i-0994ca5a178801a54	Running	t2.micro	Initializing	View alarms +	us-east-1b	ec2-54-173-58-143.co...

EC2 > Instances > i-0994ca5a178801a54 > Connect to instance

Connect to instance Info

Connect to your instance i-0994ca5a178801a54 (exp10client) using any of these options

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

Instance ID
 i-0994ca5a178801a54 (exp10client)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is nagios_exp_9.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
 chmod 400 "nagios_exp_9.pem"
4. Connect to your instance using its Public DNS:
 ec2-54-173-58-143.compute-1.amazonaws.com

Command copied

ssh -i "nagios_exp_9.pem" ubuntu@ec2-54-173-58-143.compute-1.amazonaws.com

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

Note to change the path of the .pem file.

```
Host          Client
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\Lenovo> ssh -i "C:\Users\Lenovo\Downloads\nagios_exp_9.pem" ubuntu@ec2-54-173-58-143.compute-1.amazonaws.com

The authenticity of host 'ec2-54-173-58-143.compute-1.amazonaws.com (54.173.58.143)' can't be established.
ED25519 key fingerprint is SHA256:IA3XH7f011spkO84wDcZFmqRgNn0iJZ7itI2pBMmHP4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-173-58-143.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

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

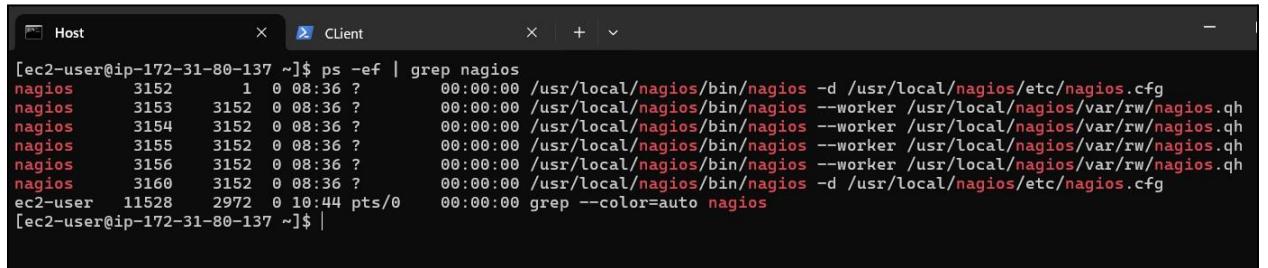
System information as of Sat Sep 28 10:43:28 UTC 2024

System load:  0.01      Processes:           107
Usage of /:   22.8% of 6.71GB  Users logged in:     0
Memory usage: 19%           IPv4 address for enX0: 172.31.82.77
```

2) Go to nagios host machine (Host machine)

Perform the following commands

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



```
[ec2-user@ip-172-31-80-137 ~]$ ps -ef | grep nagios
nagios      3152      1  0 08:36 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios      3153     3152  0 08:36 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      3154     3152  0 08:36 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      3155     3152  0 08:36 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      3156     3152  0 08:36 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      3160     3152  0 08:36 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
[ec2-user@ip-172-31-80-137 ~]$ | grep --color=auto nagios
```

```
sudo su
```

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

```
[root@ip-172-31-80-137 ec2-user]# mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-80-137 ec2-user]# ls
```

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

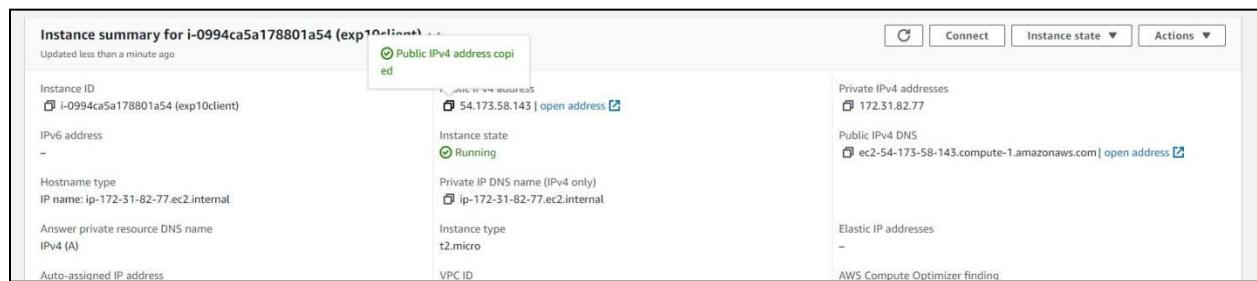
```
[root@ip-172-31-80-137 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

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

```
[root@ip-172-31-80-137 ec2-user]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

Change hostname and alias to linuxserver

Change address to public ip address of client instance (Ubuntu instance) you can get the ip address by clicking on the instance id on the instances section there you will get the public ipv4 address



```

; HOST DEFINITION
#
#####
; Define a host for the local machine

define host {

    use          linux-server           ; Name of host template to use
                                ; This host definition will in>
                                ; in (or inherited by) the lin>

    host_name    linuxserver
    alias        linuxserver
    address     54.173.58.143
}

```

Change hostgroup_name to linux-servers1

```

# Define an optional hostgroup for Linux machines

define hostgroup {

    hostgroup_name      linux-servers1      ; The name of the hostgroup
    alias               Linux Servers       ; Long name of the group
    members             localhost          ; Comma separated list of host>
}

|

```

Change the occurrences of hostname further in the document from localhost

to linuxserver

example like:

host_name	localhost
service_description	PING

changed to

```

define service {

    use          local-service           ; Name of service template
    host_name    linuxserver
    service_description PING
    check_command  check_ping!100.0,20%!500.0,60%
}

```

This is the last one

```

define service {
    use          local-service      ; Name of service template to >
    host_name    linuxserver
    service_description HTTP
    check_command check_http
    notifications enabled 0
}

```

now ctrl+O and enter to save and then ctrl+X for exiting.

Open nagios configuration file and add the line shown below
nano /usr/local/nagios/etc/nagios.cfg

```
[root@ip-172-31-80-137 ec2-user]# nano /usr/local/nagios/etc/nagios.cfg
```

##Add this line below the opened nano interface where similar lines are commented.

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

```

GNU nano 5.8                               /usr/local/nagios/etc/nagios.cfg
# These are the object configuration files in which you define hosts,
# host groups, contacts, contact groups, services, etc.
# You can split your object definitions across several config files
# if you wish (as shown below), or keep them all in a single config file.

# You can specify individual object config files as shown below:
: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/

# OBJECT CACHE FILE
# This option determines where object definitions are cached when
# Nagios starts up. The GCCE reads object definitions from
# the specified file and stores them in memory for faster access.
# The default value is /var/cache/nagios/objects.
#cfg_file=/var/cache/nagios/objects/nagios.cfg

```

ctrl+o and enter for saving and ctrl+x to exit nano editor.

Verify configuration files

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

```
[root@ip-172-31-80-137 ec2-user]# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
Nagios Core 4.5.5
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-09-17
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 0 service dependencies
  Checked 0 host dependencies
  Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...
```

```
Total Warnings: 0
Total Errors: 0
```

```
Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-80-137 ec2-user]# |
```

Restart nagios service.

```
service nagios restart
```

```
Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-80-137 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
[root@ip-172-31-80-137 ec2-user]# |
```

3) Go to client machine (ubuntu machine)

Perform the following commands

```
sudo apt update -y
```

```
sudo apt install gcc -y
```

```
sudo apt install -y nagios-nrpe-server nagios-plugins
```

The screenshot shows a terminal window with two tabs: "Host" and "Client". The "Host" tab is active and displays the following command and its output:

```
ubuntu@ip-172-31-82-77:~$ sudo apt update -y
sudo apt install gcc -y
sudo apt install -y nagios-nrpe-server nagios-plugins
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
```

Below this, the terminal continues with system status messages:

```
Running kernel seems to be up-to-date.

Restarting services...

Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart serial-getty@ttyS0.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #1: sshd[990,1101]
ubuntu @ user manager service: systemd[996]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
```

The prompt at the bottom is `ubuntu@ip-172-31-82-77:~$ |`.

Open the nrpe.cfg file in nano editor

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

Under allowed_hosts, add the nagios host ip address (public)

```
# You can either supply a username or a UID.  
#  
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd  
nrpe_user=nagios  
  
# NRPE GROUP  
# This determines the effective group that the NRPE daemon should run as.  
# You can either supply a group name or a GID.  
#  
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd  
nrpe_group=nagios  
  
# ALLOWED HOST ADDRESSES  
# This is an optional comma-delimited list of IP address or hostnames  
# that are allowed to talk to the NRPE daemon. Network addresses with a bit  
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not currently  
# supported.  
#  
# Note: The daemon only does rudimentary checking of the client's IP  
# address. I would highly recommend adding entries in your /etc/hosts.allow  
# file to allow only the specified host to connect to the port  
# you are running this daemon on.  
#  
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd  
allowed_hosts=127.0.0.1,54.224.175.95  
  
# COMMAND ARGUMENT PROCESSING  
# This option determines whether or not the NRPE daemon will allow clients
```

again save and exit the nano editor.

4) Go to nagios dashboard and click on hosts

The screenshot shows the Nagios Core 4.5.5 dashboard. At the top right, it displays "Nagios® Core™ Version 4.5.5" and the date "September 17, 2024". A green checkmark indicates "Daemon running with PID 13935". On the left, there's a sidebar with links for General, Current Status (Tactical Overview, Map, Hosts, Services, Host Groups, Grid), Service Groups (Summary, Grid), Problems (Services, Hosts, Network Outages), Reports (Availability, Trends, Alerts, History, Summary, Histogram, Notifications, Event Log), and System (Comments, Downtime, Process Info, Performance Info, Scheduling Queue, Configuration). The main content area has sections for "Get Started" (with bullet points like "Start monitoring your infrastructure", "Change the look and feel of Nagios", etc.), "Latest News" (empty), and "Don't Miss..." (empty). A "Quick Links" box on the right lists links to Nagios Library, Nagios Labs, Nagios Exchange, Nagios Support, Nagios.com, and Nagios.org. At the bottom, there's a copyright notice and a "Nagios" logo.

Click on hosts

This is a zoomed-in view of the "Tactical Overview" section from the dashboard. It shows a list of navigation links: "Current Status", "Tactical Overview" (which is highlighted in blue), "Map", "Hosts", "Services", and "Host Groups".

5) Click on linux server

Nagios®

Current Network Status
Last Updated: Sat Sep 28 11:33:24 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged in as nagiosadmin:

Host Status Totals
Up Down Unreachable Pending

2	0	0	0
All Problems	All Types		
0	2		

Service Status Totals
Ok Warning Unknown Critical Pending

12	1	0	3	0
All Problems	All Types			
4	16			

Host Status Details For All Host Groups

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	09-28-2024 11:29:10	0d 0h 8m 36s	PING OK - Packet loss = 0%, RTA = 1.18 ms
localhost	UP	09-28-2024 11:32:18	0d 3h 53m 7s	PING OK - Packet loss = 0%, RTA = 0.03 ms

Results 1 - 2 of 2 Matching Hosts

Reports
Availability Trends Alerts History Summary Histogram Notifications Event Log

Nagios®

General
Home Documentation

Current Status
Tactical Overview Map Hosts Services Host Groups Summary Grid Service Groups Summary Grid Problems Services (Unhandled) Hosts (Unhandled) Network Outages Quick Search:

Host Information
Last Updated: Sat Sep 28 11:33:39 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged in as nagiosadmin:

Host
linuxserver (linuxserver)
Member of No hostgroups
54.173.58.143

Host State Information

Host Status: UP (for 0d 0h 8m 51s)	Status Information: PING OK - Packet loss = 0%, RTA = 1.18 ms
Status Information: PING OK - Packet loss = 0%, RTA = 1.18 ms	Performance Data: rta=1.184000ms;3000.000000;5000.000000;0.000000 pl=0%;80,100,100
Performance Data: rta=1.184000ms;3000.000000;5000.000000;0.000000 pl=0%;80,100,100	Current Attempt: 1/10 (HARD state)
Current Attempt: 1/10 (HARD state)	Last Check Time: 09-28-2024 11:29:10
Last Check Time: 09-28-2024 11:29:10	Check Type: ACTIVE
Check Type: ACTIVE	Check Latency / Duration: 0.00 - 0.055 seconds
Check Latency / Duration: 0.00 - 0.055 seconds	Next Scheduled Active Check: 09-28-2024 11:34:10
Next Scheduled Active Check: 09-28-2024 11:34:10	Last State Change: 09-28-2024 11:24:48
Last State Change: 09-28-2024 11:24:48	Last Notification: N/A (notification 0)
Last Notification: N/A (notification 0)	Is This Host Flapping? NO (0.00% state change)
Is This Host Flapping? NO (0.00% state change)	In Scheduled Downtime? NO
In Scheduled Downtime? NO	Last Update: 09-28-2024 11:33:37 (0d 0h 0m 2s ago)

Host Commands

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

Host Comments
Add a new comment Delete all comments

Entry Time Author Comment Comment ID Persistent Type Expires Actions
This host has no comments associated with it

System
Comments Downtime Process Info Performance Info Scheduling Queue Configuration

6) Click on nagios services

Documentation

Current Status

Tactical Overview

Map

Hosts

Services

Host Groups

 Summary

 Grid

Service Groups

Nagios*

Current Network Status

Last Updated Sat Sep 28 11:33:58 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged in as nagiosadmin

General

Home Documentation

Current Status

Tactical Overview Map Hosts Services Host Groups Summary Grid Service Groups

Host Status Totals

Up	Down	Unreachable	Pending
2	0	0	0

All Problems All Types

0	2
---	---

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
12	1	0	3	0

All Problems All Types

4	16
---	----

Service Status Details For All Hosts

Host	Service	Status	Last Check	Duration	Attempt	Status Information
linuxserver	Current Load	OK	09-28-2024 11:30:25	0d 0h 8m 33s	1/4	OK - load average: 0.01. 0.00. 0.00
linuxserver	Current Users	OK	09-28-2024 11:31:03	0d 0h 7m 56s	1/4	USERS OK - 2 users currently logged in
HTTP		CRITICAL	09-28-2024 11:29:40	0d 0h 4m 18s	4/4	connect to address 54.173.58.143 and port 80: Connection refused
PING		OK	09-28-2024 11:32:18	0d 0h 6m 40s	1/4	PING OK - Packet loss = 0%, RTA = 1.03 ms
Root Partition		OK	09-28-2024 11:32:55	0d 0h 6m 3s	1/4	DISK OK - free space / 6105 MB (75.23% inode=98%)
SSH		OK	09-28-2024 11:33:33	0d 0h 5m 25s	1/4	SSH OK - OpenSSH_9.6p1 Ubuntu-Subuntu13.4 (protocol 2.0)
Swap Usage		CRITICAL	09-28-2024 11:32:10	0d 0h 1m 48s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
Total Processes		OK	09-28-2024 11:29:48	0d 0h 9m 10s+	1/4	PROCS OK: 37 processes with STATE = R/Z/D/T
localhost	Current Load	OK	09-28-2024 11:29:39	0d 3h 53m 5s	1/4	OK - load average: 0.02. 0.01. 0.00
localhost	Current Users	OK	09-28-2024 11:30:17	0d 3h 52m 27s	1/4	USERS OK - 2 users currently logged in
HTTP		WARNING	09-28-2024 11:29:46	0d 2h 49m 12s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.001 second response time
PING		OK	09-28-2024 11:31:32	0d 3h 5m 12s	1/4	PING OK - Packet loss = 0%, RTA = 0.03 ms
Root Partition		OK	09-28-2024 11:32:09	0d 3h 50m 35s	1/4	DISK OK - free space / 6105 MB (75.23% inode=98%)
SSH		OK	09-28-2024 11:32:47	0d 3h 49m 57s	1/4	SSH OK - OpenSSH_9.7 (protocol 2.0)
Swap Usage		CRITICAL	09-28-2024 11:31:24	0d 3h 12m 34s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
Total Processes		OK	09-28-2024 11:29:02	0d 3h 14m 56s	1/4	PROCS OK: 37 processes with STATE = R/Z/D/T

Results 1 - 16 of 16 Matching Services

Reports

Availability Trends Alerts History Activity Histogram Notifications Event Log

System

Comments

Conclusion:

In this lab, we successfully configured a monitoring setup between a Nagios host machine (referred to as "exp9 machine") and a client machine (created specifically for this experiment). The goal was to set up Nagios to monitor a remote Linux server, which involved configuring both the Nagios host and client machine (Ubuntu instance) in an EC2 environment.

ADVANCE DEVOPS EXP 11

Name:Laksh .V. Sodhai

Class: D15A

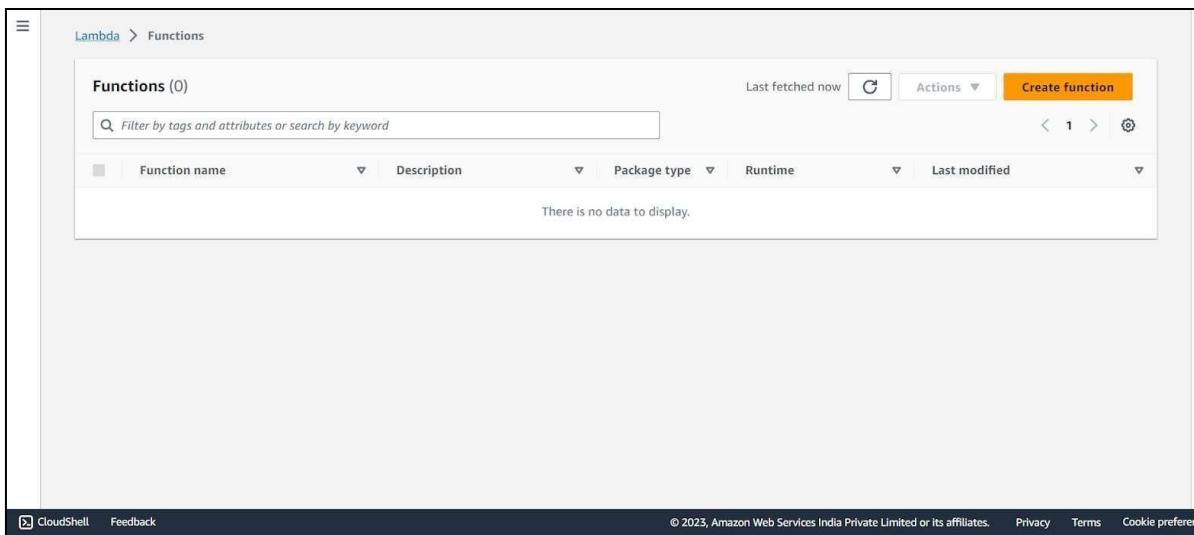
Roll No:59

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.



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
 arm64

CloudShell

© 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` ▾

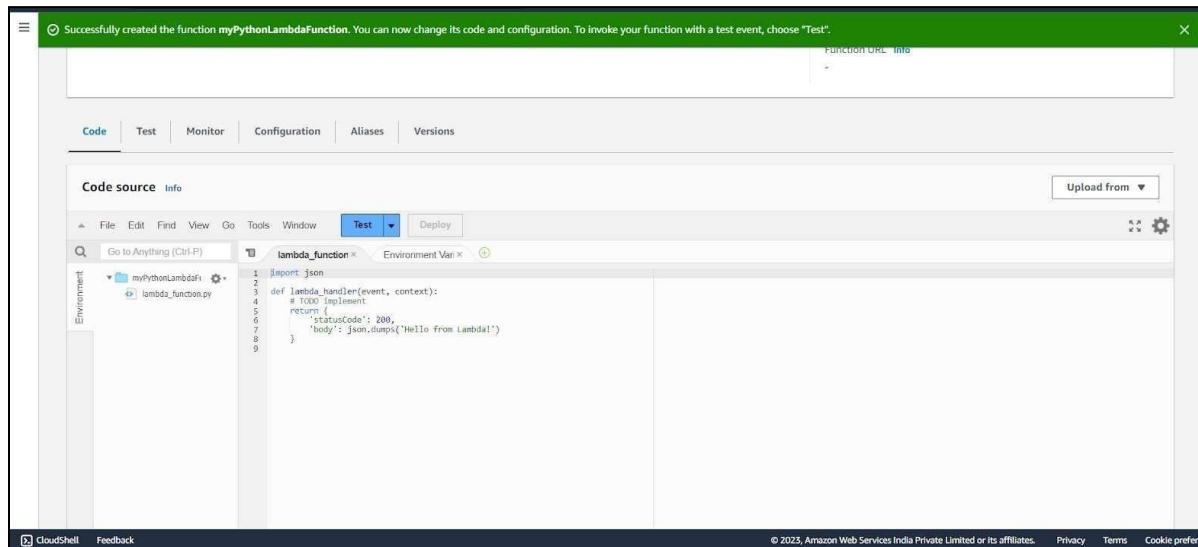
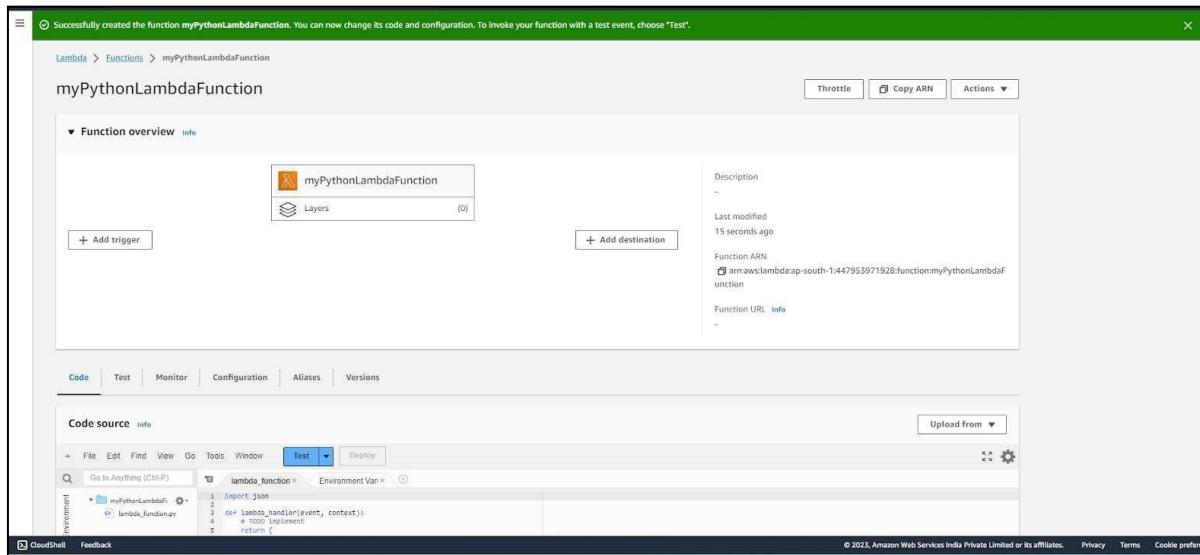
Architecture Info
Choose the instruction set architecture you want for your function code.
 x86_64
 arm64

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.

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.



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.

The screenshot shows the AWS Lambda function configuration interface. At the top, a green banner indicates that the function 'myPythonLambdaFunction' has been successfully created. The left sidebar lists various configuration tabs: General configuration, Triggers, Permissions, Destinations, Function URL, Environment variables, Tags, VPC, Monitoring and operations tools, Concurrency, Asynchronous invocation, Code signing, Database proxies, File systems, and State machines. The 'General configuration' tab is selected and expanded, showing fields for Description (empty), Memory (128 MB), Timeout (0 min 3 sec), SnapStart (None), and Ephemeral storage (512 MB). An 'Edit' button is located in the top right corner of this panel.

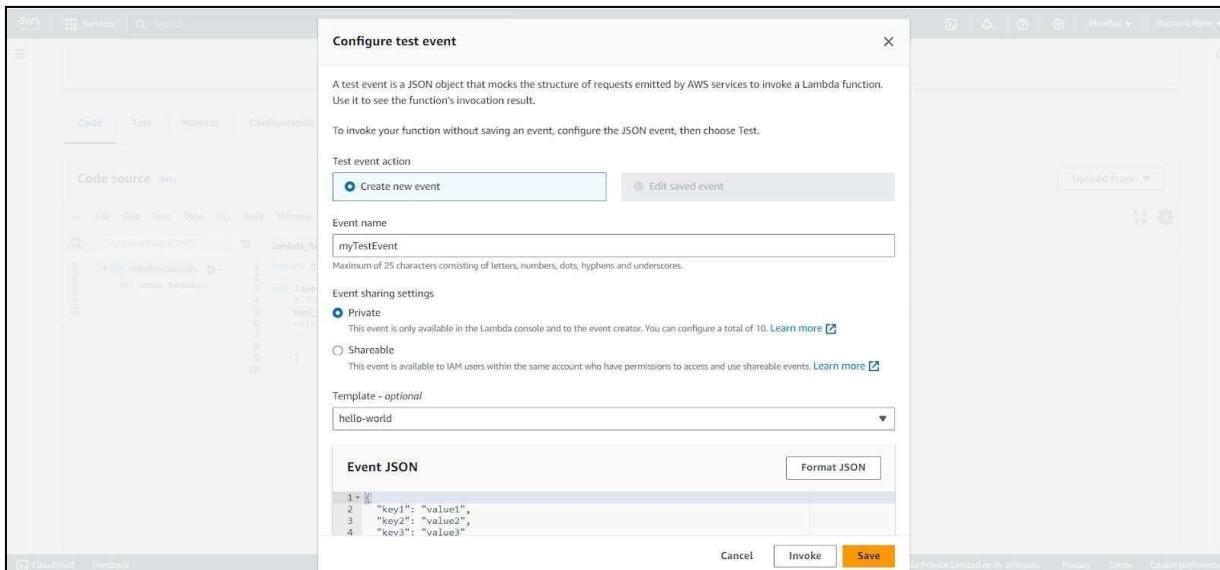
The screenshot shows the 'Edit basic settings' page for the 'myPythonLambdaFunction'. The top navigation bar includes the AWS logo, Services, a search bar, and a keyboard shortcut [Alt+S]. The breadcrumb navigation shows the path: Lambda > Functions > myPythonLambdaFunction > Edit basic settings. The main content area is titled 'Edit basic settings' and contains the 'Basic settings' section. It includes fields for Description (optional), Memory (128 MB), Ephemeral storage (512 MB), SnapStart (None), and Timeout (0 min 1 sec). The 'Execution role' field is also present. The bottom of the page features standard AWS navigation links: CloudShell and Feedback.

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.

The screenshot shows the AWS Lambda function editor interface. The top navigation bar includes tabs for Code, Test, Monitor, Configuration, Aliases, and Versions. The Code tab is selected. Below the tabs is a toolbar with File, Edit, Find, View, Go, Tools, Window, Test, Deploy, and a status message 'Changes not deployed'. A dropdown menu 'Upload from' is visible. The main area is titled 'Code source' with an 'Info' button. It displays a file tree under 'Environment' with 'myPythonLambdaFunction' expanded, showing 'lambda_function.py'. The code editor contains the following Python code:

```
1 import json
2
3 def lambda_handler(event, context):
4     # TODO implement
5     new_string="Hello! how are you?"
6     return {
7         'statusCode': 200,
8         'body': json.dumps('Hello from Lambda!')
9     }
10
```

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 test event myTestEvent was successfully saved.

File Edit Find View Go Tools Window Test Deploy Changes not deployed

Go to Anything (Ctrl-P) lambda_function Environment Var Execution result

Execution results 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  
RequestID 7d26f404-f1da-4435-9faf-8dbb2a2733cc
```

Code properties Info

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

The test event myTestEvent was successfully saved.

File Edit Find View Go Tools Window Test Deploy Changes not deployed

Go to Anything (Ctrl-P) lambda_function Environment Var Execution result

Execution results 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
```

Code source Info Upload from

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

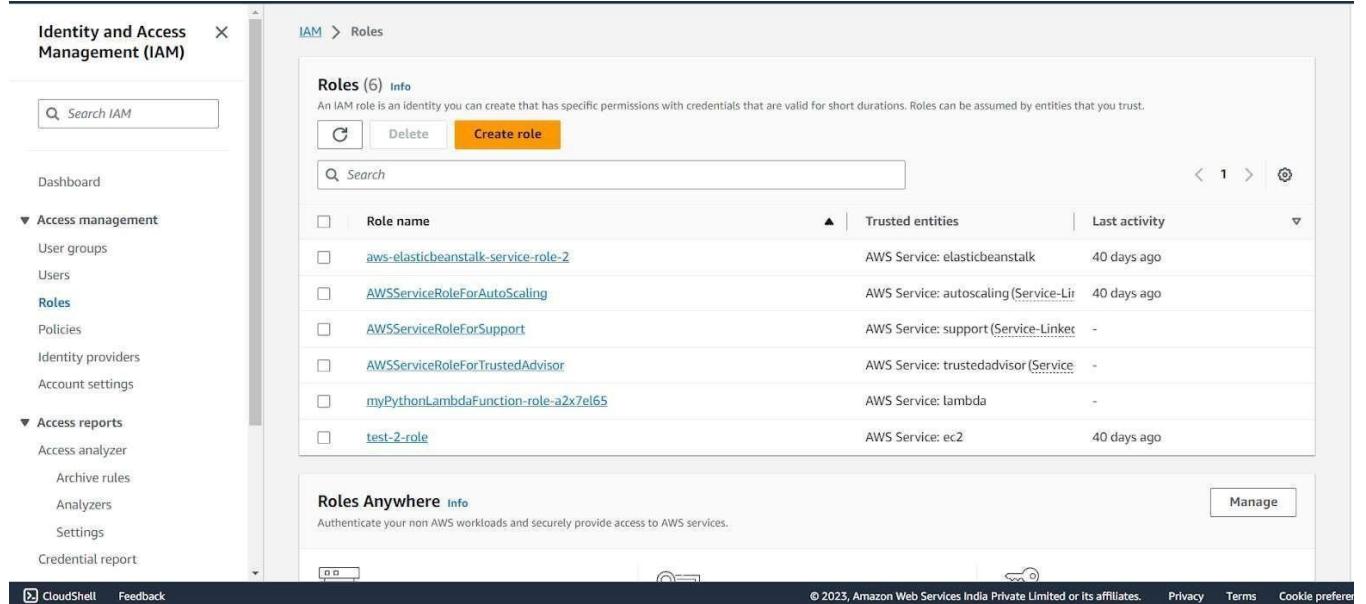
Adv. DevOps Exp. 12

Name-Laksh .V. Sodhai

Class-D15A

Roll No-59

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).

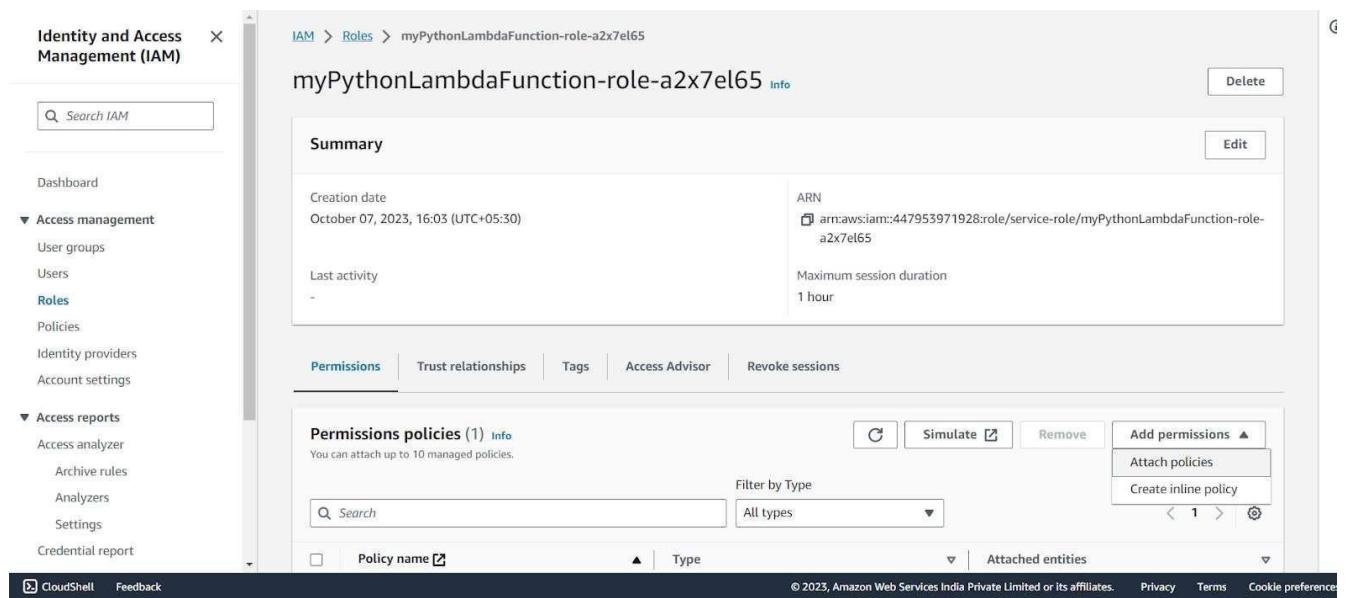


The screenshot shows the AWS IAM Roles page. On the left, there's a navigation sidebar with options like Dashboard, Access management (User groups, Roles, Policies, Identity providers, Account settings), Access reports (Archive rules, Analyzers, Settings, Credential report), CloudShell, and Feedback. The main area has a header 'Roles (6) Info' with a 'Create role' button. Below is a table listing six roles:

Role name	Trusted entities	Last activity
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	-
test-2-role	AWS Service: ec2	40 days ago

At the bottom, there's a 'Roles Anywhere' section with a 'Manage' button.

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



The screenshot shows the 'myPythonLambdaFunction-role-a2x7el65' role details page. The left sidebar is identical to the previous screenshot. The main area has a 'Summary' section with creation date (October 07, 2023, 16:03 (UTC+05:30)), ARN (arn:aws:iam::447953971928:role/service-role/myPythonLambdaFunction-role-a2x7el65), last activity (-), and maximum session duration (1 hour). Below is the 'Permissions' tab, which shows 'Permissions policies (1) Info'. It lists one policy: 'S3-ReadOnly'. There are buttons for 'Add permissions' (with options for 'Attach policies' and 'Create inline policy'), 'Simulate', and 'Remove'. A search bar and filter dropdown are also present.

S3-ReadOnly

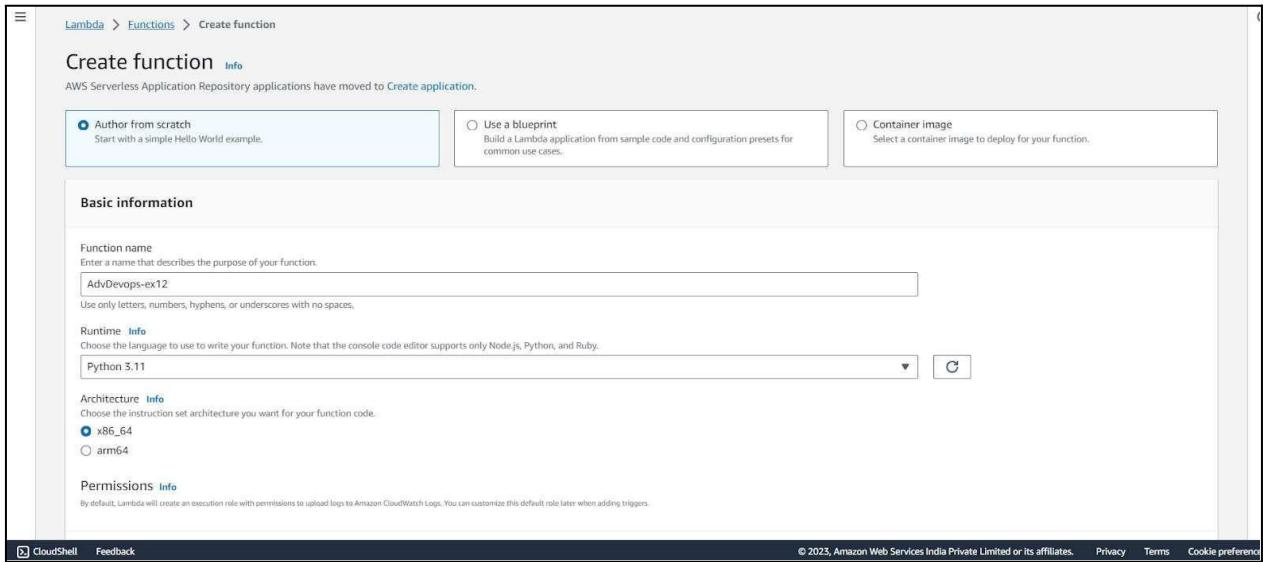
The screenshot shows the 'Add permissions' dialog in the AWS IAM console. The path is IAM > Roles > myPythonLambdaFunction-role-a2x7el65 > Add permissions. The title is 'Attach policy to myPythonLambdaFunction-role-a2x7el65'. The 'Current permissions policies' section shows one policy: 'AmazonS3ReadOnlyAccess'. The 'Other permissions policies' section has a search bar 'S3read' and a filter 'All types'. It lists two policies: 'AmazonS3ReadOnlyAccess' (AWS managed) and 'CloudWatchFullAccess' (AWS managed). The 'CloudWatchFullAccess' policy is selected. At the bottom are 'Cancel' and 'Add permissions' buttons.

CloudWatchFull

This screenshot shows the same 'Add permissions' dialog after the 'CloudWatchFullAccess' policy has been attached. The 'Current permissions policies' section now shows two policies: 'AmazonS3ReadOnlyAccess' and 'CloudWatchFullAccess'. The 'Other permissions policies' section shows a search bar 'cloudwatchfull' and a filter 'All types'. It lists two policies: 'CloudWatchFullAccess' and 'CloudWatchFullAccessV2', both of which are selected. At the bottom are 'Cancel' and 'Add permissions' buttons.

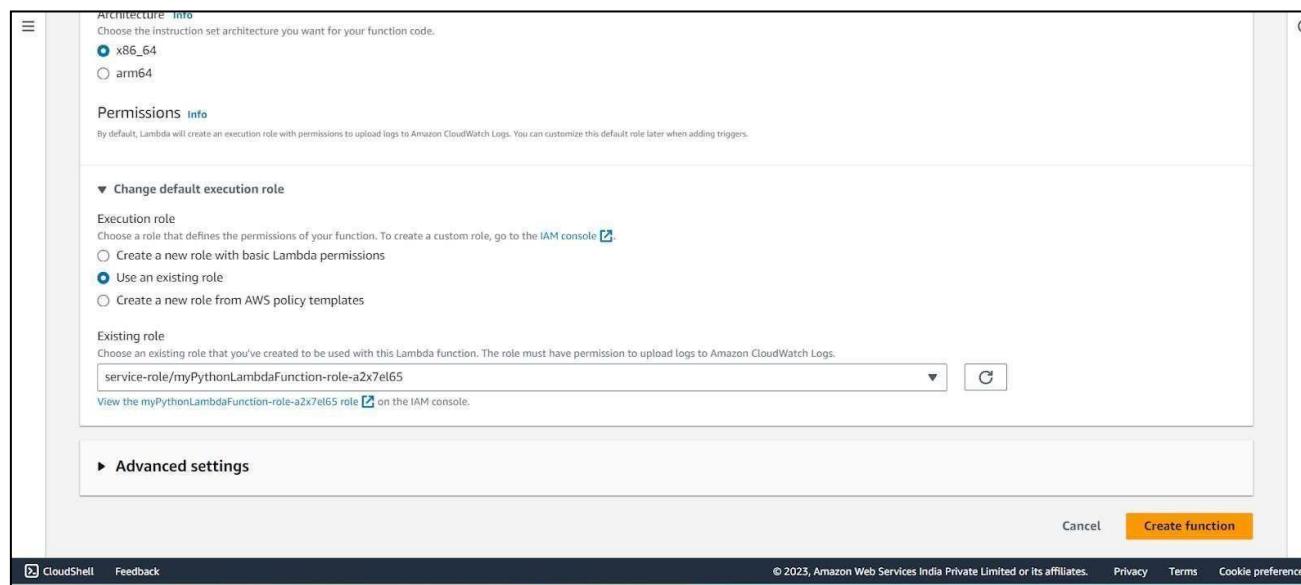
After successful attachment of policy you will see something like this you will be able to see the updated policies.

This screenshot shows the 'Permissions' tab in the AWS IAM console for the 'myPythonLambdaFunction-role-a2x7el65' role. A green success message says 'Policy was successfully attached to role.' The 'Permissions' tab is active. The 'Permissions policies' section shows three policies: 'AmazonS3ReadOnlyAccess' (AWS managed), 'AWSLambdaBasicExecutionRole-c4946a...', and 'CloudWatchFullAccess' (AWS managed). The 'CloudWatchFullAccess' policy is selected. Other tabs include 'Trust relationships', 'Tags', 'Access Advisor', and 'Revoke sessions'. At the bottom are 'Simulate', 'Remove', and 'Add permissions' buttons.

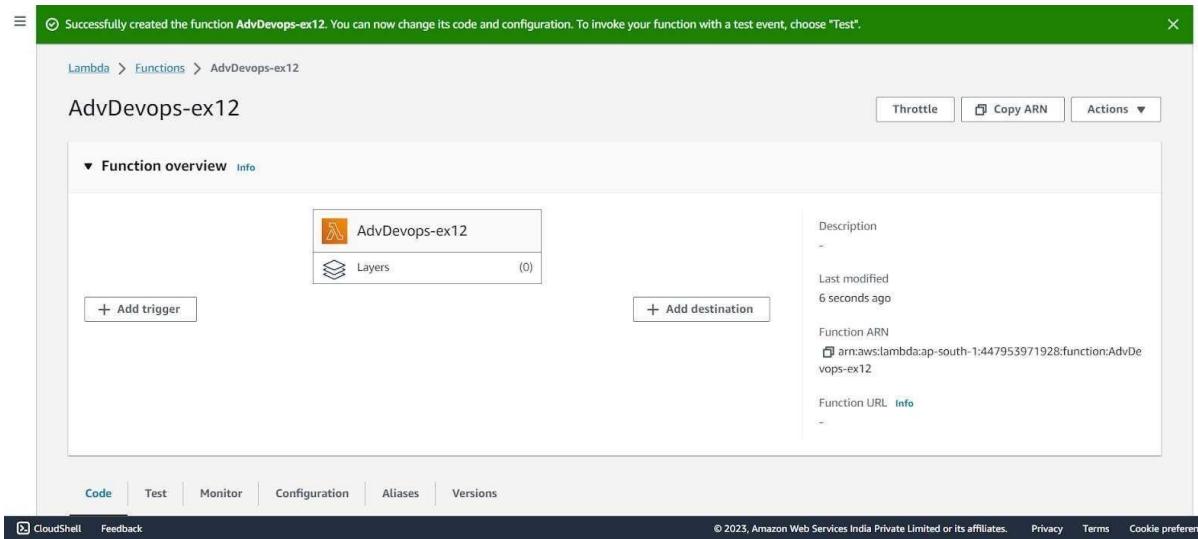


Step 3: Open up AWS Lambda and create a new Python function.

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



Step 4: The function is up and running.



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.

```
lambda_function x Environment Var x
Environment: Go to Anything (Ctrl-P)
lambda_function.py
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_unquote_plus(key, encoding='utf-8')
11    message = f'An 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
19
```

The screenshot shows the AWS Lambda Code Editor. The left sidebar shows the project structure with "lambda_function" selected. The main editor area contains the provided Python code. The status bar at the bottom right shows "18:5 Python Spaces: 4".

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

The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with 'aws' logo, 'Services' dropdown, search bar, and a key combination '[Alt+S]'. A green banner message says 'Successfully created the function AdvDevops-ex12. You can now change its code and configuration. To invoke your function, click Test.' Below the banner, there are tabs: 'Code' (selected), 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. Under the 'Code' tab, there's a sub-section titled 'Code source' with an 'Info' link. The main area is a code editor with a toolbar above it: File, Edit, Find, View, Go, Tools, Window, 'Test' (highlighted in blue), 'Deploy', and a status message 'Changes not deployed'. The code editor shows a file named 'lambda_function.py' with the following content:

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

The screenshot shows the 'Configure test event' dialog box. It has a header 'Configure test event' with a close button 'X'. The main text area says: 'A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.' Below this, a note says: 'To invoke your function without saving an event, configure the JSON event, then choose Test.' There are two radio buttons for 'Test event action': 'Create new event' (selected) and 'Edit saved event'. The 'Event name' field contains 'test'. A note below it says: 'Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.' Under 'Event sharing settings', there are two options: 'Private' (selected) and 'Shareable'. The 'Private' option note says: 'This event is only available in the Lambda console and to the event creator. You can configure a total of 10.' The 'Shareable' option note says: 'This event is available to IAM users within the same account who have permissions to access and use shareable events.' Below these are sections for 'Template - optional' (containing 's3-put') and 'Event JSON' (with a 'Format JSON' button). At the bottom right of the dialog are 'Cancel', 'Invoke' (disabled), and 'Save' buttons.

And Save it.

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

The screenshot shows the 'Buckets (3) info' section of the Amazon S3 console. It lists three buckets:

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)

At the top right, there is a 'Create bucket' button.

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

The screenshot shows the 'Create bucket' wizard. In the 'General configuration' step, the bucket name is 'AdvDevopsexp12' and the AWS Region is 'Asia Pacific (Mumbai) ap-south-1'. There is also a 'Choose bucket' button for copying settings from an existing bucket.

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

The screenshot shows the 'Event notifications' section of the bucket properties. It displays a table with columns: Name, Event types, Filters, Destination type, and Destination. A message says 'No event notifications' and 'Choose Create event notification to be notified when a specific event occurs.' There is a 'Create event notification' button.

Below this, the 'Amazon EventBridge' section is shown, with a note about using it for event-driven applications. The 'Transfer acceleration' section is also visible at the bottom.

Click on Create Event Notification.

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

General configuration

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.

Destination

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

CloudShell Feedback © 2023, Amazon Web Services India Private Limited

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 Function Overview page for a function named 'AdvDevops-ex12'. In the 'Triggers' section, there is a single entry for 'S3'. Below the triggers, there are tabs for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. On the right side, there are sections for 'Description', 'Last modified', 'Function ARN', and 'Function URL'.

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

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

The screenshot shows a terminal window with a single line of code: `{} dummy.json > ...`. Below this, a JSON object is being typed into the terminal:

```
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 a help link '[Alt+S]'. Below the navigation is a breadcrumb trail: 'Amazon S3 > Buckets > advopssexp12 > Upload'. The main area is titled 'Upload' with a 'Info' link. A note 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 [?]'. Below this is a dashed box for dragging files and a button to 'Add files' or 'Add folder'. A table titled 'Files and folders (1 Total, 89.0 B)' lists 'dummy.json' with type 'application/json' and size '89.0 B'. There are 'Remove', 'Add files', and 'Add folder' buttons above the table. A search bar 'Find by name' is at the top of the table. The 'Destination' section shows 'Destination' set to 's3://advopssexp12'. At the bottom, there are 'CloudShell' and 'Feedback' links, and a copyright notice '© 2023, Amazon Web Services India Private Limited or'.

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.

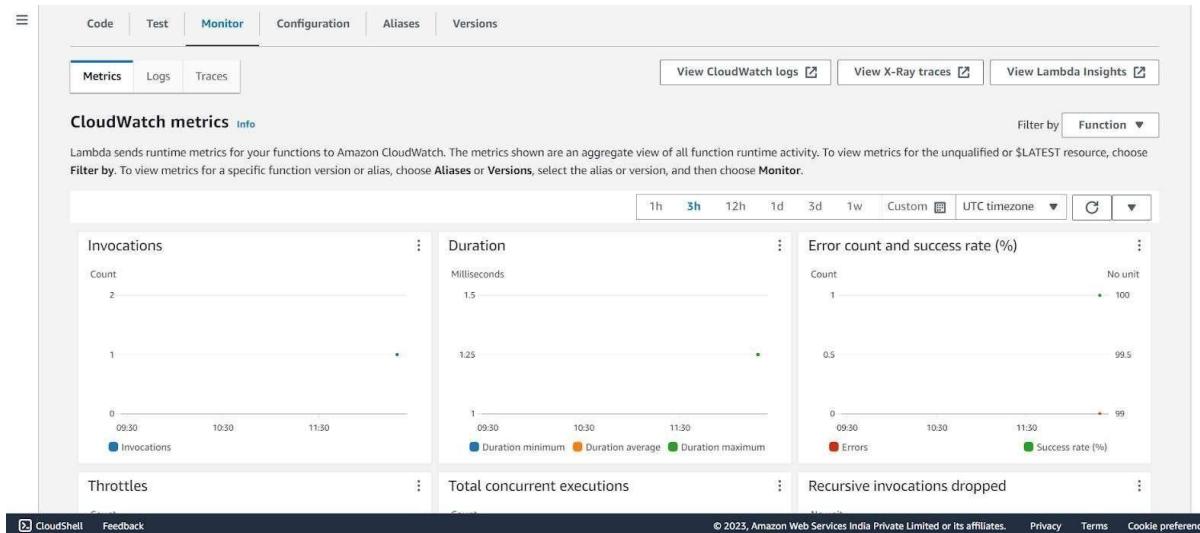
The screenshot shows the AWS Lambda 'Event JSON' editor. It displays a large JSON object with line numbers from 10 to 38. The JSON structure includes fields like 'principalId', 'requestParameters', 'responseElements', 's3', 'object', and 'sequencer'. The 'Format JSON' button is located in the top right corner. The JSON content is as follows:

```

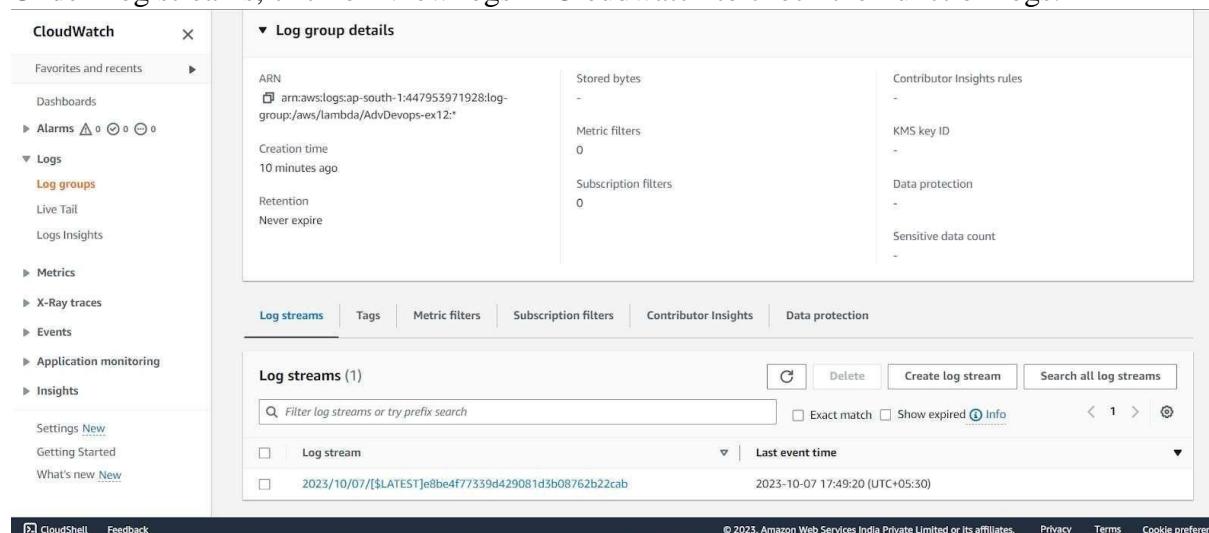
10     "principalId": "EXAMPLE"
11   },
12   "requestParameters": {
13     "sourceIPAddress": "127.0.0.1"
14   },
15   "responseElements": {
16     "x-amz-request-id": "EXAMPLE123456789",
17     "x-amz-id-2": "EXAMPLE123/5678abcdefghijklmnaqrstuvwxyzABCDEFGHIJKLMN"
18   },
19   "s3": {
20     "s3SchemaVersion": "1.0",
21     "configurationId": "testConfigRule",
22     "bucket": {
23       "name": "advopssexp12",
24       "ownerIdentity": {
25         "principalId": "EXAMPLE"
26       },
27       "arn": "arn:aws:s3:::advopssexp12"
28     },
29     "object": {
30       "key": "test%2Fkey",
31       "size": 1024,
32       "eTag": "0123456789abcdef0123456789abcdef",
33       "sequencer": "0A1B2C3D4E5F678901"
34     }
35   }
36 }
37 ]
38 }

```

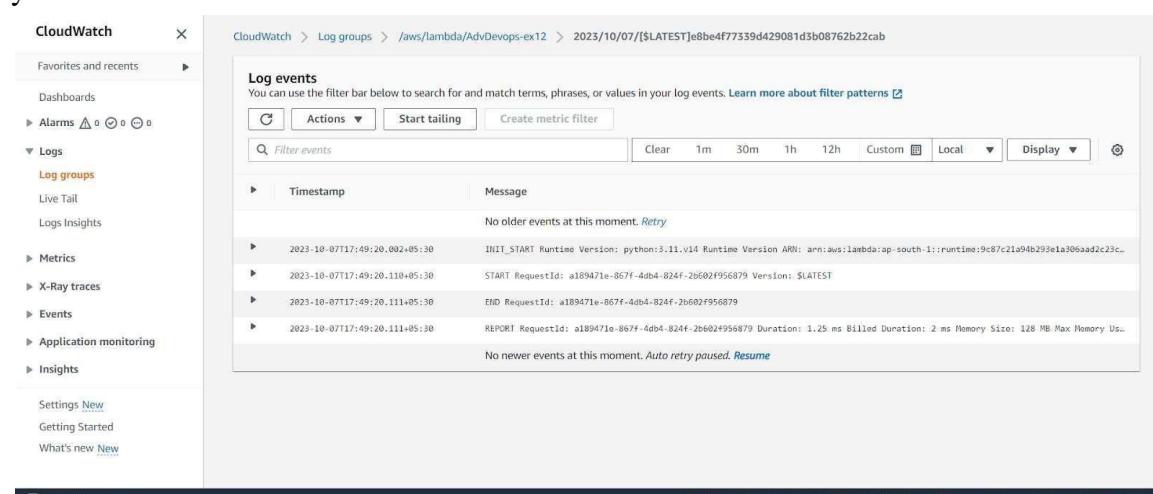
Step 16: Go back to your Lambda function , Refresh it and check the Monitor tab.



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.