

# Experiment 01

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

The screenshot shows the AWS EC2 Dashboard in the US East (N. Virginia) Region. The left sidebar includes links for EC2 Global View, Events, Console-to-Code, Instances, Images, and Elastic Block Store. The main area displays EC2 resources: 1 instance (running), 1 Auto Scaling Group, 0 Dedicated Hosts, 1 Elastic IP, 0 Key pairs, 0 Load balancers, 2 Security groups, 1 instance, 0 instances, 0 placement groups, 0 snapshots, 0 volumes, and 1 volume.

In the center, there's a "Launch instance" section with a "Launch instance" button and a "Migrate a server" option. Below it, a note says "Note: Your instances will launch in the US East (N. Virginia) Region". To the right, the "Service health" section shows "AWS Health Dashboard" and indicates "This service is operating normally".

The right sidebar contains sections for "EC2 Free Tier", "Account attributes", "Default VPC" (vpc-0fc2f5f7a22f1a71), "Settings" (Data protection and security, Zones, EC2 Serial Console, Default credit specification, EC2 console preferences), and "Additional information".

At the bottom, a detailed view of an AMI selection is shown. It lists "Amazon Linux", "macOS", "Ubuntu", "Windows", "Red Hat", and "SUSE Linux Enterprise Server". The "Ubuntu" AMI is selected. The "Amazon Machine Image (AMI)" section shows "Ubuntu Server 24.04 LTS (HVM), SSD Volume Type" with an AMI ID of "ami-04a81a99f5ec58529". It notes "Free tier eligible" and provides details about the instance type (t2.micro), security group (New security group), and storage (1 volume(s) - 8 GiB). A tooltip explains the "Free tier" includes 750 hours of t2.micro or t3.micro usage in the regions where the instance is launched. The "Launch instance" button is prominently displayed at the bottom right.

**Configure storage**

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 24.04 LTS, ami-04a81a99fsec58529

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

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

**Advanced details**

**Launch instance**

**Success**  
Successfully initiated launch of instance (i-0f3bd5bcbcc35b81a)

**Next Steps**

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

**Instances (1/2)**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
Mynewapp-env	i-0365889e6b3d22cdb	Running	t3.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-44-
<b>vedant_Instance</b>	<b>i-0f3bd5bcbcc35b81a</b>	<b>Running</b>	<b>t2.micro</b>	<b>0/2 checks passed</b>	<b>View alarms +</b>	<b>us-east-1b</b>	<b>ec2-54-</b>

**i-0f3bd5bcbcc35b81a (vedant\_Instance)**

**Details**

Instance ID: i-0f3bd5bcbcc35b81a (vedant\_Instance)

Public IPv4 address: 54.159.13.162 | [open address](#)

Private IP4 address: 172.31.33.103

Public IPv4 DNS: ec2-54-159-13-162.compute-1.amazonaws.com | [open address](#)

AWS Services Search [Alt+S]

EC2 > Instances > i-0365889e6b3d22cdb > Connect to instance

## Connect to instance info

Connect to your instance i-0365889e6b3d22cdb (Mynewapp-env) using any of these options

EC2 Instance Connect Session Manager SSH client EC2 serial console

**⚠ Port 22 (SSH) is not authorized**  
Port 22 (SSH) is currently not authorized by your security group. To use EC2 Instance Connect, you must authorize port 22 for the EC2 Instance Connect service IP addresses in your Region: 18.206.107.24/29.  
[Learn more.](#)

Instance ID [i-0365889e6b3d22cdb \(Mynewapp-env\)](#)

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 [44.205.155.152](#)

Username Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, root.

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

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

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

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

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

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

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

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

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

```
ubuntu@ip-172-31-6-34:~$ sudo apt update && sudo apt upgrade -y
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:13 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [317 kB]
Get:14 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [82.7 kB]
Get:15 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [5640 B]
Get:16 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [318 kB]
Get:17 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [133 kB]
Get:18 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [45.0 kB]
Get:19 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [12.5 kB]
Get:20 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [288 kB]
Get:21 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [40.7 kB]
Get:22 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 c-n-f Metadata [416 kB]
Get:23 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [14.1 kB]
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [3608 B]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [212 B]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [532 B]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 B]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [112 B]
Get:29 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [18.3 kB]
Get:30 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [10.5 kB]
Get:31 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [17.6 kB]
Get:32 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1016 B]
Get:33 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]
Get:34 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]
```

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-6-34:~$ sudo apt install apache2 -y
Reading package lists... Done
Building dependency tree...
Enabling module auth_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /usr/lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /usr/lib/systemd/system/apache-htcacheclean.service.
Processing triggers for ufw (0.36.2-6) ...
Processing triggers for man-db (2.12.8-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.2) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Pending kernel upgrade!
```

```

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

```



## DevSync

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

Welcome to DevSync

### Introduction



### Promotional Video



Vedant Sanap  
D15A 48

## ADVANCE DEVOPS EXP-2

Aim: To build your application using AWS Codebuild and deploy on S3 using AWS CodePipeline  
deploy sample application on EC2 instance using AWS codedeploy. Code and Output : Using

### Elastic Beanstalk:

The screenshot shows the AWS Elastic Beanstalk console interface for creating a new environment. It consists of three main sections:

- Environment information**:
  - Environment name: Vedantapp-env
  - Domain: .us-east-1.elasticbeanstalk.com
  - Check availability button
- Platform**:
  - Platform type: Managed platform (selected)
  - Custom platform option
  - Platform dropdown: PHP
  - Platform branch dropdown: PHP 8.3 running on 64bit Amazon Linux 2023
  - Platform version dropdown: 4.3.2 (Recommended)
- Application code**:
  - Sample application option

us-east-1.console.aws.amazon.com/elasticbeanstalk/home?region=us-east-1#/create-environment

Services Search [Alt+S] 80% - +

Step 1  
[Configure environment](#)

Step 2  
**Configure service access**

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

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

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

Step 6  
[Review](#)

### Configure service access Info

**Service access**  
IAM roles, assumed by Elastic Beanstalk as a service role, and EC2 instance profiles allow Elastic Beanstalk to create and manage your environment. Both the IAM role and instance profile must be attached to IAM managed policies that contain the required permissions. [Learn more](#)

**Service role**  
 Create and use new service role  
 Use an existing service role

**Existing service roles**  
Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.

aws-elasticbeanstalk-service-role [▼](#) [C](#)

**EC2 key pair**  
Select an EC2 key pair to securely log in to your EC2 instances. [Learn more](#)

newKey [▼](#) [C](#)

**EC2 instance profile**  
Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.

aws-elasticbeanstalk-ec2-role [▼](#) [C](#)

[View permission details](#)

[Cancel](#) [Skip to review](#) [Previous](#) **Next**

**Lifecycle**

false	Log streaming	Allow URL fopen
	Deactivated	On

**Display errors**

Off	Document root	Max execution time
	-	60

**Memory limit**

256M	Zlib output compression	Proxy server
	Off	nginx

**Logs retention**

7	Rotate logs	Update level
	Deactivated	minor

**X-Ray enabled**

Deactivated
-------------

**Environment properties**

Key	Value
No environment properties	
There are no environment properties defined	

[Cancel](#) [Previous](#) **Submit**

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

The screenshot displays two main sections of the AWS console.

**Elastic Beanstalk Environment Dashboard:**

- Left Sidebar:** Shows navigation links for Applications, Environments, Change history, Application: vedantapp (with sub-links for Application versions, Saved configurations), Environment: Vedantapp-env (with sub-links for Go to environment, Configuration, Events, Health, Logs, Monitoring, Alarms, Managed updates, Tags), and Recent environments.
- Central Content:** A banner states "Elastic Beanstalk is launching your environment. This will take a few minutes." Below it, the "Vedantapp-env" environment is shown in a card format. The "Environment overview" section includes fields for Health (Unknown), Environment ID (e-anz36tmjjp), Domain (-), Application name (vedantapp), and Platform (PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2). The "Events" tab shows two INFO-level events from August 20, 2024, detailing environment setup and creation.

**Search Results for 'codeArtifact':**

- Search Bar:** Shows the query "codeArtifact".
- Search Results:** A search bar at the top right says "Search results for 'code'". The results are categorized into Services and Features.
  - Services:** Includes Amazon Q Developer (Including Amazon CodeWhisperer), CodeCommit, CodePipeline, and AWS Signer.
  - Features:** Includes Full repository analysis and Pull request code review.
- Footer:** Shows copyright information: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

[Alt+S]

Developer Tools > CodePipeline > Pipelines

Introducing the new V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model. Learn more

Pipelines

Name	Latest execution status	Latest source revisions	Latest execution started	Most recent executions
newpipeline (Type: V2   Execution mode: QUEUED)	Failed	Source - fb61b094  Update index.html	6 days ago	

AWS Services Search [Alt+S]

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose pipeline settings   
Step 1 of 5

**Pipeline settings**

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

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

Execution mode  
Choose the execution mode for your pipeline. This determines how the pipeline is run.  
 Superseded  
A more recent execution can overtake an older one. This is the default.  
 Queued (Pipeline type V2 required)  
Executions are processed one by one in the order that they are queued.  
 Parallel (Pipeline type V2 required)  
Executions don't wait for other runs to complete before starting or finishing.

Service role  
 New service role [Create a service role in your account](#)   
 Existing service role [Choose an existing service role from your account](#)

us-east-1.console.aws.amazon.com/codesuite/settings/connections/create/github... ?

N. Virginia VedantSanap

Developer Tools > Connections > Create connection

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

## Connect to GitHub

**GitHub connection settings** [Info](#)

Connection name

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

or [Install a new app](#)

▶ Tags - *optional*

[Connect](#)

CloudShell Feedback Privacy Terms Cookie preferences © 2024, Amazon Web Services, Inc. or its affiliates. Default branch Default branch will be used only when pipeline execution starts from a different source or ma

AWS Services Search [Alt+S] N. Virginia VedantSampath

Step 2 Add source stage Step 3 Add build stage Step 4 Add deploy stage Step 5 Review

### Source

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

GitHub (Version 2)

New GitHub version 2 (app-based) action To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. Learn more

Connection Choose an existing connection that you have already configured, or create a new one and then return to this task.

arn:aws:codeconnections:us-east-1:314146309670:connection/7e91c9d2-bb X or Connect to GitHub

Ready to connect Your GitHub connection is ready for use.

Repository name Choose a repository in your GitHub account.

Morphious0110/aws-codepipeline-s3-codedeploy-linux-2.0 X You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

Default branch Default branch will be used only when pipeline execution starts from a different source or manually started.

master X

### Output artifact format

Choose the output artifact format.

CodePipeline default AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.

Full clone AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

### Trigger

Trigger type Choose the trigger type that starts your pipeline.

No filter Starts your pipeline on any push and clones the HEAD.

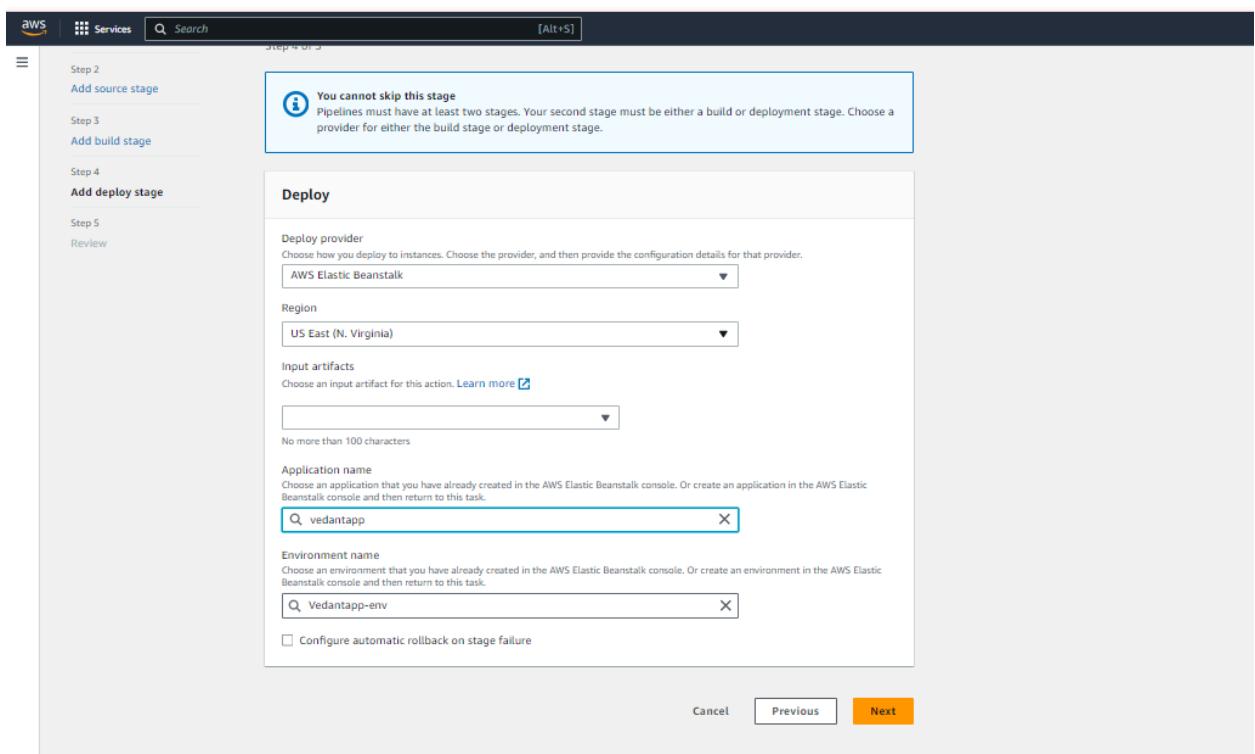
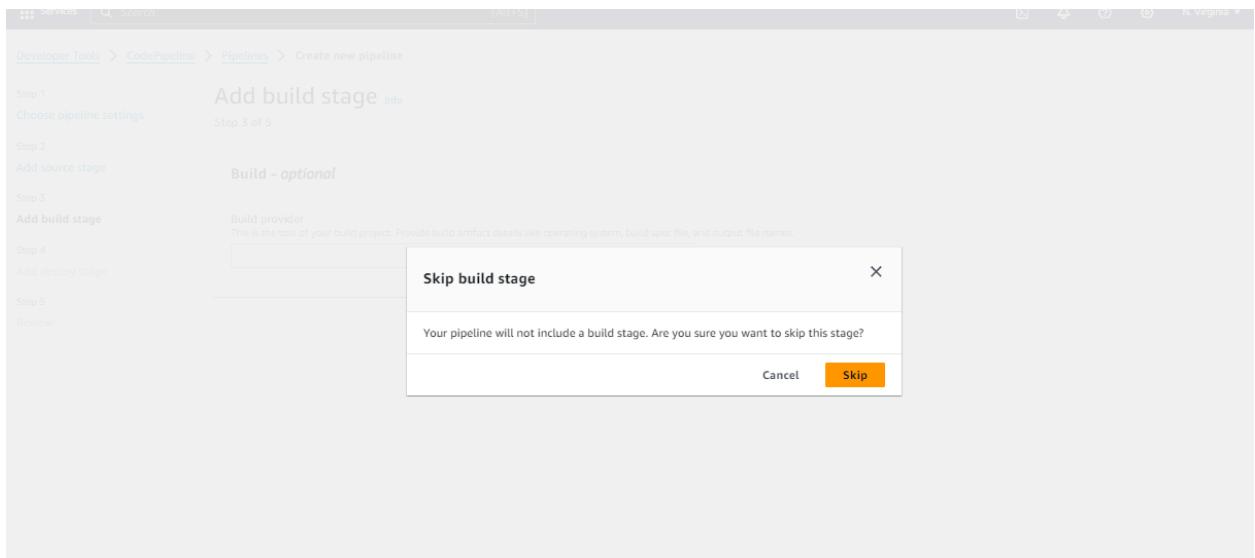
Specify filter Starts your pipeline on a specific filter and clones the exact commit. Pipeline type V2 is required.

Do not detect changes Don't automatically trigger the pipeline.

You can add additional sources and triggers by editing the pipeline after it is created.

Cancel Previous Next

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



us-east-1.console.aws.amazon.com/codesuite/codepipeline/pipeline/new?region=us-east-1

**Trigger configuration**  
You can add additional pipeline triggers after the pipeline is created.

Trigger type  
No filter

**Step 3: Add build stage**

Build action provider

Build stage  
No build

**Step 4: Add deploy stage**

Deploy action provider

Deploy action provider  
AWS Elastic Beanstalk

ApplicationName  
vedantapp

EnvironmentName  
Vedantapp-env

Configure automatic rollback on stage failure  
Disabled

**Create pipeline**

Success  
Congratulations! The pipeline vedant-pipeline has been created.

CodePipeline

- Source • CodeCommit
- Artifacts • CodeArtifact
- Build • CodeBuild
- Deploy • CodeDeploy
- Pipeline • CodePipeline
- Getting started
- Pipelines
- Settings
- Feedback

vedant-pipeline

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded Pipeline execution ID: d20db2c4-228e-4e0f-9e00-497633d23011

GitHub (Version 2) Succeeded - 1 minute ago  
#615094 View details

Disable transition

Deploy Succeeded Pipeline execution ID: d20db2c4-228e-4e0f-9e00-497633d23011

AWS Elastic Beanstalk Succeeded - Just now

Start rollback

CloudShell Feedback

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

Environments (2) <small>Info</small>								
Environment name		Health	Application name	Platform	Domain	Running versions	Tier name	Date created
<a href="#">Mynewapp-env (terminated)</a>	<small>Unknown</small>	<small>MyNewApp</small>	PHP 8.3 running on 64bit Amazon Linux 2 v2.14.0	<a href="#">Mynewapp-env.eba-pfz7h5nw...</a>	-	-	WebServer	August 13, 2024 14:08...
<a href="#">Vedantapp-env</a>	<small>Ok</small>	<small>vedantapp</small>	PHP 8.3 running on 64bit Amazon Linux 2 v2.14.0	<a href="#">Vedantapp-env.eba-5xm9gupn...</a>	code-pipeline-172413...	code-pipeline-172413...	WebServer	August 20, 2024 12:56...

Not secure vedantapp-env.eba-5xm9gupn.us-east-1.elasticbeanstalk.com

# Congratulations!

You have successfully created a pipeline that retrieved this source application from an Amazon S3 bucket and deployed it to three Amazon EC2 instances using AWS CodeDeploy.

Updated CHanges

## S3 Bucket :

### General configuration

AWS Region  
US West (Oregon) us-west-2

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

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

**Files and folders (1 Total, 505.0 B)**

All files and folders in this table will be uploaded.

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

**Destination** [Info](#)

Destination  
<s3://vedantawsbucke>

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

[Feedback](#) [© 2024, Amazon Web Services](#)

## Edit Block public access (bucket settings) [Info](#)

### Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#) 

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

- Block public access to buckets and objects granted through *new* access control lists (ACLs)**  
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through *any* access control lists (ACLs)**  
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through *new* public bucket or access point policies**  
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**  
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

## Object Ownership

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

**ACLs disabled (recommended)**

All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

**ACLs enabled**

Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

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



**Enabling ACLs turns off the bucket owner enforced setting for Object Ownership**

Once the bucket owner enforced setting is turned off, access control lists (ACLs) and their associated permissions are restored. Access to objects that you do not own will be based on ACLs and not the bucket policy.

I acknowledge that ACLs will be restored.

## Object Ownership

**Upload succeeded**

View details below.

### Upload: status

[Close](#)

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

#### Summary

Destination  
s3://vedantawsbucke

Succeeded

1 file, 505.0 B (100.00%)

Failed

0 files, 0 B (0%)

[Files and folders](#)

[Configuration](#)

#### Files and folders (1 Total, 505.0 B)

## 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

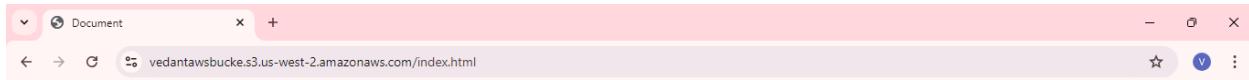
- Host a static website

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

- Redirect requests for an object

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

**i** 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](#)



## Hello World

Hello World Lorem ipsum dolor sit amet consectetur, adipisicing elit. Rem voluptatibus sint ipsam, iure eligendi velit laboriosam vitae nisi facilis ipsa recusandae nulla quia assumenda rerum quos, exercitationem doloribus consectetur voluptate.

EC2 :



### Experiment No 3

Vedant Sanap  
D15A 48  
Batch C

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

#### 1) Launch 2 EC2 instance and select Ubuntu in AMI

The screenshot shows the AWS Lambda console with the 'Create new key pair' dialog open. The dialog has fields for 'Key pair name' (containing 'vedantsanap') and 'Key pair type' (with 'RSA' selected). It also includes sections for 'Private key file format' (with '.pem' selected) and a warning message about storing the private key securely. At the bottom are 'Cancel' and 'Create key pair' buttons.

#### 2) Create new key pair

The screenshot shows the AWS Lambda console with the 'Create new key pair' dialog open. The dialog has fields for 'Key pair name' (containing 'vedantsanap') and 'Key pair type' (with 'RSA' selected). It also includes sections for 'Private key file format' (with '.pem' selected) and a warning message about storing the private key securely. At the bottom are 'Cancel' and 'Create key pair' buttons.

- 3) In Security group select all checkbox and launch instance**
- 4) Go to security group and edit inbound rules of both instance**
- 5) Delete all the rules and add new rule with All traffic and Anywhere-IPv4**

Inbound rules (1/1)					
<input type="button" value="Filter security group rules"/> <input type="button" value="Manage tags"/> <input type="button" value="Edit inbound rules"/>					
Name	Security group rule...	IP version	Type	Protocol	Port range
<input checked="" type="checkbox"/> -	sgr-07c7275a5903c511f	IPv4	All traffic	All	All

- 6) Now in running instances click on master instance and click on connect**

EC2 Instance Connect    Session Manager    SSH client    EC2 serial console

Instance ID  
 i-0d7f35ca2039c23b1 (Master)

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  
 3.110.175.160

User name  
Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ubuntu.

**Note:** In most cases, the default user name, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

- 8) Similarly connect the worker**
- 9) Set hostname to master and worker respectively**

```
ubuntu@ip-172-31-12-130:~$ sudo hostnamectl set-hostname master
ubuntu@ip-172-31-12-130:~$
```

```
ubuntu@master:~$
```

```
ubuntu@worker:~$
```

## 10) Use command sudo apt-get update on both master and worker CLI

sudo apt-get update on both

```
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [10.5 kB]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [1]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [24.3 kB]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.4 kB]
Get:29 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [644 B]
Get:30 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [1]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [765 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [165 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [11.3 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [826 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [133 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [536 B]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [781 kB]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [143 kB]
Get:39 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.7 kB]
Get:40 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [36.5 kB]
Get:41 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7060 B]
Get:42 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Fetched 27.1 MB in 5s (5664 kB/s)
Reading package lists... Done
ubuntu@master:~$ 
```

## 11) Installing Docker on both CLI

sudo apt-get install docker.io on both

```
Done.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@master:~$ 
```

## 12) Enable Docker on both CLI and check its status

sudo systemctl enable docker

sudo systemctl status docker on both

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@master:~$ sudo systemctl enable docker
ubuntu@master:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2023-09-17 17:23:49 UTC; 3min 30s ago
TriggeredBy: ● docker.socket
   Docs: https://docs.docker.com
 Main PID: 3050 (dockerd)
    Tasks: 7
   Memory: 33.2M
      CPU: 291ms
     CGroup: /system.slice/docker.service
             └─3050 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

ubuntu@master:~$ 
```

**DOCKER INSTALLED SUCCESSFULLY**

### 13) Now for Installing Kubernetes (On both CLI)

```
sudo apt-get update
```

```
ubuntu@master:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
ubuntu@master:~$ 
```

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

```
Updating certificates in /etc/ssl/certs...
rehash: warning: skipping ca-certificates.crt, it does not contain exactly one certificate or CRL
19 added, 6 removed; done.
Setting up libcurl4:amd64 (7.81.0-1ubuntu1.13) ...
Setting up curl (7.81.0-1ubuntu1.13) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Processing triggers for ca-certificates (20230311ubuntu0.22.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@master:~$ 
```

### 14) Download Google cloud public signing key

```
sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg https://dl.k8s.io/apt/doc/apt-key.gpg
```

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@master:~$ sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg https://dl.k8s.io/apt/doc/apt-key.gpg
ubuntu@master:~$ 
```

### 15) Adding kubernetes apt repository

```
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@master:~$ 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
deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-xenial main
ubuntu@master:~$ 
```

### 16) Run this 3 commands

```
sudo apt-get update
```

```
ubuntu@master:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:5 https://packages.cloud.google.com/apt kubernetes-xenial InRelease [8993 B]
Get:6 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 Packages [69.9 kB]
Fetched 78.9 kB in 1s (53.8 kB/s)
Reading package lists... Done
ubuntu@master:~$ 
```

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

```
Setting up conntrack (1:1.4.6-2build2) ...
Setting up kubectl (1.28.2-00) ...
Setting up ebtables (2.0.11-4build2) ...
Setting up socat (1.7.4.1-3ubuntu4) ...
Setting up cri-tools (1.26.0-00) ...
Setting up kubernetes-cni (1.2.0-00) ...
Setting up kubelet (1.28.2-00) ...
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /lib/systemd/system/kubelet.service.
Setting up kubeadm (1.28.2-00) ...
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@master:~$ 
```

sudo apt-mark hold kubelet kubeadm kubectl

```
ubuntu@master:~$ sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
ubuntu@master:~$ 
```

## KUBERNETES INSTALLED SUCCESSFULLY

### 17) Kubernetes Deployment

sudo swapoff -a

```
ubuntu@master:~$ sudo swapoff -a
ubuntu@master:~$ 
```

### 18) Initialize kubernetes on Master

sudo touch "/etc/docker/daemon.json"

sudo nano "/etc/docker/daemon.json" Run this command and copy paste this

```
{
"exec-opts": ["native.cgroupdriver=systemd"]
}
```

Then press ctrl + O and enter then ctrl + X

sudo cat "/etc/docker/daemon.json"

sudo systemctl daemon-reload

sudo systemctl restart docker

sudo systemctl restart kubelet

sudo kubeadm reset

```
ubuntu@master:~$ sudo touch "/etc/docker/daemon.json"
ubuntu@master:~$ sudo nano "/etc/docker/daemon.json" 
```

```
ubuntu@master:~$ sudo cat "/etc/docker/daemon.json"
{
    "exec-opts": ["native.cgroupdriver=systemd"],
    "log-driver": "json-file",
    "log-opts": {
        "max-size": "100m"
    },
    "storage-driver": "overlay2"
}
ubuntu@master:~$ 
```

```
ubuntu@master:~$ sudo systemctl daemon-reload
ubuntu@master:~$ sudo systemctl restart docker
ubuntu@master:~$ sudo systemctl restart kubelet
ubuntu@master:~$ sudo kubeadm reset
W0917 18:15:57.371540 10839 preflight.go:56] [reset] WARNING: 
[preflight] Are you sure you want to proceed? [y/N]: y
[preflight] Running pre-flight checks
W0917 18:16:01.329044 10839 removeetcdmember.go:106] [reset] No
[preflight] Deleted contents of the etcd data directory: /var/lib/etcd
[preflight] Stopping the kubelet service
[preflight] Unmounting mounted directories in "/var/lib/kubelet"
```

```
$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16 --ignore-preflight-errors=all
ubuntu@master:~$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16 --ignore-preflight-er
[init] Using Kubernetes version: v1.28.2
[preflight] Running pre-flight checks
    [WARNING NumCPU]: the number of available CPUs 1 is less than the required 2
    [WARNING Mem]: the system RAM (965 MB) is less than the minimum 1700 MB
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your internet conn
[preflight] You can also perform this action in beforehand using 'kubeadm config images p
W0917 18:17:31.346348 10860 checks.go:835] detected that the sandbox image "registry.k8
    It is recommended that using "registry.k8s.io/pause:3.9" as the CRI sandbox image.
[certs] Using certificateDir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [kubernetes kubernetes.default kub
    172.31.12.130]
[certs] Generating "apiserver-kubelet-client" certificate and key
[certs] Generating "front-proxy-ca" certificate and key
[certs] Generating "front-proxy-client" certificate and key
[certs] Generating "etcd/ca" certificate and key
[certs] Generating "etcd/server" certificate and key
[certs] etcd/server serving cert is signed for DNS names [localhost master] and IPs [172.
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [localhost master] and IPs [172.31
[certs] Generating "etcd/healthcheck-client" certificate and key
[certs] Generating "apiserver-etcd-client" certificate and key
[certs] Generating "sa" key and public key
kubeadm join 172.31.12.130:6443 --token 67nba2.98zekjx1ogwtrr29 --discovery-token-ca-cert-hash
sha256:5d3403f5221016f77cbed1757a266467af45dc41b765ebe535f15ee058baf883
```

```
Your Kubernetes control-plane has initialized successfully!
```

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

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

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

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

```
You should now deploy a pod network to the cluster.
```

```
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
```

```
https://kubernetes.io/docs/concepts/cluster-administration/addons/
```

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

```
kubeadm join 172.31.12.130:6443 --token 67nba2.98zekjxlogwtrr29 \  
  --discovery-token-ca-cert-hash sha256:5d3403f5221016f77cb1757a266467af45dc41b765ebe535f15ee058baf883  
ubuntu@master:~$ []  
ubuntu@master:~$ mkdir -p $HOME/.kube  
ubuntu@master:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config  
ubuntu@master:~$ sudo chown $(id -u):$(id -g) $HOME/.kube/config  
ubuntu@master:~$ []
```

```
ubuntu@master:~$ kubectl apply -f https://github.com/flannel-io/flannel/releases/latest/download/kube-flannel.yaml  
namespace/kube-flannel created  
serviceaccount/flannel created  
clusterrole.rbac.authorization.k8s.io/flannel created  
clusterrolebinding.rbac.authorization.k8s.io/flannel created  
configmap/kube-flannel-cfg created  
daemonset.apps/kube-flannel-ds created  
ubuntu@master:~$ []
```

```
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"  
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"  
[kubelet-start] Starting the kubelet  
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...  
I0917 18:31:31.967826    11348 kubelet.go:220] [kubelet-start] preserving the crisocket information for th  
I0917 18:31:31.968119    11348 patchnode.go:31] [patchnode] Uploading the CRI Socket information "unix:///  
as an annotation  
I0917 18:31:31.968407    11348 cert_rotation.go:137] Starting client certificate rotation controller
```

```
This node has joined the cluster:
```

```
* Certificate signing request was sent to apiserver and a response was received.  
* The Kubelet was informed of the new secure connection details.
```

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

```
ubuntu@worker:~$ []
```

```
ubuntu@master:~$ kubectl get pods --all-namespaces  
The connection to the server 172.31.12.130:6443 was refused - did you specify the right host or port?  
ubuntu@master:~$ []
```

```
ubuntu@master:~$ kubectl get nodes  
NAME      STATUS      ROLES      AGE      VERSION  
master    Ready       control-plane   24m     v1.28.2  
worker    Ready       <none>        11m     v1.28.2  
ubuntu@master:~$ []
```

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

```
Last login: Sun Sep 17 18:56:17 2023 from 13.233.177.5
ubuntu@master:~$ kubectl get pods --all-namespaces
NAMESPACE      NAME              READY   STATUS    RESTARTS   AGE
kube-flannel   kube-flannel-ds-f4469   1/1    Running   12 (80s ago)   36m
kube-flannel   kube-flannel-ds-nt47t   1/1    Running   13 (81s ago)   28m
kube-system    coredns-5dd5756b68-5kjv4   1/1    Running   8 (80s ago)   40m
kube-system    coredns-5dd5756b68-stpx6   1/1    Running   8 (80s ago)   40m
kube-system    etcd-master           1/1    Running   12 (3m6s ago)  41m
kube-system    kube-apiserver-master  1/1    Running   16 (80s ago)   41m
kube-system    kube-controller-manager 1/1    Running   15 (3m6s ago)  41m
kube-system    kube-proxy-97gjz       1/1    Running   18 (29s ago)   40m
kube-system    kube-proxy-mm6xl       1/1    Running   13 (81s ago)   28m
kube-system    kube-scheduler-master 1/1    Running   16 (2m49s ago)  41m
ubuntu@master:~$ kubectl get nodes
NAME        STATUS   ROLES     AGE   VERSION
master      Ready    control-plane   43m   v1.28.2
worker      Ready    <none>    30m   v1.28.2
ubuntu@master:~$
```

## Conclusion:

Thus we have understood the Kubernetes Cluster Architecture, installed and spun a Kubernetes Cluster on AWS Cloud Platform.

## Error:

The status of all namespaces was not showing running in the first try, but when I rebooted the master and worker instances it was successful.

## Experiment No 4

Vedant Sanap  
D15A 48  
Batch C

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

- Running the application on the cluster

```
kubectl create deployment nginx --image=nginx
```

```
Last login: Sun Sep 17 18:58:53 2023 from 13.233.177.4
ubuntu@master:~$ kubectl create deployment nginx --image=nginx
deployment.apps/nginx created
ubuntu@master:~$
```

- Verifying the deployment using command

```
kubectl get deployments
```

```
ubuntu@master:~$ kubectl get deployments
NAME      READY    UP-TO-DATE   AVAILABLE   AGE
nginx     1/1      1           1           47s
ubuntu@master:~$
```

- Run the following command to create a service named nginx that will expose the app publicly.

```
kubectl expose deploy nginx --port 80 --target-port 80 --type NodePort
```

```
ubuntu@master:~$ kubectl expose deploy nginx --port 80 --target-port 80 --type NodePort
service/nginx exposed
ubuntu@master:~$
```

- Run this command to see the summary of the service and ports exposed.

```
kubectl get services
```

```
ubuntu@master:~$ kubectl get services
NAME      TYPE        CLUSTER-IP   EXTERNAL-IP  PORT(S)          AGE
kubernetes  ClusterIP  10.96.0.1   <none>       443/TCP         4d14h
nginx      NodePort    10.103.96.233  <none>       80:30816/TCP   67s
ubuntu@master:~$
```

- Add the port which is displayed i.e 30816 (will differ for each device) in the inbound rules of the security group of the worker.

Inbound rules (2)						
	Name	Security group rule...	IP version	Type	Protocol	Port range
<input type="checkbox"/>	-	sgr-067c4ec19a6dc863c	IPv4	Custom TCP	TCP	30816
<input type="checkbox"/>	-	sgr-043a60f4b25fe2c26	IPv4	All traffic	All	

- We can verify that the nginx page is accessible on all nodes using curl command(Worker)

1. sudo su
2. curl worker:30816

```
Last login: Fri Sep 22 13:48:46 2023 from 13.233.177.4
ubuntu@worker:~$ sudo su
root@worker:/home/ubuntu# curl worker:30816
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
root@worker:/home/ubuntu# 
```

**Open a new tab in browser and paste the public IP address followed by :port number (30816 in my case)**



## Welcome to nginx!

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

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

*Thank you for using nginx.*

## Conclusion:

Thus, we have studied and implemented how to install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy.

## Errors:

I was facing an error because I forgot to make changes in the security group of worker node.

# ADVANCE DEVOPS EXP-5

Vedant Sanap

D15A/48

The screenshot shows a software download interface with two main sections: Windows and Linux.

**Windows** section:

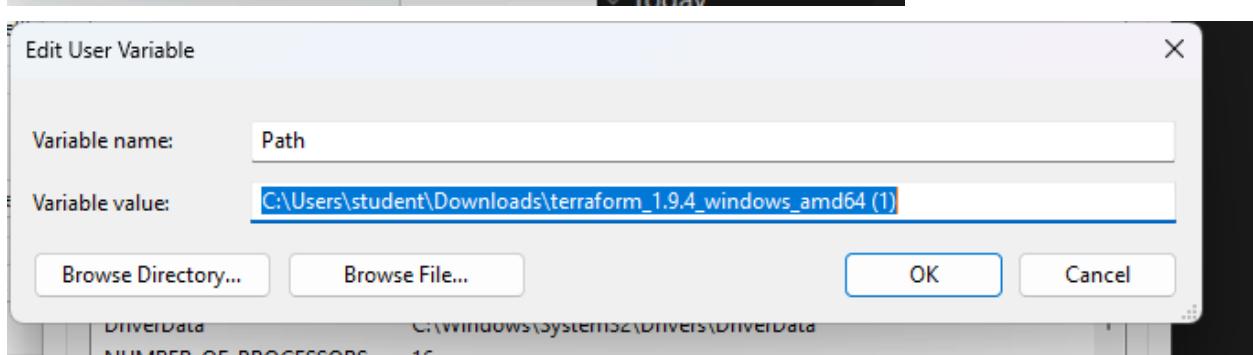
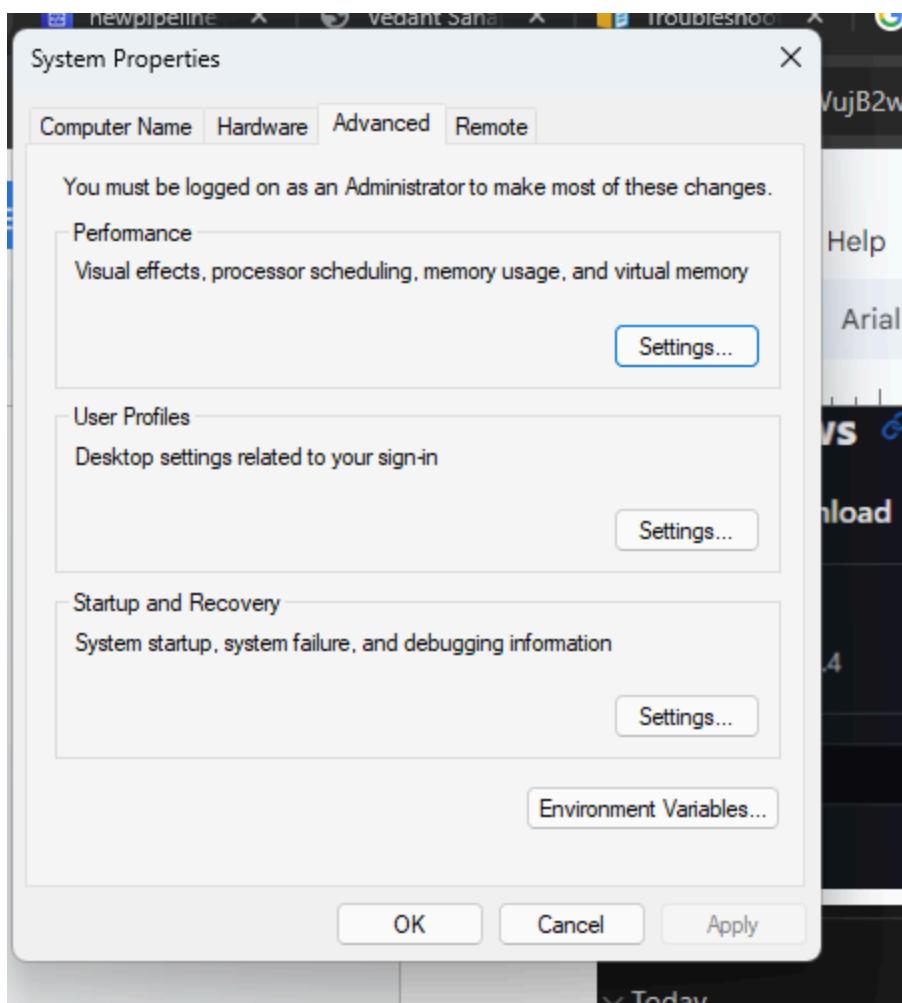
- Version: 1.9.4
- [Download](#)
- Version: 1.9.4
- [Download](#)

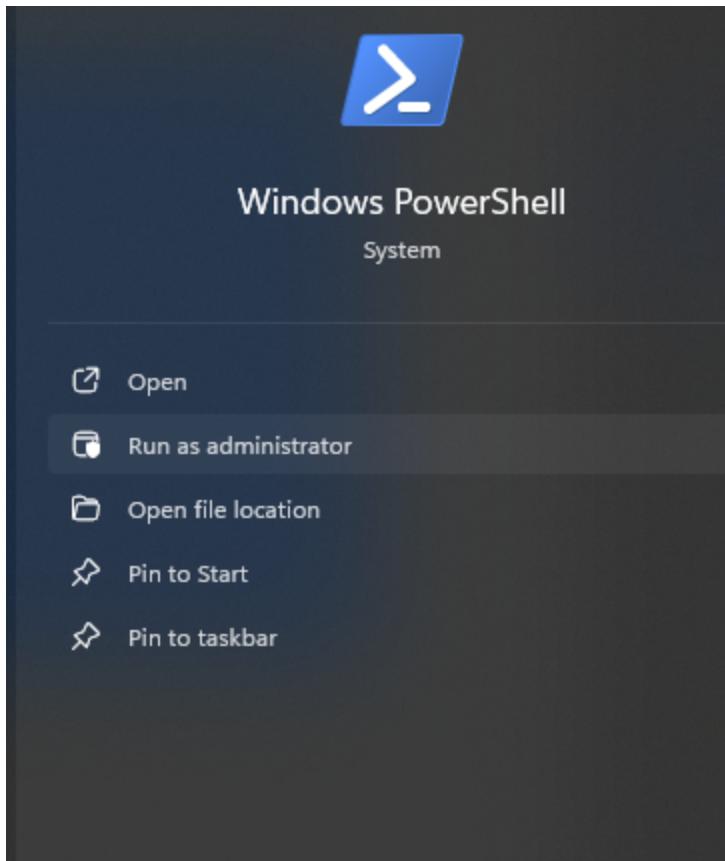
**Binary download**

- 386**: Version: 1.9.4, [Download](#)
- AMD64**: Version: 1.9.4, [Download](#)

**Linux** section:

Name	Date modified	Type	Size
/LICENSE	13-08-2024 14:00	Text Document	5 KB
terraform	13-08-2024 14:00	Application	88,918 KB





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

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

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

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

## Experiment 6

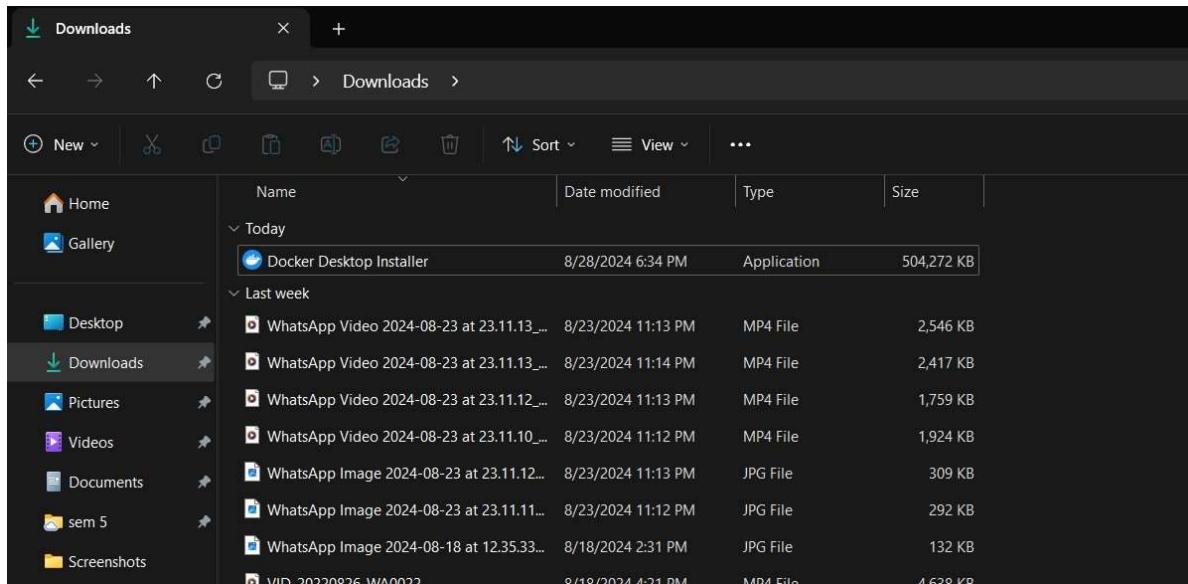
**Aim:**

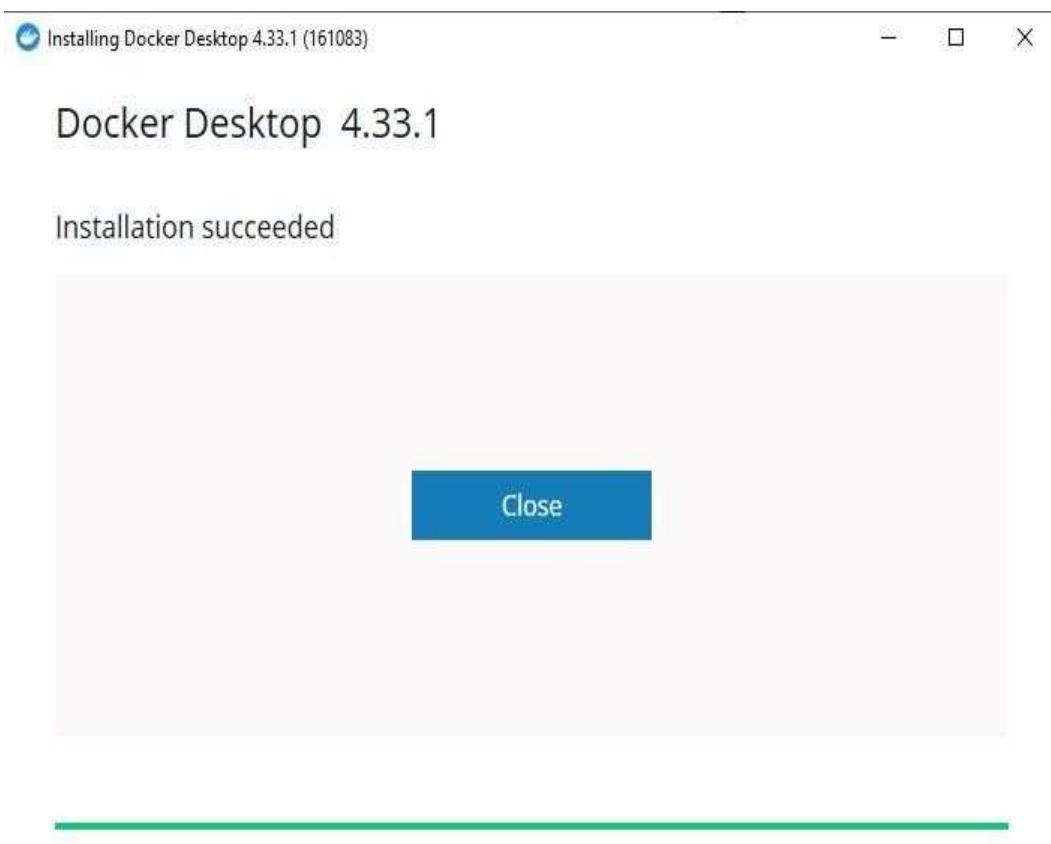
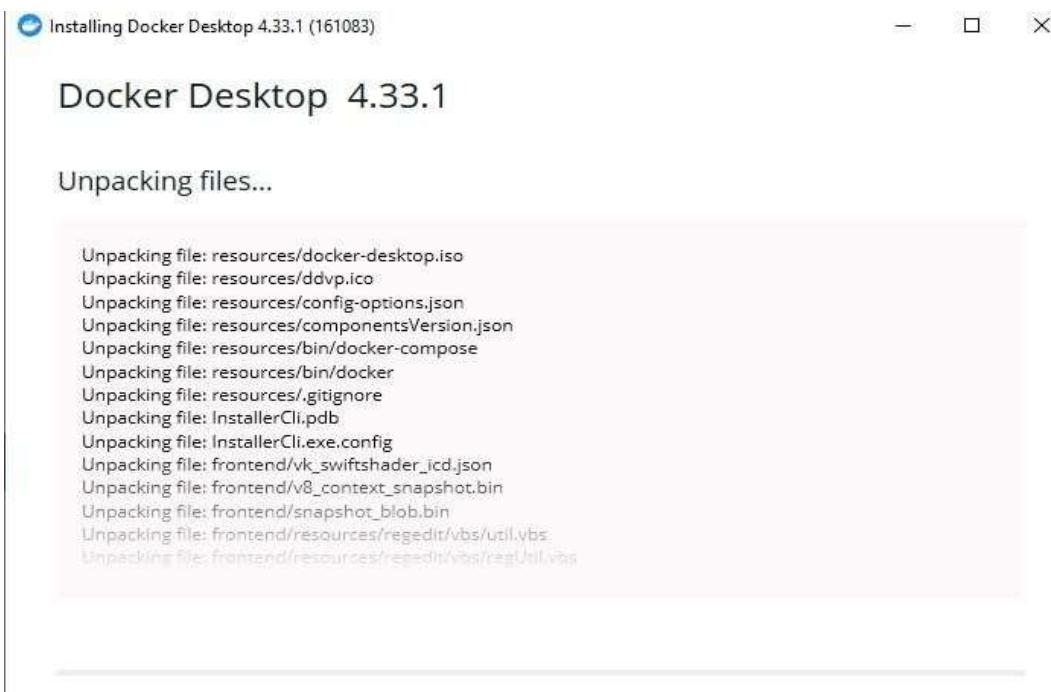
To Build, change, and destroy AWS / GCP /Microsoft Azure/ DigitalOcean infrastructure Using Terraform.(S3 bucket or Docker)

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

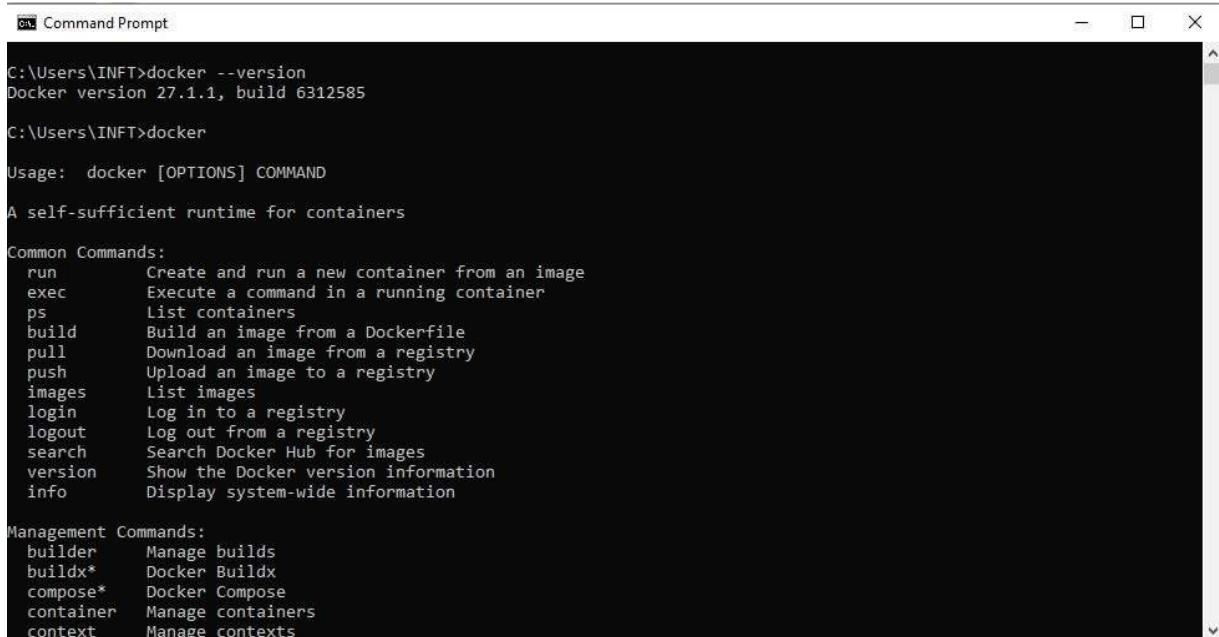


Step 2: The Docker is successfully downloaded. Now, run the docker installer and complete the installation.





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



```
Command Prompt

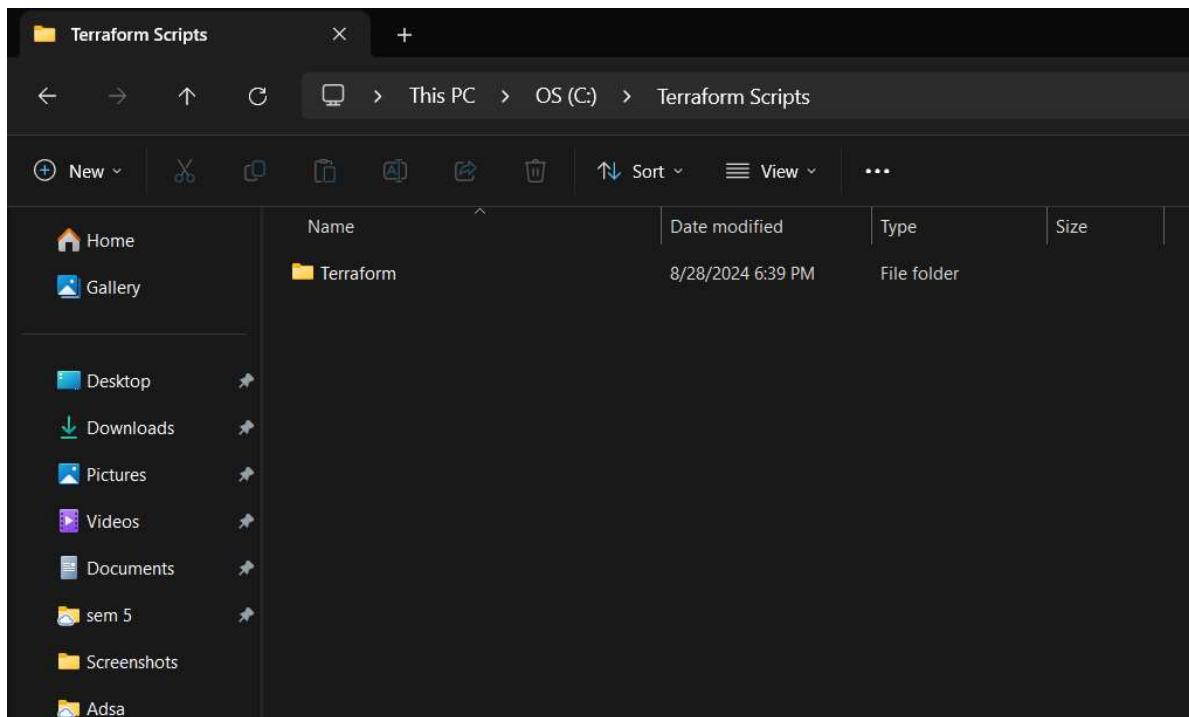
C:\Users\INFT>docker --version
Docker version 27.1.1, build 6312585

C:\Users\INFT>docker
Usage: docker [OPTIONS] COMMAND
A self-sufficient runtime for containers

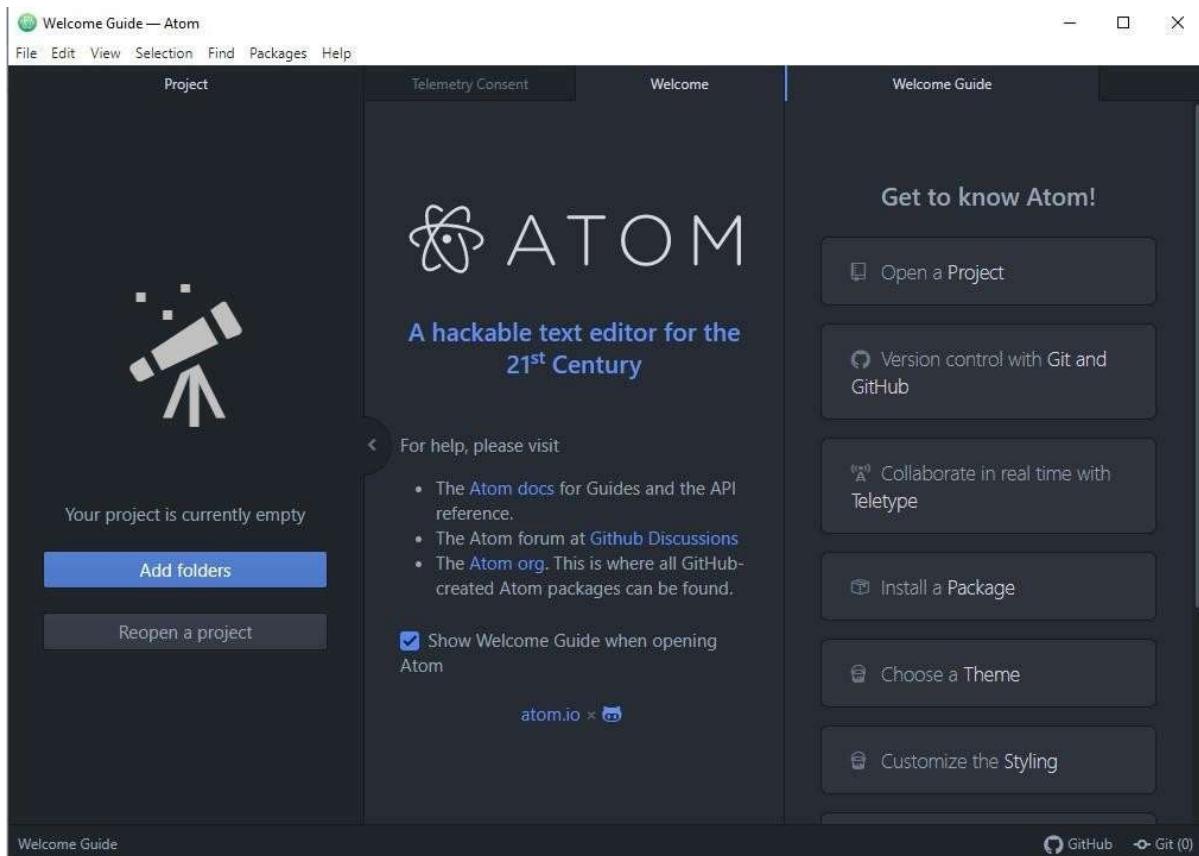
Common Commands:
  run      Create and run a new container from an image
  exec    Execute a command in a running container
  ps       List containers
  build   Build an image from a Dockerfile
  pull    Download an image from a registry
  push    Upload an image to a registry
  images  List images
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

Management Commands:
  builder  Manage builds
  buildx*  Docker Buildx
  compose*  Docker Compose
  container  Manage containers
  context   Manage contexts
```

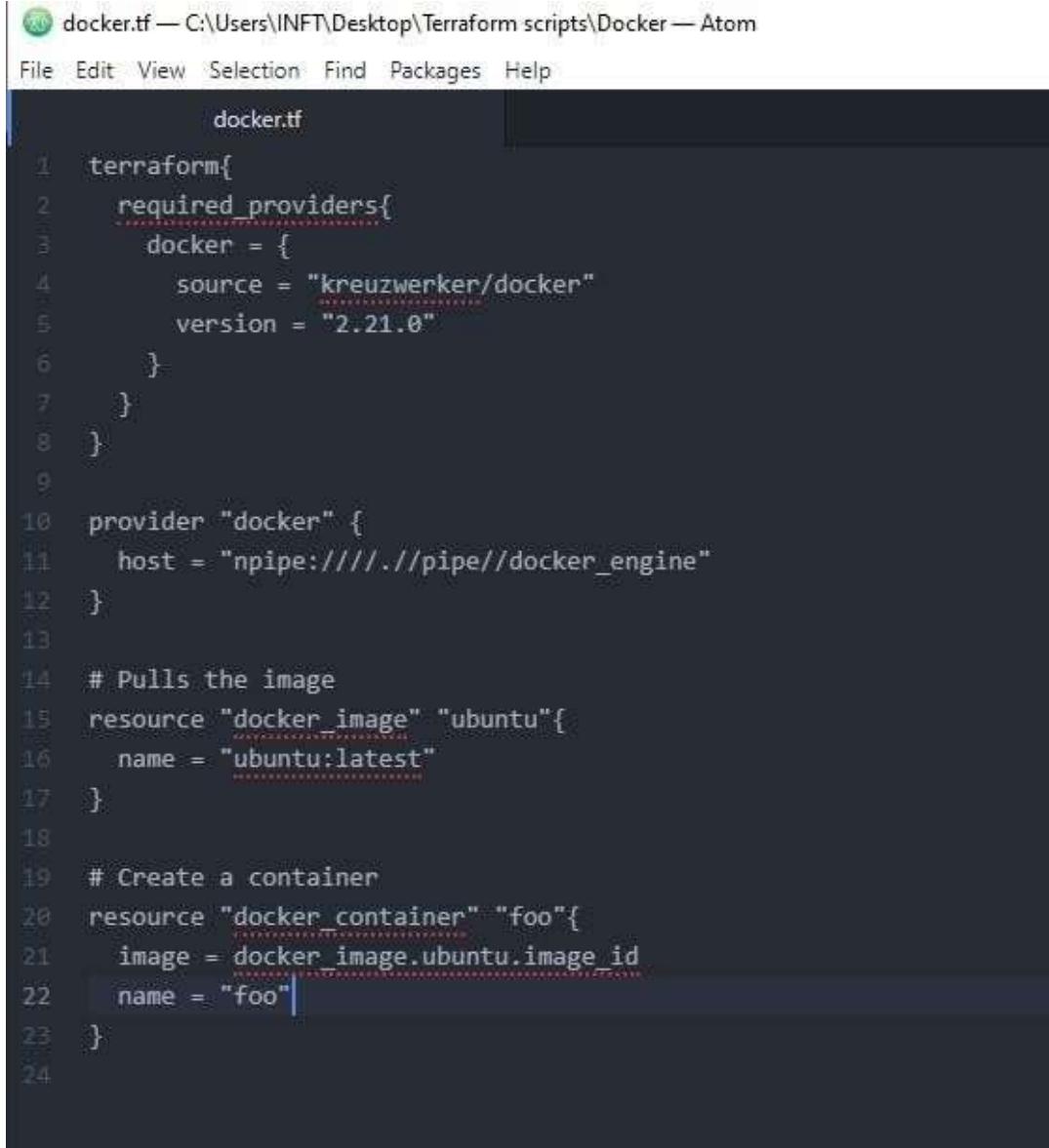
Step 4: Create a folder Terraform\_scripts and inside it create a folder named Docker.



## Step 5: Download Atom Editor.



Step 6: Run the following script in the Atom Editor

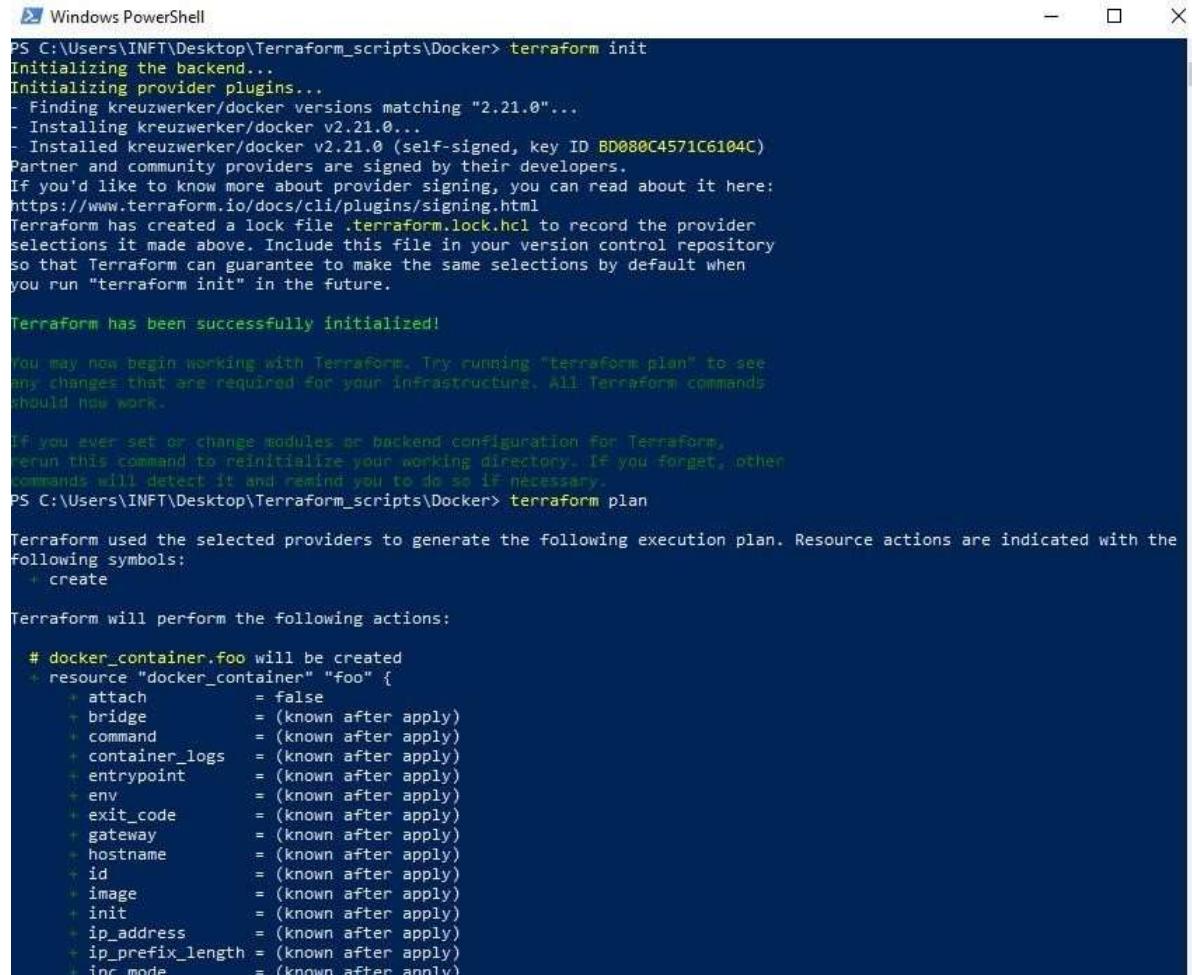


The screenshot shows the Atom code editor interface with a dark theme. The title bar reads "docker.tf — C:\Users\INFT\Desktop\Terraform scripts\Docker — Atom". The menu bar includes File, Edit, View, Selection, Find, Packages, and Help. The main editor area displays a Terraform configuration file named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

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

The cursor is positioned at the end of the "name = "foo"" line in the "docker\_container" block.

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



```
PS C:\Users\INFT\Desktop\Terraform_scripts\Docker> terraform init
Initializing the backend...
Initializing provider plugins...
- Finding kreuzwerker/docker versions matching "2.21.0"...
- Installing kreuzwerker/docker v2.21.0...
- Installed kreuzwerker/docker v2.21.0 (self-signed, key ID BD000C4571C6104C)
  Partner and community providers are signed by their developers.
If you'd like to know more about provider signing, you can read about it here:
  https://www.terraform.io/docs/cli/plugins/signing.html
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

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

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS C:\Users\INFT\Desktop\Terraform_scripts\Docker> terraform plan

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

Terraform will perform the following actions:

# docker_container.foo will be created
+ resource "docker_container" "foo" {
  + attach          = false
  + bridge          = (known after apply)
  + command         = (known after apply)
  + container_logs = (known after apply)
  + entrypoint      = (known after apply)
  + env             = (known after apply)
  + exit_code       = (known after apply)
  + gateway         = (known after apply)
  + hostname        = (known after apply)
  + id              = (known after apply)
  + image           = (known after apply)
  + init            = (known after apply)
  + ip_address      = (known after apply)
  + ip_prefix_length = (known after apply)
  + ipc_mode        = (known after apply)
```

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

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

Terraform will perform the following actions:

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

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

Enter a value: yes

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

Error: container exited immediately

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

```
Windows PowerShell

PS C:\Users\INFT\Desktop\Terraform_scripts\Docker> terraform destroy
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]

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

Terraform will perform the following actions:

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

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

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

Enter a value: yes

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

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

```
Windows PowerShell

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

## Experiment No 7

Vedant Sanap  
D15A 48  
Batch C

### AIM: Installing SonarQube from the Docker Image

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

Sonarqube:latest

```
PS D:\Desktop\DockeFile> 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
44ba2882f8eb: Pull complete
2cabec57fa36: Pull complete
c20481384b6a: Pull complete
bf7b17ee74f8: Pull complete
38617faac714: Pull complete
b795b715553d: Pull complete
c5244f6c9231: Pull complete
Digest: sha256:1ffd122cfb37ce982289dc7f5d38bb702ba05af7b5a50f7cb077ae25e60b5b9a
Status: Downloaded newer image for sonarqube:latest
1442c4e613b25aaedec05c060f020a00802b1c6dbaa27e8c5c0dad4ed8fc1f76
```

The screenshot shows the Docker Hub interface with the 'Images' tab selected. At the top, there are tabs for 'Local', 'Hub', 'Artifactory', and 'EARLY ACCESS'. Below the tabs, it says '715.97 MB / 0 Bytes in use 1 images' and 'Last refresh: 30 minutes ago'. A search bar is present. The main table lists one image:

Name	Tag	Status	Created	Size	Actions
sonarqube bb91606199eb	latest	In use	17 days ago	715.97 MB	

go to the SonarQube page by typing:

<http://localhost:9000/> on your browser.

Installation is successful if you see this page

The screenshot shows a web browser window with the URL 'localhost:9000/sessions/new?return\_to=%2F'. The page title is 'Log in to SonarQube'. It contains two input fields: one for 'admin' and one for '.....'. Below the fields are 'Log in' and 'Cancel' buttons.

## Update to new password

The screenshot shows the SonarQube interface at [localhost:9000/projects/create](http://localhost:9000/projects/create). The top navigation bar includes links for Gmail, YouTube, Translate, and What's New in Dev... The main content area is titled "How do you want to create your project?". It asks if the user wants to benefit from SonarQube's features like repository import and Pull Request decoration. Below this, it says "Create your project from your favorite DevOps platform." A note states: "First, you need to set up a DevOps platform configuration." There are six options for importing projects:

- Import from Azure DevOps (Setup)
- Import from Bitbucket Cloud (Setup)
- Import from Bitbucket Server (Setup)
- Import from GitHub (Setup)
- Import from GitLab (Setup)
- Create project manually

Below these options, a note says: "Are you just testing or have an advanced use-case? Create a project manually." A "Create project manually" button is shown.

## Create project manually:

Here project name is “AdDevops”

The screenshot shows the "Create project" step of the SonarQube setup process. The title is "Create a project". The first field is "Project display name \*", which contains "AdDevops". A note below says: "Up to 255 characters. Some scanners might override the value you provide." The next field is "Project key \*", which also contains "AdDevops". A note below says: "The project key is a unique identifier for your project. It may contain up to 400 characters. Allowed characters are alphanumeric, '-' (dash), '\_' (underscore), '.' (period) and ':' (colon), with at least one non-digit." The third field is "Main branch name \*", which contains "main". A note below says: "The name of your project's default branch [Learn More](#)". A "Next" button is visible at the bottom.

The screenshot shows the "Set up project for Clean as You Code" step. The title is "Set up project for Clean as You Code". A note says: "The new code definition sets which part of your code will be considered new code. This helps you focus attention on the most recent changes to your project, enabling you to follow the Clean as You Code methodology. Learn more: [Defining New Code](#)". The next section is "Choose the baseline for new code for this project". It has three options:

- 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 become part of the overall code.  
Recommended for projects following continuous delivery.
  - Reference branch

Choose a branch as the baseline for the new code.  
Recommended for projects using feature branches.

A "Create project" button is at the bottom.

## open Jenkins

Go to Dashboard ->Manage Jenkins -> Plugin Manager and search for SonarQube Scanner under Available plugins for Jenkins and install without restart.

### Plugins

The screenshot shows the Jenkins Plugin Manager interface. A search bar at the top contains the text "sonarqube". Below it, a table lists available plugins. The "SonarQube Scanner" plugin is highlighted with a checked checkbox in the "Install" column. The table includes columns for "Install", "Name", and "Released". The "SonarQube Scanner" entry has a release date of "10 mo ago". Other visible entries include "Sonar Gerrit" (released 20 days ago) and "SonarQube Generic Coverage" (released 4 yr 1 mo ago). At the bottom of the page, there are two buttons: "Install without restart" (highlighted in blue) and "Download now and install after restart". A status message indicates "Update information obtained: 1 day 11 hr ago" and a "Check now" button.

## Download progress

### Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

### SSH server

Success

### Deploy to container

Success

### Loading plugin extensions

Success

### SonarQube Scanner

Success

### Loading plugin extensions

Success

→ [Go back to the top page](#)

(you can start using the installed plugins right away)

→  Restart Jenkins when installation is complete and no jobs are running

## Under Jenkins ,

Dashboard -> Manage Jenkins -> Configure System ,  
Look for SonarQube Servers and enter the details. Enter the Server Authentication Token if needed.

## SonarQube servers

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

Environment variables Enable injection of SonarQube server configuration as build environment variables

### SonarQube installations

List of SonarQube installations

Name	AdDevops
Server URL	Default is http://localhost:9000 <input type="text"/>
Server authentication token	SonarQube authentication token. Mandatory when anonymous access is disabled. <input type="text" value="- none -"/> <input type="button" value="Add"/>
Advanced <input type="button" value="▼"/>	
<input type="button" value="Save"/>	<input type="button" value="Apply"/>

Search SonarQube Scanner under Dashboard -> Manage Jenkins -> Global Tool Configuration.

**Choose the latest configuration and choose Install Automatically.**

## SonarQube Scanner

### SonarQube Scanner installations

List of SonarQube Scanner installations on this system

<input type="button" value="Add SonarQube Scanner"/>	
<h4>≡ SonarQube Scanner</h4>	
Name	AdDevops
<input checked="" type="checkbox"/> Install automatically <small>?</small>	<input type="checkbox"/>
<h4>≡ Install from Maven Central</h4>	
Version	SonarQube Scanner 5.0.1.3006
<input type="button" value="Add Installer ▾"/>	
<input type="button" value="Add SonarQube Scanner"/>	

## Ant

**create a New Item in Jenkins, choose a freestyle project.**

The screenshot shows the Jenkins 'Create New Item' dialog. At the top, there is a field labeled 'Enter an item name' containing 'AdDevopsLab7'. Below this, a list of project types is shown:

- Freestyle project**: This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used
- 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) an
- Multi-configuration project**: Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific bu
- 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 : as they are in different folders.
- Multibranch Pipeline**: Creates a set of Pipeline projects according to detected branches in one SCM repository.
- Organization Folder**: Creates a set of multibranch project subfolders by scanning for repositories.

At the bottom right of the dialog, there is a blue 'OK' button.

**Choose this GitHub repository in Source Code Management.**

[https://github.com/shazforiot/MSBuild\\_firstproject.git](https://github.com/shazforiot/MSBuild_firstproject.git)

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

sonar.projectKey=AdDevops  
sonar.login=admin  
sonar.password=abc  
sonar.hosturl=<http://localhost:9000/>

Git

Repositories ?

Repository URL ?  
https://github.com/PrajaktaUpadhye6/MSBuild\_firstproject.git

Credentials ?  
- none -

Add Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?  
\*/master

Add Branch

Repository browser ?  
(Alt+)

Save Apply

Build Steps

Execute SonarQube Scanner

Task to run ?

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

Path to project properties ?

Analysis properties ?  
sonar.projectKey=AdDevops  
sonar.login=admin  
sonar.password=abc  
sonar.hosturl=http://localhost:9000/

Additional arguments ?

VM Options ?

Save Apply

Go to http://localhost:9000/ and enter your previously created username.  
**Go to Permissions and grant the Admin user Execute Permissions.**

User/Group	Administer Issues	Administer Security Hotspots	Execute Analysis
sonar-administrators System administrators	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
sonar-users Every authenticated user automatically belongs to this group	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Anyone <small>DEPRECATED</small> 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"/>	<input type="checkbox"/>
Administrator admin	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

## Build and Run:

Project AdDevopsLab7

Status: This is Lab 7.

Build History: trend

Build #1 | Sep 18, 2023, 10:31 PM

Atom feed for all | Atom feed for failures

Build #1 (Sep 18, 2023, 10:31:52 PM)

Status: Success

Changes: No changes.

Console Output: Started by user Prajakta Upadhye

Edit Build Information: Revision: f2bc042c04c6e72427c380bcaee6d6fee7b49adf

Delete build '#1': Repository: [https://github.com/PrajaktaUpadhye6/MSBuild\\_firstproject.git](https://github.com/PrajaktaUpadhye6/MSBuild_firstproject.git)

## Console Output:

Dashboard > AdDevopsLab7 > #1 > Console Output

### Console Output

```

Started by user Prajaka Upadhye
Running as SYSTEM
Building on the built-in node in workspace C:\ProgramData\Jenkins\workspace\AdDevopsLab7
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/PrajakaUpadhye/MSBuild_FirstProject.git
> git.exe init C:\ProgramData\Jenkins\workspace\AdDevopsLab7 # MSBuild_FirstProject.git
Fetching upstream changes from https://github.com/PrajakaUpadhye/MSBuild_FirstProject.git
> git.exe --version # timeout=10
> git.exe -v --version # git version 2.39.2.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/PrajakaUpadhye/MSBuild_FirstProject.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe config remote.origin.url https://github.com/PrajakaUpadhye/MSBuild_FirstProject.git # timeout=10
> git.exe config --add remote.origin.fetch refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision f2bc042c0d6c72427c3800caeddfecf79d9adf (refs/remotes/origin/master)
> git.exe checkout -f f2bc042c0d6c72427c3800caeddfecf79d9adf # timeout=10
> git.exe checkout -f f2bc042c0d6c72427c3800caeddfecf79d9adf # timeout=10
Commit message: "updated"
First time build. Skipping changelog.
Unpacking https://repo.maven.org/maven2/org/sonarsource/scanner/cli/5.8.1.3000/sonar-scanner-cli-5.8.1.3000.zip to
C:\ProgramData\Jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\AdDevops on Jenkins
[AdDevopsLab7] $ C:\ProgramData\Jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\AdDevops\bin\sonar-scanner.bat -Dsonar.projectKey=AdDevops -Dsonar.login=admin -Dsonar.hostUrl=http://localhost:9000/ -Dsonar.password=d1c1c\ProgramData\Jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\AdDevops\bin..\conf\sonar-scanner.properties
INFO: Scanner configuration file: C:\ProgramData\Jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\AdDevops\bin..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: SonarScanner 5.8.1.3000
INFO: Java 18.0.2.1 Oracle Corporation (64-bit)

Dashboard > AdDevopsLab7 > #1 > Console Output

```

INFO: Sensor IaC Docker Sensor [iac] (done) | time=7ms  
INFO: ----- Run sensors on project  
INFO: Sensor CM [csharp]  
WARN: Your project contains CM files which cannot be analyzed with the scanner you are using. To analyze CM or VB.NET, you must use the SonarScanner for .NET 5.x or higher, see <https://redirect.sonarsource.com/doc/install-configure-scanner-msbuild.html>  
INFO: Sensor CM [csharp] (done) | time=0ms  
INFO: Sensor Analysis Warnings Import [csharp]  
INFO: Sensor Analysis Warnings Import [csharp] (done) | time=15ms  
INFO: Sensor CM File Caching Sensor [csharp]  
WARN: Incremental analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir' property.  
INFO: Sensor CM File Caching Sensor [csharp] (done) | time=0ms  
INFO: Sensor Zero Coverage Sensor  
INFO: Sensor Zero Coverage Sensor (done) | time=2ms  
INFO: SCM Publisher SCM provider for this project is: git  
INFO: SCM Publisher 4 source files to be analyzed  
INFO: SCM Publisher 4/4 source files have been analyzed (done) | time=792ms  
INFO: CPD Executor Calculating CPD for 0 files  
INFO: CPD Executor CPD calculation finished (done) | time=0ms  
INFO: Analysis report generated in 83ms, dir size=139.0 kB  
INFO: Analysis report compressed in 12ms, zip size=19.2 kB  
INFO: Analysis report uploaded in 24ms  
INFO: ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=AdDevops  
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report  
INFO: More about the report processing at <http://localhost:9000/api/ce/task?id=AyqP1tkG6Xzcb174cp>  
INFO: Analysis total time: 18.844 s  
INFO: -----  
INFO: EXECUTION SUCCESS  
INFO: -----  
INFO: Total time: 31.521s  
INFO: Final Memory: 20M/74M  
INFO: -----  
Finished: SUCCESS

## Project on sonarque:

sonarcube

Projects Issues Rules Quality Profiles Quality Gates Administration More

The last analysis has warnings. See details Version not provided

AdDevops / main ✓ 3 Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

Quality Gate Status: Passed

Measures

New Code	Overall Code
Reliability	Maintainability
0 Bugs	0 Code Smells
Security	Security Review
0 Vulnerabilities	0 Security Hotspots

Duplications: 0.0% Duplications

Duplicated Blocks: 0

Enjoy your sparkling clean code!

**Conclusion:** Thus, we have successfully installed SonarQube from Docker image.

## Experiment No 8

Vedant Sanap  
D15A 48  
Batch C

### **AIM: Integrating Jenkins with SonarQube.**

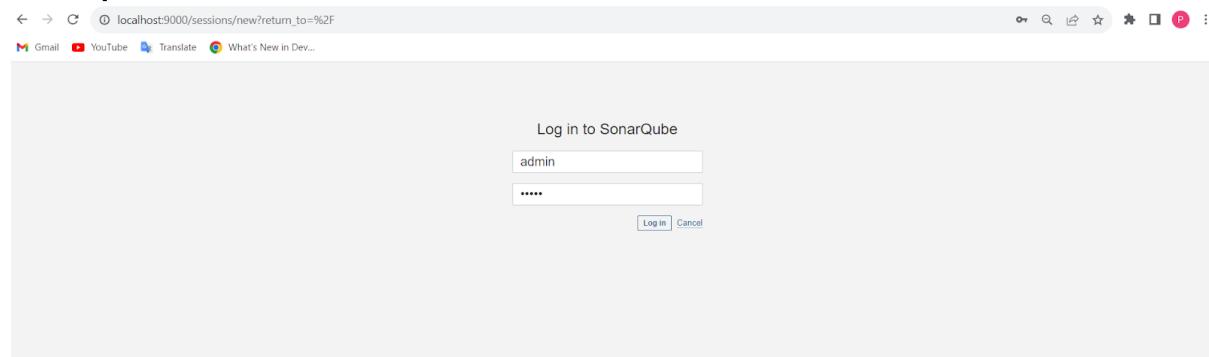
Open up Jenkins Dashboard on localhost, port 8080

Run SonarQube in a Docker container using this command -

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

```
PS D:\Desktop\DockeFile> 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  
44ba2882f8eb: Pull complete  
2cabec57fa36: Pull complete  
c20481384b6a: Pull complete  
bf7fb17ee74f8: Pull complete  
38617faac714: Pull complete  
b795b715553d: Pull complete  
c5244f6c9231: Pull complete  
Digest: sha256:1fffd122cfb37ce982289dc7f5d38bb702ba05af7b5a50f7cb077ae25e60b5b9a  
Status: Downloaded newer image for sonarqube:latest  
1442c4e613b25aaedec05c060f020a00802b1c6dbaa27e8c5c0dad4ed8fc1f76
```

### **sonarqube**



### **Create project manually**

#### **Create a project**

Project display name \*

Up to 255 characters. Some scanners might override the value you provide.

Project key \*

The project key is a unique identifier for your project. It may contain up to 400 characters.  
Allowed characters are alphanumeric, '-' (dash), '\_' (underscore), '.' (period) and ':' (colon), with at least one non-digit.

Main branch name \*

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

**Next**

Congratulations! Your project has been created.

AdDevopsLab8 / main

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

### Analysis Method

Use this page to manage and set-up the way your analyses are performed.

How do you want to analyze your repository?

- With Jenkins
- With GitHub Actions
- With Bitbucket Pipelines
- With GitLab CI
- With Azure Pipelines
- Other CI  
SonarQube integrates with your workflow no matter which CI tool you're using.
- Locally  
Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI environment.

In Jenkins create a pipeline here named “SonarQube”

Enter an item name

SonarQube

» Required field

**Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system.

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

**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (for example, a CI/CD pipeline).

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments.

**Folder**  
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is a list of items, folder is a container where items are in different folders.

**Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected branches in one SCM repository.

**Organization Folder**  
Creates a set of multibranch project subfolders by scanning for repositories.

OK

Enter the following in pipeline script:

```
node {
    stage('Cloning the GitHub Repo') {
        git 'https://github.com/PrajaktaUpadhye6/MSBuild_firstproject.git'
    }
    stage('SonarQube analysis') {
        withSonarQubeEnv('sonarqube') {
            bat "D:/sonar-scanner-cli-5.0.1.3006-windows/sonar-scanner-5.0.1.3006-windows/bin/sonar-scanner.bat \
                -D sonar.login=admin \
                -D sonar.password=abc \
                -D sonar.projectKey=AdDevops \
                -D sonar.exclusions=vendor/**,resources/**,/**/.java \
                -D sonar.host.url=http://127.0.0.1:9000/"
        }
    }
}
```

```
}
```

## Pipeline

### Definition

Pipeline script

#### Script ?

```
1 node {
2   stage('Cloning the GitHub Repo') {
3     git 'https://github.com/PrajaktaUpadhye6/HTML_TRIAL.git'
4   }
5   stage('SonarQube analysis') {
6     withSonarQubeEnv('sonarqube') {
7       bat "D:/sonar-scanner-cli-5.0.1.3006-windows/sonar-scanner-5.0.1.3006-windows/bin/sonar-scanner.bat \
8           -D sonar.login=admin \
9           -D sonar.password=abc \
10          -D sonar.projectKey=AdDevops \
11          -D sonar.exclusions=vendor/**,resources/**,*/*.java \
12          -D sonar.host.url=http://127.0.0.1:9000"
13     }
14   }
15 }
```

Use Groovy Sandbox ?

Pipeline Syntax

Save

Apply

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

Build and run:

Status Pipeline SonarQube

</> Changes Lab 8

▷ Build Now

⚙ Configure

>Delete Pipeline

Full Stage View

GitHub

SonarQube

Rename

Pipeline Syntax

Build History trend ▾

Filter builds... /

#8 Sep 20, 2023 12:36 PM No Changes

Atom feed for all Atom feed for failures

Stage View

Cloning the GitHub Repo	SonarQube analysis
Average stage times: (Average full run time: ~36s) Sep 20 12:36 No Changes	2s 33s

Permalinks

- Last build (#8), 1 min 44 sec ago
- Last stable build (#8), 1 min 44 sec ago
- Last successful build (#8), 1 min 44 sec ago
- Last completed build (#8), 1 min 44 sec ago

Console output:

**Console Output**

```

Started by user Prajaka Upadhye
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\ProgramData\Jenkins\jenkins\workspace\SonarQube
[Pipeline] }
[Pipeline] stage
[Pipeline] { (Cloning the GitHub Repo)
[Pipeline] }
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\jenkins\workspace\SonarQube\.git # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/PrajaktaUpadhye/HBuild_FirstProject.git # timeout=10
Fetching upstream changes from https://github.com/PrajaktaUpadhye/HBuild_FirstProject.git
> git.exe fetch --tags --progress
> git.exe --version # git version 2.39.2.windows.1
> git.exe fetch --tags --progress -- https://github.com/PrajaktaUpadhye/HBuild_FirstProject.git +refs/heads/*:refs/remotes/*
> git.exe rev-parse --abbrev-ref HEAD # timeout=10
Checking out Revision f2bd02d42c04de72427c380bcace6ddfee7b49adf (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2bd02d42c04de72427c380bcace6ddfee7b49adf # timeout=10
> git.exe branch -a -v --no-abrev # timeout=10
> git.exe branch -D master # timeout=10
> git.exe checkout -b master f2bd02d42c04de72427c380bcace6ddfee7b49adf # timeout=10
Commit message: "updated"
> git.exe rev-list --no-walk f2bd02d42c04de72427c380bcace6ddfee7b49adf # timeout=10
[Pipeline] }
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] End of Pipeline
Finished: SUCCESS

```

INFO: Sensor Analysis Warnings Import [csharp] (done) | time=0ms  
INFO: Sensor C# File Caching Sensor [csharp]  
WARN: Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.project  
INFO: Sensor C# File Caching Sensor [csharp] (done) | time=0ms  
INFO: Sensor C# Coverage Sensor [csharp]  
INFO: Sensor C# Coverage Sensor [csharp] (done) | time=1ms  
INFO: SCM Publisher SCM provider for this project is: git  
INFO: SCM Publisher 4 source files to be analyzed  
INFO: SCM Publisher 4/4 source files have been analyzed (done) | time=885ms  
INFO: CPD Executor Calculating CPD for 0 files  
INFO: CPD Executor CPD calculation finished (done) | time=0ms  
INFO: Analysis report generated in 94ms, dir size=139.1 KB  
INFO: Analysis report compressed in 47ms, zip size=19.0 KB  
INFO: Analysis report uploaded in 29ms  
INFO: ANALYSIS SUCCESSFUL, you can find the results at: http://127.0.0.1:9000/dashboard?id=AdDevopsLab0  
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report  
INFO: More about the report processing at http://127.0.0.1:9000/api/ce/task?id=AQwaxV1S9hGVDy2L  
INFO: Analysis total time: 25.831 s

## sonarqube:

**AdDevops PUBLIC** Passed

Last analysis: 3 minutes ago • 543 Lines of Code • HTML, CSS

E 7	A 0	E 0.0%	A 8	—	22.3%
Bugs	Vulnerabilities	Hotspots Reviewed	Code Smells	Coverage	Duplications

★ AdDevops / main ? The last analysis has warnings. See details Version not provided

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

**Quality Gate Status** ? Passed



Enjoy your sparkling clean code!

**Measures**

New Code Overall Code

<b>Reliability</b>	<b>Maintainability</b>
7 Bugs	8 Code Smells
<b>Security</b>	<b>Security Review</b>
0 Vulnerabilities	6 Security Hotspots

**Duplications**

22.3% Duplications

## Reliability:

AdDevops / main

The last analysis has warnings. See details Version not provided

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

**Project Overview**

**Reliability**

- Overview
- New Code
- Bugs: 0
- Rating: A
- Remediation Effort: 0
- Overall Code
- Bugs: 7

**AdDevops**

View as List Select files Navigate 15 files

**Reliability Rating** See history New Code: Since September 20, 2023

File	Rating
scroll.css	E
trial.css	C
verticaltable.html	C
demo.html	B
index.html	B
payment.html	B

There are 9 hidden components with a score of A. Show Them

## Bugs:

AdDevops / main

The last analysis has warnings. See details Version not provided

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

**Bugs**

- Bugs: 0
- Rating: A
- Remediation Effort: 0
- Overall Code
- Bugs: 7
- Rating: E
- Remediation Effort: 28min

**Bugs** See history New Code: Since September 20, 2023

File	Rating
box.html	0
class.css	0
demo.html	1
element.css	0
float property.html	0
home.html	0
id.css	0
index.html	1
payment.html	1
scroll.css	1

## Maintainability:

AdDevops / main

The last analysis has warnings. See details Version not provided

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

**Project Overview**

**Reliability**

**Security**

**Security Review**

**Maintainability**

- Overview
- New Code
- Code Smells: 0

**AdDevops**

View as List Select files Navigate 15 files

**Maintainability Rating** See history New Code: Since September 20, 2023

File	Rating
box.html	A
class.css	A
demo.html	A
element.css	A
float property.html	A
home.html	A
id.css	A
index.html	A

## Security:

[AdDevops / main](#) ?

The last analysis has warnings. [See details](#) Version not provided

Project Settings Project Information

Overview Issues Security Hotspots **Measures** Code Activity

Project Overview

**Reliability** ? >

**Security** ? >

**Security Review** ? >

New Code

    Security Hotspots 0

Rating A

Overall Code

    Security Hotspots 6

AdDevops

View as List Select files Navigate 15 files

**Security Review Rating** E [See history](#)

New Code: Since September 20, 2023

demo.html	E
home.html	E
index.html	E

There are 12 hidden components with a score of A. [Show Them](#)

## Duplications:

[AdDevops / main](#) ?

The last analysis has warnings. [See details](#) Version not provided

Project Settings Project Information

Overview Issues Security Hotspots **Measures** Code Activity

Project Overview

**Reliability** ? >

**Security** ? >

**Security Review** ? >

**Maintainability** ? >

**Coverage** >

**Duplications** >

Overview

AdDevops

View as List Select files Navigate 15 files

**Duplicated Lines (%) 22.3%** [See history](#)

New Code: Since September 20, 2023

	Duplicated Lines (%)	Duplicated Lines
demo.html	100%	67
index.html	100%	67
box.html	0.0%	0
class.css	0.0%	0
element.css	0.0%	0
float property.html	0.0%	0
home.html	0.0%	0

## Cyclomatic complexity:

[AdDevops / main](#) ?

The last analysis has warnings. [See details](#) Version not provided

Project Settings Project Information

Overview Issues Security Hotspots **Measures** Code Activity

New Lines 0

Lines of Code 543

Lines 602

Files 15

Comment Lines 47

Comments (%) 8.0%

**Complexity** ? >

    Cyclomatic Complexity 9

Issues >

AdDevops

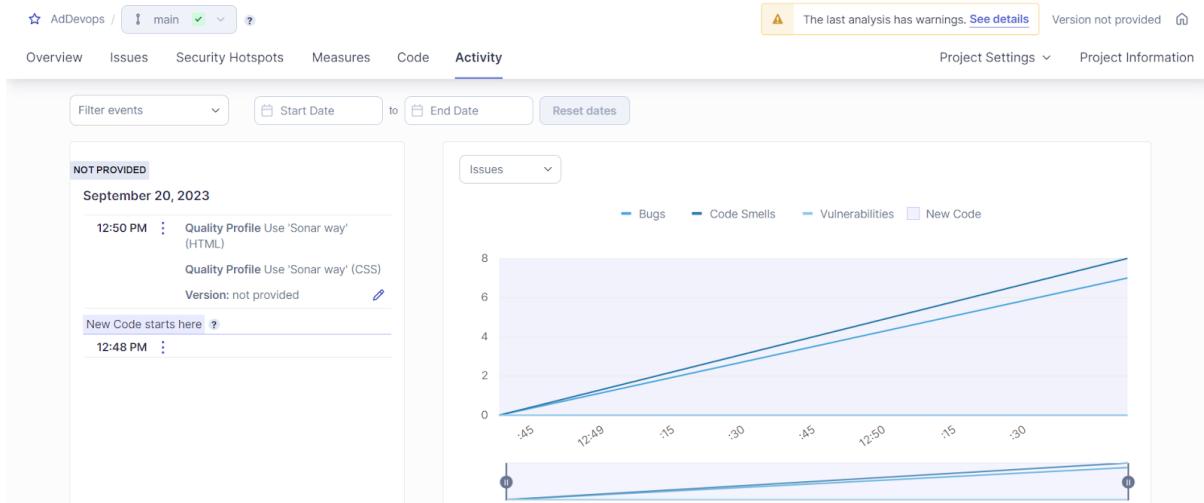
View as List Select files Navigate 9 files

**Cyclomatic Complexity 9** [See history](#)

New Code: Since September 20, 2023

box.html	1
demo.html	1
float property.html	1
home.html	1
index.html	1
payment.html	1
start.html	1
trial.html	1
verticaltable.html	1

## Activity:



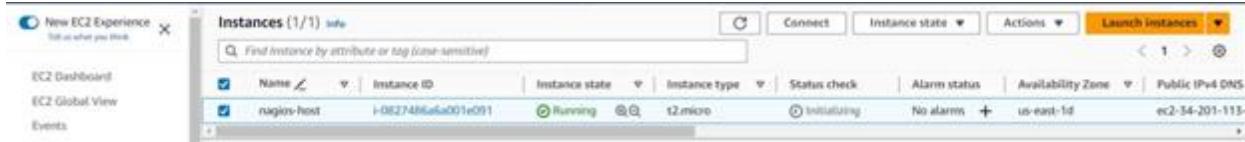
**Conclusion:** Thus, we have successfully integrated Jenkins with SonarQube.

## Experiment No 9

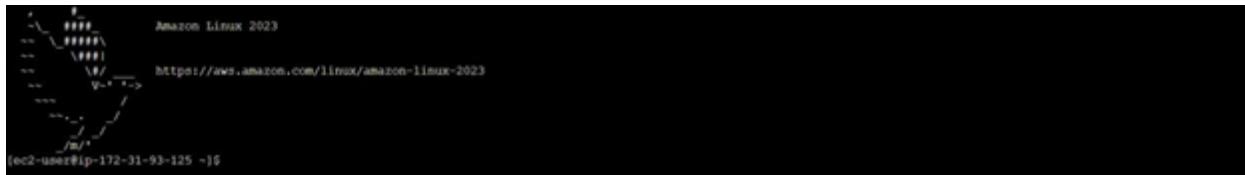
Vedant Sanap  
D15A 48  
Batch C

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

1)



2)



3)

```
git:(dev) [7]:7-697.amzn2023.0.2.x86_64
graphite2-2.1.3-14-7.amzn2023.0.2.x86_64
harfbuzz-devel-7.0.0-2.amzn2023.0.1.x86_64
langpacks-core-font-en-3.0-21.amzn2023.0.4.search
libcurl-1.7.2-3.amzn2023.0.3.x86_64
libcurl-xcb-1.7.2-3.amzn2023.0.3.x86_64
libcurlx-1.3.4-6.amzn2023.0.2.x86_64
libcurlz-0.9.10-14.amzn2023.0.2.x86_64
libcurlz-devel-0.9.10-14.amzn2023.0.2.x86_64
libcurlz-devel-1.4.4-1.amzn2023.0.1.x86_64
libcurlz-pegturbo-2.1.4-2.amzn2023.0.5.x86_64
libcurlng-2.1.6-37-10.amzn2023.0.6.x86_64
libcurlpol-devel-9.4-3.amzn2023.0.3.x86_64
libcurlng-1.2.4-1.amzn2023.0.4.x86_64
libcurlc-devel-1.13.1-7.amzn2023.0.2.x86_64
pcrutils-utf16-10.40-1.amzn2023.0.3.x86_64
pcrutils-capture-devel-3.40-1.2.amzn2023.0.2.x86_64
libcurl-devel-1.7.11-33.amzn2023.0.4.x86_64

Completed!
[ec2-user@ip-172-31-93-125 ~]$ sudo adduser -m nagios
[ec2-user@ip-172-31-93-125 ~]$ jatini
dash: jatini: command not found
[ec2-user@ip-172-31-93-125 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
Re-type new password:
passwd: all authentication tokens updated successfully.

google-noto-fonts-common-20201205-2.amzn2023.0.2.x86_64
graphite2-devel-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz-icu-7.0.0-2.amzn2023.0.1.x86_64
libcurl-1.7.2-3.amzn2023.0.3.x86_64
libcurlx-common-1.7.2-3.amzn2023.0.3.search
libcurlx-1.0.9-4.amzn2023.0.2.x86_64
libcurlz-devel-1.0.9-4.amzn2023.0.2.x86_64
libcurlzpm-devel-3.5.15-2.amzn2023.0.1.x86_64
libcurlx-1.2.0-4.amzn2023.0.2.x86_64
libcurlz-0.9.10-14.amzn2023.0.2.x86_64
libcurlz-devel-1.4.4-1.amzn2023.0.3.x86_64
libcurlz-pegturbo-devel-2.1.4-2.amzn2023.0.5.x86_64
libcurlng-devel-2.1.6-37-10.amzn2023.0.6.x86_64
libcurlpol-devel-9.4-3.amzn2023.0.3.x86_64
libcurlng-devel-1.2.4-1.amzn2023.0.6.x86_64
libcurlc-devel-1.13.1-7.amzn2023.0.2.x86_64
pcrutils-utf16-10.40-1.amzn2023.0.3.x86_64
pixman-0.40-0.9.amzn2023.0.9.x86_64
korg-all-pegturbo-devel-2021.4-1.amzn2023.0.7.search

[ec2-user@ip-172-31-93-125 ~]$
```

4)

```
[ec2-user@ip-172-31-93-125 ~]$ mkdir ~/downloads
[ec2-user@ip-172-31-93-125 ~]$ cd ~/downloads
[ec2-user@ip-172-31-93-125 downloads]$ wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
--2023-10-04 15:54:00-- http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2659772 (2.5M) [application/x-gzip]
Saving to: 'nagios-plugins-2.0.3.tar.gz'

nagios-plugins-2.0.3.tar.gz      100%[=====]   2.50K  6.04MB/s    in 0.4s

2023-10-04 15:54:01 (6.04 MB/s) - 'nagios-plugins-2.0.3.tar.gz' saved [2659772/2659772]
```

## 5)

```
[ec2-user@ip-172-31-43-1 ~]$ mkdir downloads
[ec2-user@ip-172-31-43-1 ~]$ cd downloads
[ec2-user@ip-172-31-43-1 downloads]$ wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
--2021-10-23 22:38:58-- http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
Resolving prdownloads.sourceforge.net (prdownloads.sourceforge.net)... 204.68.111.105
Connecting to prdownloads.sourceforge.net (prdownloads.sourceforge.net)|204.68.111.105|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz [following]
--2021-10-23 22:38:58-- http://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz
Resolving downloads.sourceforge.net (downloads.sourceforge.net)... 204.68.111.105
```

## 6)

```
[ec2-user@ip-172-31-43-1 nagios-4.0.8]$ ./configure --with-command-group=nagcmd
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
checking for gcc... gcc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
```

## 7)

```
GNU nano 2.9.8          /usr/local/nagios/etc/objects/c

#####
# CONTACTS.CFG - SAMPLE CONTACT/CONTACTGROUP DEFINITIONS
#
# NOTES: This config file provides you with some example contact and contact
# group definitions that you can reference in host and service
# definitions.
#
# You don't need to keep these definitions in a separate file from your
# other object definitions. This has been done just to make things
# easier to understand.
#
#####
# CONTACTS
#
#####

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

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

## 8)

```
*** Support Notes *****
```

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

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

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

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

For more information on obtaining support for Nagios, visit:

<http://support.nagios.com>

```
*****
```

Enjoy.

```
[ec2-user@ip-172-31-43-1 nagios-4.0.8]$ |
```

9)

```
[ec2-user@ip-172-31-46-218 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 ***
```

10)

```
[ec2-user@ip-172-31-46-218 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-172-31-46-218 nagios-4.0.8]$ |
```

11)

```
[ec2-user@ip-172-31-46-218 ~]$ cd ~/downloads/  
[ec2-user@ip-172-31-46-218 downloads]$ tar zxvf nagios-plugins-2.0.3.tar.gz|
```

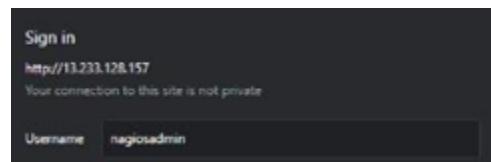
12)

```
:2-user@ip-172-31-46-218 ~]$ sudo service nagios start
Starting nagios (via systemctl): [ OK ]
:2-user@ip-172-31-46-218 ~]$ |
```

13)

```
[ec2-user@ip-172-31-46-218 ~]$ sudo systemctl status nagios
● nagios.service - LSB: Starts and stops the Nagios monitoring server
  Loaded: loaded (/etc/rc.d/init.d/nagios; bad; vendor preset: disabled)
  Active: active (running) since Sun 2021-10-24 08:05:00 UTC; 1min 21s ago
    Docs: man:systemd-sysv-generator(8)
 Process: 30073 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status
 CGroup: /system.slice/nagios.service
         ├─30094 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagi
         ├─30096 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/va
         ├─30097 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/va
         ├─30098 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/va
         ├─30099 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/va
         └─30100 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagi
```

14)



15)

**Conclusion:** Thus, we have understood continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

## Experiment No 10

Vedant Sanap  
D15A 48  
Batch C

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

### Output-

**Step 1:** To Confirm that Nagios is running on the server side, run this sudo systemctl status nagios on the “NAGIOS HOST”.

```
● nagios.service - Nagios Core 4.4.14
   Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-09-30 08:54:01 UTC; 20s ago
     Docs: https://www.nagios.org/documentation
 Process: 55285 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Process: 55286 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 55287 (nagios)
   Tasks: 6 (limit: 1141)
  Memory: 5.3M
    CPU: 252ms
   CGroup: /system.slice/nagios.service
           ├─55287 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           ├─55288 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─55289 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─55290 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─55291 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           └─55292 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 08:54:01 ip-172-31-44-151 nagios[55287]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
lines 1-19]
```

**Step 2:** To monitor a Linux machine, create an Ubuntu 20.04 server EC2 Instance in AWS.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
cutenagios_se...	i-09d6b0d2e181a7287	Running	t2.micro	-	No alarms	ap-south-
cutenagios_cli...	i-0e36968400dac0991	Running	t2.micro	-	No alarms	ap-south-
cutenagios_se...	i-03a3e79fc5ab0a056	Stopped	t2.micro	-	No alarms	ap-south-

**Step 3:** On client side Step-03 Make a package index update and install gcc, nagios-nrpe-server and the plugins.

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

```
*** System restart required ***
Last login: Sat Sep 30 08:31:30 2023 from 13.233.177.3
ubuntu@ip-172-31-44-151:~$ sudo apt install gcc -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gcc is already the newest version (4:11.2.0-1ubuntu1).
gcc set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
ubuntu@ip-172-31-44-151:~$ 
root@ip-172-31-44-151:/home/ubuntu# sudo apt install nagios-nrpe-server nagios-plugins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'monitoring-plugins' instead of 'nagios-plugins'.
monitoring-plugins is already the newest version (2.3.1-1ubuntu4).
nagios-nrpe-server is already the newest version (4.0.3-1ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
```

```

Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Fetched 229 kB in 1s (290 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
2 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-44-151:/home/ubuntu# sudo apt install gcc -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gcc is already the newest version (4:11.2.0-1ubuntu1).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
root@ip-172-31-44-151:/home/ubuntu# sudo apt install -y nagios-nrpe-server nagios-plugins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'monitoring-plugins' instead of 'nagios-plugins'
monitoring-plugins is already the newest version (2.3.1-1ubuntu4).
nagios-nrpe-server is already the newest version (4.0.3-1ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.

```

#### Step 4: Open nrpe.cfg file to make changes.

sudo nano /etc/nagios/nrpe.cfg

```

GNU nano 6.2                               /etc/nagios/nrpe.cfg
# SERVER ADDRESS
# Address that nrpe should bind to in case there are more than one interface
# and you do not want nrpe to bind on all interfaces.
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

server_address=127.0.0.1

# LISTEN QUEUE SIZE
# Listen queue size (backlog) for serving incoming connections.
# You may want to increase this value under high load.

listen_queue_size=5

^G Help      ^C Write Out    ^W Where Is     ^K Cut          ^T Execute      ^C Location    M-U Undo
^X Exit      ^R Read File    ^V Replace      ^U Paste        ^J Justify     ^Y Go To Line  M-P Redo
M-A Set Mark M-6 Copy

: 127.0.0.1,::1,13.235.0.144           /etc/nagios/nrpe.cfg *
95 # that are allowed to talk to the NRPE daemon. Network addresses with a bit mask
96 # (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not currently
97 # supported.
98 #
99 # Note: The daemon only does rudimentary checking of the client's IP
100 # address. I would highly recommend adding entries in your /etc/hosts.allow
101 # file to allow only the specified host to connect to the port
102 # you are running this daemon on.
103 #
104 # NOTE: This option is ignored if NRPE is running under either inetd or xinetd
105
106 allowed_hosts=127.0.0.1,::1,13.235.0.144
107 server_address=0.0.0.0
108
109
110

^G Help      ^C Write Out    ^W Where Is     ^K Cut          ^T Execute      ^C Location    M-U Undo
^X Exit      ^R Read File    ^V Replace      ^U Paste        ^J Justify     ^Y Go To Line  M-P Redo
M-A Set Mark M-6 Copy

```

#### Step 5: Restart the NRPE server

sudo systemctl restart nagios-nrpe-server

```

Restarting services...
Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service
systemctl restart user@1000.service

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-41-41:/home/ubuntu# sudo nano /etc/nagios/nrpe.cfg
root@ip-172-31-41-41:/home/ubuntu# sudo systemctl restart nagios-nrpe-server
root@ip-172-31-41-41:/home/ubuntu# sudo systemctl status nagios-nrpe-server
* nagios-nrpe-server.service - Nagios Remote Plugin Executor

```

```

root@ip-172-31-41-41:/home/ubuntu# sudo systemctl status nagios-nrpe-server
● nagios-nrpe-server.service - Nagios Remote Plugin Executor
   Loaded: loaded (/lib/systemd/system/nagios-nrpe-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-09-30 09:27:17 UTC; 6s ago
     Docs: http://www.nagios.org/documentation
 Main PID: 7349 (nrpe)
    Tasks: 1 (limit: 1141)
   Memory: 1.5M
      CPU: 9ms
     CGroup: /system.slice/nagios-nrpe-server.service
             └─7349 /usr/sbin/nrpe -c /etc/nagios/nrpe.cfg -f

Sep 30 09:27:17 ip-172-31-41-41 systemd[1]: nagios-nrpe-server.service: Deactivated successfully.
Sep 30 09:27:17 ip-172-31-41-41 systemd[1]: Stopped Nagios Remote Plugin Executor.
Sep 30 09:27:17 ip-172-31-41-41 systemd[1]: Started Nagios Remote Plugin Executor.
Sep 30 09:27:17 ip-172-31-41-41 nrpe[7349]: Starting up daemon
Sep 30 09:27:17 ip-172-31-41-41 nrpe[7349]: Server listening on 0.0.0.0 port 5666.
Sep 30 09:27:17 ip-172-31-41-41 nrpe[7349]: Listening for connections on port 5666
Sep 30 09:27:17 ip-172-31-41-41 nrpe[7349]: Allowing connections from: 127.0.0.1,::1,13.235.0.144
root@ip-172-31-41-41:/home/ubuntu# 

```

### Step 6: On the server run this command

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

```

root@ip-172-31-44-151:/home/ubuntu# ps -ef | grep nagios
nagios  55287      1  0 08:54 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios  55288  55287  0 08:54 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios  55289  55287  0 08:54 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios  55290  55287  0 08:54 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios  55291  55287  0 08:54 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios  55292  55287  0 08:54 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios  56327      1  0 08:58 ?        00:00:00 /usr/sbin/nrpe -c /etc/nagios/nrpe.cfg -f
root  60903  60158  0 09:32 pts/1    00:00:00 grep --color=auto nagios
root@ip-172-31-44-151:/home/ubuntu# sudo su
root@ip-172-31-44-151:/home/ubuntu# mkdir /usr/local/nagios/etc/objects/monitorhosts
root@ip-172-31-44-151:/home/ubuntu# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts

```

### Step 7: Become a root user and create 2 folders 1.sudo su 2.mkdir

```
/usr/local/nagios/etc/objects/monitorhosts 3.mkdir
```

```
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts Copy the sample localhost.cfg file to
linuxhost folder 4.cp /usr/local/nagios/etc/objects/localhost.cfg
```

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

```

root@ip-172-31-44-151:/home/ubuntu# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
root@ip-172-31-44-151:/home/ubuntu# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

```

### Step 8: Open linuxserver.cfg using nano and make the following changes

nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg Change the hostname  
to linux server (EVERYWHERE ON THE FILE) Change address to the public IP address of your  
LINUX CLIENT.

```

GNU nano 6.2          /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

# Define a host for the local machine

define host {

    use           linux-server      ; Name of host template to use
                                ; This host definition will inherit all variables that are defined
                                ; in (or inherited by) the linux-server host template definition.

    host_name     localhost
    alias         localhost
    address       127.0.0.1
}

[...]

```

**i-03a3e79fc5ab0a056 (cutenagios\_server)**

```

GNU nano 6.2          /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg *

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

# Define an optional hostgroup for Linux machines

define hostgroup {

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

[...]

```

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

### Step 9: Open the Nagios Config file and add the following line nano

/usr/local/nagios/etc/nagios.cfg Add this line cfg\_dir=/usr/local/nagios/etc/objects/monitorhosts/

```

GNU nano 6.2          /usr/local/nagios/etc/nagios.cfg *

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

#cfg_dir=/usr/local/nagios/etc/servers
#cfg_dir=/usr/local/nagios/etc/printers
#cfg_dir=/usr/local/nagios/etc/switches
#cfg_dir=/usr/local/nagios/etc/routers
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts

# OBJECT CACHE FILE
# This option determines where object definitions are cached when
# Nagios starts/restarts. The CGIs read object definitions from
Save modified buffer?
Y Yes
N No      ^C Cancel

```

### Step 10: Verify the configuration files.

```

root@ip-172-31-44-151:/home/ubuntu# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
root@ip-172-31-44-151:/home/ubuntu#   /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.4.14
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2023-08-01
License: GPL

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

Running pre-flight check on configuration data...

Checking objects...
  Checked 8 services.
  Checked 1 hosts.
  Checked 1 host groups.

  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 1 hosts
  Checked 0 service dependencies
  Checked 0 host dependencies
  Checked 5 timperiods

Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors: 0

Things look okay - No serious problems were detected during the pre-flight check
root@ip-172-31-44-151:/home/ubuntu# [nano /usr/local/nagios/etc/nagios.cfg]

```

## Step 11: Restart the nagios service service nagios restart

Sudo systemctl status nagios

```

● nagios.service - Nagios Core 4.4.14
   Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat Sep 30 08:54:01 UTC; 20s ago
     Docs: https://www.nagios.org/documentation
   Process: 55285 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Process: 55286 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 55287 (nagios)
   Tasks: 6 (limit: 1141)
    Memory: 5.3M
      CPU: 252ms
     CGroup: /system.slice/nagios.service
             ├─55287 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─55288 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─55289 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─55290 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─55291 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─55292 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 08:54:01 ip-172-31-44-151 nagios[55287]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
lines 1-15 |
```

## Step 12: Now, check your nagios dashboard and you'll see a new host being added.

Host **	Status **	Last Check **	Duration **	Status Information
localhost	<span style="color: green;">OK</span> UP	09-30-2023 18:17:06	0d 0h 5m 3s	PING OK - Packet loss = 0%, RTA = 0.62 ms
centos	<span style="color: green;">OK</span> UP	09-30-2023 18:20:14	0d 9h 28m 7s	PING OK - Packet loss = 0%, RTA = 0.04 ms

Results 1 - 2 of 2 Matching Hosts

Not secure | 13.233.247.135/nagios/

# Nagios®

**General**

- Home
- Documentation

**Current Status**

- Tactical Overview
- Map (Legacy)
- Host
- Services
- Host Groups

  - Summary
  - Grid

- Service Groups

  - Summary
  - Grid

- Problems

  - Services (Unhandled)
  - Hosts (Unhandled)
  - Network Outages

- Quick Search:

**Reports**

- Availability
- Trends (Legacy)
- Alerts

  - History
  - Summary
  - Histogram (Legacy)

- Notifications
- Event Log

**System**

- Comments
- Downtime
- Process Info
- Performance Info
- Schedule Queue
- Configuration

**Current Network Status**

Last check time: 03 Mar 23:38:11 UTC 2023  
Updated every 90 seconds  
Nagios® Core™ 4.4.14 - 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
2	0	0	0

[All Problems](#) [All Types](#)

**Service Status Totals**

OK	Warning	Unknown	Critical	Pending
13	0	0	3	0

[All Problems](#) [All Types](#)

**Service Status Details For All Hosts**

Limit Results: 100 ▾

Host **	Service **	Status **	Last Check **	Duration **	Attempt **	Status Information
linuxserver	Current Load	OK	10-03-2023 23:34:51	3d 13h 47m 10s	1/4	OK - load average: 0.00, 0.02, 0.00
	Current Users	OK	10-03-2023 23:35:29	3d 13h 46m 32s	1/4	USERS OK - 2 users currently logged in
	HTTP	CRITICAL	10-03-2023 23:36:06	0d 0h 12m 5s	4/4	CRITICAL - Socket timeout!
	PING	CRITICAL	10-03-2023 23:36:44	0d 0h 1m 27s	1/4	PING OK - Packet loss = 0%, RTA = 0.60 ms
	Root Partition	OK	10-03-2023 23:37:21	3d 13h 44m 40s	1/4	DISK OK - free space: / 4859 MB (62.78% inode=88%)
	SSH	OK	10-03-2023 23:37:59	0d 0h 0m 12s	1/4	SSH OK - OpenSSH_8.9p1 Ubuntu-3ubuntu0.1 (protocol 2.0)
	Swap Usage	CRITICAL	10-03-2023 23:33:36	3d 13h 43m 25s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
Total Processes	OK	10-03-2023 23:34:14	3d 13h 42m 47s	1/4	PROCS OK: 39 processes with STATE = R/SZDT	
localhost	Current Load	OK	10-03-2023 23:35:10	3d 14h 43m 33s	1/4	OK - load average: 0.00, 0.02, 0.00
	Current Users	OK	10-03-2023 23:35:47	3d 14h 42m 55s	1/4	USERS OK - 2 users currently logged in
	HTTP	CRITICAL	10-03-2023 23:36:25	3d 14h 42m 18s	1/4	HTTP OK: HTTP/1.1 200 OK - 10945 bytes in 0.000 second response time
	PING	OK	10-03-2023 23:37:02	3d 14h 41m 40s	1/4	PING OK - Packet loss = 0%, RTA = < 0.04 ms
	Root Partition	OK	10-03-2023 23:37:40	3d 14h 41m 3s	1/4	DISK OK - free space: / 4859 MB (62.78% inode=88%)
	SSH	OK	10-03-2023 23:33:17	3d 14h 40m 25s	1/4	SSH OK - OpenSSH_8.9p1 Ubuntu-3ubuntu0.4 (protocol 2.0)
	Swap Usage	CRITICAL	10-03-2023 23:33:55	3d 14h 36m 48s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
Total Processes	OK	10-03-2023 23:33:24	3d 14h 39m 10s	1/4	PROCS OK: 40 processes with STATE = R/SZDT	

Results 1 - 16 of 16 Matching Services

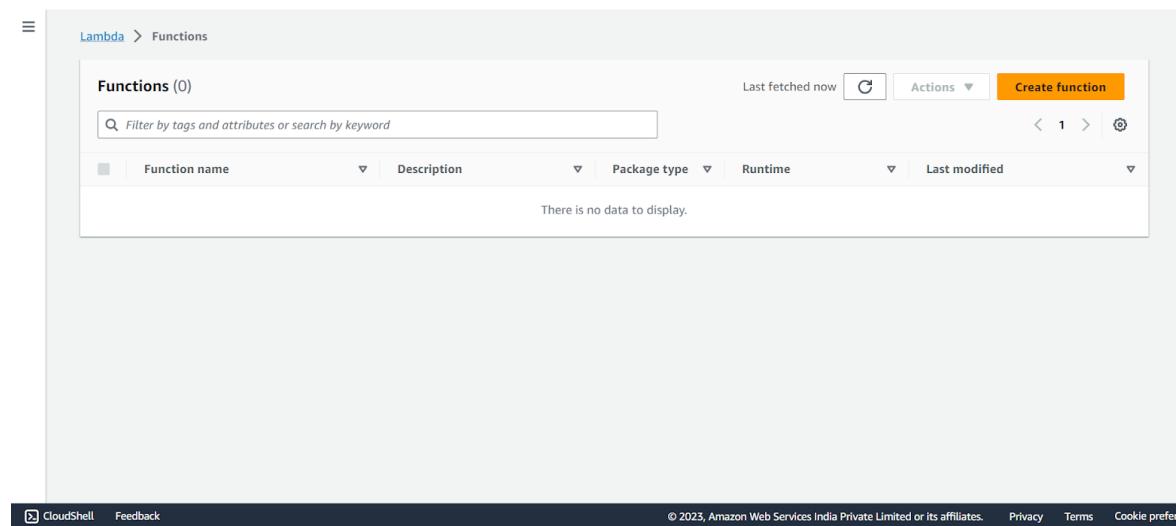
# Experiment No 11

Vedant Sanap  
D15A 48  
Batch C

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

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

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



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

2. Choose to create a function from scratch or use a blueprint, i.e templates defined by AWS for you with all configuration presets required for the most common use cases.  
Then, choose a runtime env for your function, under the dropdown, you can see all the options AWS supports, Python, Nodejs, .NET and Java being the most popular ones.  
After that, choose to create a new role with basic Lambda permissions if you don't have an existing one.

Lambda > Functions > Create function

## Create function Info

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

Author from scratch  
Start with a simple Hello World example.

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

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

### Basic information

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

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

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

Lambda > Functions > Create function Info

## Create function Info

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

Author from scratch  
Start with a simple Hello World example.

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

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

### Basic information

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

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

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

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

arm64

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

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

Lambda > Functions > Create function

## Create function Info

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

Author from scratch  
Start with a simple Hello World example.

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

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

### Basic information

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

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

arm64

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

▶ Change default execution role

▶ Advanced settings

Cancel Create function

Click on the Create button.

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

The screenshot shows the AWS Lambda Functions console. A green banner at the top indicates "Successfully created the function myPythonLambdaFunction. You can now change its code and configuration. To invoke your function with a test event, choose 'Test'." Below the banner, the function name "myPythonLambdaFunction" is displayed. The "Function overview" section shows a thumbnail of the function icon, a "Layers" section (empty), and buttons for "+ Add trigger" and "+ Add destination". On the right, there are sections for "Description", "Last modified" (15 seconds ago), "Function ARN" (arn:aws:lambda:ap-south-1:447953971928:function:myPythonLambdaFunction), and "Function URL" (info). Below this, tabs for "Code", "Test", "Monitor", "Configuration", "Aliases", and "Versions" are visible. The "Code source" tab is selected, showing a code editor with the following Python code:

```
1 import json
2
3 def lambda_handler(event, context):
4     # TODO Implement
5     return [
6         {
7             'statusCode': 200,
8             'body': json.dumps('Hello from Lambda!')
9         }
10 ]
```

This screenshot is identical to the one above, showing the AWS Lambda Functions console with the "Code source" tab selected. The code editor displays the same Python code as before:

```
1 import json
2
3 def lambda_handler(event, context):
4     # TODO Implement
5     return [
6         {
7             'statusCode': 200,
8             'body': json.dumps('Hello from Lambda!')
9         }
10 ]
```

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

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

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

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

[Edit](#)

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

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

## Edit basic settings

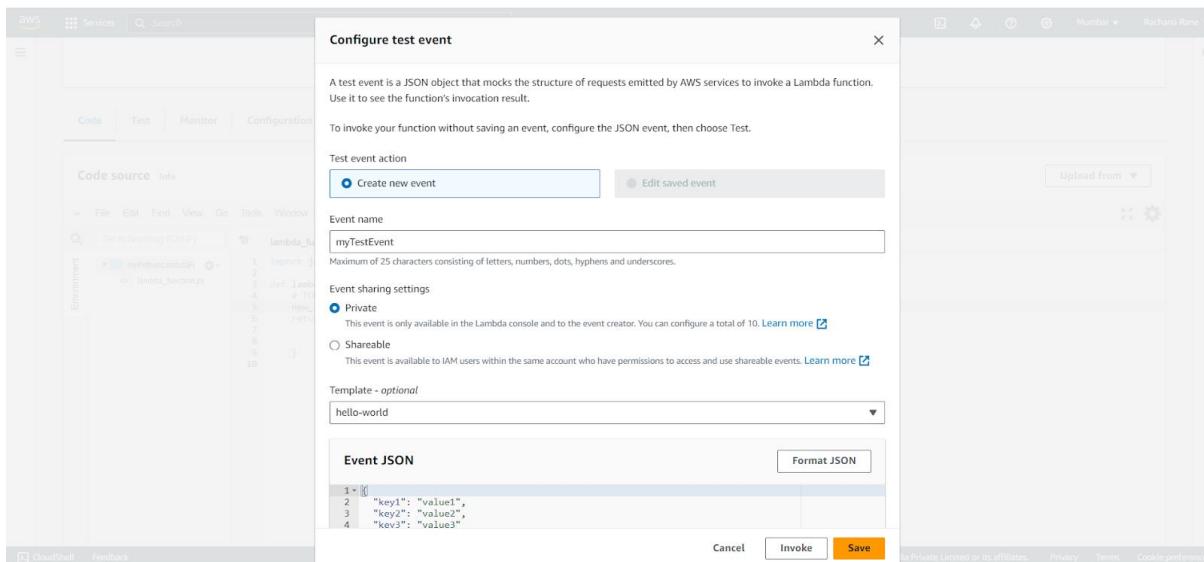
Basic settings <a href="#">Info</a>
Description - optional <input type="text"/>
Memory <a href="#">Info</a> Your function is allocated CPU proportional to the memory configured. <input type="text" value="128"/> MB Set memory to between 128 MB and 10240 MB
Ephemeral storage <a href="#">Info</a> You can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

CloudShell Feedback

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

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

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



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

The screenshot shows the AWS Lambda Test interface. At the top, a green banner says "The test event myTestEvent was successfully saved." Below this is a navigation bar with File, Edit, Find, View, Go, Tools, Window, Test (selected), Deploy, and Changes not deployed. The main area has tabs for Go to Anything (Ctrl-P), lambda\_function (selected), Environment Vari, and Execution result. The Execution result tab shows the following details:

- Test Event Name: myTestEvent
- Response:

```
{ "statusCode": 200, "body": "Hello from Lambda!" }
```
- Function Logs:

```
START RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc Version: $LATEST
END RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc
REPORT RequestId: 7d26f404-f1da-4435-9faf-8dbb2a2733cc Duration: 1.66 ms Billed Duration: 2 ms Memory Size: 128 MB Max Memory Used: 40 MB Init Duration: 110.05 ms
```
- Request ID: 7d26f404-f1da-4435-9faf-8dbb2a2733cc

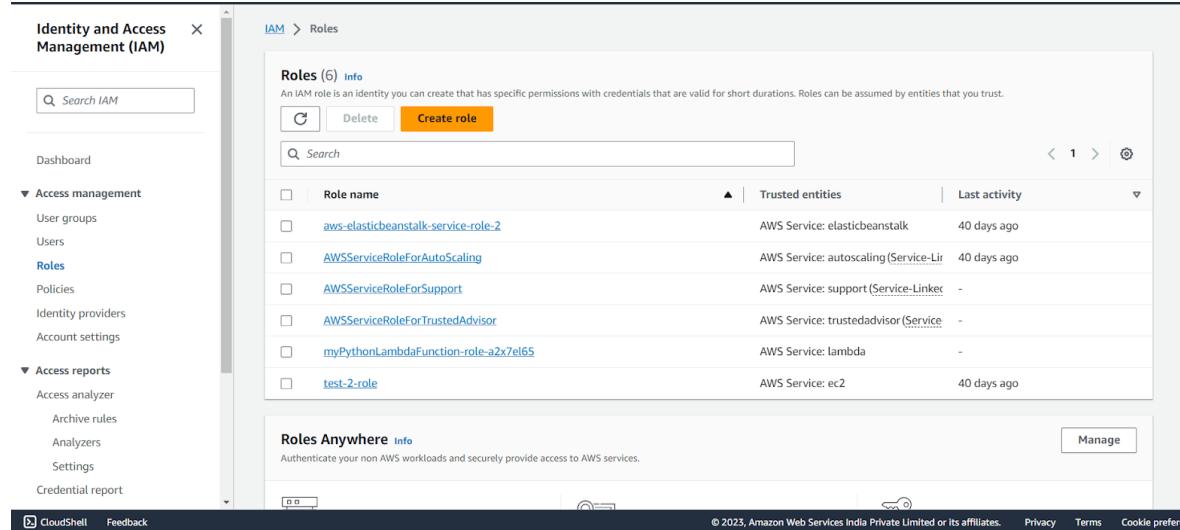
At the bottom, there's a "Code properties" section with an Info link, CloudShell, Feedback, and a copyright notice: © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences.

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

## Experiment No 12

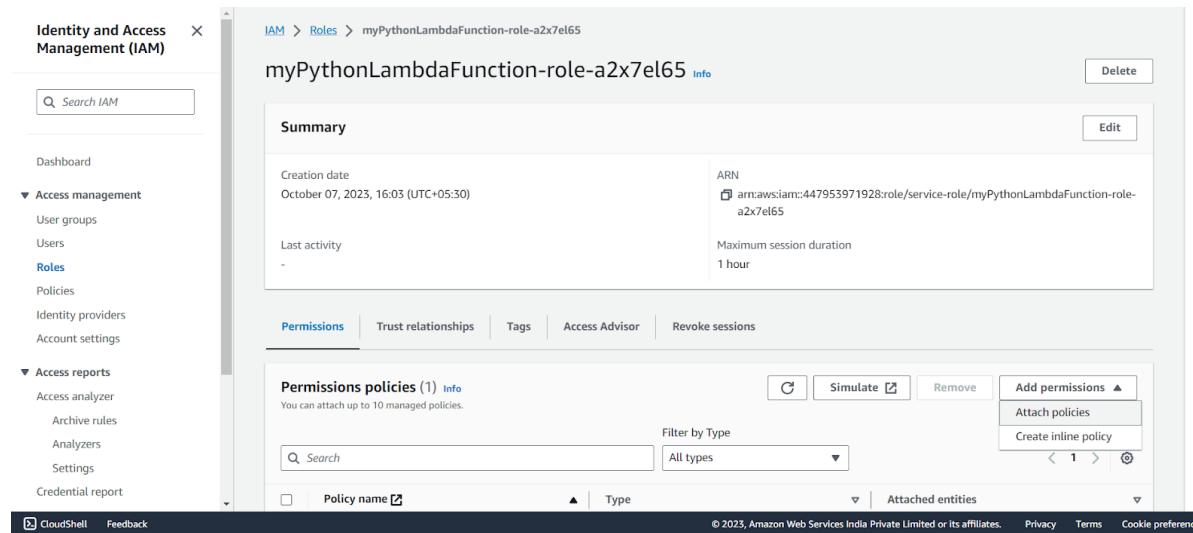
Vedant Sanap  
D15A 48  
Batch C

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



The screenshot shows the AWS IAM Roles page. On the left, there's a navigation sidebar with options like Dashboard, Access management, Policies, and Access reports. The main area displays a table of roles with columns for Role name, Trusted entities, and Last activity. The roles listed are: aws-elasticbeanstalk-service-role-2, AWSServiceRoleForAutoScaling, AWSServiceRoleForSupport, AWSServiceRoleForTrustedAdvisor, myPythonLambdaFunction-role-a2x7el65, and test-2-role. The 'myPythonLambdaFunction-role-a2x7el65' role was selected.

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



The screenshot shows the detailed view of the 'myPythonLambdaFunction-role-a2x7el65' role. The left sidebar is identical to the previous screenshot. The main area has tabs for Summary, Permissions, Trust relationships, Tags, Access Advisor, and Revoke sessions. The Permissions tab is active, showing one managed policy attached: 'arn:aws:iam::447953971928:role/service-role/myPythonLambdaFunction-role-a2x7el65'. Below this, there are buttons for Add permissions (with options for Attach policies or Create inline policy), Simulate, Remove, and a search bar for Policy name.

S3-ReadOnly

The screenshot shows the 'Add permissions' dialog in the AWS IAM console. The path is IAM > Roles > myPythonLambdaFunction-role-a2x7el65 > Add permissions. The title is 'Attach policy to myPythonLambdaFunction-role-a2x7el65'. The 'Current permissions policies (1)' section is expanded, showing one policy: 'AmazonS3ReadOnlyAccess'. The 'Other permissions policies (882)' section is also visible, with a search bar for 'S3read' and a table showing 'AmazonS3ReadOnlyAccess' as an AWS managed policy. Buttons for 'Cancel' and 'Add permissions' are at the bottom.

## CloudWatchFull

The screenshot shows the 'Add permissions' dialog in the AWS IAM console. The path is IAM > Roles > myPythonLambdaFunction-role-a2x7el65 > Add permissions. The title is 'Attach policy to myPythonLambdaFunction-role-a2x7el65'. The 'Current permissions policies (2)' section is expanded, showing two policies: 'CloudWatchFullAccess' and 'CloudWatchFullAccessV2'. The 'Other permissions policies (881)' section is also visible, with a search bar for 'cloudwatchfull' and a table showing 'CloudWatchFullAccess' and 'CloudWatchFullAccessV2' as AWS managed policies. Buttons for 'Cancel' and 'Add permissions' are at the bottom.

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

The screenshot shows the 'Permissions' tab in the AWS IAM console for the 'myPythonLambdaFunction-role-a2x7el65' role. A green success message box is displayed: 'Policy was successfully attached to role.' It shows the last activity and maximum session duration (1 hour). The 'Permissions' tab is selected, showing three attached policies: 'AmazonS3ReadOnlyAccess', 'AWSLambdaBasicExecutionRole-c4946a...', and 'CloudWatchFullAccess'. A 'Permissions boundary (not set)' section is also present. Buttons for 'Simulate', 'Remove', and 'Add permissions' are available.

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

Lambda > Functions > Create function

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

Author from scratch  
Start with a simple Hello World example.

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

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

**Basic information**

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

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

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

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

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

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

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

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

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

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

▼ Change default execution role

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

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

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

► Advanced settings

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

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

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

Step 4: The function is up and running.

The screenshot shows the AWS Lambda Function Overview page for a function named 'AdvDevops-ex12'. At the top, a green banner indicates 'Successfully created the function AdvDevops-ex12. You can now change its code and configuration. To invoke your function with a test event, choose "Test".' Below the banner, the function name 'AdvDevops-ex12' is displayed along with a 'Layers' section showing '(0)'. There are buttons for '+ Add trigger' and '+ Add destination'. On the right, there's a 'Description' field with a '-' sign, 'Last modified' (6 seconds ago), 'Function ARN' (arn:aws:lambda:ap-south-1:447953971928:function:AdvDe vops-ex12), and a 'Function URL' link. At the bottom, navigation tabs include 'Code' (selected), 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. A footer bar at the bottom includes 'CloudShell', 'Feedback', and copyright information: '© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences'.

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

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

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

The editor interface includes tabs for 'Environment', 'lambda\_function', and 'Environment Var'. At the bottom, it shows '18.5 Python Spaces: 4' and the standard AWS footer with 'CloudShell', 'Feedback', and copyright information.

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

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

The "Code" tab is selected. The code editor shows the following Python code:

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

A modal window titled "Configure test event" is open. It contains the following fields:

- Test event action:** A radio button group where "Create new event" is selected.
- Event name:** The input field contains "test".
- Event sharing settings:** A radio button group where "Private" is selected. A note below states: "This event is only available in the Lambda console and to the event creator. You can configure a total of 10." with a "Learn more" link.
- Template - optional:** A dropdown menu currently set to "s3-put".
- Event JSON:** A text area containing the following JSON template:

```
{"s3":>{>>> "Bucket": "my-new-bucket", "Key": "my-new-key", "Type": "ObjectCreated:Put", "Event": "ObjectCreated:Put"}}
```
- Buttons:** "Format JSON", "Cancel", "Invoke", and "Save" (highlighted in orange).

And Save it.

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

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

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

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

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

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

The screenshot shows the 'Event notifications' section of the bucket properties. It indicates 'No event notifications' and provides a 'Create event notification' button. Below this, there is an 'Amazon EventBridge' section with a 'Send notifications to Amazon EventBridge for all events in this bucket' toggle switch set to 'Off'.

Click on Create Event Notification.

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

The screenshot shows the 'General configuration' section with the 'Event name' field set to 'S3putrequest'. It also shows the 'Event types' section where the 'Put' checkbox is selected under 'Object creation'.

**General configuration**

Event name: S3putrequest  
Event name can contain up to 255 characters.

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

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

**Event types**

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

**Object creation**

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

CloudShell Feedback © 2023, Amazon Web Services India Private Limited

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

The screenshot shows the 'Destination' section with the 'Lambda function' option selected. The 'Specify Lambda function' section has 'Choose from your Lambda functions' selected. The 'Lambda function' dropdown contains 'AdvDevops-ex12'.

**Destination**

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

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

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

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

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

**Specify Lambda function**

**Choose from your Lambda functions**

**Enter Lambda function ARN**

**Lambda function**  
AdvDevops-ex12

Cancel **Save changes**

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

The screenshot shows the AWS Lambda Functions overview for the function 'AdvDevops-ex12'. In the 'Triggers' section, there is one entry for 'S3'. The details pane on the right shows the ARN of the function and its last modified time as '1 minute ago'. The tabs at the bottom include 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'.

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

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

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

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

The screenshot shows the AWS S3 'Upload' interface. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, a search bar containing 'Search', and a keyboard shortcut '[Alt+S]'. Below the navigation is a breadcrumb trail: 'Amazon S3 > Buckets > advopssexp12 > Upload'. The main title is 'Upload' with an 'Info' link. A note below says: 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)'.

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

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

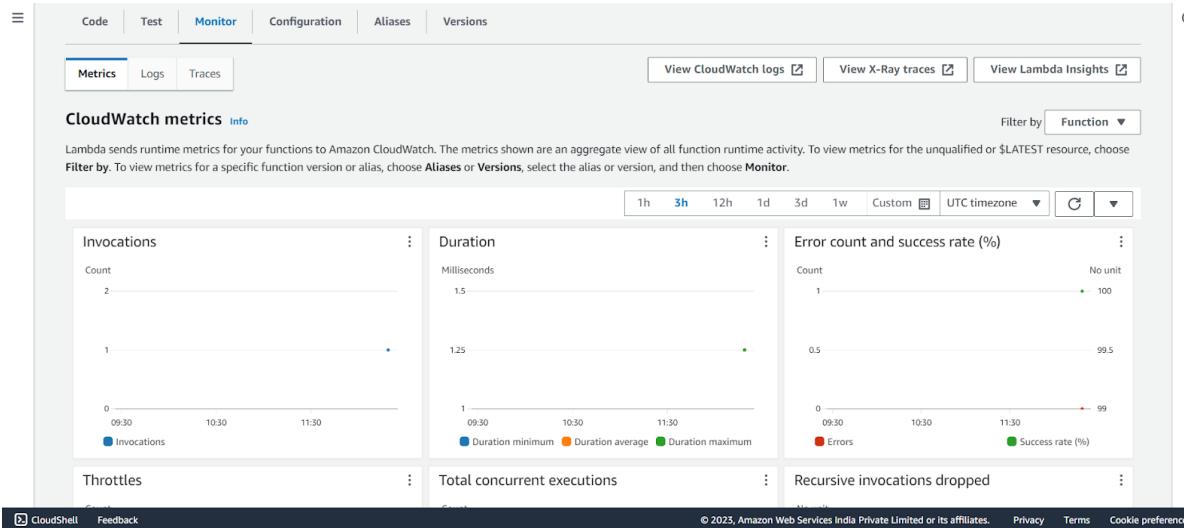
The screenshot shows a JSON editor with the title 'Event JSON' and a 'Format JSON' button. The JSON code is a multi-line string with line numbers 10 through 38. The code defines an event structure with fields like 'principalId', 'requestParameters', 'responseElements', and 's3' which points to a bucket named 'advopssexp12'.

```

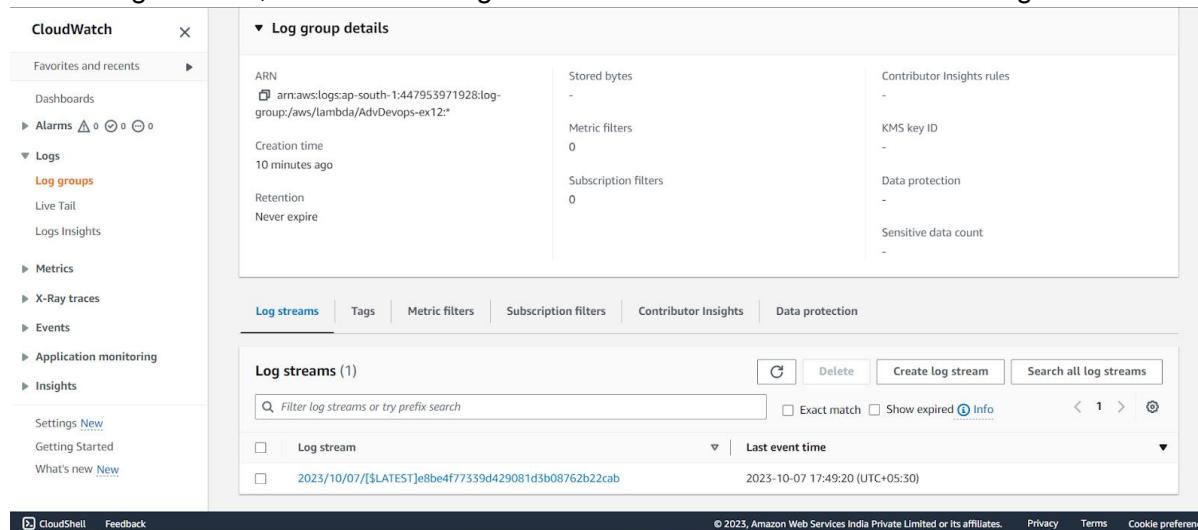
10     "principalId": "EXAMPLE"
11 },
12 "requestParameters": {
13   "sourceIPAddress": "127.0.0.1"
14 },
15 "responseElements": {
16   "x-amz-request-id": "EXAMPLE123456789",
17   "x-amz-id-2": "EXAMPLE123/5678abcdefghijklmabdaisawesome/mnopqrstuvwxyzABCDEFGH"
18 },
19 "s3": {
20   "s3SchemaVersion":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

```

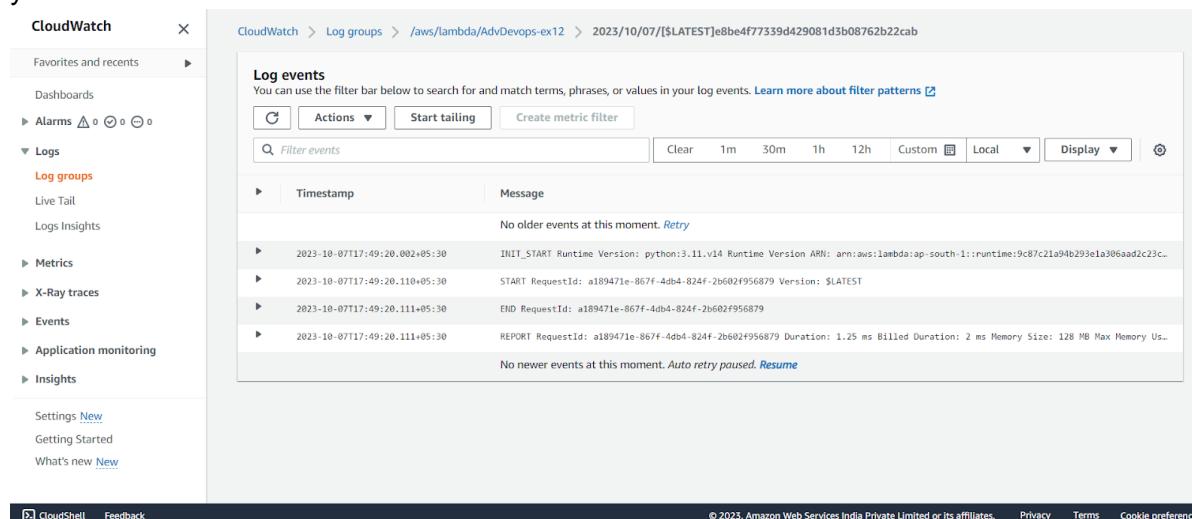
**Step 16:** Go back to your Lambda function , Refresh it and check the Monitor tab.



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



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



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

Assignment 5

- Q1) Use S3 bucket to host video streaming using AWS S3 bucket  
- To host video streaming, follow these steps:
- 1) Creating S3 bucket:  
    - login to AWS management console  
    - navigate to S3 & create new bucket. Give bucket a unique name & choose a region.
  - 2) Enable static website hosting:  
    - After creating bucket, go to bucket properties tab under static website hosting, enable the option & provide index document.
  - 3) Set Bucket policy for public access  
    - To make your videos publicly accessible go to permission tab.  
    - Set a bucket policy to allow public access to files.
  - 4) Upload your video files (mp4, webm, etc) to S3 bucket using AWS console or AWS CLI.
  - 5) USE HLS for adaptive bitrate streaming. Convert adaptive: HLS format.
  - 6) Enable CORS config if you are using JS language for frontend to access videos, configure on S3 bucket.
  - 7) To improve video streaming performance, especially for global users set up Amazon CloudFront.  
    - Create new cloud front distribution. Select S3 bucket as origin, enable caching for smooth playback.
  - 8) Use video player to embed video on your site.

7. Amazon Fi

Q) Visit S3 website URL from S3 buckets or Cloud front to view eg. video streaming. Similarly

Q2) Discuss BMW and Hofstar using AWS

→ Both BMW & Hofstar have used AWS to scale their operations, chance customer experience & innovation in respective industries.

① BMW Case Study using AWS

- 1) BMW uses AWS to build a connected car platform. The platform collects & process data from millions of vehicles data from millions of vehicles in real time enables services like remote diagnosis, navigation, etc.
- 2) Cars drove a IoT device & vehicle provides real time vehicle insights & maintains alerts enhancing both safety & user experience.

3) With Services like Amazon Sage, NLP, BMW is leveraging AWS to provide vehicle production process & improve in cost experiences for user.

② Hofstar Case Study using AWS:

- 1) Scalability during high traffic events like SOL, Hofstar with uses of unpredicted user traffic spikes up millions of concurrent user streams from service.
- 2) Hofstar relies on Amazon Cloud front, global content delivery network to distribute video contents to viewers at high speeds. Thus ensuring a smooth video during high demand period.

Q3) Why Kubernetes? Advantages & disadvantages of Kubernetes  
 Explain how it adds value to DevOps.

→ 1) Advantages of Kubernetes:

- ① Portability: Works across various environments.
- ② Scalability: Automatically scales app based on demand.
- ③ High availability: Ensures uptime with self-healing & load balancing.
- ④ Automated update: Manages rollbacks & rollbacks with zero downtime.
- ⑤ Efficient Resource Management: Optimizes CPU & memory usage.

2) Disadvantages of Kubernetes:

- ① Steep learning curve: Requires exposure to manage effectively.
- ② Resource intensive: High operation & high infrastructure cost.
- ③ Complexity: Setup & management can be challenging.
- ④ Security config: Needs config setup to avoid vulnerabilities.
- ⑤ Operational overhead: Managing clusters & scaling requires dedicated resources.

3) Adidas uses Kubernetes to modernize its infrastructure & accelerate its digital transformation journey.

- ① Scalability: Adidas handles traffic spike during events with automatic resource scaling via Kubernetes.

- (2) Global management: ~~keeps track of standard usage~~  
Adequate deployment for constituents' global performance
- (3) Agile deployment: Releases updates faster  
Deploying microservices independently within kill cycles

Q4 | What is Nagios & explain how Nagios can be used in E-Services?

- It is a tool that widely used to monitor IT infrastructure. It helps the status of network devices, servers, applications & services in real-time.
- 2) It is used to monitor large range of network services including HTTP, SMTP, POP3, etc.
- 3) It also sends notifications through emails when issue arises or unresolved.
- 4) Custom plugins: supports variety of plugins extend functionality. One can also custom plugin as per requirements.
- 5) Use of Nagios:

- (1) Service availability: Checks availability of e-Services like websites, databases, payment gateway, etc.
- (2) Performance monitoring: Ensures services don't overload & network function smoothly.
- (3) User experience: Ensures reliable, fast service by monitoring key infra components.

Name : Vedant Sonap  
Class : D15A  
Roll No: 48

66

04  
03

## Assignment - 3

- 1) Create a Rest API with serverless framework.
- 2) Creating REST API with serverless framework is an efficient way to deploy serverless application that can scale automatically without managing servers.
- 3) Serverless framework: A powerful tool that deployment of services and serverless applications across various cloud providers such as AWS, Azure and Google Cloud.  
~~Serverless architecture~~: This design model allows developers to build application without worrying about underlying infrastructure, enabling focus on code and business logic.
- 4) REST API: Representational State Transfer's architecture style for designing network application.

Steps for creating REST API for serverless framework:

Install serverless framework:

You start by installing Serverless framework CLI globally using node package manager (npm). This allow you to manage Serverless applications directly from your terminal.

2) Creating a Node.js serverless project:  
A directory is created for your project where you will initialize a Serverless service (project). The source will have all your lambda functions, config and Cloud resources. Using the command Create you set up a template for the Node.js microservices that will eventually deploy to AWS Lambda.

3) Project Structure:

The project creates essential files handler.js (which contains code for lambda functions) and service.yml

4) Create a REST API Resource.

In the service.yml file you define functions that handle post requests of HTTP.

5) Deploy the Service:

With the 'Sls deploy' command Serverless packages your applications, uploads necessary to AWS and setup the infrastructure.

6) Testing the API: Once deployed you can test API using tools like curl or Postman by making post requests to generated API.

1. Storing data in DynamoDB: To store submitted candidate data you integrate AWS DynamoDB as a DB.

### 1. AWS IAM Permissions:

You need to ensure that Serverless framework is given right permissions to interact with AWS resources like Dynamo DB.

### 2. Monitoring and Maintenance:

After deployment Serverless framework provides source information like deployed endpoints, API key, log streams.

### 2. Case Studies for SonarQube.

~~Creating your own profile in SonarQube for testing project quality. Use SonarQube to analyze your GitHub code. Install SonarLink in your Java IntelliJ IDE and analyze Java code. Analyze Python project with SonarQube.~~

→ SonarQube is an open source platform used for continuous integration of code quality. It detects bugs, code smell and security vulnerabilities in project across various programming languages.

### Profile Creation in SonarQube:

Quality profiles in SonarQube are essential config that define rules applied during code analysis. Each project has a quality profile for every

Supported language with default being Sonar profile comes built in for all languages. New profiles can be created by copying or extending existing ones. Copying creates an independent profile, while extending existing ones. Copying an independent profile, while extending inherits from parent profile and reflects its changes automatically. SonarQube allows for comparison of two profiles to check for differences in activated rules and uses can track via event log.

### e) Using Sonarcloud to fetch code:

SonarQube is cloud-based counterpart of Sonar that integrates directly with Github, Bitbucket, Azure and Gitlab repos. To get started with cloud product page and connect your Github Next Import repo into your Sonarcloud, where each Github repo becomes a Sonarcloud. Define 'recode' to focus on recent changes and choose between Automatic, Analysis or CI-based analysis. Automatic analysis happens directly in Sonarcloud, while CI based analysis integrates with your build process and the analysis is completed. Results can be viewed in both sonarcloud and github including Security import issue.

## Sonarlint in Java IDE:

Sonarlint is an IDE that performs on-the-fly code analysis as you edit code. It helps developers detect bugs, security vulnerabilities and code smells directly in the development environment such as IntelliJ Idea or Eclipse. This approach ensures immediate feedback on code quality, promoting clean and maintainable code from beginning.

## Analyzing Python Projects with SonarQube:

SonarQube supports Python test coverage reporting but it requires third party tool like coverage.py to generate the coverage part. To enable coverage adjust your build process so that coverage tool runs before Sonar Scanner and ensures report file is saved in different path.

For setup you can use Tox, PyTest and coverage to configure & run test. In your tox.ini include config for pytest and coverage report in XML format. The build process can also be automated using GitHub action which install dependencies, run tests and invoke SonarQube Scan.

## Analyzing Node.js project with SonarQube:

For Node.js project, SonarQube can analyze JavaScript and TypeScript code. Similar to Python script, you can configure SonarScanner to scan

the code against Industry Standard rule book practice, flagging issues related to vulnerabilities, bugs & performance optimization.

Q3) At a large organization, your centralized ops team may get repetitive infra requests. You can use Terraform to build a self-service infrastructure.

→ Terraform's self-service infrastructure provides a powerful use case in large organizations.

i) Self-service infra: By using Terraform modules, you can create reusable and standardized config. Module creation in Terraform, mainly variable.tf and output.tf.

Also after module creation, its standardization is equally important.

(ii) Enabling self-service for product teams: Create a self-service or vocation control access and provide products teams and their important RBAC for preventing unauthorized access.

(iii) Automate Infrastructure Requests via ticketing System:

Integrate Terraform Cloud or Terraform Enterprise Connect Terraform with the ticketing system, automate approval workflows and monitor & log requests.

paces  
Workshop Setup for environment Segregation ?  
To manage different environments, Terraform  
workspaces were set up. This ensured that  
teams could deploy the same infrastructure across  
different environments without overlap.

