

Adv Devops exp-1.a

Prajwal Pandey

D15A - 33

1. Creating key-pair

Create key pair X

Key pair name
Key pairs allow you to connect to your instance securely.

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

RSA
RSA encrypted private and public key pair

ED25519
ED25519 encrypted private and public key pair

Private key file format

.pem
For use with OpenSSH

.ppk
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)

[Cancel](#) **Create key pair**

2. Launching Instance

The screenshot shows the AWS EC2 Instances "Launch an instance" page. At the top, there's a green success message: "Successfully initiated launch of instance (i-07b730b5298e1b71f)". Below it, a "Next Steps" section lists several options:

- Create billing and free tier usage alerts
- Connect to your instance
- Connect an RDS database
- Create EBS snapshot policy

Each option has a brief description and a "Learn more" link or button.

3. Connecting using EC2 instance

This screenshot shows the "Connect instance" configuration page for the launched EC2 instance (i-07b730b5298e1b71f). It includes fields for:

- Instance ID: i-07b730b5298e1b71f (Portfolio-Website)
- Connection Type:
 - Connect using EC2 Instance Connect: "Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address."
 - Connect using EC2 Instance Connect Endpoint: "Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint."
- Public IP address: 13.60.234.82
- Username: ec2-user (search bar)

A note at the bottom states: "Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username."

At the bottom right are "Cancel" and "Connect" buttons.

4. Launching wizard

The screenshot shows the AWS Security Groups page. It displays one security group named "launch-wizard-3". The table columns are:

Name	Security group ID	Security group name	VPC ID
-	sg-04b57c53725489c2d	launch-wizard-3	vpc-0207b9f94c8d7b0

The screenshot shows the AWS CloudFormation console interface. At the top, there is a green success message: "Inbound security group rules successfully modified on security group (sg-04b57c53725489c2d | launch-wizard-3)". Below the message, there is a "Details" link. The main area is titled "Security Groups (5)" with an "Info" link. It includes a search bar with placeholder text "Find resources by attribute or tag", a "Actions" dropdown, and a "Create security group" button. A CSV export option is also present. The table lists five security groups:

<input type="checkbox"/>	Name	Security group ID	Security group name	VPC ID
<input type="checkbox"/>	Pepper-env	sg-08f6882ccab7b2e43	awseb-e-j9fgrnthpe-stack-AWSEBSecu...	vpc-0207b9f94c8d7
<input type="checkbox"/>	-	sg-04b57c53725489c2d	launch-wizard-3	vpc-0207b9f94c8d7
<input type="checkbox"/>	-	sg-032ddf0486073b10b	launch-wizard-1	vpc-0207b9f94c8d7

5. Adding Inbound rules

Inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info
sgr-09afaa6f2408666b4	SSH ▼	TCP	22	Cus... ▼	<input type="text" value="0.0.0.0"/> X
-	HTTP ▼	TCP	80	An... ▼	<input type="text" value="Web Port"/> X
-	HTTPS ▼	TCP	443	An... ▼	<input type="text" value="Web Port"/> X

[Add rule](#)

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP [X](#)

6. Starting

```

drwxr-xr-x. 2 root root 16384 Aug 12 10:16 IP_exp_1
[root@ip-172-31-40-210 HTML-ECOMERCE-main]# mv * /var/www/html/
[root@ip-172-31-40-210 HTML-ECOMERCE-main]# cd /var/www/html
[root@ip-172-31-40-210 html]# ls -lrt
total 16
drwxr-xr-x. 2 root root 16384 Aug 12 10:16 IP_exp_1
[root@ip-172-31-40-210 html]# cd IP_exp_1
[root@ip-172-31-40-210 IP_exp_1]# mv * /var/www/html/
[root@ip-172-31-40-210 IP_exp_1]# cd /var/www/html
[root@ip-172-31-40-210 html]# ls -lrt
total 15036
-rw-r--r--. 1 root root 128534 Aug 12 10:16 vision.webp
-rw-r--r--. 1 root root 4244 Aug 12 10:16 styles.css
-rw-r--r--. 1 root root 162478 Aug 12 10:16 mac.png
-rw-r--r--. 1 root root 1961 Aug 12 10:16 logo.png
-rw-r--r--. 1 root root 392986 Aug 12 10:16 ipad.jpeg
-rw-r--r--. 1 root root 11613 Aug 12 10:16 index.html
-rw-r--r--. 1 root root 44107 Aug 12 10:16 iPhone-14.jpg
-rw-r--r--. 1 root root 217676 Aug 12 10:16 'home page.png'
-rw-r--r--. 1 root root 11511 Aug 12 10:16 airpods.jpg
-rw-r--r--. 1 root root 14401920 Aug 12 10:16 180917_01_09_720p_5000br.mp4
drwxr-xr-x. 2 root root 6 Aug 23 11:34 IP_exp_1
[root@ip-172-31-40-210 html]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled)
     Active: inactive (dead)
       Docs: man:httpd.service(8)

[root@ip-172-31-40-210 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-40-210 html]# systemctl start httpd
[root@ip-172-31-40-210 html]#

```

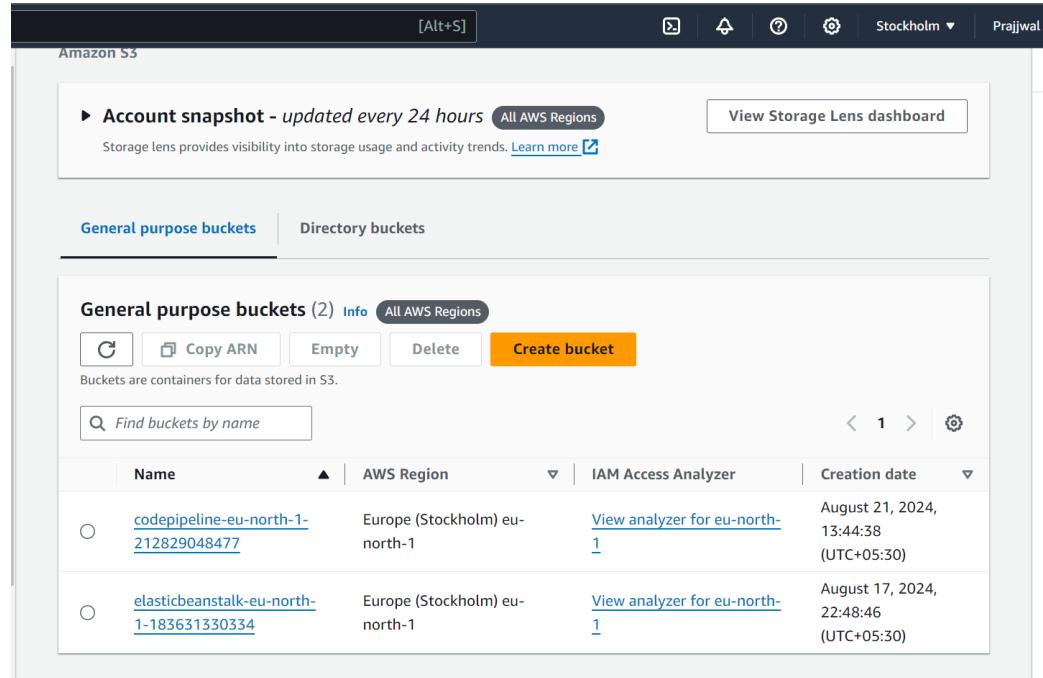
[Store](#)[Mac](#)[iPad](#)[iPhone](#)[Airpods](#)

APPLE VISION PRO



B) Hosting a static website on Amazon S3.

1. Creating a bucket.



The screenshot shows the AWS S3 console under the 'General purpose buckets' tab. It displays two buckets: 'codepipeline-eu-north-1' and 'elasticbeanstalk-eu-north-1'. Both buckets were created in the 'Europe (Stockholm) eu-north-1' region on August 21, 2024, at 13:44:38 UTC+05:30. The 'Create bucket' button is visible at the top right of the list.

Name	AWS Region	IAM Access Analyzer	Creation date
codepipeline-eu-north-1 Copy ARN	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 21, 2024, 13:44:38 (UTC+05:30)
elasticbeanstalk-eu-north-1- 183631330334	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 17, 2024, 22:48:46 (UTC+05:30)

General configuration

AWS Region
Europe (Stockholm) eu-north-1

Bucket type [Info](#)

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

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

Bucket name [Info](#)

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

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

[Choose bucket](#)

Format: s3://bucket/prefix

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

View details X

Amazon S3 > Buckets

► Account snapshot - updated every 24 hours All AWS Regions

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

View Storage Lens dashboard

General purpose buckets Directory buckets

General purpose buckets (3) Info All AWS Regions

Buckets are containers for data stored in S3.

Find buckets by name

Name AWS Region IAM Access Analyzer Creation date

Name	AWS Region	IAM Access Analyzer	Creation date
applevision	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 25, 2024, 22:09:42 (UTC+05:30)
codepipeline-eu-north-1-212829048477	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 21, 2024, 13:44:38 (UTC+05:30)

Create bucket

2. Enable static website hosting

Edit static website hosting [Info](#)

Static website hosting

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

Static website hosting

- Disable
 Enable

Hosting type

- Host a static website

Use the bucket endpoint as the web address. [Learn more](#)

- Redirect requests for an object

Redirect requests to another bucket or domain. [Learn more](#)

ⓘ For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see [Using Amazon S3 Block Public Access](#)

3. Upload your files

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 (9 Total, 14.7 MB)

[Remove](#)

[Add files](#)

[Add folder](#)

All files and folders in this table will be uploaded.

Find by name

< 1 >

<input type="checkbox"/>	Name	Folder	Type
<input type="checkbox"/>	180917_01_09_720p_5000br.mp4	IP_exp_1/	video/mp4
<input type="checkbox"/>	airpods.jpg	IP_exp_1/	image/jpeg
<input type="checkbox"/>	home page.png	IP_exp_1/	image/png
<input type="checkbox"/>	index.html	IP_exp_1/	text/html
<input type="checkbox"/>	ipad.jpeg	IP_exp_1/	image/jpeg
<input type="checkbox"/>	iPhone-14.jpg	IP_exp_1/	image/jpeg

 **Upload succeeded**

[View details below.](#)

Files and folders (9 Total, 14.7 MB)

Find by name

Name	Folder	Type	Size	Status	Error
180917_01...	IP_exp_1/	video/mp4	13.7 MB	 Succeeded	-
airpods.jpg	IP_exp_1/	image/jpeg	11.2 KB	 Succeeded	-
home page.p...	IP_exp_1/	image/png	212.6 KB	 Succeeded	-
index.html	IP_exp_1/	text/html	9.9 KB	 Succeeded	-
ipad.jpeg	IP_exp_1/	image/jpeg	383.8 KB	 Succeeded	-
iPhone-14.jp...	IP_exp_1/	image/jpeg	43.1 KB	 Succeeded	-
logo.png	IP_exp_1/	image/png	1.9 KB	 Succeeded	-
mac.png	IP_exp_1/	image/png	158.7 KB	 Succeeded	-
vision.webp	IP_exp_1/	image/webp	125.5 KB	 Succeeded	-

4. Add bucket policy

Edit bucket policy [Info](#)

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply

Bucket ARN

arn:aws:s3:::applevision

Policy

```
1▼ {
2    "Version": "2012-10-17",
3▼   "Statement": []
4▼     {
5       "Sid": "PublicReadGetObject",
6       "Effect": "Allow",
7▼         "Principal": {
8           "AWS": "*"
9         },
10        "Action": "s3:GetObject",
11        "Resource": "arn:aws:s3:::applevision/*"
12      }
13 }
```

5. Go to URL in permissions

[!\[\]\(96407c3f85010637bf157ab4762568af_img.jpg\) Store](#) [Mac](#) [iPad](#) [iPhone](#) [Airpods](#)

APPLE VISION PRO



Book a demo >

AdvDevops

Exp-2 Elastic Beanstalk

Prajwal Pandey D15A - 33

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

The screenshot shows the 'Configure environment' step of the AWS Elastic Beanstalk setup wizard. On the left, a sidebar lists steps from 1 to 6. Step 1 is 'Configure environment', which is currently active. Step 2 is 'Configure service access'. Step 3 is 'optional: Set up networking, database, and tags'. Step 4 is 'optional: Configure instance traffic and scaling'. Step 5 is 'optional: Configure updates, monitoring, and logging'. Step 6 is 'Review'. The main content area is titled 'Configure environment' and contains two sections: 'Environment tier' and 'Application information'. In the 'Environment tier' section, 'Web server environment' is selected. In the 'Application information' section, the 'Application name' field is set to 'pepper'. Below it, there's a section for 'Application tags (optional)'. At the bottom, there's a 'Environment information' section.

2. Choose PHP from the drop-down menu and then click Create Application.

The screenshot shows the 'Platform' configuration step of the AWS Elastic Beanstalk setup wizard. The sidebar shows the user is on step 2, 'Configure service access'. The main content area is titled 'Platform' and contains three sections: 'Platform type', 'Platform', and 'Application code'. In the 'Platform type' section, 'Managed platform' is selected. In the 'Platform' section, 'PHP' is chosen from the dropdown. In the 'Application code' section, the dropdown shows 'PHP 8.3 running on 64bit Amazon Linux 2023'.

3. Give Key pair

The screenshot shows the AWS Elastic Beanstalk configuration interface. On the left, a sidebar lists steps: 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), and Step 6 (Review). The main panel is titled "Service access" and describes IAM roles. It includes fields for "Service role" (radio buttons for "Create and use new service role" or "Use an existing service role"), "Service role name" (text input "aws-elasticbeanstalk-service-role"), "View permission details" button, "EC2 key pair" (dropdown "AWSLinux" with a "View permission details" button), and "EC2 instance profile" (dropdown "Prajjwal-admin" with a "View permission details" button).

4. Instance settings

The screenshot shows the AWS Elastic Beanstalk configuration interface. The sidebar lists steps: 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), and Step 6 (Review). The main panel is titled "Instance settings" and describes launching in a custom VPC. It includes a dropdown for "VPC" set to "vpc-0207b9f94c8d7b0b2 | (172.31.0.0/16)" and a "Create custom VPC" link. Below this is a section for "Public IP address" with a checked "Activated" checkbox. The final section is "Instance subnets", which contains a table:

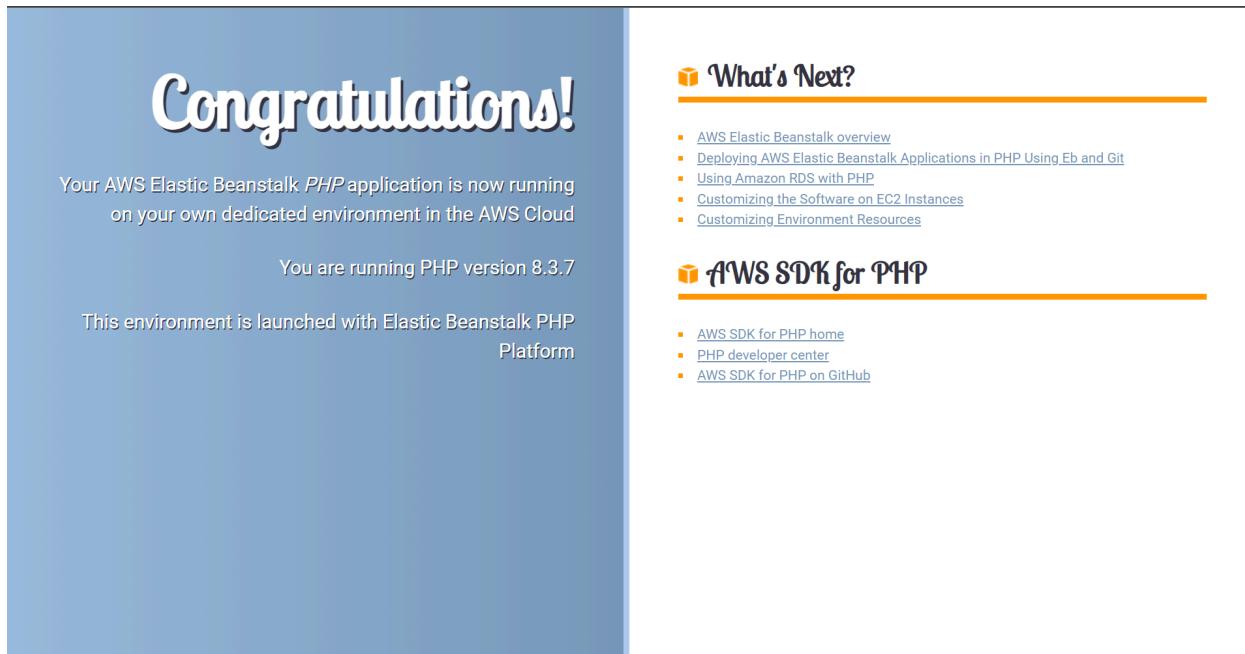
Availability Zone	Subnet	CIDR	Name
<input checked="" type="checkbox"/> eu-north-1a	subnet-032aeec06...	172.31.16.0/20	
<input type="checkbox"/> eu-north-1c	subnet-0492ff0d6...	172.31.0.0/20	

5. Select security groups

The screenshot shows the AWS CloudWatch Metrics console. At the top, there's a navigation bar with the AWS logo, 'Services' button, search bar, and user information ('Stockholm' and 'Prajwal'). Below the navigation is a section titled 'Instance metadata service (IMDS)'. It contains a note about IMDSv1 and IMDSv2, with a checkbox labeled 'Deactivated' checked. Underneath is a section for 'EC2 security groups' with the sub-section 'Select security groups to control traffic'. A table titled 'EC2 security groups (3)' lists three groups: 'default' (Group ID: sg-04591dc7af87ad805), 'launch-wizard-1' (selected, Group ID: sg-032ddf0486073b10b), and 'launch-wizard-2' (Group ID: sg-010ef87791b520839). The bottom of the page includes standard AWS footer links: CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

6. Configure updates, monitoring and logging

The screenshot shows the AWS Elastic Beanstalk configuration wizard. On the left, a sidebar lists steps: 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), and Step 6 (Review). The current step is Step 5. The main content area is titled 'Configure updates, monitoring, and logging - optional'. It has a section for 'Monitoring' with a sub-section 'Health reporting'. It explains enhanced health reporting and provides options for 'System' (Basic or Enhanced, Enhanced is selected). There are dropdown menus for 'CloudWatch Custom Metrics - Instance' and 'CloudWatch Custom Metrics - Environment', both set to 'Choose metrics'. At the bottom, there's a section for 'Health event streaming to CloudWatch Logs' with a note about configuring Elastic Beanstalk to stream environment health events to CloudWatch Logs. The bottom of the page includes standard AWS footer links: CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.



The screenshot shows the AWS Elastic Beanstalk console interface. On the left, a sidebar lists "Applications", "Environments", and "Change history". Under "Application: pepper", there are links for "Application versions" and "Saved configurations". Under "Environment: Pepper-env", there are links for "Go to environment", "Configuration", "Events", "Health", "Logs", "Monitoring", and "Alarms". A "Managed updates" section is also present.

The main content area displays a green banner stating "Environment successfully launched." Below it, the "Environment overview" section shows the "Health" status as "Ok" and the "Domain" as "Pepper-env.eba-mbimu2dh.eu-north-1.elasticbeanstalk.com". It also shows the "Environment ID" as "e-j9fgrnthe" and the "Application name" as "pepper".

The "Platform" section shows the "Platform" as "PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2" and the "Running version" as "-". The "Platform state" is listed as "Supported".

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

7. Choose pipeline settings

The screenshot shows the 'Choose pipeline settings' step in the AWS CodePipeline console. The pipeline name is set to 'pipeline1'. A note indicates that V1 pipelines are no longer supported and recommends using V2. The execution mode is set to 'Superseded'. The pipeline type is 'Standard (Pipeline type V2 required)'. The sidebar shows steps 1 through 5: 'Choose pipeline settings', 'Add source stage', 'Add build stage', 'Add deploy stage', and 'Review'.

8. Connect to GitHub

The screenshot shows the 'Connect to GitHub' step in the AWS CodePipeline console. The provider selected is 'GitHub (Version 1)'. The repository is 'Prajwal-pep/HTML-ECOMERCE' and the branch is 'main'. A note states that the GitHub (Version 1) action is not recommended due to security concerns and suggests using GitHub (Version 2). The change detection options are set to 'GitHub webhooks (recommended)'.

9. Choose input artifacts and application name

The screenshot shows the AWS Elastic Beanstalk Step 5 configuration screen. It includes fields for Deploy provider (set to AWS Elastic Beanstalk), Region (set to Europe (Stockholm)), Input artifact (set to SourceArtifact), Application name (set to pepper), and Environment name (set to Pepper-env). A checkbox for Configure automatic rollback on stage failure is also present.

Deploy provider: AWS Elastic Beanstalk

Region: Europe (Stockholm)

Input artifact: SourceArtifact

Application name: pepper

Environment name: Pepper-env

Configure automatic rollback on stage failure

The screenshot shows the AWS CodePipeline pipeline details page for pipeline1. The pipeline was saved successfully, and the most recent change will re-run through the pipeline. The pipeline type is V2 and the execution mode is QUEUED. The Source step is listed as succeeded, with a GitHub (Version 1) link and a timestamp of Just now. There are buttons for Notify, Edit, Stop execution, Clone pipeline, and Release change.

Success: Pipeline was saved successfully.

Success: The most recent change will re-run through the pipeline. It might take a few moments for the status of the run to show in the pipeline view.

Developer Tools > CodePipeline > Pipelines > pipeline1

pipeline1

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded
Pipeline execution ID: d1dbb2d1-f0ef-45d8-9f85-81eaa5f02f11

Source
GitHub (Version 1)
Succeeded - Just now
b4894808

ADVANCE DEVOPS EXPERIMENT 3

Prajwal Pandey – D15A/33

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

1.1 Create 3 EC2 instances master, node 1 and node 2.

The screenshot shows the AWS Lambda console interface. At the top, there's a search bar with the placeholder "Search our full catalog including 1000s of application and OS images". Below it, there are two tabs: "Recents" and "Quick Start", with "Quick Start" being the active tab. Under the "Quick Start" tab, there are five pre-defined Lambda functions listed: "Amazon Linux" (with the AWS logo), "macOS" (with a Mac logo), "Ubuntu" (with the Ubuntu logo), "Windows" (with the Microsoft logo), and "Red Hat" (with the Red Hat logo). To the right of these, there's a "Browse more AMIs" button with a magnifying glass icon, followed by the text "Including AMIs from AWS, Marketplace and the Community".

1.2 Create a new key pair (use the same key pair for all 3 instances)

▼ 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*

Select [Create new key pair](#)

|

Proceed without a key pair (Not recommended)	Default value
aws-ubuntu-3 Type: rsa	Edit
AWSLinux Type: rsa	
aws-ubuntu Type: rsa	
AWS Type: rsa	

Auto-assign public IP [Info](#)

Enable

Instances (3) [Info](#) Last updated less than a minute ago [C](#) Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive) All states < 1 > [@](#)

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
<input type="checkbox"/>	Master	i-04f405ec11fee2774	Running @ Q	t2.medium	2/2 checks passed View alarms +	
<input type="checkbox"/>	Node 2	i-02431bc279743d749	Running @ Q	t2.micro	2/2 checks passed View alarms +	
<input type="checkbox"/>	Node 1	i-0b13a0153f7300ccc	Running @ Q	t2.medium	2/2 checks passed View alarms +	

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

The screenshot shows the AWS CloudWatch Metrics interface. The top navigation bar has tabs: EC2 Instance Connect, Session Manager, SSH client (which is selected), and EC2 serial console. Below the tabs, the Instance ID is listed as i-0e3930ceb2d892d01 (Worker-2). A numbered list of steps is provided:

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is two-tier-app-k8s.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 "two-tier-app-k8s.pem"
4. Connect to your instance using its Public DNS:
ec2-13-234-226-219.ap-south-1.compute.amazonaws.com

Below the steps, there is an Example section with the command:

```
ssh -i "two-tier-app-k8s.pem" ubuntu@ec2-13-234-226-219.ap-south-1.compute.amazonaws.com
```

```
acer@TMP214-53 MINGW64 ~/Downloads
$ ssh -i "two-tier-app-k8s.pem" ubuntu@ec2-13-232-36-34.ap-south-1.compute.amazonaws.com
The authenticity of host 'ec2-13-232-36-34.ap-south-1.compute.amazonaws.com (13.232.36.34)' can't be established.
ED25519 key fingerprint is SHA256:uVGEO+FwYefj60j0ft70Sralv8NrzEi/IwxAtBY+EPE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-232-36-34.ap-south-1.compute.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 Wed Sep 11 14:07:10 UTC 2024

System load: 0.0          Processes:      106
Usage of /: 20.7% of 7.57GB  Users logged in:   0
Memory usage: 5%           IPv4 address for eth0: 172.31.45.227
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

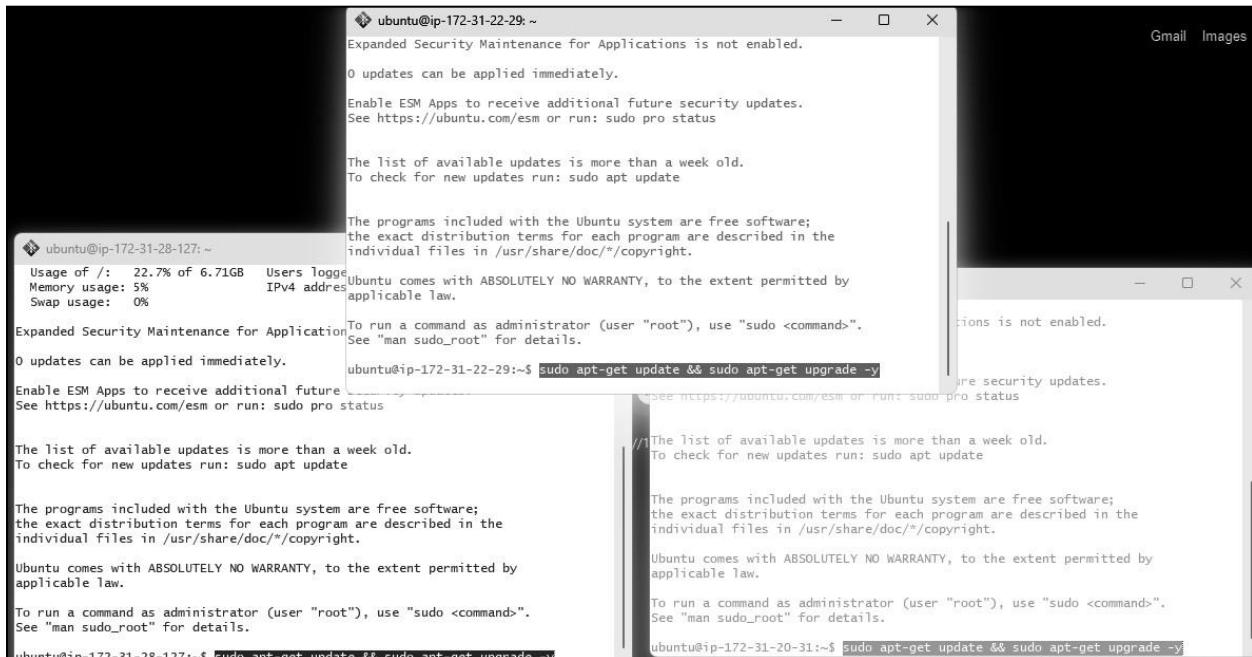
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

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

Step 2: Prepare Nodes

2.1. Update the package manager on all nodes:

```
sudo apt-get update && sudo apt-get upgrade -y
```



The image shows three separate terminal windows side-by-side, each displaying the output of the command `sudo apt-get update && sudo apt-get upgrade -y`. The top window has a title bar "ubuntu@ip-172-31-22-29: ~". The middle window has a title bar "ubuntu@ip-172-31-28-127: ~". The bottom window has a title bar "ubuntu@ip-172-31-20-31: ~". Each window displays system status, update availability, and the execution of the upgrade command.

```
ubuntu@ip-172-31-22-29: ~
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

ubuntu@ip-172-31-28-127: ~
Usage of /: 22.7% of 6.71GB Users logge Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
Memory usage: 5% IPv4 addre applicable law.
Swap usage: 0% See "man sudo_root" for details.

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

0 updates can be applied immediately. ubuntu@ip-172-31-22-29:~$ sudo apt-get update && sudo apt-get upgrade -y

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

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

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

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

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

ubuntu@ip-172-31-28-127:~$ sudo apt-get update && sudo apt-get upgrade -y

ubuntu@ip-172-31-20-31: ~
options is not enabled.
are security updates.
See https://ubuntu.com/esm or run: sudo pro status

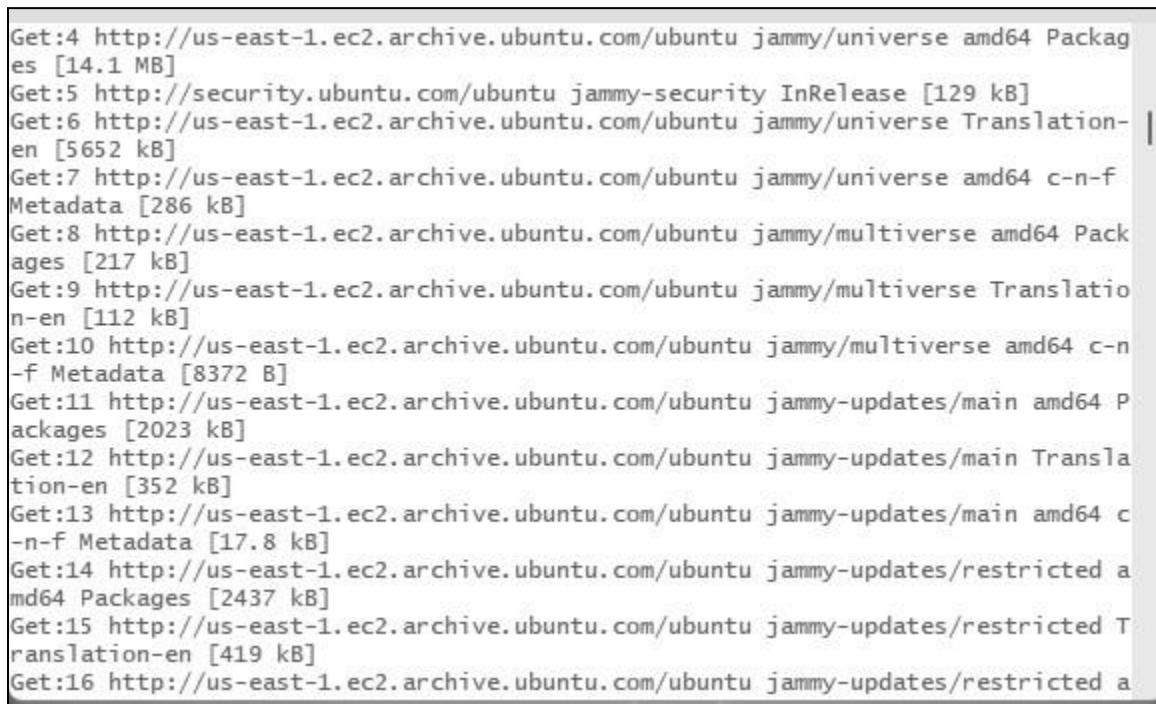
The list of available updates is more than a week old.
To check for new updates run: sudo apt update

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

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

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

ubuntu@ip-172-31-20-31:~$ sudo apt-get update && sudo apt-get upgrade -y
```



The image shows a single terminal window displaying the detailed output of the `sudo apt-get update && sudo apt-get upgrade -y` command. The output is a long list of GET requests for various Ubuntu packages from the US-East-1 archive, including jammy/universe, jammy-security, jammy-updates/main, and jammy-updates/restricted. Each request includes the URL, package name, version, and file size.

```
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2023 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [352 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [17.8 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [2437 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [419 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted a
```

2.2. Disable Swap (Kubernetes requires swap to be off):

```
sudo swapoff -a sudo sed -i '/ swap / s/^/#/' /etc/fstab
```

```
ubuntu@ip-172-31-22-29:~$ sudo swapoff -a  
sudo sed -i '/ swap / s/^/#/' /etc/fstab
```

2.3. Load necessary kernel modules for networking and iptables:

```
cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
```

```
overlay br_netfilter
```

```
EOF
```

```
sudo modprobe overlay
```

```
sudo modprobe br_netfilter
```

```
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf  
overlay  
br_netfilter  
EOF  
sudo modprobe overlay  
sudo modprobe br_netfilter  
overlay  
br_netfilter
```

2.4. Configure sysctl settings for Kubernetes networking:

```
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf net.bridge.bridge-nf-call-ip6tables = 1 net.bridge.bridge-nf-call-iptables = 1
```

```
EOF
```

```
sudo sysctl --system
```

```
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf  
overlay  
br_netfilter  
EOF  
sudo modprobe overlay  
sudo modprobe br_netfilter  
overlay  
br_netfilter  
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf  
net.bridge.bridge-nf-call-ip6tables = 1  
net.bridge.bridge-nf-call-iptables = 1  
EOF  
sudo sysctl --system  
net.bridge.bridge-nf-call-ip6tables = 1  
net.bridge.bridge-nf-call-iptables = 1  
* 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
```

Step 3: Install Docker

Kubernetes uses container runtimes like Docker. Install Docker on all nodes.

```
sudo apt-get update
sudo apt-get install -y apt-transport-https ca-certificates curl
software-properties-common curl -fsSL https://download.docker.com/linux/ubuntu/gpg |
sudo apt-key add sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu
$(lsb_release -cs) stable" sudo apt-get update sudo apt-get
install -y docker-ce docker-ce-cli containerd.io
```

```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y apt-transport-https ca-certificates curl software-proper
ties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubun
tu $(lsb_release -cs) stable"
sudo apt-get update
sudo apt-get install -y docker-ce docker-ce-cli containerd.io
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
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Fetched 129 kB in 1s (241 kB/s)
Reading package lists... Done
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.
curl is already the newest version (7.81.0-1ubuntu1.17).
curl set to manually installed.
software-properties-common is already the newest version (0.99.22.9).
software-properties-common set to manually installed.
```

Configure Docker for Kubernetes:

```
cat <<EOF | sudo tee /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opt": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
```

EOF

sudo systemctl restart docker

```
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opt": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF
sudo systemctl restart docker
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opt": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
```

Step 4: Install kubeadm, kubelet, kubectl

Install Kubernetes tools on all nodes.

4.1. Add Kubernetes APT repository:

```
sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg
https://packages.cloud.google.com/apt/doc/apt-key.gpg echo "deb [signed-
by=/usr/share/keyrings/kubernetes-archive-keyring.gpg]
https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee
/etc/apt/sources.list.d/kubernetes.list
```

```
ubuntu@ip-172-31-22-29:~$ sudo curl -fsSLo /usr/share/keyrings/kubernetes-archiv
e-keyring.gpg https://packages.cloud.google.com/apt/doc/apt-key.gpg
echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://
apt.kubernetes.io/ kubernetes-xenial main" | sudo tee /etc/apt/sources.list.d/k
ubernetes.list
deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.k
ubernetes.io/ kubernetes-xenial main
```

4.2. Install kubeadm, kubelet, and kubectl:

```
sudo apt-get update
sudo apt-get install -y
kubelet kubeadm kubectl
sudo apt-mark hold
kubelet kubeadm kubectl
```

```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
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
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
```

Step 5: Initialize the Kubernetes Cluster on Master Node

On the master node:

```
sudo kubeadm init --pod-network-cidr=10.244.0.0/16
```

```
ubuntu@ip-172-31-22-29:~$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16 --v
=5
Found multiple CRI endpoints on the host. Please define which one do you wish to
use by setting the 'criSocket' field in the kubeadm configuration file: unix://
/var/run/containerd/containerd.sock, unix:///var/run/crio/crio.sock
k8s.io/kubernetes/cmd/kubeadm/app/util/runtime.detectCRISocketImpl
    cmd/kubeadm/app/util/runtime/runtime.go:167
k8s.io/kubernetes/cmd/kubeadm/app/util/runtime.DetectCRISocket
    cmd/kubeadm/app/util/runtime/runtime.go:175
k8s.io/kubernetes/cmd/kubeadm/app/util/config.SetNodeRegistrationDynamicDefaults
    cmd/kubeadm/app/util/config/initconfiguration.go:118
k8s.io/kubernetes/cmd/kubeadm/app/util/config.SetInitDynamicDefaults
    cmd/kubeadm/app/util/config/initconfiguration.go:64
k8s.io/kubernetes/cmd/kubeadm/app/util/config.DefaultedInitConfiguration
    cmd/kubeadm/app/util/config/initconfiguration.go:248
k8s.io/kubernetes/cmd/kubeadm/app/util/config.LoadOrDefaultInitConfiguration
    cmd/kubeadm/app/util/config/initconfiguration.go:282
k8s.io/kubernetes/cmd/kubeadm/app/cmd.newInitData
    cmd/kubeadm/app/cmd/init.go:319
k8s.io/kubernetes/cmd/kubeadm/app/cmd.newCmdInit.func3
    cmd/kubeadm/app/cmd/init.go:170
k8s.io/kubernetes/cmd/kubeadm/app/cmd/phases/workflow.(*Runner).InitData
    cmd/kubeadm/app/cmd/phases/workflow/runner.go:183
k8s.io/kubernetes/cmd/kubeadm/app/cmd.newCmdInit.func1
```

5.1. Set up kubectl on the master node:

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

Step 6: Install a Pod Network Add-on

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

Install Flannel:

```
ubuntu@ip-172-31-22-29:~$ sudo kubeadm config images pull
sudo kubeadm init
mkdir -p "$HOME"/.kube
sudo cp -i /etc/kubernetes/admin.conf "$HOME"/.kube/config
sudo chown "$(id -u)": "$(id -g)" "$HOME"/.kube/config

# Network Plugin = calico
kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml

kubeadm token create --print-join-command --vvv
Found multiple CRI endpoints on the host. Please define which one do you wish to use by setting the 'criSocket' field in the kubeadm configuration file: unix:///var/run/containerd/containerd.sock, unix:///var/run/crio/crio.sock
To see the stack trace of this error execute with --v=5 or higher
Found multiple CRI endpoints on the host. Please define which one do you wish to use by setting the 'criSocket' field in the kubeadm configuration file: unix:///var/run/containerd/containerd.sock, unix:///var/run/crio/crio.sock
kubectl apply -f
https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml
```

Step 7: Join Worker Nodes to the Cluster

On the **worker nodes**, run the command provided by the master node during initialization . It looks something like this:

```
ubuntu@ip-172-31-22-29:~$ kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml --validate=false
E0913 15:35:04.261458 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080: connect: connection refused
E0913 15:35:04.261902 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080: connect: connection refused
E0913 15:35:04.263424 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080: connect: connection refused
E0913 15:35:04.263795 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080: connect: connection refused
E0913 15:35:04.265840 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080: connect: connection refused
E0913 15:35:04.266524 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080: connect: connection refused
unable to recognize "https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml": Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080: connect: connection refused
ubuntu@ip-172-31-22-29:~$ sudo kubeadm join <master-ip>:6443 --token <token> --discovery-token-ca-cert-hash sha256:<hash>
clusterrolebinding.rbac.authorization.k8s.io/calico-cni-plugin created
daemonset.apps/calico-node created
deployment.apps/calico-kube-controllers created
kubeadm join 172.31.62.216:6443 --token br7fe5.hq28adbm1mu17ky --discovery-token-ca-cert-hash sha256:2bc469a8d14fbebe8f879328d2b416fad32b29a8505d3f448b98783fff3b814d9
```

Step 8: Verify the Cluster

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

```
ubuntu@ip-172-31-43-109:~$ kubectl get nodes
NAME           STATUS    ROLES      AGE     VERSION
ip-172-31-39-183   Ready    <none>    8s      v1.29.0
ip-172-31-43-109   Ready    control-plane  4m21s   v1.29.0
ip-172-31-46-153   Ready    <none>    2m54s   v1.29.0
ubuntu@ip-172-31-43-109:~$
```

ADVANCE DEVOPS EXPERIMENT 4

Prajjwal Pandey – D15A/33

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

1. **Install prerequisites:** sudo apt-get update sudo apt-get install -y apt-transport-https ca-certificates curl

- ## 2. Add the GPG key for Kubernetes:

```
sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg  
https://packages.cloud.google.com/apt/doc/apt-key.gpg
```

- ### 3. Add the Kubernetes repository:

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

```
ubuntu@ip-172-31-22-29:~$ sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg https://packages.cloud.google.com/apt/doc/apt-key.gpg
```

```
ubuntu@ip-172-31-22-29:~$ echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-focal main" | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-focal main
```

1.2 Install kubectl

```
sudo apt-get update
```

```
sudo apt-get install -y kubectl
```

```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y kubectl
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]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:6 https://prod-cdn.packages.k8s.io/repositories/isv:/addons:/cri-o:/prerelease:/main/deb InRelease
Ign:7 https://packages.cloud.google.com/apt kubernetes-focal InRelease
Err:8 https://packages.cloud.google.com/apt kubernetes-focal Release
  404 Not Found [IP: 172.253.62.138 443]
Reading package lists... Done
E: The repository 'https://apt.kubernetes.io kubernetes-focal Release' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
kubectl is already the newest version (1.29.0-1.1).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
```

Step 2: Deploying Your Application on Kubernetes

2.1 Set up Kubernetes Cluster

```
kubectl get nodes
```

```
ubuntu@ip-172-31-45-227:~$ kubectl get nodes
NAME           STATUS   ROLES      AGE    VERSION
ip-172-31-43-211 Ready    <none>    50s   v1.29.0
ip-172-31-45-13 Ready    <none>    34s   v1.29.0
ip-172-31-45-227 Ready    control-plane   5m17s  v1.29.0
ubuntu@ip-172-31-45-227:~$ |
```

Step 3: Create the Deployment YAML file

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

b) 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-172-31-45-227: ~
GNU nano 6.2                                     nginx-deployment.yaml
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
```

Step 4:Create the Service YAML File

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

```
ubuntu@ip-172-31-45-227:~$ nano nginx-service.yaml
```

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

```
ubuntu@ip-172-31-45-227: ~
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
```

Step 5:Apply the YAML Files

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

```
ubuntu@ip-172-31-45-227:~$ kubectl apply -f nginx-deployment.yaml
kubectl apply -f nginx-service.yaml
deployment.apps/nginx-deployment created
service/nginx-service created
```

- b)Verify the Deployment: Check the status of your Deployment,Pods and Services.

```
ubuntu@ip-172-31-45-227:~$ kubectl get deployments
kubectl get pods
kubectl get services
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   2/2     2           2           40s
NAME           READY   STATUS      RESTARTS   AGE
nginx-deployment-6b4d6fdbf-6k84m   1/1     Running    0          40s
nginx-deployment-6b4d6fdbf-9d8j6   1/1     Running    0          40s
NAME           TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)      AGE
kubernetes     ClusterIP   10.96.0.1     <none>        443/TCP     40m
nginx-service   LoadBalancer 10.106.182.152 <pending>    80:32317/TCP 40s
```

Describe the deployment(Extra)

```
ubuntu@ip-172-31-45-227:~$ kubectl get deployments
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment  1/1     1           1          14h
ubuntu@ip-172-31-45-227:~$ kubectl describe deployment
Name:            nginx-deployment
Namespace:       default
CreationTimestamp:  Wed, 11 Sep 2024 17:16:17 +0000
Labels:          <none>
Annotations:    deployment.kubernetes.io/revision: 2
Selector:        app=nginx
Replicas:        1 desired | 1 updated | 1 total | 1 available | 0 unavailable
StrategyType:   RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
    nginx:
      Image:      nginx:latest
      Port:       80/TCP
      Host Port:  0/TCP
      Environment:  <none>
      Mounts:
        /usr/share/nginx/html from website-volume (rw)
  Volumes:
    website-volume:
      Type:      ConfigMap (a volume populated by a ConfigMap)
      Name:      nginx-website
      Optional:  false
Conditions:
  Type    Status  Reason
  ----  -----
  Available  True    MinimumReplicasAvailable
  Progressing  True    NewReplicaSetAvailable
OldReplicaSets: nginx-deployment-6b4d6fdbf (0/0 replicas created)
NewReplicaSet:  nginx-deployment-776b8fd845 (1/1 replicas created)
Events:         <none>
```

Step 6:Ensure Service is Running

kubectl get service

```
ubuntu@ip-172-31-45-227:~$ kubectl get service
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes  ClusterIP  10.96.0.1      <none>        443/TCP      16h
nginx     NodePort   10.106.0.176    <none>        80:32618/TCP  76m
nginx-service  NodePort   10.106.182.152  <none>        80:30007/TCP  15h
nginx2     NodePort   10.99.32.156    <none>        80:31421/TCP  8s
```

Step 7:Forward the Service Port to Your Local Machine

kubectl port-forward allows you to forward a port from your local machine to a port on a service running in the Kubernetes cluster.

1. **Forward the Service Port:** Use the following command to forward a local port to the service's target port. `kubectl port-forward service/nginx-service 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
```

This command will forward local port 8080 on your machine to port 80 of the service nginx-service running inside the cluster.

2. This means port forwarding is now active, and any traffic to localhost:8080 will be routed to the nginx-service on port 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
```

Step 8: Access the Application Locally

1. **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.

In case the port 8080 is unavailable, try using a different port like 8081



ADVANCE DEVOPS EXP 5

Name:Prajwal Pandey

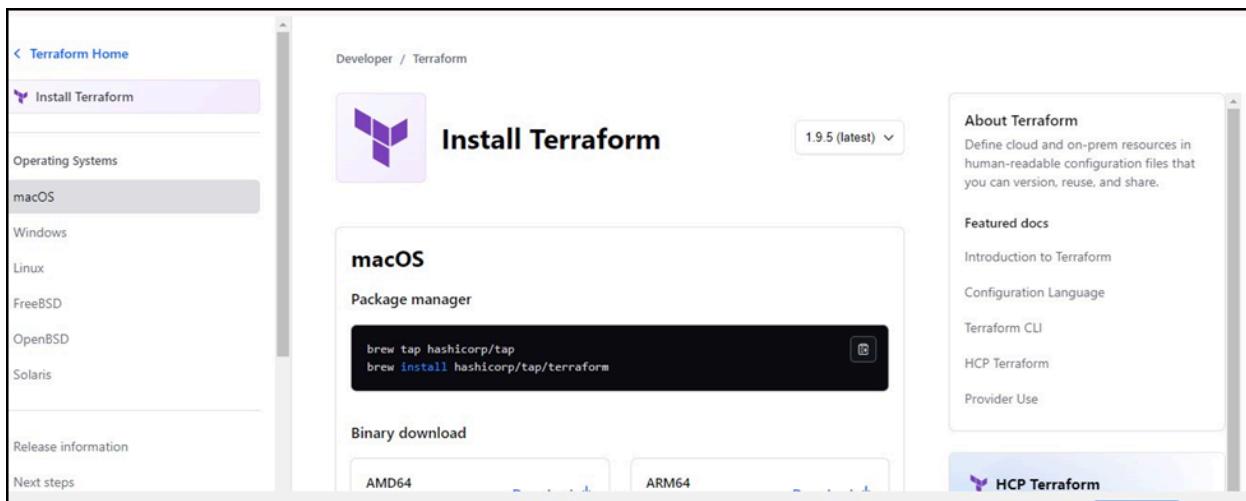
Class:D15A

Roll No:33

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

Installation for Windows:

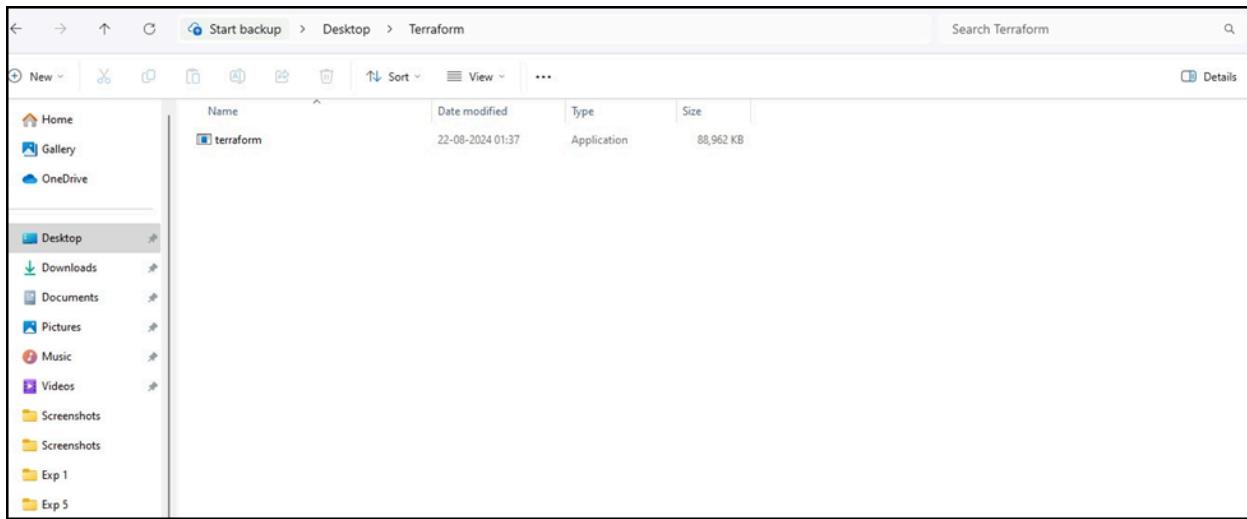
Step 1:Go to the website [terraform.io](https://www.terraform.io) and install Terraform from there..Select the AMD64 option for Windows and download Terraform.



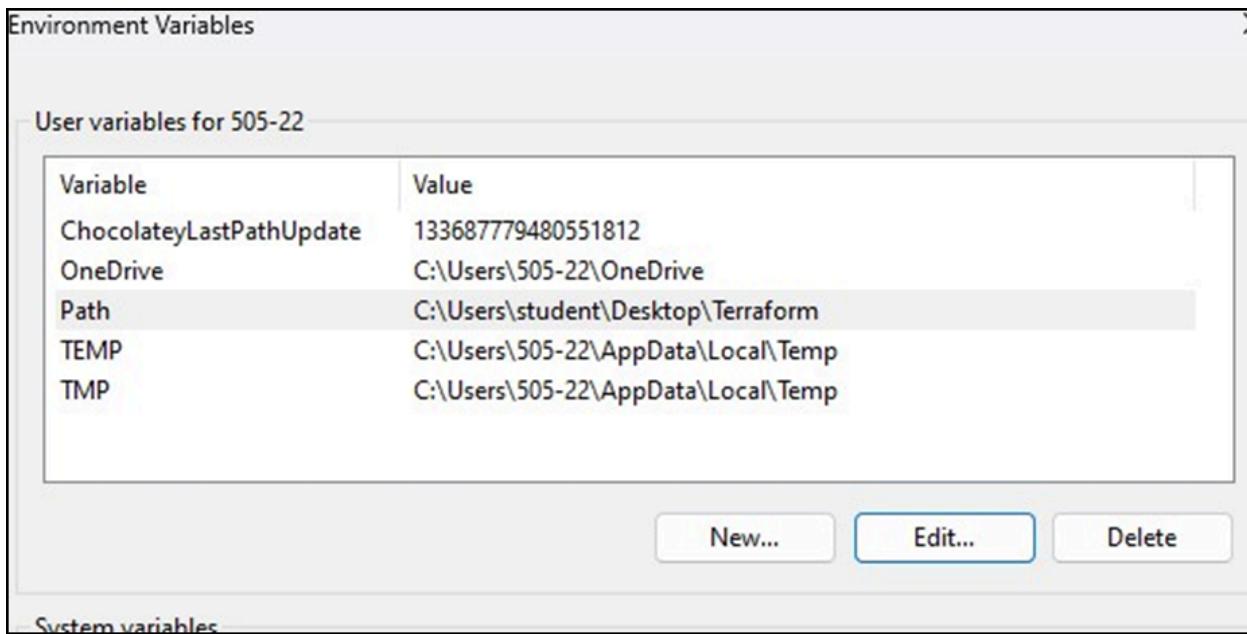
Step 2:Go to the zip file where Terraform is installed.

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.

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

C:\Users\prajj>
```

ADVANCE DEVOPS EXP6

Name:Prajwal Pandey

Class:D15A

Roll No:33

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, Check for the docker version with the following command.

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

C:\Users\prajj>docker --version
Docker version 27.0.3, build 7d4bcd8

C:\Users\prajj>
```

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:

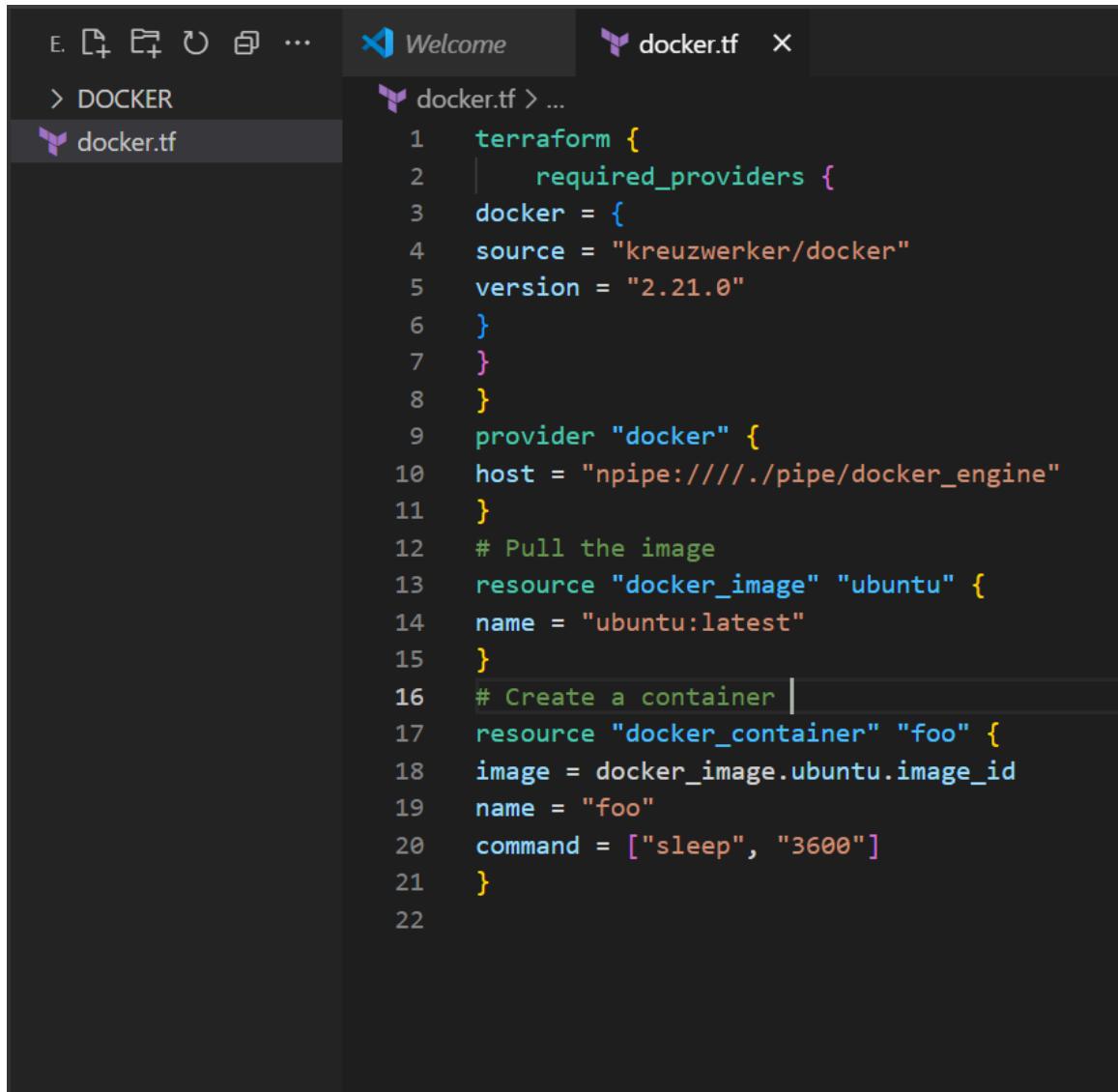
```
terraform { required_providers {
  docker = {
    source = "kreuzwerker/docker" version =
    "2.21.0"
  }
}

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"]
}
```



The screenshot shows a code editor interface with a dark theme. At the top, there's a toolbar with icons for file operations like Open, Save, and Close. To the right of the toolbar is a 'Welcome' button and a tab labeled 'docker.tf > ...'. The main area of the editor displays a Terraform configuration file. The code is color-coded, with keywords in blue and identifiers in purple. The configuration defines a provider 'docker' and a resource 'docker_container'.

```
1  terraform {
2    required_providers {
3      docker = {
4        source = "kreuzwerker/docker"
5        version = "2.21.0"
6      }
7    }
8  }
9  provider "docker" {
10    host = "npipe:///./pipe/docker_engine"
11  }
12 # Pull the image
13 resource "docker_image" "ubuntu" {
14   name = "ubuntu:latest"
15 }
16 # Create a container |
17 resource "docker_container" "foo" {
18   image = docker_image.ubuntu.image_id
19   name = "foo"
20   command = ["sleep", "3600"]
21 }
22
```

Step 3: Execute Terraform Init command to initialize the resources

The screenshot shows a VS Code interface with the following details:

- File Explorer:** Shows files: .terraform, DOCKER, .terraform.lock.hcl, and docker.tf.
- Editor:** The docker.tf file is open, displaying Terraform configuration code:

```
1 terraform {  
2   required_providers {  
3     docker = {  
4       source = "kreuzwerker/docker"  
5       version = "2.21.0"  
6     }  
7   }  
8 }  
9 provider "docker" {  
10   host = "npipe:////./pipe/docker_engine"  
11 }  
12 # Pull the image  
13 resource "docker_image" "ubuntu" {
```

- Terminal:** Shows the output of the Terraform init command:

```
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 fol
+ 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
+ stdio_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 :

```
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.

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

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

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

Enter a value: yes

docker_container.foo: Destroying... [id=01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24]
docker_container.foo: Destruction complete after 0s
docker_image.ubuntu: Destroying... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu: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

Prajwal Pandey - D15A - 33

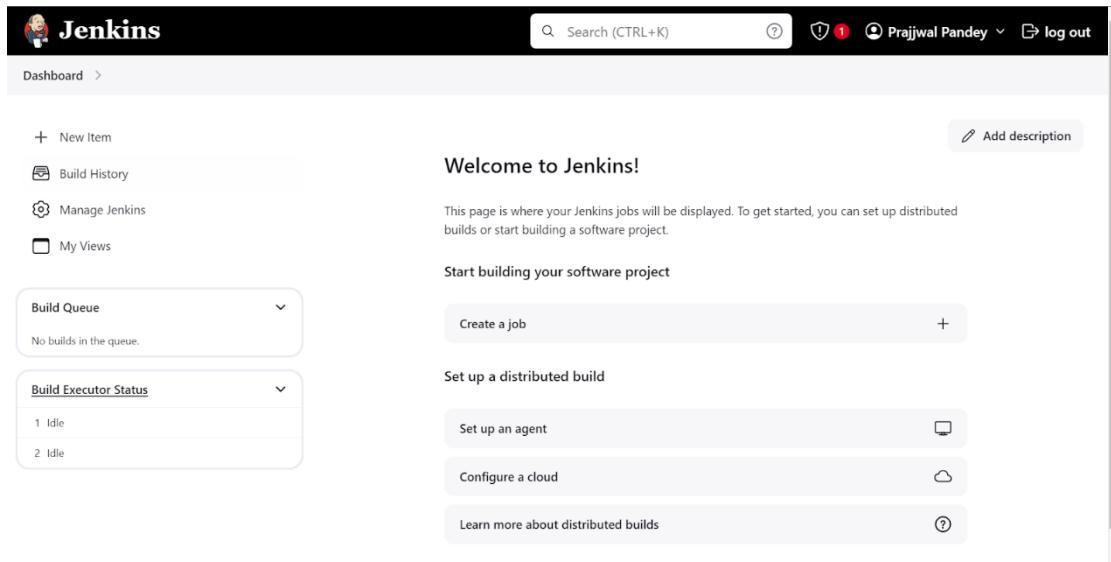
Aim: To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

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.

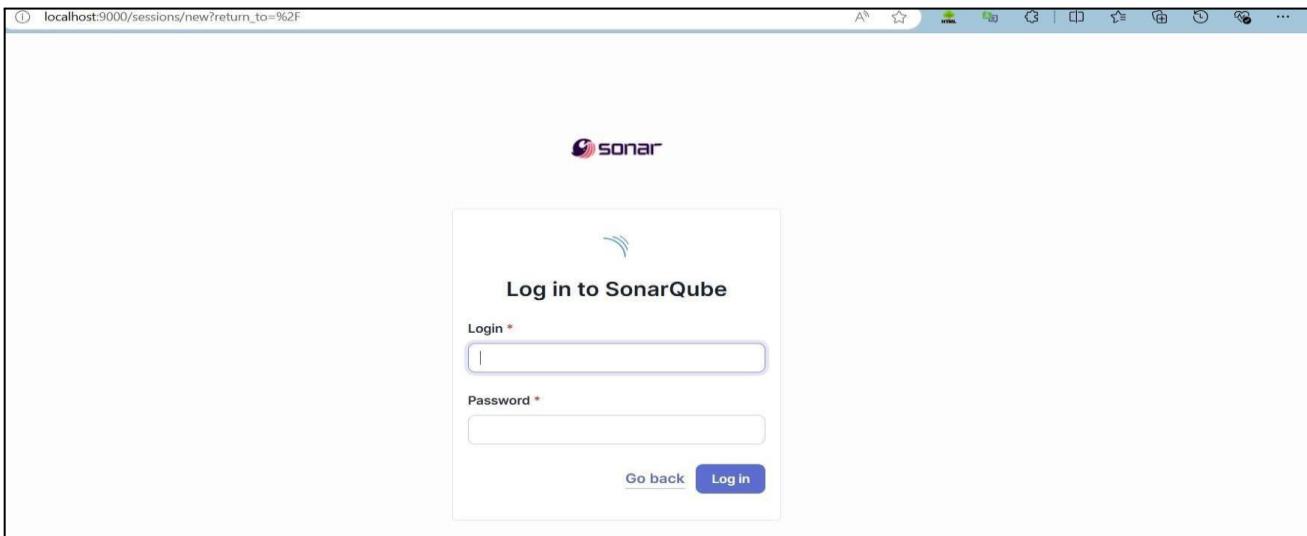


2. Run SonarQube in a Docker container using this command **docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest**

```
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
5bf67120a14329b758ad55b8a8e7495495f619936e93dd5aefd3e624c65e43b6

C:\Windows\System32>
```

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 Setup Import from Bitbucket Cloud Setup Import from Bitbucket Server Setup

Import from GitHub Setup Import from GitLab Setup

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

Setup the project and come back to Jenkins Dashboard.

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

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. 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 considered old. Recommended for projects following continuous delivery.

The screenshot shows the Jenkins 'Plugins' page under 'Manage Jenkins'. The left sidebar has links for 'Updates' (25), 'Available plugins', 'Installed plugins', 'Advanced settings', and 'Download progress' (selected). The main area is titled 'Download progress' and shows the 'SonarQube Scanner' plugin is being loaded. It lists 'Preparation' steps: 'Checking internet connectivity', 'Checking update center connectivity', and 'Success'. Below that, it shows 'Loading plugin extensions' with two 'Success' status indicators. At the bottom, there are links to 'Go back to the top page' and '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>

The screenshot shows the 'SonarQube servers' configuration page. It includes sections for 'Environment variables', 'SonarQube installations' (with a 'List of SonarQube installations' link), 'Name' (set to 'adv_devops_7_sonarqube'), 'Server URL' (set to 'https://localhost:9000'), 'Server authentication token' (set to '- none -'), and an 'Advanced' dropdown.

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 under 'Manage Jenkins'. It includes sections for 'Gradle installations', 'SonarScanner for MSBuild installations', 'SonarQube Scanner installations', and 'Ant installations'. Each section has a 'Add [Tool]' button.

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

Check the “Install automatically” option.

The screenshot shows the 'SonarQube Scanner installations' configuration dialog. It includes fields for 'Name' (set to 'sonarqube_exp') and 'Install automatically' (checkbox checked). A sub-section for 'Install from Maven Central' shows the version 'SonarQube Scanner 6.1.0.4477' selected. There are also 'Add Installer' and 'Add SonarQube Scanner' buttons.

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.

OK

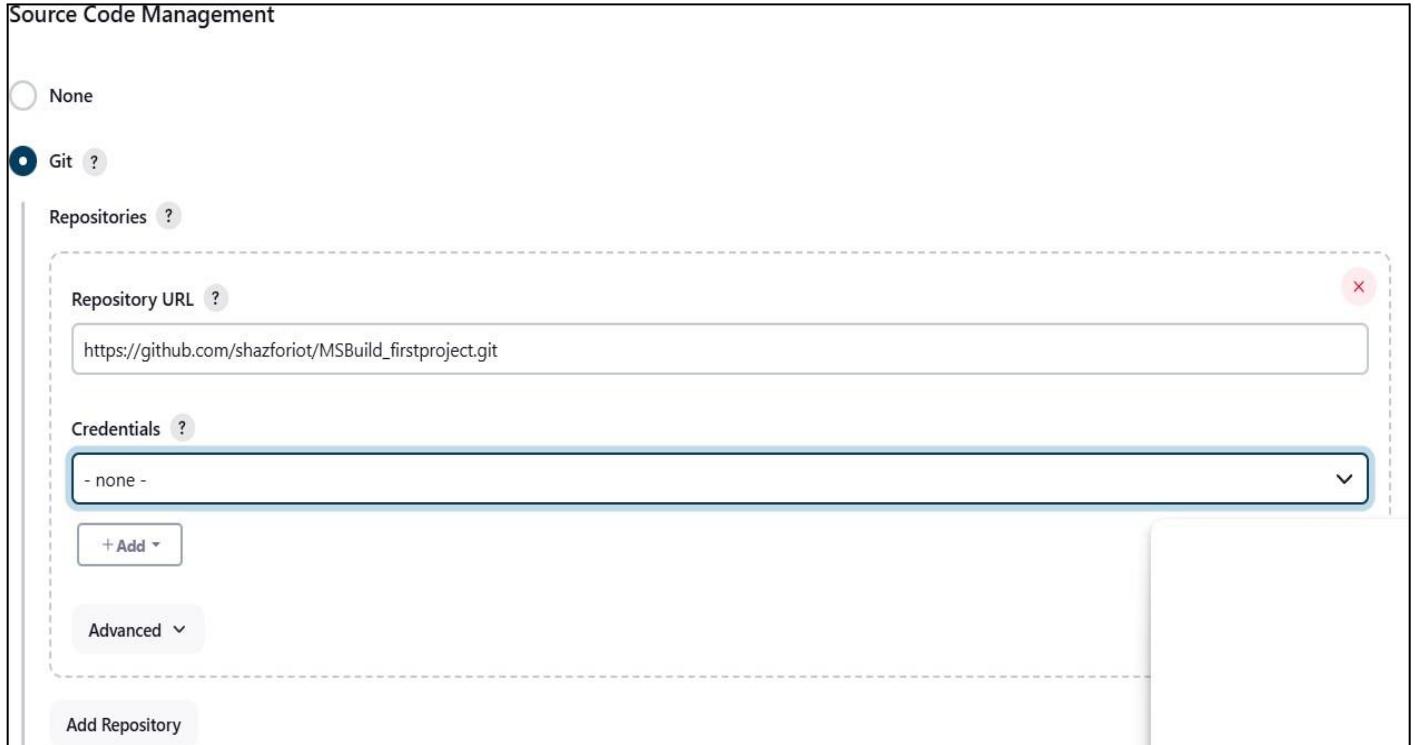
branch Pipeline

Creates a branch of Pipeline projects according to detected branches in one SCM repository.

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.



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.

Configure

-  General
-  Source Code Management
-  Build Triggers
-  Build Environment
-  Build Steps
-  Post-build Actions

Build Environment

- Filter
- 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
 - SonarScanner for MSBuild - End Analysis

Add build step ^

Post-build Actions

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
sonar.login=admin
sonar.sources=.
```

Additional arguments ?

JVM Options ?

Then save

Status **adv_devops_exp7** Add description Disable Project

Changes Workspace Build Now Configure Delete Project SonarQube Rename

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

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More Q

Administration Configuration Security Projects System Marketplace

Global Permissions

Grant and revoke permissions to make changes at the global level. These permissions include editing Quality Profiles, executing analysis, and performing global system administration.

	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

4 of 4 shown

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:

- Click on your **username** in the top-right corner of the SonarQube dashboard.
- Select **My Account** from the dropdown menu.
- Go to the **Security** tab.
- Under **Generate Tokens**, type a name for the token (e.g., "Jenkins-SonarQube").
- Click **Generate**.
- Copy the token and save it securely. You will need it in Jenkins.

Step 2: Update the Token in Jenkins

1. Go to Jenkins Dashboard:

- Open Jenkins and log in with your credentials.

2. Configure the Jenkins Job:

- Go to the job that is running the SonarQube scanner (**adv_devops_exp7**).
- Click **Configure**.

3. Update the SonarQube Token:

- In the SonarQube analysis configuration (either in the pipeline script or under "Build" section, depending on your job type), update the **sonar.login** parameter with the new token.

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=sq_568834b7b5e77a92843e4b3072e044643ce921c1
sonar.sources=.

Additional arguments ?

JVM Options ?

12. Run the Jenkins build.

Status ✓ adv_devops_exp7

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

Permalinks

- Last build (#10), 19 sec ago
- Last stable build (#10), 19 sec ago
- Last successful build (#10), 19 sec ago
- Last failed build (#8), 22 min ago
- Last unsuccessful build (#8), 22 min ago
- Last completed build (#10), 19 sec ago

Build History trend

Filter... /

#10 Sep 18, 2024, 2:36PM

Check the console Output

Dashboard > adv_devops_exp7 > #3 > Console Output

Status ✓ Console Output

- </> Changes
- Console Output**
- Edit Build Information
- Delete build '#3'
- Timings
- Git Build Data

Started by user Prajjwal Pandey
Running as SYSTEM
Building 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
> git.exe config remote.origin.url https://github.com/shazforiot/MSBuild_firstproject.git # timeout=10
Fetching upstream changes from https://github.com/shazforiot/MSBuild_firstproject.git
> git.exe --version # timeout=10

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

The screenshot shows the SonarQube interface for the project 'adv_devops_7.sonarqube'. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. Below the navigation is a breadcrumb trail: star icon → adv_devops_7.sonarqube / main. The main content area has tabs for Overview, Issues, Security Hotspots, Measures, Code, and Activity. The Overview tab is selected. A banner at the top of the main content area encourages users to take a tour or skip it. To the right of the banner, there's a 'Project Settings' dropdown and a 'Project Information' link. The central part of the page displays a large green checkmark icon with the word 'Passed' next to it. A note below the checkmark says 'The last analysis has warnings. See details'. Below this, there are three tabs: 'New Code' (selected), 'Overall Code', 'Security', 'Reliability', and 'Maintainability'. The status bar at the bottom indicates 'Last analysis 5 minutes ago'.

ADVANCE DEVOPS EXP 8

Prajwal Pandey – D15A / 33

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), "Scanners" (which is expanded), "Scanner environment", "SonarScanner CLI", "SonarQube extension for Azure DevOps", "SonarQube extension for Jenkins", "SonarScanner for .NET", and "SonarScanner for Maven". The main content area features a section for "SonarScanner" and "Issue Tracker", with a "Show more" link. A specific release is highlighted: "6.1" (published on "2024-06-27") which supports "macOS and Linux AArch64 distributions". It provides download links for "Linux x64", "Linux AArch64", "Windows x64", "macOS x64", "macOS AArch64", and "Docker Any (Requires a pre-installed JVM)". Below this, there are "Release notes" and two paragraphs of text explaining the use of the SonarScanner CLI and its compatibility with ARM architecture.

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

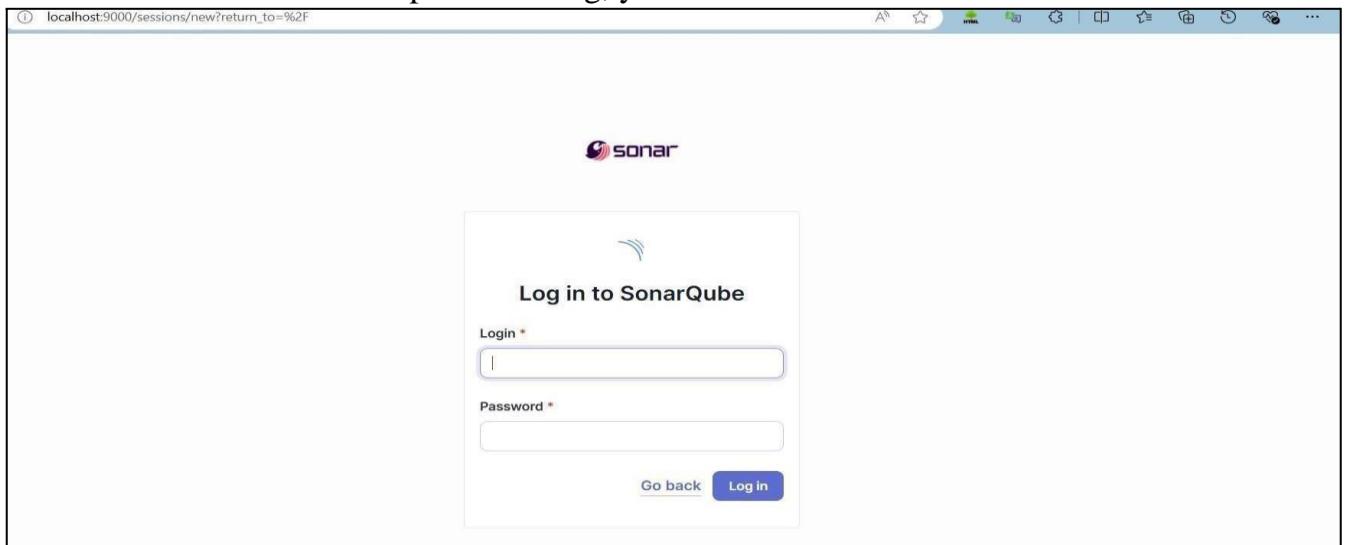
Extract the downloaded zip file in a folder.

The screenshot shows a file explorer window with a dark theme. In the center, there is a list of files and folders. At the top, there are various icons for file operations like New, Cut, Copy, Paste, Sort, View, and Extract all. Below these are two tabs: "Home" and "Gallery". Under the "Home" tab, there is a single item: a folder icon followed by the path "sonar-scanner-6.2.0.4584-windows..." and the type "File folder".

1. Install sonarqube image Command: **docker pull sonarqube**

```
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindc
PS C:\Users\Soham Satpute> 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
```

2. 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.

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 Setup Import from Bitbucket Cloud Setup Import from Bitbucket Server Setup

Import from GitHub Setup Import from GitLab Setup

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

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 *

Sonarqube-test



Project key *

Sonarqube-test



Main branch name *

main

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 attention 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 Jenkins dashboard displays three projects: 'mn' (Last Success: 7 days 10 hr, Last Failure: N/A, Last Duration: 2.6 sec), 'my-app-pipeline' (Last Success: 28 days, Last Failure: N/A, Last Duration: 8.8 sec), and 'my-Maven' (Last Success: 28 days, Last Failure: N/A, Last Duration: 1 min 37 sec). The interface includes a sidebar with options like 'New Item', 'Build History', 'Project Relationship', 'Check File Fingerprint', 'Manage Jenkins', and 'My Views'. A search bar at the top right allows users to search for Jenkins items.

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

The 'Manage Jenkins > Plugins' page shows the search results for 'sonarq'. The 'Available plugins' section is selected. A search bar contains 'sonarq'. The results list the 'SonarQube Scanner' plugin, version 2.17.2, released 6 months 29 days ago. The plugin description states: 'This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.' An 'Install' button is visible on the right.

The 'Manage Jenkins > Plugins' page shows the 'Download progress' section. The 'Available plugins' section is selected. The 'SonarQube Scanner' plugin is listed under 'Preparation' with status 'Success'. Below it, 'Loading plugin extensions' is also marked as 'Success'. A link to 'Go back to the top page' and a checkbox to 'Restart Jenkins when installation is complete and no jobs are running' are shown at the bottom.

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>

8. Search for SonarQube Scanner under Global Tool Configuration.

Name

Server URL

Default is <http://localhost:9000>

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

+ Add ▾

Advanced ▾

Choose the latest configuration and choose Install automatically.

Dashboard > Manage Jenkins > Tools

Dashboard > Manage Jenkins > Tools

Add Git ▾

Gradle installations

Add Gradle

SonarScanner for MSBuild installations

Add SonarScanner for MSBuild

SonarQube Scanner installations

Add SonarQube Scanner

Ant installations

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

Add SonarQube Scanner

SonarQube Scanner**Name**

SonarQube

 Install automatically ?**Install from Maven Central****Version**

SonarQube Scanner 6.2.0.4584

Add Installer ▾

Add SonarQube Scanner

Save**Apply**

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

New Item

Enter an item name

AdDevops-8

Select an item type

**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.

OK

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_Jen
kins>') { 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.

The screenshot shows the Jenkins Pipeline Configuration screen. The left sidebar has tabs for 'General', 'Advanced Project Options', and 'Pipeline'. The 'Pipeline' tab is selected. The main area is titled 'Definition' with a dropdown set to 'Pipeline script'. A large text area contains the Groovy script for the pipeline. At the bottom of the script area, there is a checked checkbox labeled 'Use Groovy Sandbox'. Below the script area are 'Save' and 'Apply' buttons.

```

1 node {
2   stage('Cloning the GitHub Repo') {
3     git 'https://github.com/shazforiot/GOL.git'
4   }
5
6   stage('SonarQube analysis') {
7     withSonarQubeEnv('sonarqube') {
8       bat """
9         cd /Users/Athava_Prabhu/Downloads/sonar-scanner-clt-6.2.0.4584-windows-x64/sonar-scanner-6.2.0.4584-windows-x64/bin/sonar-scanner \
10        -Dsonar.projectKey=Pipeline \
11        -Dsonar.sources=. \
12        -Dsonar.exclusions=**/*_java \
13        -Dsonar.host.url=http://localhost:9000 \
14        -Dsonar.login=admin \
15        -Dsonar.password=admin10 \
16        """
17     }
18   }
19 }
20

```

Use Groovy Sandbox ?

Save Apply

Dashboard > sonarpipe >

Status **sonarpipe** Add description

Stage View

	Cloning the GitHub Repo	SonarQube analysis
Average stage times:	3s	52s
(Average full run time: ~6min 30s)		
#23 Oct 03 09:11 No Changes	3s	6min 26s
#22 Oct 03 09:09 No Changes	4s	1s failed
#21 Oct 03 09:05 No Changes	3s	49s failed
#20 Oct 03 09:01 No Changes	2s	4s failed

Build History trend ▾ Filter... #23 Oct 3, 2024, 9:11 AM #22

11. Check console

Dashboard > sonarpipe > #23

Status **Console Output** Download Copy View as plain text

Console Output

Skipping 4,247 KB.. Full Log

```

09:17:22.526 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/testelement/AbstractScopedAssertion.html for block at line 17. Keep only the first 100 references.
09:17:22.526 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/testelement/AbstractScopedAssertion.html for block at line 529. Keep only the first 100 references.
09:17:22.526 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/testelement/AbstractScopedAssertion.html for block at line 75. Keep only the first 100 references.
09:17:22.579 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/testelement/AbstractScopedAssertion.html for block at line 232. Keep only the first 100 references.
09:17:22.579 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/testelement/AbstractScopedAssertion.html for block at line 353. Keep only the first 100 references.
09:17:22.579 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/testelement/AbstractScopedAssertion.html for block at line 355. Keep only the first 100 references.
09:17:22.579 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/testelement/AbstractScopedAssertion.html for block at line 232. Keep only the first 100 references.
09:17:22.579 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/testelement/AbstractScopedAssertion.html for block at line 32. Keep only the first 100 references.

```

12. Now, check the project in SonarQube:

SonarQube-test / main

Quality Gate Passed Last analysis 26 minutes ago

New Code: Since September 26, 2024 Started 4 days ago

New issues: 0 Required = 0

Accepted issues: 0 Valid issues that were not fixed

Coverage: 0% (Green circle)

Duplications: 0% (Grey circle)

Security Hotspots: 0 (Green circle)

13. code problems consistency:

My Issues All

Filters

Issues in new code

Clean Code Attribute

- Consistency: 197k
- Intentionality: 14k
- Adaptability: 0
- Responsibility: 0

Software Quality

gameoflife-acceptance-tests/Dockerfile

- Use a specific version tag for the image. **Maintainability** Intentionality No tags + L1 • 5min effort • 4 years ago • ⚙ Code Smell • ⚙ Major
- Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. **Maintainability** Intentionality No tags + L12 • 5min effort • 4 years ago • ⚙ Code Smell • ⚙ Major
- Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. **Maintainability** Intentionality No tags + L12 • 5min effort • 4 years ago • ⚙ Code Smell • ⚙ Major

14. Intentionality:

This screenshot shows a code review interface with a sidebar and a main content area. The sidebar contains filters like 'My Issues' and 'All', and sections for 'Issues in new code' and 'Clean Code Attribute'. Under 'Clean Code Attribute', 'Intentionality' is selected, showing 14k issues. The main content area displays three specific code smell findings under the file 'gameoflife-acceptance-tests/Dockerfile'. Each finding includes a checkbox, a description, an 'Intentionality' button, and a status bar with tags and effort information.

gameoflife-acceptance-tests/Dockerfile

Use a specific version tag for the image. Intentionality

Maintainability (1)

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 (1)

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 (1)

No tags +

15. Bugs

This screenshot shows a bug tracking interface with a sidebar and a main content area. The sidebar contains filters for 'Software Quality' (Security, Reliability, Maintainability) and 'Type' (Bug, Vulnerability, Code Smell). Under 'Type', 'Bug' is selected, showing 14k issues. The main content area displays two specific bug findings under files 'gameoflife-core/build/reports/tests/all-tests.html' and 'gameoflife-core/build/reports/tests/allclasses-frame.html'. Each finding includes a checkbox, a description, an 'Intentionality' button, and a status bar with tags and effort information.

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

Add "lang" and/or "xml:lang" attributes to this "<html>" element Intentionality

Reliability (1)

accessibility wcag2-a +

Open Not assigned L1 ~ 2min effort ~ 4 years ago ⚡ Bug ⚡ Major

Add "<th>" headers to this "<table>". Intentionality

Reliability (1)

accessibility wcag2-a +

Open Not assigned L9 ~ 2min effort ~ 4 years ago ⚡ Bug ⚡ Major

gameoflife-core/build/reports/tests/allclasses-frame.html

Embedded database should be used for evaluation purposes only

Code smells:

The screenshot shows the SonarQube interface with the following details:

- Project:** SonarQube-test / main
- Navigation:** Overview, Issues (selected), Security Hotspots, Measures, Code, Activity, Project Settings, Project Information
- Issues List:** Type: Bug (14k), Vulnerability (0), Code Smell (253). The "Code Smell" item is highlighted.
- Issue Details (Top):** gameoflife-web/tools/jmeter/printable_docs/building.html
 - Code Smell:** Add an "alt" attribute to this image.
 - Reliability:** Reliability (dropdown set to Reliability)
 - Intentionality:** accessibility, wcag2-a
 - Status:** Open (radio button selected), Not assigned (radio button)
 - Details:** L29 - 5min effort - 4 years ago - Code Smell - Minor
- Issue Details (Bottom):** gameoflife-web/tools/jmeter/printable_docs/changes.html
 - Code Smell:** Add an "alt" attribute to this image.
 - Reliability:** Reliability (dropdown set to Reliability)
 - Intentionality:** accessibility, wcag2-a
 - Status:** Open (radio button selected), Not assigned (radio button)
 - Details:** L31 - 5min effort - 4 years ago - Code Smell - Minor

Duplications:

Sonarqube-test / main ✓ ?

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

Coverage >

Duplications >

Overview

New Code

Duplicated Lines 0

Duplicated Blocks 0

Overall Code

Density 50.6%

Duplicated Lines 384,007

Duplications Overview ?
(Only showing data for the first 500 files)
See the data presented on this chart as a list

Size: Duplicated Blocks

Zoom: 100%

Duplicated Lines

localhost:9000/component_measures/metric=Duplications&id=Sonarqube-test#

Cyclomatic Complexities:

The screenshot shows the SonarQube interface for the project "Sonarqube-test". The top navigation bar includes links for Overview, Issues, Security Hotspots, Measures (which is selected), Code, and Activity, along with Project Settings and Project Information.

The left sidebar displays various metrics:

- Duplicated Blocks: 0
- Overall Code: 50.6%
- Duplicated Lines: 384,007
- Duplicated Blocks: 42,799
- Duplicated Files: 979

Under the Complexity section, "Cyclomatic Complexity" is highlighted at 1,112. Other sections include Size, Complexity (with a dropdown menu), and Issues.

The main content area shows the "Cyclomatic Complexity" report for the "Sonarqube-test" project. It includes a summary table and a detailed tree view of complexity by component:

Component	Cyclomatic Complexity
gameoflife-acceptance-tests	—
gameoflife-build	—
gameoflife-core	18
gameoflife-deploy	—
gameoflife-web	1,094

A message in the top right corner indicates "New Code: Since September 26, 2024".

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

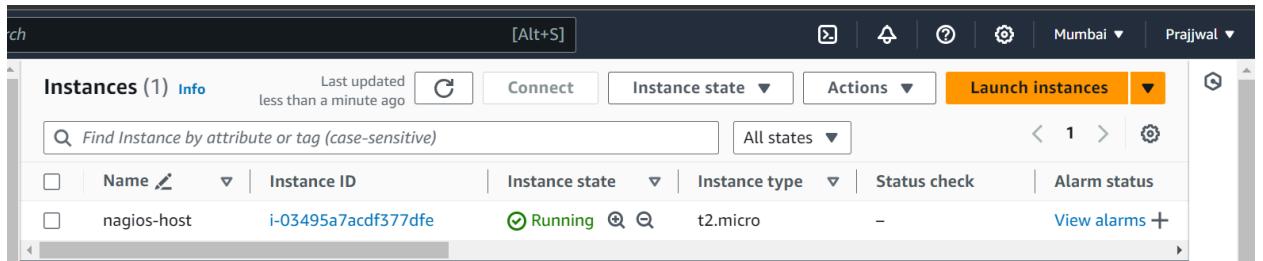
ADVANCE DEVOPS EXPERIMENT - 9

Prajjwal Pandey D15A / 33

AIM: Installation of Nagios

Prerequisites: AWS Free Tier

1. Create an Amazon Linux EC2 Instance in AWS and name it - nagios-host



- Under Security Group, make sure HTTP, HTTPS, SSH, ICMP are open from everywhere.

Inbound rules (7)					
	Name	Security group rule...	IP version	Type	Protocol
1	-	sgr-09141f2ec2b58d69b	IPv4	All UDP	UDP
2	-	sgr-06aae366ef5b84ecc	IPv4	All TCP	TCP
3	-	sgr-0dc9a4fd0834b714c	IPv4	SSH	TCP
4	-	sgr-0ee934299f240201b	IPv4	All ICMP - IPv6	IPv6 ICMP
5	-	sgr-0efe5f88357cea453	IPv4	HTTP	TCP
6	-	sgr-0f1b8e6c4e039e97b	IPv4	HTTPS	TCP
7	-	sgr-0e9a8c015cd45756a	IPv4	All ICMP - IPv4	ICMP

3. SSH into Your EC2 instance or simply use EC2 Instance Connect from the browser.

```
'      #
~\_\_ #####\          Amazon Linux 2023
~~ \_\_ #####\_
~~ \_\_ #####|_
~~ \_\_ #####|_
~~ \_\_ #####|_ https://aws.amazon.com/linux/amazon-linux-2023
~~ \_\_ #####|_ v~' _-->
~~ \_\_ #####|_ /
~~ \_\_ #####|_ /`_
~~ \_\_ #####|_ /`_
[ec2-user@ip-172-31-36-38 ~]$
```

4. Update the package indices and install the following packages using yum

```
sudo yum update
```

```
sudo yum install httpd php
```

```
sudo yum install gcc glibc glibc-common
```

```
sudo yum install gd gd-devel
```

```
libcurl-4.4.0-4.amzn2023.0.18.x86_64
libwebp-1.2.4-1.amzn2023.0.6.x86_64
libxcb-1.13.1-7.amzn2023.0.2.x86_64
libxml2-devel-2.10.4-1.amzn2023.0.6.x86_64
pcre2-utf16-10.40-1.amzn2023.0.3.x86_64
pixman-0.40.0-3.amzn2023.0.3.x86_64
xml-common-0.6.3-56.amzn2023.0.2.noarch
xz-devel-5.2.5-9.amzn2023.0.2.x86_64
```

```
libcurl
libwebp
libxcb
pcre2-d
pcre2-u
sysprof
xorg-x11
zlib-de
```

```
Complete!
```

```
[ec2-user@ip-172-31-36-38 ~]$
```

i-03495a7acdf377dfe (nagios-host)

Public IPs: 43.205.233.133 Private IPs: 172.31.36.38

5. Create a new Nagios User with its password. You'll have to enter the password twice for Confirmation.

```
sudo adduser -m nagios
```

```
sudo passwd nagios
```

```
Sorry, passwords do not match.
```

```
New password:
```

```
BAD PASSWORD: The password contains the user name in some form
Retype new password:
```

```
passwd: all authentication tokens updated successfully.
```

```
[ec2-user@ip-172-31-36-38 ~]$
```

6. Create a new user group

```
sudo groupadd nagcmd
```

7. Use these commands so that you don't have to use sudo for Apache and Nagios

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

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

8. Create a new directory for Nagios downloads

```
mkdir ~/downloads
```

```
cd ~/downloads
```

9. Use wget to download the source zip files.

```
wget
```

<http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz>

```
wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
```

```
nagios-plugins-2.0.3.tar.gz      100%[=====>] 2024-09-29 14:00:39 (1.10 MB/s) - `nagios-plugins-2.0.3.tar.gz' saved [2659772/2659772]  
[ec2-user@ip-172-31-36-38 ~]$
```

i-03495a7acdf377dfe (nagios-host)

Public IPs: 43.205.233.133 Private IPs: 172.31.36.38

10. Use tar to unzip and change to that directory.

```
tar zxvf nagios-4.0.8.tar.gz
```

11. Run the configuration script with the same group name you previously created.

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

```
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.

12. Compile the source code.

```
make all
```

13. Install binaries, init script and sample config files. Lastly, set permissions on the external command directory.

```
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 ***

```

14. Edit the config file and change the email address.

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

```

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

define contact{
    contact_name           nagiosadmin          ; Short name of user
    use                   generic-contact        ; Inherit default values from generic-contact template
    alias                Nagios Admin         ; Full name of user
}

^O      email            prajjwal0904@gmail.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
}

^X

```

15. Configure the web interface.

```
sudo make install-webconf
```

```

*** External command directory configured ***

[ec2-user@ip-172-31-36-38 nagios-4.0.8]$ sudo nano /usr/local/nagios/etc/objects/contacts.cfg
[ec2-user@ip-172-31-36-38 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 ***

```

16. Create a nagiosadmin account for nagios login along with password. You'll have to specify the password twice.

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

```

[ec2-user@ip-172-31-36-38 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
[ec2-user@ip-172-31-36-38 nagios-4.0.8]$ 

```

17. Restart Apache

```
sudo service httpd restart
```

18. Go back to the downloads folder and unzip the plugins zip file.

```
cd ~/downloads
```

```
tar zxvf nagios-plugins-2.0.3.tar.gz
```

19. Compile and install plugins

```
cd nagios-plugins-2.0.3  
.configure --with-nagios-user=nagios --with-nagios-group=nagios  
make  
sudo make install
```

20. Start Nagios

Add Nagios to the list of system services

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

```
sudo chkconfig nagios on
```

Verify the sample configuration files

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

If there are no errors, you can go ahead and start Nagios.

```
sudo service nagios start
```

```
Total Warnings: 0  
Total Errors: 0  
  
Things look okay - No serious problems were detected during the pre-flight check  
[ec2-user@ip-172-31-36-38 bin]$ sudo service nagios start  
Reloading systemd: [ OK ]  
Starting nagios (via systemctl): [ OK ]  
[ec2-user@ip-172-31-36-38 bin]$ █
```

21. Check the status of Nagios

```
sudo systemctl status nagios
```

```
Starting nagios (via systemctl): [ OK ]  
[ec2-user@ip-172-31-36-38 bin]$ sudo systemctl status nagios  
● nagios.service - LSB: Starts and stops the Nagios monitoring server  
    Loaded: loaded (/etc/rc.d/init.d/nagios; generated)  
    Active: active (running) since Sun 2024-09-29 15:15:32 UTC; 45s ago  
      Docs: man:systemd-sysv-generator(8)  
    Process: 72490 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)  
     Tasks: 6 (limit: 1112)  
    Memory: 2.2M  
       CPU: 56ms  
      CGroup: /system.slice/nagios.service
```

22. Go back to EC2 Console and copy the Public IP address of this instance

The screenshot shows the AWS CloudWatch Instances console. At the top, there's a search bar with placeholder text 'Find Instance by attribute or tag (case-sensitive)' and a dropdown menu set to 'All states'. Below the search bar, there are filters for 'Instance state = running' and 'Clear filters'. The main table lists one instance: 'nagios-host' with Instance ID 'i-03495a7acdf377dfe'. The instance is shown as 'Running' with a green checkmark. It's of type 't2.micro' and has 2/2 checks passed. There are buttons for 'View alarms' and '+'. On the right side of the table, there are navigation arrows and a settings gear icon.

i-03495a7acdf377dfe (nagios-host)

Instance ID: i-03495a7acdf377dfe (nagios-host)

Public IPv4 address: 43.205.233.133 | [open address](#)

IPv6 address: -

Instance state: Running

Public IPv4 DNS: 172.31.36.38

23. Open up your browser and look for `http://<your_public_ip_address>/nagios`
Enter username as nagiosadmin and password which you set in Step 16.

24. After entering the correct credentials, you will see this page.

Nagios® Core

General

Current Status

- Tactical Overview
- Map
- Hosts
- Services
- Host Groups
- Service Groups
- Grid
- Service Grids
- Inventory
- Grid
- Plugins
- Services (Unitored)
- Hosts (Unitored)
- Network Checks
- Quick Search

Reports

- Availability
- Timezone
- Alerts
- History
- Inventory
- Integrations
- Notifications
- Event Log

Systems

- Components
- Downtime
- Process Info
- Performance Data
- Scheduling Queue
- Configuration

Nagios® Core™ Version 4.0.8
August 12, 2014
Check for updates

A new version of Nagios Core is available!
Visit [nagios.org](#) to download Nagios 4.4.6.

Nagios XI
Easy Configuration Advanced Reporting
[Download](#)

Nagios Log Server
Monitor and analyze logs from anywhere
[Download](#)

Nagios Network Analyzer
Real-time network and bandwidth analysis
[Download](#)

Get Started

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

Latest News

- Nagios Update: 4.0.8.5
- Nagios Update: 4.0.8.4
- Nagios Update: 4.0.8.3
- More news...

Quick Links

- Nagios Links (Materials and docs)
- Nagios Labs (development tools)
- Nagios Exchange (plugins and add-ons)
- Nagios Support (tech support)
- Nagios.com (company)
- Nagios.org (greggit)

Don't Miss...

- Monitoring Log Data with Nagios - Nagios Log Server can handle all log data
- Can Nagios monitor network? - Yes! Nagios Network Analyzer can take in a variety of flow data. Learn More
- Nagios XI 5 Available Now! - Easier configuration. Advanced Reporting. Get started Today!

Copyright © 2010-2014 Nagios Core Development Team and Community Contributors. Copyright © 1998-2009 Ethan Galstad. See the THINNAK license for more information on contributions.

This means that Nagios was correctly installed and configured with its plugins so far.

ADVANCE DEVOPS EXPERIMENT - 10

Prajwal Pandey D15A / 33

Prerequisites

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

Steps

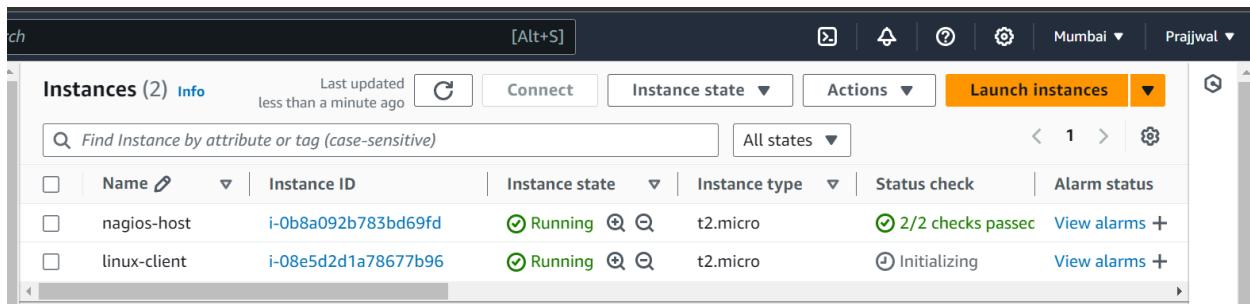
1. Confirm Nagios is Running

```
sudo systemctl status nagios
```

```
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Tue 2024-10-01 16:21:57 UTC; 8s ago
     Docs: https://www.nagios.org/documentation
 Process: 67457 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
 Process: 67458 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (o
 Main PID: 67459 (nagios)
   Tasks: 6 (limit: 1112)
  Memory: 2.0M
    CPU: 17ms
   CGroup: /system.slice/nagios.service
           ├─67459 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           ├─67460 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─67461 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─67462 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─67463 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           └─67464 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
```

2. Create a Linux Client

- Create an Ubuntu 20.04 server EC2 instance in AWS.
- Assign it the same security group as the Nagios Host.
- Name it linux-client.



3. Verify Nagios Process

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

```
[ec2-user@ip-172-31-37-184 nagios-plugins-2.3.3]$ ps -ef | grep nagios
nagios   67459      1  0 16:21 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios   67460  67459  0 16:21 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   67461  67459  0 16:21 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   67462  67459  0 16:21 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   67463  67459  0 16:21 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   67464  67459  0 16:21 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user  68320     2728  0 16:37 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-37-184 nagios-plugins-2.3.3]$
```

4. Create Directories for Monitoring Hosts

```
sudo su
```

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

5. Copy and Modify Configuration File

```
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
```

- Change hostname to linuxserver everywhere in the file.
- Change address to the public IP address of your linux-client.
- Change hostgroup_name under hostgroup to linux-servers1.

6. Update Nagios Configuration

```
Add the following line: cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/
```

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

7. Verify Configuration Files

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

```
Checked 2 hosts.
Checked 2 host groups.
Checked 0 service groups.
Checked 1 contacts.
Checked 1 contact groups.
Checked 24 commands.
Checked 5 time periods.
Checked 0 host escalations.
Checked 0 service escalations.
Checking for circular paths...
    Checked 2 hosts
    Checked 0 service dependencies
    Checked 0 host dependencies
    Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors: 0
```

8. Restart Nagios Service

```
sudo systemctl restart nagios
```

9. Switch to Client Machine

- SSH into the linux-client machine or use the EC2 Instance Connect feature.

10. Install Required Packages on Client.

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

11. Configure NRPE

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

- Under allowed_hosts, add your Nagios host IP address.

```
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 mask
# (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=13.201.125.190
```

12. Restart NRPE Server

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

13. Check Nagios Dashboard

- Go to your Nagios dashboard.
- Click on Hosts and then on `linuxserver` to see the host details.
- Click on Services to see all services and ports being monitored.

Host Information

Last Updated: Tue Oct 1 16:52:19 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as *nagiosadmin*

[View Status Detail For This Host](#)
[View Alert History For This Host](#)
[View Trends For This Host](#)
[View Alert Histogram For This Host](#)
[View Availability Report For This Host](#)
[View Notifications For This Host](#)

Host
localhost
(linuxserver)

Member of
No hostgroups

13.234.202.182

Host State Information

Host Status:	DOWN (for 0d 0h 6m 37s)
Status Information:	PING CRITICAL - Packet loss = 100%
Performance Data:	rta=5000.000000ms;3000.000000;5000.000000;0.000000 pl=100%;80;100;4/10 (SOFT state)
Current Attempt:	10-01-2024 16:50:12
Last Check Time:	ACTIVE
Check Type:	0.000 / 30.003 seconds
Next Scheduled Active Check:	10-01-2024 16:51:42
Last State Change:	10-01-2024 16:45:42
Last Notification:	N/A (notification 0)
Is This Host Flapping?	NO (5.86% state change)
In Scheduled Downtime?	NO
Last Update:	10-01-2024 16:52:11 (0d 0h 0m 8s ago)
Active Checks:	ENABLED
Passive Checks:	ENABLED
Obsessing:	ENABLED
Notifications:	ENABLED
Event Handler:	ENABLED
Flap Detection:	ENABLED

Current Network Status

Last Updated: Tue Oct 1 16:53:13 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as *nagiosadmin*

[View History For all hosts](#)
[View Notifications For All Hosts](#)
[View Host Status Detail For All Hosts](#)

Host Status Totals

Up	Down	Unreachable	Pending
1	1	0	0
All Problems			All Types
1	2		

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
6	1	0	1	0
All Problems			All Types	
2	8			

Service Status Details For All Hosts

Service Status Details For All Hosts						
Host		Service		Status		Status Information
Host	Service	Status	Last Check	Duration	Attempt	
localhost	Current Load	OK	10-01-2024 16:52:34	0d 0h 30m 39s	1/4	OK - load average: 0.00, 0.00, 0.00
localhost	Current Users	OK	10-01-2024 16:48:12	0d 0h 30m 1s	1/4	USERS OK - 2 users currently logged in
localhost	HTTP	WARNING	10-01-2024 16:51:49	0d 0h 26m 24s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.001 second response time
localhost	PING	OK	10-01-2024 16:49:27	0d 0h 28m 46s	1/4	PING OK - Packet loss = 0%, RTA = 0.05 ms
localhost	Root Partition	OK	10-01-2024 16:50:04	0d 0h 28m 9s	1/4	DISK OK - free space: / 6080 MB (74.92% inode=98%).
localhost	SSH	OK	10-01-2024 16:51:57	0d 0h 27m 31s	1/4	SSH OK - OpenSSH_8_7 (protocol 2.0)
localhost	Swap Usage	CRITICAL	10-01-2024 16:49:19	0d 0h 23m 54s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
localhost	Total Processes	OK	10-01-2024 16:51:57	0d 0h 26m 16s	1/4	PROCS OK: 39 processes with STATE = R/SZDT

Results 1 - 8 of 8 Matching Services

ADVANCE DEVOPS EXPERIMENT - 11

Prajwal Pandey D15A / 33

Steps:

1. Open up the Lambda Console and click on the Create button.
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.

Create function Info

Choose one of the following options to create your function.

Author from scratch
Start with a simple Hello World example.

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

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

Basic information

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

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

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

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

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

Lambda > Functions > lambdaPrajwal

lambdaPrajwal

Throttle Copy ARN Actions ▾

▼ Function overview Info

Diagram Template

 **lambdaPrajwal**
 Layers (0)

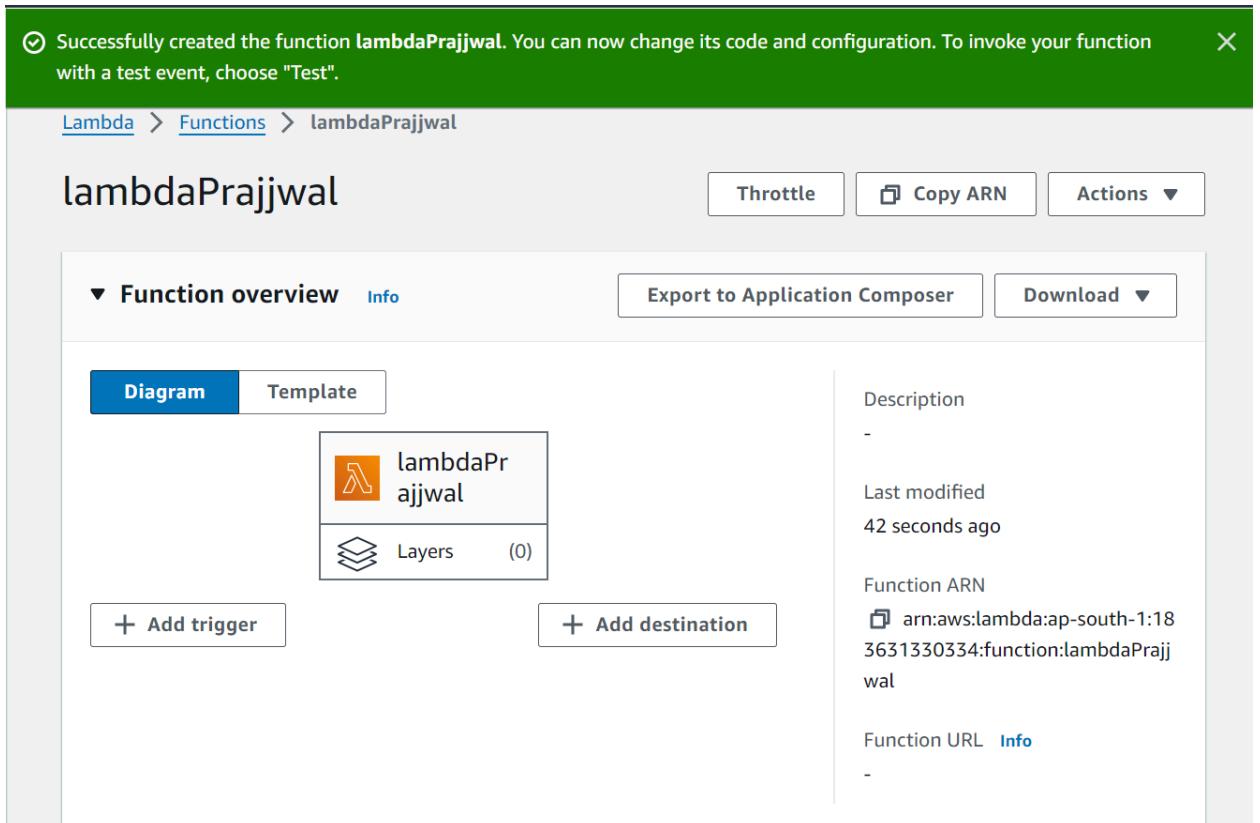
+ Add trigger + Add destination

Description
-

Last modified
42 seconds ago

Function ARN
arn:aws:lambda:ap-south-1:183631330334:function:lambdaPrajwal

Function URL Info
-



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

File Edit Find View Go Tools Window Test Deploy

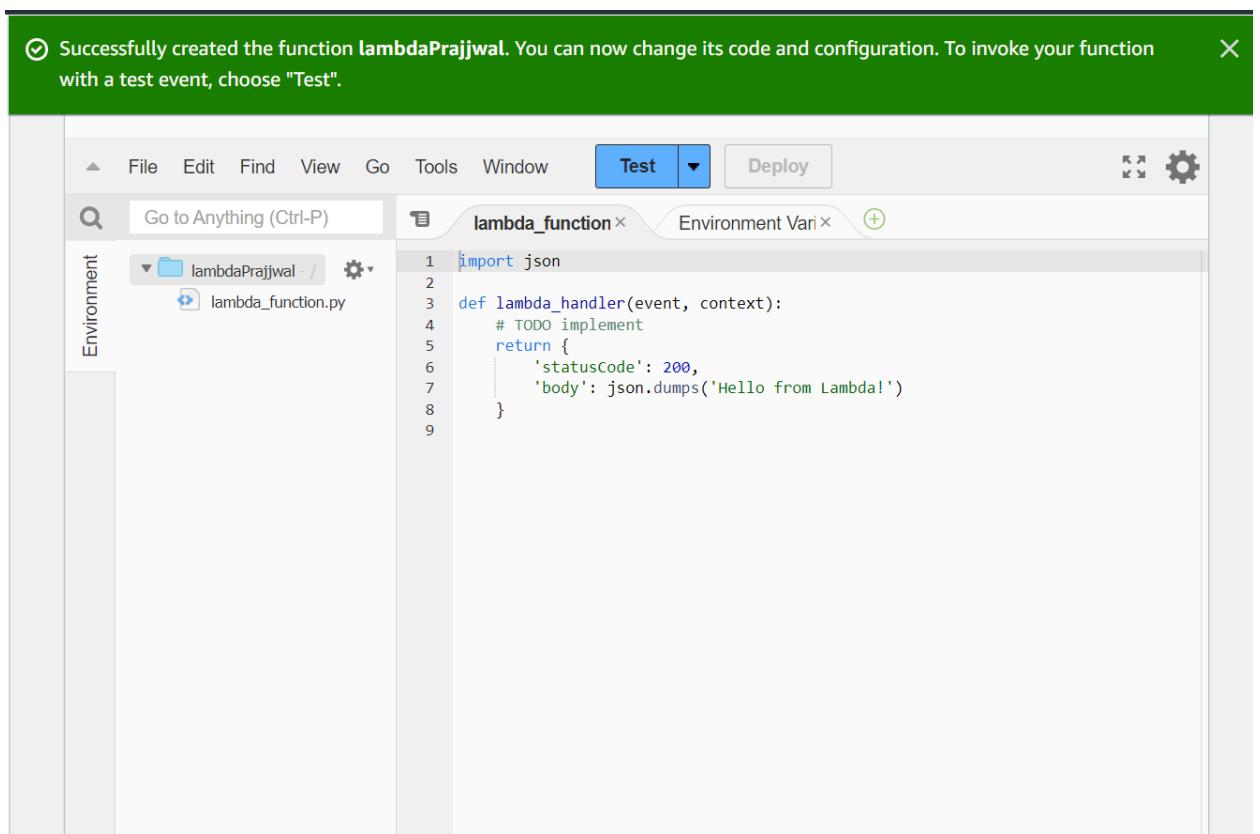
Go to Anything (Ctrl-P) lambda_function Environment Vari Environment

Environment

lambdaPrajwal /

lambda_function.py

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



3. 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 Configuration page. The top navigation bar has tabs: Code, Test, Monitor, Configuration (which is highlighted in blue), Aliases, and Versions. On the left, there's a sidebar with links: General configuration (which is also highlighted in blue), Triggers, Permissions, Destinations, Function URL, Environment variables, and Tags. The main content area is titled "General configuration" with an "Edit" button. It contains the following settings:

	Description	Value
Memory	-	128 MB
Ephemeral storage	512 MB	Timeout
SnapStart	None	0 min 3 sec

The screenshot shows the AWS Lambda Configuration page with modifications made in the General configuration section:

- Memory:** Set to between 128 MB and 10240 MB.
- Ephemeral storage:** Set to 512 MB.
- SnapStart:** Set to None.
- Timeout:** Set to 1 second.
- Execution role:** Set to "Use an existing role" with the value "service-role/lambdaPrajwal-role-upwhq7h6".

- 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.

Successfully updated the function lambdaPrajwal.

Code **Test** Monitor Configuration Aliases Versions

Test event [Info](#)

To invoke your function without saving an event, configure the JSON event, then choose Test.

Test event action

Create new event Edit saved event

Event name

PrajwalEvent

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

Event sharing settings

Private
This event is only available in the Lambda console and to the event creator. You can configure a total of 10. [Learn more](#)

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

Template - *optional*

hello-world

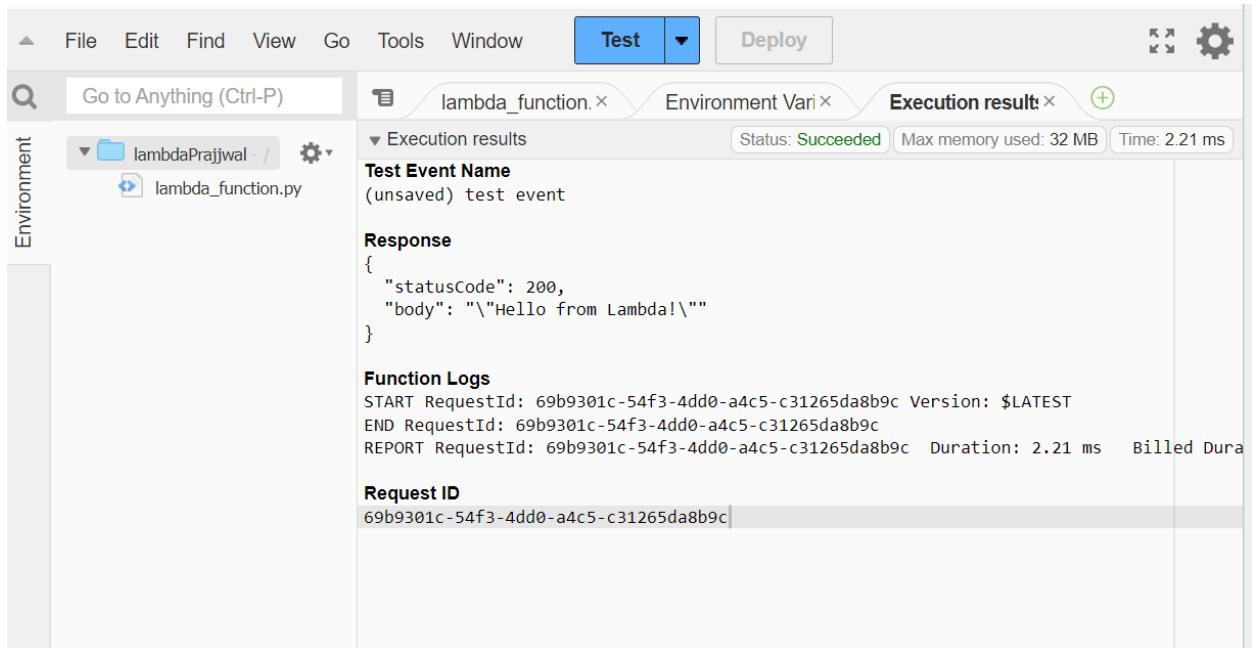
Event JSON

1 ↴ {
2 "key1": "value1",
3 "key2": "value2",
4 "key3": "value3"
5 }

Format JSON

1:1 JSON Spaces: 2

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



ADVANCE DEVOPS EXPERIMENT - 12

Prajwal Pandey D15A / 33

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

Theory:

AWS Lambda and S3 Integration: AWS Lambda allows you to execute code in response to various events, including those triggered by Amazon S3. When an object is added to an S3 bucket, it can trigger a Lambda function to execute, allowing for event-driven processing without managing servers.

Workflow:

1. Create an S3 Bucket:

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

2. Create the Lambda Function:

- Set up a new Lambda function using AWS Lambda’s console. You can choose a runtime environment like Python, Node.js, or Java.

- Write code that logs a message like “An Image has been added” when triggered.

3. Set Up Permissions:

- Ensure that the Lambda function has the necessary permissions to access S3.

You can do this by attaching an IAM role with policies that allow reading from the bucket and writing logs to CloudWatch.

4. Configure S3 Trigger:

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

5. Test the Setup:

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

STEPS:-

1. Create an S3 bucket of the same location as that of the Lambda function.

Create bucket [Info](#)

Buckets are containers for data stored in S3.

General configuration

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

Bucket name [Info](#)

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

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

[Choose bucket](#)

Format: s3://bucket/prefix

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

Bucket Versioning
 Disable
 Enable

Tags - *optional* (0)
You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

No tags associated with this bucket.
[Add tag](#)

Default encryption [Info](#)
Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)
 Server-side encryption with Amazon S3 managed keys (SSE-S3)
 Server-side encryption with AWS Key Management Service keys (SSE-KMS)
 Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the [Storage](#) tab of the [Amazon S3 pricing page](#).

Bucket Key
Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-

The screenshot shows the Amazon S3 console with the path [Amazon S3](#) > [Buckets](#) > [prajjbucket](#). The main title is **prajjbucket** [Info](#). Below it is a navigation bar with tabs: **Objects** (selected), Properties, Permissions, Metrics, Management, and Access Points. The main content area is titled **Objects (0)** [Info](#). It contains a toolbar with buttons for **C**, **Copy S3 URI**, **Copy URL**, **Download**, **Open**, **Delete**, **Actions** (with a dropdown arrow), **Create folder**, and **Upload** (highlighted in orange). A note below the toolbar states: "Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)". There is a search bar with placeholder text "Find objects by prefix". Below the search bar is a table header with columns: **Name**, **Type**, **Last modified**, **Size**, and **Storage class**. The table body displays the message "No objects" and "You don't have any objects in this bucket.". At the bottom right of the table area is a **Upload** button.

2. Add roles while creating the Lambda function and give permissions for accessing the S3 bucket.

The screenshot shows the AWS Lambda function creation wizard. It starts with three options:

- Author from scratch**: Start with a simple Hello World example. This option is selected.
- 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.

Below these options is a section titled **Basic information**.

Function name
Enter a name that describes the purpose of your function.
 (The input field is highlighted with a blue border.)
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.
 (The input field is highlighted with a blue border.) [▼](#) [C](#)

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

- x86_64** (Selected, indicated by a blue dot)
- arm64**

▼ Change default execution role

Execution role

Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

- Create a new role with basic Lambda permissions
- Use an existing role
- Create a new role from AWS policy templates

i Role creation might take a few minutes. Please do not delete the role or edit the trust or permissions policies in this role.

Role name

Enter a name for your new role.

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

Policy templates - optional [Info](#)

Choose one or more policy templates.

▼

Amazon S3 object read-only permissions X
S3

✓ Successfully created the function **prajjwallambda01**. You can now change its code and configuration. To invoke your function with a test event, choose "Test". X

[Lambda](#) > [Functions](#) > prajjwallambda01

prajjwallambda01

[Throttle](#)[Copy ARN](#)[Actions ▾](#)

▼ Function overview [Info](#)

[Export to Application Composer](#)[Download ▾](#)[Diagram](#)[Template](#)[+ Add trigger](#)[+ Add destination](#)

Description

-

Last modified

3 seconds ago

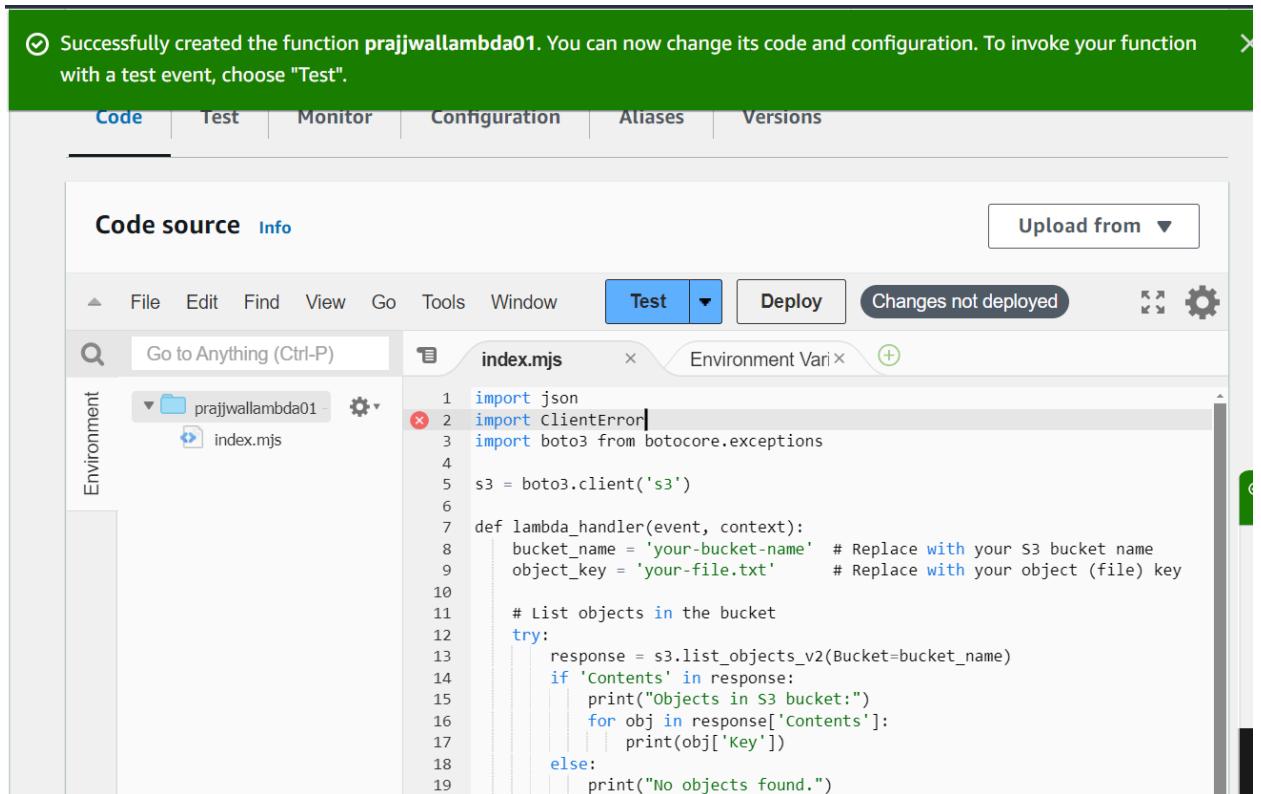
Function ARN

arn:aws:lambda:ap-south-1:183631330334:function:prajjwallambda01

Function URL [Info](#)

-

3. After creating the Lambda function copy a code available on the internet which allows the Lambda function to access the S3 bucket contents.



Successfully created the function **prajjwallambda01**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

Code source [Info](#)

[Upload from](#)

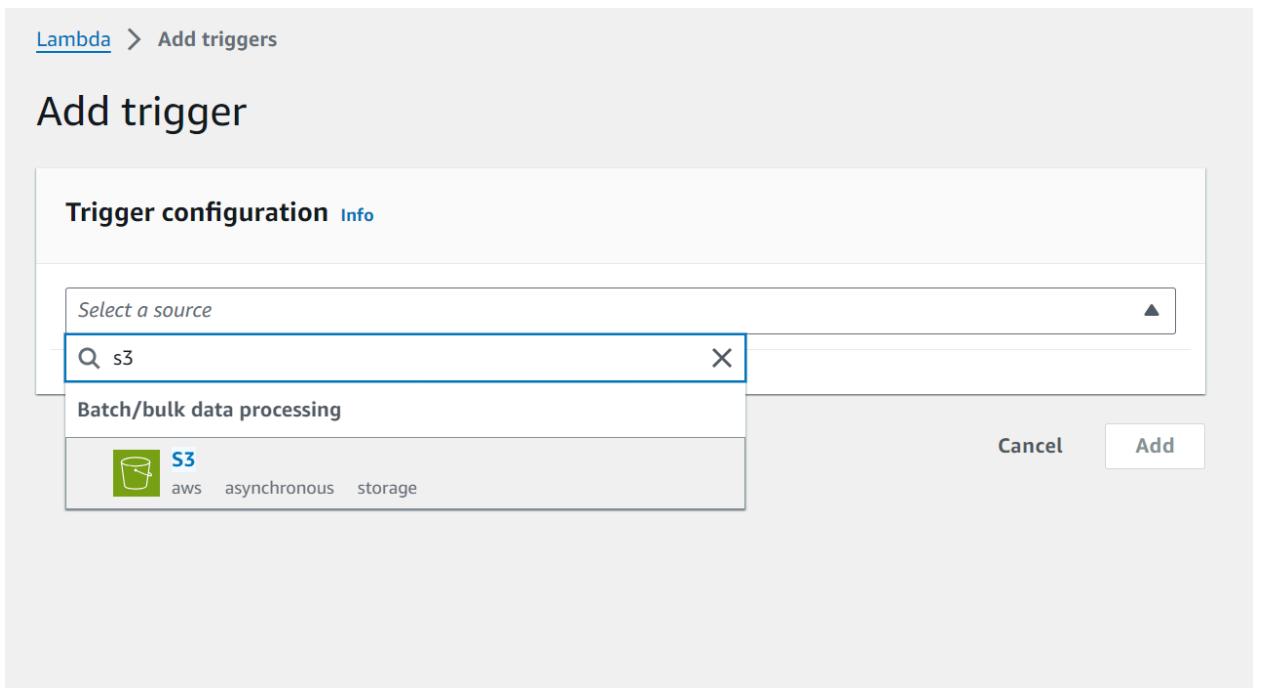
index.mjs

```

1 import json
2 import ClientError
3 import boto3 from botocore.exceptions
4
5 s3 = boto3.client('s3')
6
7 def lambda_handler(event, context):
8     bucket_name = 'your-bucket-name' # Replace with your S3 bucket name
9     object_key = 'your-file.txt' # Replace with your object (file) key
10
11     # List objects in the bucket
12     try:
13         response = s3.list_objects_v2(Bucket=bucket_name)
14         if 'Contents' in response:
15             print("Objects in S3 bucket:")
16             for obj in response['Contents']:
17                 print(obj['Key'])
18         else:
19             print("No objects found.")

```

4. Add a trigger to the Lambda function so any changes in the S3 bucket will be first visible to the user.



[Lambda](#) > [Add triggers](#)

Add trigger

Trigger configuration [Info](#)

Select a source

s3

Batch/bulk data processing

S3

aws asynchronous storage

Cancel Add

 S3
aws asynchronous storage

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

Bucket region: ap-south-1

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

All object create events X

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

Suffix - optional
Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters. Any [special characters](#) must be URL encoded.

5. In the event notification of the S3 bucket we can see that it has been connected to the Lambda function .

Event notifications (1)					Edit	Delete	Create event notification
Name	Event types	Filters	Destination type	Destination			
a9e8e939-							
9989-4f83-	All object create events	-	Lambda function	prajwallambda01			
804c-							
142405582ad4							

6. Upload a photo to the S3 bucket

The screenshot shows the AWS S3 'Upload' interface. At the top, it says 'Upload' and 'Info'. Below that, a message says 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)'.

A large dashed box area is labeled 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.'

Below this, a table titled 'Files and folders (1 Total, 964.6 KB)' shows one item: 'Screenshot 2023-12-21 194914.p...' which is an image/png file.

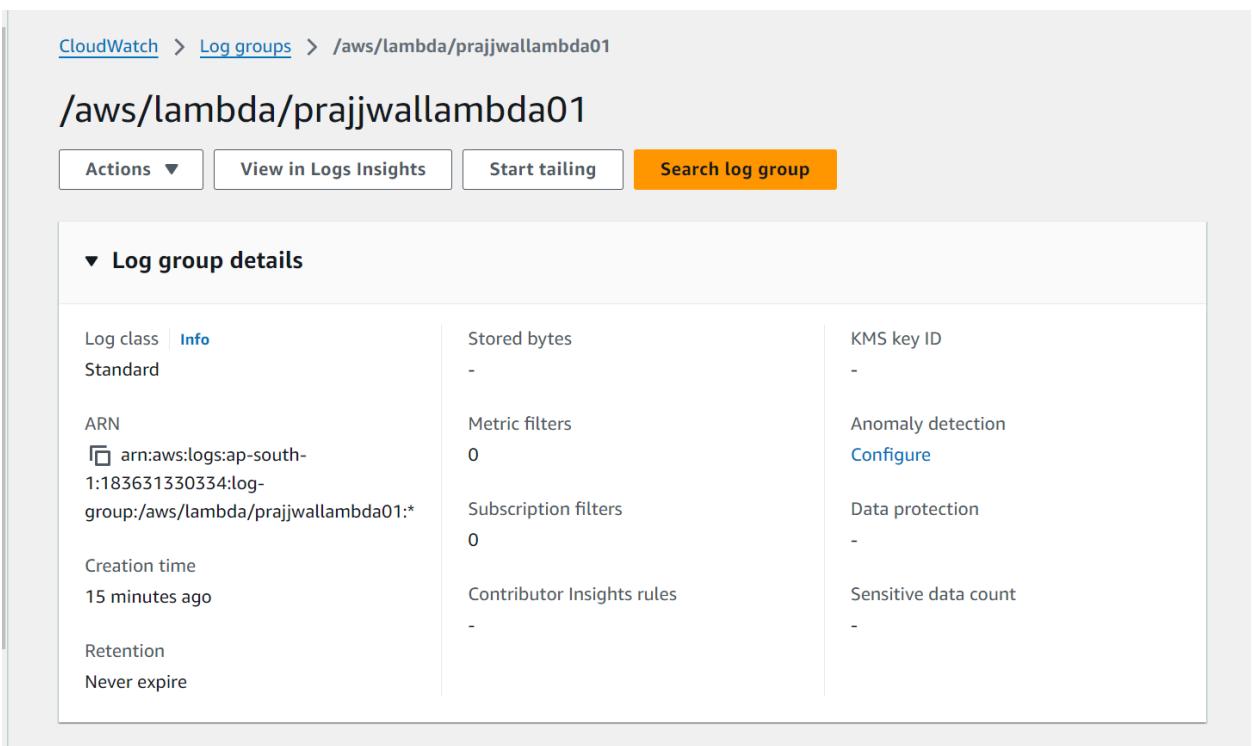
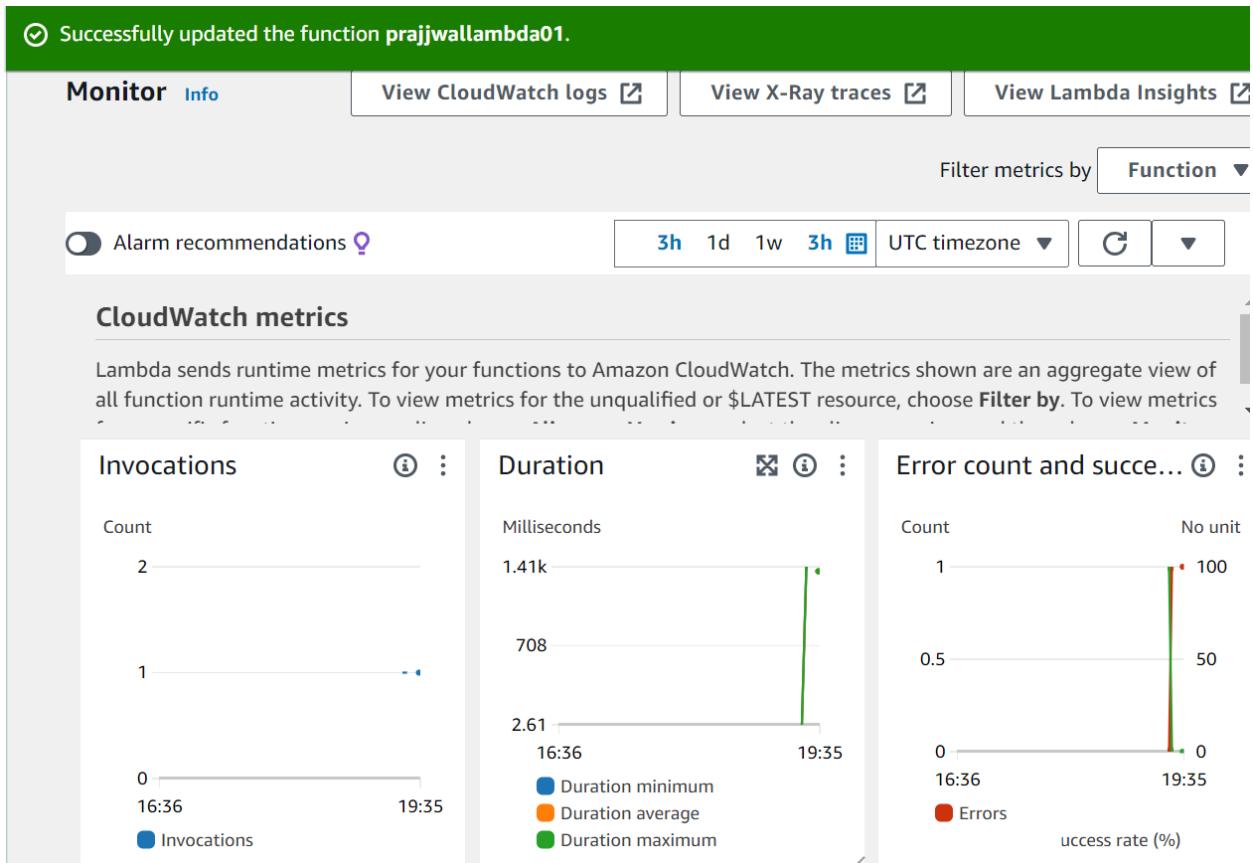
Buttons for 'Remove', 'Add files', and 'Add folder' are available.

The 'Destination' section shows 's3://prajjbucket'.

At the bottom, a green bar indicates 'Upload succeeded' with 'View details below.' Below this, a summary table shows 'Destination s3://prajjbucket' with 'Succeeded' (1 file, 964.6 KB (100.00%)) and 'Failed' (0 files, 0 B (0%)).

The 'Files and folders' tab is selected, showing the same table as above: 'Screenshot 2023-12-21 194914.p...' (image/png, 964.6 KB, Status: Succeeded).

7. Now run the function and in the cloud watch logs of AWS you can see the message printed and all the other details of the working of the Lambda function.



Log events		Actions ▾	Start tailing	Create metric filter		
Filter events - press enter to search		1m	1h	UTC timezone ▾	Display ▾	⚙️
▶	Timestamp	Message				
No older events at this moment. Retry						
▶	2024-10-07T19:36:29.642Z	INIT_START Runtime Version: python:3.12.v36 Runtime Version ARN: arn:aws:lambda:ap-south...				
▶	2024-10-07T19:36:29.721Z	[ERROR] Runtime.ImportModuleError: Unable to import module 'index': No module named 'ind...				
▶	2024-10-07T19:36:29.780Z	INIT_REPORT Init Duration: 138.91 ms Phase: init Status: error Error Type: Runtime.Impor...				
▶	2024-10-07T19:36:30.489Z	[ERROR] Runtime.ImportModuleError: Unable to import module 'index': No module named 'ind...				
▶	2024-10-07T19:36:31.208Z	INIT_REPORT Init Duration: 1417.13 ms Phase: invoke Status: error Error Type: Runtime.Im...				
▶	2024-10-07T19:36:31.208Z	START RequestId: ea6fa0b4-63e5-4ac9-a947-9fd8c90a96b2 Version: \$LATEST				
▶	2024-10-07T19:36:31.239Z	END RequestId: ea6fa0b4-63e5-4ac9-a947-9fd8c90a96b2				
▶	2024-10-07T19:36:31.239Z	REPORT RequestId: ea6fa0b4-63e5-4ac9-a947-9fd8c90a96b2 Duration: 1447.34 ms Billed Durat...				