

# Advance Devops Exp -1A

Niraj S. Kothawade  
D15A - 24

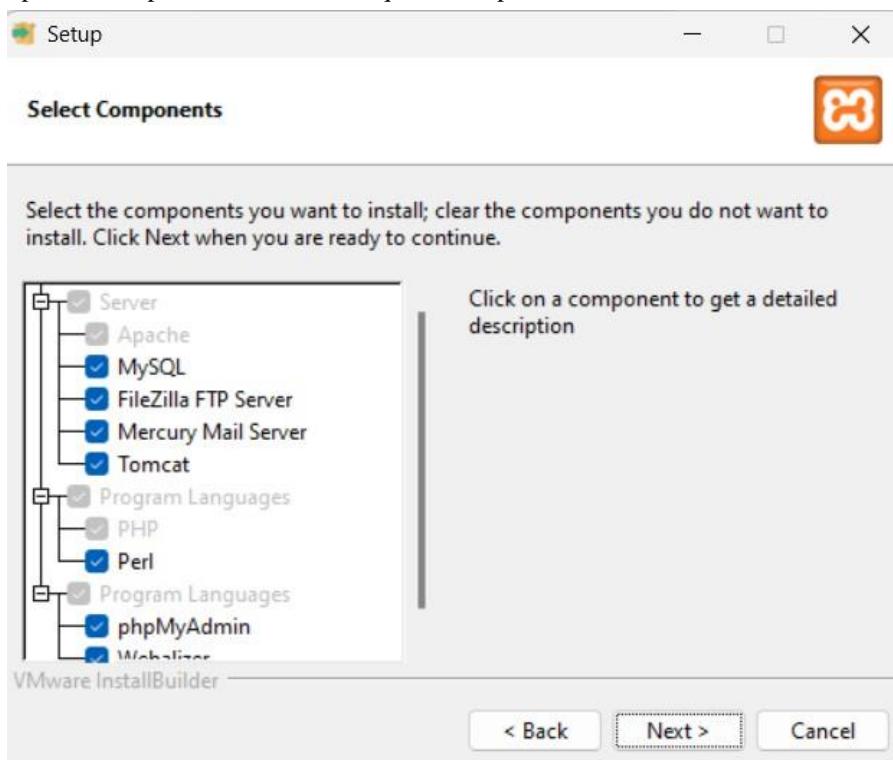
## 1) On local server (XAMPP)

Step 1: Install XAMPP from <https://www.apachefriends.org/>

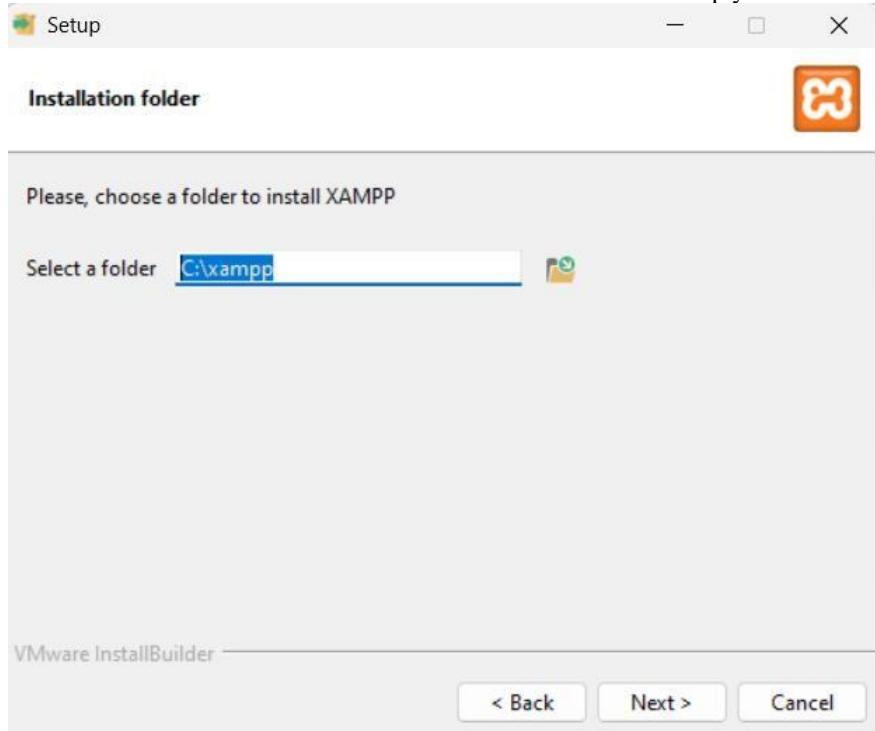
- 1) Select your OS. It will automatically start downloading.



- 2) Open the setup file. Select all the required components and click next



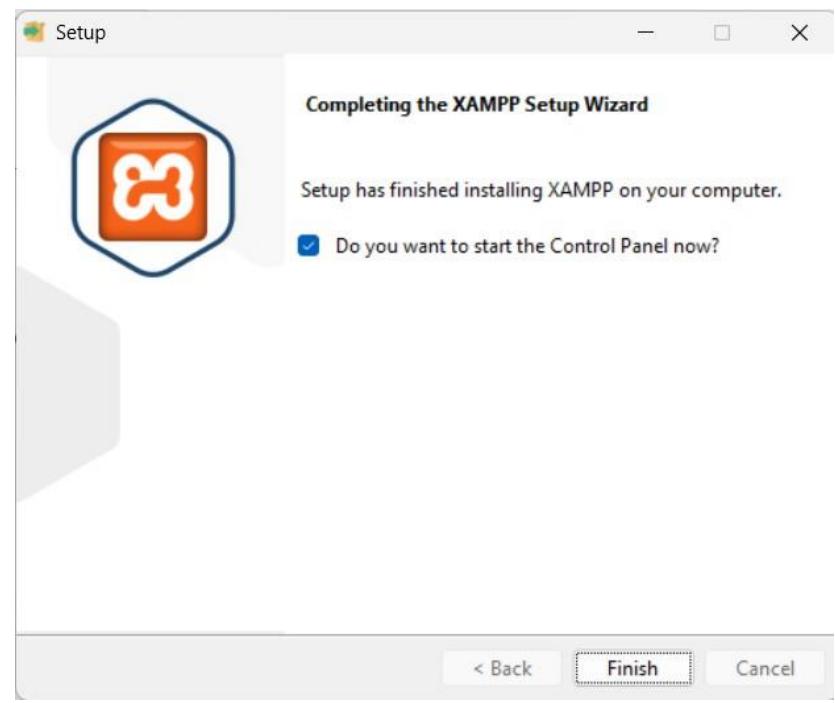
- 3) Choose the folder to install XAMPP in. Make sure the folder is empty. Click next



- 4) Select the language, click next. XAMPP starts to install



5) The installation is complete. Click Finish



6)

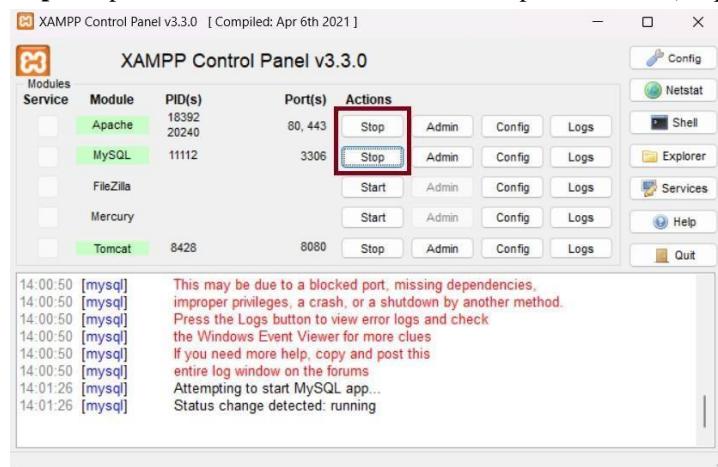
**Step 2:** Setup a file that is to be hosted on the server. Make sure the file has extension .php

test1	06-08-2024 22:48	PHP Source File	1 KB
-------	------------------	-----------------	------

**Step 3:** Go to the directory where XAMPP was installed. Go to **htdocs** folder. Place your folder in this directory.

Name	Date modified	Type	Size
dashboard	07-08-2024 11:47 PM	File folder	
img	07-08-2024 11:47 PM	File folder	
webalizer	07-08-2024 11:47 PM	File folder	
xampp	07-08-2024 11:47 PM	File folder	
applications	15-06-2022 09:37 PM	Chrome HTML Do...	
bitnami	15-06-2022 09:37 PM	CSS Source File	
favicon	16-07-2015 09:02 PM	ICO File	
file1	07-08-2024 11:54 PM	PHP Source File	
index	16-07-2015 09:02 PM	PHP Source File	

**Step 4:** Open XAMPP Control Panel, start the Apache service (Required) and mySQL service (if needed)



**Step 5:** Open your web browser. Type localhost/YOUR\_FILENAME.php. This will open your website on your browser.

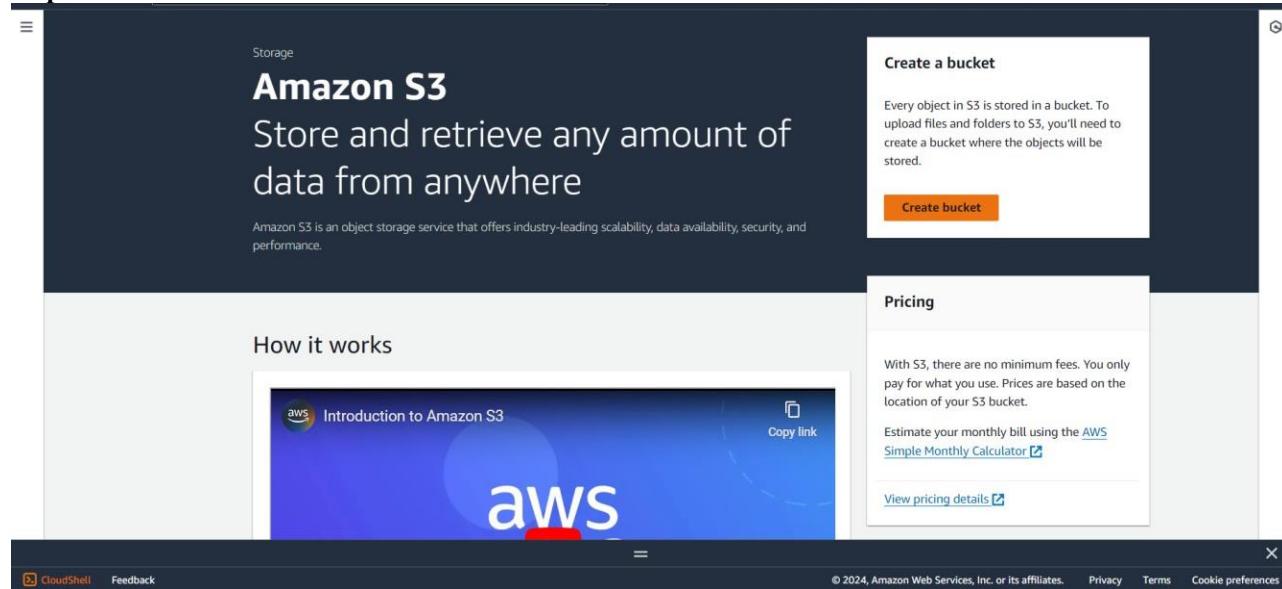


## 2) AWS S3

**Step 1:** Login to your AWS account. Go to services and open S3.

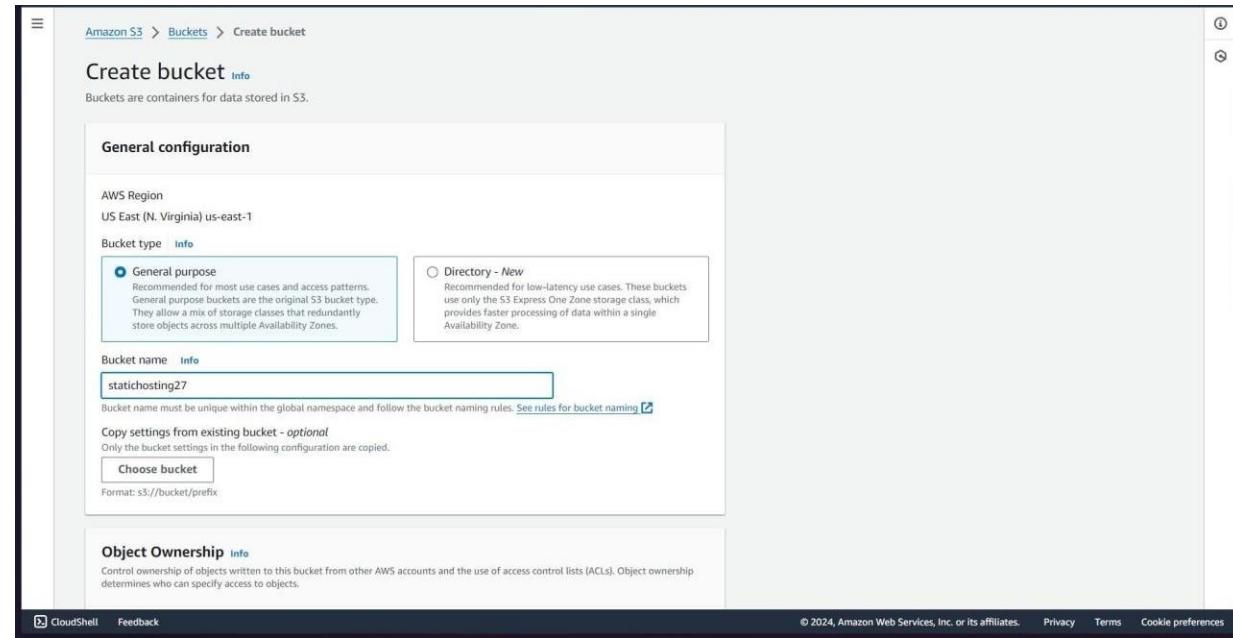
A screenshot of the AWS Management Console Services menu. On the left, there's a sidebar with sections like 'Recently visited', 'Favorites', and 'All services'. Under 'All services', a long list of AWS services is shown with icons. To the right, a modal window titled 'Recently visited' is open, listing services with their descriptions. The 'S3' service is highlighted with a red box. Other listed services include IAM, IAM Identity Center, Resource Access Manager, Cloud9, EC2, and others.

## Step 2: Click on Create Bucket



The screenshot shows the Amazon S3 homepage under the Storage section. The main heading is "Amazon S3" with the subtext "Store and retrieve any amount of data from anywhere". Below this, a description states: "Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance." To the right, there is a call-to-action box titled "Create a bucket" with the subtext: "Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored." At the bottom left, there is a "How it works" section featuring a video thumbnail titled "Introduction to Amazon S3". On the right side, there is a "Pricing" section with the subtext: "With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket." It includes a link to the "AWS Simple Monthly Calculator" and a "View pricing details" link. The footer contains links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

## Step 3: Give a name to your bucket, keeping other options default, scroll down and click on Create Bucket



The screenshot shows the "Create bucket" configuration page. The top navigation bar shows "Amazon S3 > Buckets > Create bucket". The main title is "Create bucket" with an "Info" link. A note below says: "Buckets are containers for data stored in S3." The "General configuration" section is expanded, showing:

- AWS Region:** US East (N. Virginia) us-east-1
- Bucket type:** [Info](#)
  - General purpose: Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.
  - Directory - New: Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.
- Bucket name:** [Info](#) statichosting27
  - Note: 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

The "Object Ownership" section is also visible at the bottom, with the note: "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." The footer contains links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

**Step 4:** Click on the name of your bucket and goto Properties

The screenshot shows the AWS S3 Buckets page. At the top, a green banner indicates that a bucket named "statichosting09" has been successfully created. Below the banner, there's an "Account snapshot" section with a link to "View details". A "General purpose buckets" tab is selected, showing one bucket named "statichosting09" in the list. The bucket details include its name, region (US East (N. Virginia) us-east-1), and creation date (August 8, 2024). There are also links to "View analyzer for us-east-1" and "Create bucket".

Name	AWS Region	IAM Access Analyzer	Creation date
statichosting09	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	August 8, 2024, 22:26:28 (UTC+05:30)

**Step 5:** Scroll down till you find Static website hosting, click on edit

**Step 6:** Enable static website hosting, in Index document, write the name of your document and in error document, give name as 404.html. Save your changes.

**Step 7:** Go to Objects tab and click on upload file.

The image consists of three vertically stacked screenshots of the AWS S3 console.

**Screenshot 1: Edit static website hosting**

This screenshot shows the "Edit static website hosting" configuration page for the bucket "statichosting09".

- Static website hosting:** The "Enable" radio button is selected.
- Hosting type:** The "Host a static website" radio button is selected, with a note explaining that customers must make all content publicly readable using S3 Block Public Access.
- Index document:** The input field contains "index.html".

**Screenshot 2: Bucket details**

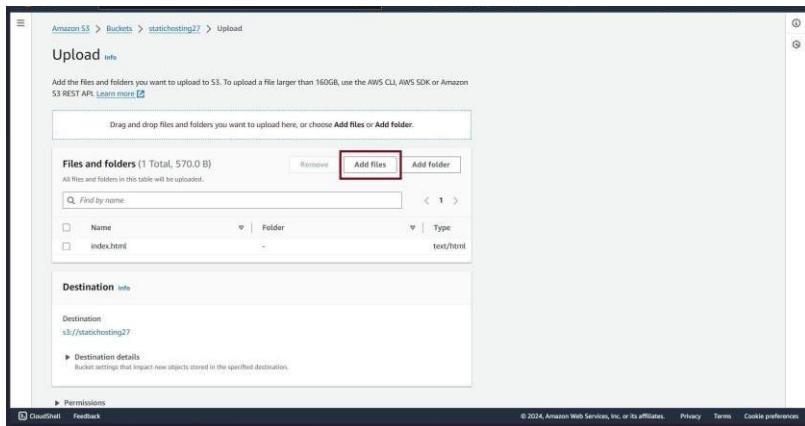
This screenshot shows the main bucket details page for "statichosting09".

- Objects:** The "Objects" tab is selected, showing 0 objects.
- Actions:** A prominent orange "Upload" button is located at the top right of the object list.

**Screenshot 3: Desktop Taskbar**

This screenshot shows the Windows taskbar at the bottom of the screen, displaying various pinned icons and the current date and time (10:27 PM, 08-08-2024).

**Step 8:** Click on Add files. Add all the files you want to upload. Then scroll down and click on Upload



The screenshot shows the 'Upload' interface for an S3 bucket named 'statichosting27'. The 'Files and folders' section displays one item: 'index.html' (Total, 570.0 B). The 'Add files' button is highlighted with a red box. The 'Destination' section shows the path 's3://statichosting27'. The 'Permissions' section is partially visible at the bottom.

Amazon S3 > Buckets > statichosting27 > Upload

Upload info

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

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

Files and folders (1 Total, 570.0 B)

All files and folders in this table will be uploaded.

Find by name: < 1 >

Add files Add folder

Name	Folder	Type
index.html	-	text/html

Destination info

Destination  
s3://statichosting27

Destination details

Bucket settings that impact new objects stored in the specified destination.

Permissions

CloudWatch Feedback

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

**Step 9:** This will take you to the Objects screen. Switch to Properties, scroll down to Static web hosting. There you would find the link (Bucket website endpoint) to your website.

The screenshot shows the 'Static website hosting' configuration for a bucket. It includes fields for 'Static website hosting' (Enabled), 'Hosting type' (Bucket hosting), and 'Bucket website endpoint' (http://statichosting09.s3-website-us-east-1.amazonaws.com). A note states that when configured, the website is available at the AWS Region-specific website endpoint of the bucket.

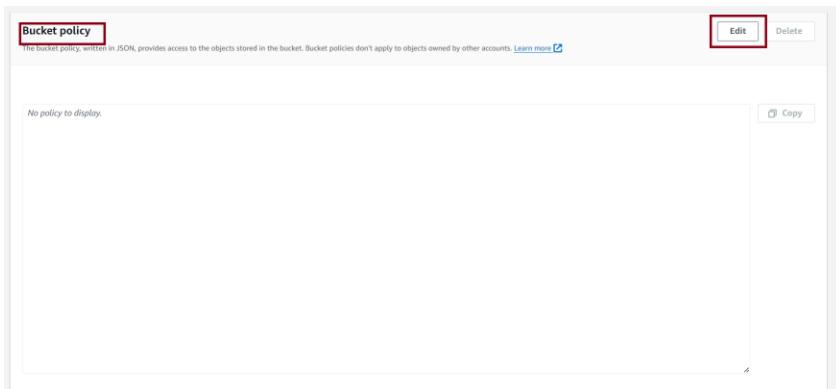
**Step 10:** Open the link. It will show a 403 forbidden error screen as the contents of the bucket are not available for the public users. To change this, go to Permissions tab, go to Block public access and click on edit

The screenshot shows a 403 Forbidden error page. The URL is statichosting27.s3-website-us-east-1.amazonaws.com. The error message is "An Error Occurred While Attempting to Retrieve a Custom Error Document". Below it, there is a list of error details: Code: AccessDenied, Message: Access Denied, RequestId: 8TQ4EGP4TK06MVPB, HostId: hF+ToadQUoCuDM8H+iFRsXda28TGp+xikYbjb4CICS/t+3it4ihA/tvgA1Xrlxo+JL5AhkT6hJs=.

**Step 11:** Uncheck the Block all public access checkbox and click on save changes

The screenshot shows the 'Edit Block public access (bucket settings)' page. The 'Block all public access' checkbox is currently checked. Below it, there are four additional settings: 'Block public access to buckets and objects granted through new access control lists (ACLs)', 'Block public access to buckets and objects granted through any access control lists (ACLs)', 'Block public access to buckets and objects granted through new public bucket or access point policies', and 'Block public cross-account access to buckets and objects through any public bucket or access point policies'. At the bottom, there are 'Cancel' and 'Save changes' buttons.

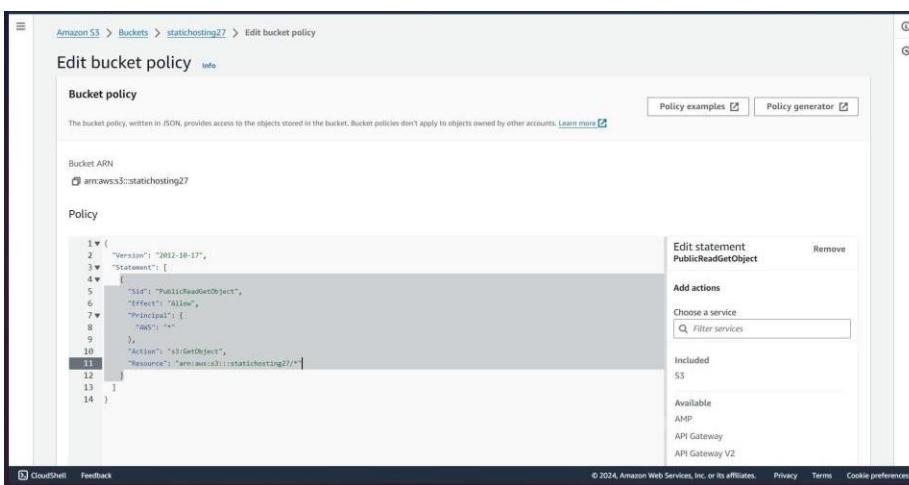
## Step 12: Scroll down to bucket policy and click edit



## Step 13:

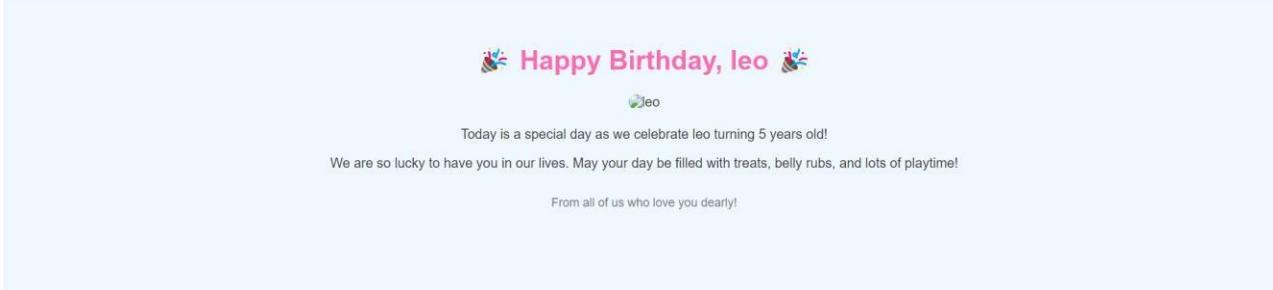
```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "PublicReadGetObject",  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "*"  
      },  
      "Action": "s3:GetObject",  
      "Resource": "arn:aws:s3:::YOUR-BUCKET-NAME-HERE/*"  
    }  
  ]  
}
```

Paste this code snippet in the policy textarea. Replace YOUR-BUCKET-NAME-HERE with the name you have given to your bucket. Save the changes.



**Step 14:** Now reload the website. You can see your website

---



# ADVANCE DEVOPS EXPERIMENT NO.1

Niraj S. Kothawade

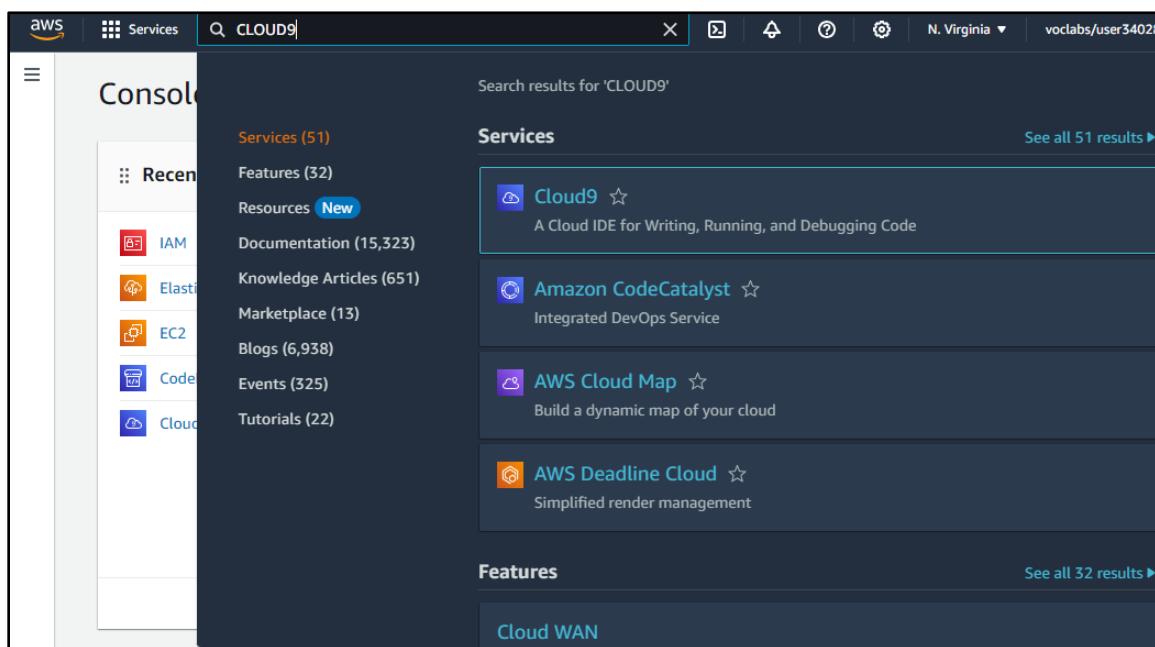
D15A - 24

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

## Cloud9

### Steps:

1. Open your AWS account and search for Cloud9 service inside Developer tools. Create a new Cloud9 environment by filling in the required details. Make sure you use an EC2 instance to create your environment.



The screenshot shows the AWS Cloud9 landing page. At the top, there's a navigation bar with the AWS logo, a 'Services' dropdown, a search bar, and account information ('N. Virginia' and 'voclabs/user3402848'). Below the header, the page title 'AWS Cloud9' is displayed in large, bold letters, followed by the subtitle 'A cloud IDE for writing, running, and debugging code'. A descriptive text block explains that AWS Cloud9 allows you to write, run, and debug your code with just a browser, providing immediate access to a rich code editor, integrated debugger, and built-in terminal with preconfigured AWS CLI. A prominent orange 'Create environment' button is located on the right side of the main content area.

The screenshot shows the 'Create environment' form. The first section is titled 'Details'.

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

**Description - optional**  
Limit 200 characters.

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

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

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

## New EC2 instance

### Instance type [Info](#)

The memory and CPU of the EC2 instance that will be created for Cloud9 to run on.

t2.micro (1 GiB RAM + 1 vCPU)

Free-tier eligible. Ideal for educational users and exploration.

t3.small (2 GiB RAM + 2 vCPU)

Recommended for small web projects.

m5.large (8 GiB RAM + 2 vCPU)

Recommended for production and most general-purpose development.

Additional instance types

Explore additional instances to fit your need.

### Platform [Info](#)

This will be installed on your EC2 instance. We recommend Amazon Linux 2023.

Amazon Linux 2023



### Timeout

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

30 minutes



## Network settings [Info](#)

### Connection

How your environment is accessed.

AWS Systems Manager (SSM)

Accesses environment via SSM without opening inbound ports (no ingress).

Secure Shell (SSH)

Accesses environment directly via SSH, opens inbound ports.

### ► VPC settings [Info](#)

### ► Tags - optional [Info](#)

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.



The following IAM resources will be created in your account

- **AWSServiceRoleForAWSCloud9** - AWS Cloud9 creates a service-linked role for you. This allows AWS Cloud9 to call other AWS services on your behalf. You can delete the role from the AWS IAM console once you no longer have any AWS Cloud9 environments. [Learn more](#)

Successfully created Test123. To get the most out of your environment, see [Best practices for using AWS Cloud9](#)

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

AWS Cloud9 > Environments

Environments (1)

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

Q iam

Search results for 'iam'

Services (11)

Features (24)

Resources [New](#)

Documentation (59,458)

Knowledge Articles (467)

Marketplace (856)

Blogs (1,843)

Events (12)

Tutorials (1)

**Services**

[See all 11 results ▾](#)

IAM <a href="#">☆</a> Manage access to AWS resources
IAM Identity Center <a href="#">☆</a> Manage workforce user access to multiple AWS accounts and cloud applications
Resource Access Manager <a href="#">☆</a> Share AWS resources with other accounts or AWS Organizations

Identity and Access Management (IAM) X

IAM > Users

Users (0) [Info](#)

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

User name	Path	Group	Last activity	MFA	Password age
No resources to display					

User name

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ \_ - (hyphen)

**Provide user access to the AWS Management Console - optional**  
If you're providing console access to a person, it's a best practice [IAM Identity Center](#) to manage their access.

Console password

Autogenerated password  
You can view the password after you create the user.

Custom password  
Enter a custom password for the user.

- Must be at least 8 characters long
- Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), @ # \$ % ^ & \* () \_ + - (hyphen) = [ ] { } | '

Show password

**Users must create a new password at next sign-in - Recommended**  
Users automatically get the [IAMUserChangePassword](#) policy to allow them to change their own password.

### User details

User name	Console password type	Require password reset
niraj	Custom password	Yes

### Permissions summary

Name	Type	Used as
<a href="#">IAMUserChangePassword</a>	AWS managed	Permissions policy

**Tags - optional**  
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

Add user to an existing group or create a new one. Using groups is a best practice way to manage users' permissions by job function. [Learn more](#)

[more](#)

## Permissions options

### Add user to group

Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

### Copy permissions

Copy all group memberships, attached managed policies, and inline policies from an existing user.

### Attach policies directly

Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.



### Get started with groups

Create a group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. [Learn more](#)

[Create group](#)

User name  
niraj

Console password type  
None

Require password reset  
No

## Permissions summary

< 1 >

Name	Type	Used as
No resources		

## Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

[Cancel](#)

[Previous](#)

[Create user](#)

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

Here the environment has been successfully created

The screenshot shows the AWS Cloud9 Environments page. At the top, there is a blue header bar with the text "For capabilities similar to AWS Cloud9, explore AWS Toolkits in your own IDE and AWS CloudShell in the AWS Management Console. Learn more" and a close button. Below the header, the page title is "AWS Cloud9 > Environments". The main content area has a table titled "Environments (1)". The table columns are: Name, Cloud9 IDE, Environment type, Connection, Permission, and Owner ARN. There is one row in the table with the following values: Name - Test123, Cloud9 IDE - Open, Environment type - EC2 instance, Connection - Secure Shell (SSH), Permission - Owner, and Owner ARN - arn:aws:sts::554378108602:assumed-role/voclabs/user3402848=PATANKAR\_ARYAN\_ANIL. The "Create environment" button is located at the top right of the table.

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

**2.**We have successfully set up and launched our Cloud9 environment. Over here, we can build and develop programs as per our desire. We are also allowed to collaborate with multiple other users and access shared resources.

The screenshot shows the AWS Cloud9 IDE interface. At the top, there's a navigation bar with File, Edit, Find, View, Go, Run, Tools, Window, Support, Preview, and Run buttons. A Share icon is also present. Below the navigation bar is a file explorer window titled "Welcome" showing a directory structure with "Test123 - /home/e", "IPLab-02" (containing "download (1).jfif", "download (1).png", "download (2).jfif", "download (2).png", "download (3).png", "download (4).png", "download.png", "index.html", "introduction.mp3", "promotional-video.m", "style.css", and "README.md"), and "README.md". To the right of the file explorer is a "Developer Tools" panel. The main workspace displays the title "AWS Cloud9" and the sub-header "Welcome to your development environment". Below this, a "Toolkit for AWS Cloud9" section provides information about the toolkit, and a "Getting started" sidebar offers options like "Create File", "Upload Files...", and "Clone from GitHub". At the bottom of the workspace, there are two tabs: "bash - ip-172-31-11-129.x" and "Immediate". On the far right, there's a "Preview" tab showing a browser window with the URL "/IPLab-02/index.html". The browser preview shows a basic HTML page with an Amazon logo and a "Our Services" section. The terminal at the bottom shows the command "voclabs:~/environment \$".

Further, we are supposed to login from another browser using the credentials of the IAM user, to access the shared cloud9 environment with us. These steps could not be completed because Cloud9 services have been disrupted and there is no access to the IAM user from the remote login.

## Advance DevOps Exp - 2

Niraj S. Kothawade  
D15A -22

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

The image shows two screenshots of the AWS Management Console. The top screenshot displays the search results for 'elastic beanstalk'. The search bar at the top contains the query 'elastic beanstalk'. Below it, the 'Services' section lists 'Elastic Beanstalk' under 'Features (48)', described as 'Run and Manage Web Apps'. The bottom screenshot shows the 'Amazon Elastic Beanstalk' landing page. It features the title 'Amazon Elastic Beanstalk' and the subtitle 'End-to-end web application management.' A 'Get started' button is prominently displayed, along with a description of how Elastic Beanstalk handles deployment and scaling. Another 'Create application' button is located on the right side of the landing page.

AWS Services Search [Alt+S] Stockholm Niraj017

Configure environment 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 environment** Info

**Environment tier** Info  
Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications.

**Web server environment**  
Run a website, web application, or web API that serves HTTP requests. [Learn more](#)

**Worker environment**  
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. [Learn more](#)

**Application information** Info

Application name  Maximum length of 100 characters.

▶ Application tags (optional)

**Environment information** Info  
Choose the name, subdomain and description for your environment. These cannot be changed later.

https://eu-north-1.console.aws.amazon.com/console/home?region=eu-north-1 © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS Services Search [Alt+S] Stockholm Niraj017

▶ Application tags (optional)

**Environment information** Info  
Choose the name, subdomain and description for your environment. These cannot be changed later.

Environment name  Must be from 4 to 40 characters in length. The name can contain only letters, numbers, and hyphens. It can't start or end with a hyphen. This name must be unique within a region in your account.

Domain  .eu-north-1.elasticbeanstalk.com

Environment description

AWS Services Search [Alt+S] Stockholm Niraj017

**Platform** Info

Platform type  
 Managed platform Platforms published and maintained by Amazon Elastic Beanstalk. Learn more [\[?\]](#)  
 Custom platform Platforms created and owned by you. This option is unavailable if you have no platforms.

Platform  
Python

Platform branch  
Python 3.11 running on 64bit Amazon Linux 2023

Platform version  
4.1.3 (Recommended)

AWS Services Search [Alt+S] Stockholm Niraj017

**Application code** Info

Sample application  
  
Existing version Application versions that you have uploaded.  
 Upload your code Upload a source bundle from your computer or copy one from Amazon S3.

**Presets** Info

Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

Configuration presets  
 Single instance (free tier eligible)  
 Single instance (using spot instance)  
 High availability  
 High availability (using spot and on-demand instances)  
 Custom configuration

Cancel **Next**

Screenshot of the AWS IAM search results page. The search term 'iam' is entered in the search bar. The results are categorized into Services and Features.

**Services** (11):

- IAM: Manage access to AWS resources
- IAM Identity Center: Manage workforce user access to multiple AWS accounts and cloud applications
- Resource Access Manager: Share AWS resources with other accounts or AWS Organizations

**Features** (24):

- Groups: IAM feature
- Roles: IAM feature
- Roles Anywhere: IAM feature

Navigation buttons at the bottom: Cancel, Skip to review, Previous, Next.

Screenshot of the AWS IAM Roles management page.

**Roles (2) Info:**

- Role name: AWSServiceRoleForSupport (AWS Service: support)
- Role name: AWSServiceRoleForTrustedAdvisor (AWS Service: trustedadvisor)

**Roles Anywhere Info:**

- Access AWS from your non AWS workloads: Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.
- X.509 Standard: Use your own existing PKI infrastructure or use AWS Certificate Manager Private Certificate Authority to authenticate identities.
- Temporary credentials: Use temporary credentials with ease and benefit from the enhanced security they provide.

Navigation buttons at the bottom: CloudShell, Feedback.

Screenshot of the 'Create role' wizard Step 1: Select trusted entity.

**Step 1: Select trusted entity**

**Step 2: Add permissions**

**Step 3: Name, review, and create**

**Trusted entity type:**

- AWS service: Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- AWS account: Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- Web identity: Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- SAML 2.0 federation: Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy: Create a custom trust policy to enable others to perform actions in this account.

**Use case**  
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

**Service or use case**  
EC2

Choose a use case for the specified service.

**Use case**

- EC2**  
Allows EC2 instances to call AWS services on your behalf.
- EC2 Role for AWS Systems Manager**  
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
- EC2 Spot Fleet Role**  
Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.
- EC2 - Spot Fleet Auto Scaling**  
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.
- EC2 - Spot Fleet Tagging**  
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.
- EC2 - Spot Instances**  
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.
- EC2 - Spot Fleet**  
Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.
- EC2 - Scheduled Instances**  
Allows EC2 Scheduled Instances to manage instances on your behalf.

**Cancel** **Next**

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

**Step 2 Add permissions**

**Step 3 Name, review, and create**

**Permissions policies (3/947) Info**  
Choose one or more policies to attach to your new role.

Filter by Type: All types | 14 matches

Policy name	Type	Description
<input checked="" type="checkbox"/> AWSElasticBeanstalkWebTier	AWS managed	Provide the instances in your web server environment access to upload log files to Amazon S3.
<input checked="" type="checkbox"/> AWSElasticBeanstalkWorkerTier	AWS managed	Provide the instances in your worker environment access to upload log files to Amazon S3, to use Amazon SQS to mon...
<input checked="" type="checkbox"/> AWSElasticBeanstalkMulticontainerDoc...	AWS managed	Provide the instances in your multicontainer Docker environment access to use the Amazon EC2 Container Service to ...
<input type="checkbox"/> AWSElasticBeanstalkEnhancedHealth	AWS managed	AWS Elastic Beanstalk Service policy for Health Monitoring system
<input type="checkbox"/> AWSElasticBeanstalkCustomPlatform...	AWS managed	Provide the instance in your custom platform builder environment permission to launch EC2 instance, create EBS snap...
<input type="checkbox"/> AWSElasticBeanstalkRoleWorkerTier	AWS managed	(Elastic Beanstalk operations role) Allows a worker environment tier to create an Amazon DynamoDB table and an Am...
<input type="checkbox"/> AWSElasticBeanstalkRoleSNS	AWS managed	(Elastic Beanstalk operations role) Allows an environment to enable Amazon SNS topic integration.
<input type="checkbox"/> AWSElasticBeanstalkRoleRDS	AWS managed	(Elastic Beanstalk operations role) Allows an environment to integrate an Amazon RDS instance.
<input type="checkbox"/> AWSElasticBeanstalkRoleECS	AWS managed	(Elastic Beanstalk operations role) Allows a multicontainer Docker environment to manage Amazon ECS clusters.
<input type="checkbox"/> AWSElasticBeanstalkRoleCore	AWS managed	AWSElasticBeanstalkRoleCore (Elastic Beanstalk operations role) Allows core operation of a web service environment.
<input type="checkbox"/> AWSElasticBeanstalkRoleCWL	AWS managed	(Elastic Beanstalk operations role) Allows an environment to manage Amazon CloudWatch Logs log groups.
<input type="checkbox"/> AWSElasticBeanstalkReadOnly	AWS managed	Grants read-only permissions. Explicitly allows operators to gain direct access to retrieve information about resources r...
<input type="checkbox"/> AdministratorAccess-AWSElasticBeanst...	AWS managed	Grants account administrative permissions. Explicitly allows developers and administrators to gain direct access to reso...
<input type="checkbox"/> AWSElasticBeanstalkManagedUpdates...	AWS managed	This policy is for the AWS Elastic Beanstalk service role used to perform managed updates of Elastic Beanstalk environ...

**Set permissions boundary - optional**

**Cancel** **Previous** **Next**

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

Screenshot of the AWS IAM 'Create role' wizard, Step 1: Name, review, and create.

**Role details**

**Role name:** aws-elasticbeanstalk-ec2

**Description:** Allows EC2 instances to call AWS services on your behalf.

**Step 1: Select trusted entities**

**Trust policy:**

```
1 - [ {  
2 -   "Version": "2012-10-17",  
3 -   "Statement": [  
4 -     {  
5 -       "Effect": "Allow",  
6 -       "Action": "sts:AssumeRole"  
7 -     }  
8 -   ]  
9 - }]
```

**Step 2: Add permissions**

**Permissions policy summary:**

Policy name	Type	Attached as
AWS-ElasticBeanstalk-MulticontainerDocker	AWS managed	Permissions policy
AWS-ElasticBeanstalk-WebTier	AWS managed	Permissions policy
AWS-ElasticBeanstalk-WorkerTier	AWS managed	Permissions policy

**Step 3: Add tags**

Add tags - optional Info  
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel Previous Create role

**Role aws-elasticbeanstalk-ec2 created.**

**Roles (3) Info**

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
AWSServiceRoleForSupport	AWS Service: support (Service-Linker)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service)	-

**Roles Anywhere Info**

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

**Access AWS from your non AWS workloads**

Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.

**X.509 Standard**

Use your own existing PKI infrastructure or use [AWS Certificate Manager Private Certificate Authority](#) to authenticate identities.

**Temporary credentials**

Use temporary credentials with ease and benefit from the enhanced security they provide.

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

**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-ec2

**EC2 key pair**

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

Choose a key pair

**EC2 instance profile**

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

aws-elasticbeanstalk-ec2

[View permission details](#)

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

The screenshot displays three sequential views of AWS services, showing the deployment and configuration of an Elastic Beanstalk environment.

**Initial View (Top):** The user is in the Elastic Beanstalk console, navigating through the "Niraj24-env" environment. A message at the top states, "Elastic Beanstalk is launching your environment. This will take a few minutes." The left sidebar shows navigation options for Applications, Environments, and Change history, with "Application: Niraj24" and "Environment: Niraj24-env" expanded. The main panel shows "Environment overview" details like Health (Pending), Environment ID (e-vm4nnzhtmr), Domain (Niraj24-env.eba-q5bn9pg.eu-north-1.elasticbeanstalk.com), and Application name (Niraj24). On the right, the "Platform" section indicates Python 3.11 running on 64bit Amazon Linux 2023/4.1.3, with a "Supported" status. A "Events" tab is visible below.

**Second View (Middle):** The user has switched to the CloudFormation service. The search bar shows "cloudfomation". The left sidebar remains the same. The main panel displays search results for "CloudFormation" and "Application Composer". The "CloudFormation" card is selected, showing its features: StackSets, IaC Generator, Stacks, Exports, and Application Composer. The "Application Composer" card is also shown. The right panel shows the same "Platform" details as the first view.

**Final View (Bottom):** The user is now in the CloudFormation "Stacks" view. The search bar shows "CloudFormation > Stacks". The left sidebar shows "CloudFormation" with "Stacks" selected. The main panel lists a single stack named "awseb-e-vm4nnzhtmr-stack" with a status of "CREATE\_COMPLETE" and a creation time of 2024-08-22 19:04:43 UTC+0530. The right panel shows the same "Platform" details.

**CloudFormation**

**Stacks**

- Stack details
- Drifts
- StackSets
- Exports

**Application Composer** New

IaC generator

**Registry**

- Public extensions
- Activated extensions
- Publisher

**Spotlight**

**Feedback**

**CloudShell** **Feedback**

**CloudFormation** > **Stacks** > awseb-e-vm4nnzhtmr-stack

**awseb-e-vm4nnzhtmr-stack**

**Stack info**

**Overview**

**Stack ID:** arn:aws:cloudformation:eu-north-1:147997146975:stack/awseb-e-vm4nnzhtmr-stack/4a800800-608b-11ef-9672-0a3c26256ef3

**Description:** AWS Elastic Beanstalk environment (Name: 'Niraj24-env' Id: 'e-vm4nnzhtmr')

**Status:** CREATE\_COMPLETE

**Status reason:** -

**Parent stack:** -

**Created time:** 2024-08-22 19:04:43 UTC+0530

**Updated time:** -

**Deleted time:** -

**Drift status:** -

**CloudFormation** > **Application Composer**

**Application Composer**

**Resources**

**Canvas**

**Template**

**Arrange**

**Standard Component AWSEBAutoScalingLaunchConfiguration**

**Standard Component AWSEBInstanceLaunchWaitHandle**

**Standard Component AWSEBEIP**

**Standard Component AWSEBBeanstalkMetadata**

**Standard Component AWSEBInstanceLaunchWaitCondition**

**CloudShell** **Feedback**

**Instances (1) Info**

**Find Instance by attribute or tag (case-sensitive)**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 IP
Niraj24-env	i-0fbaa7e346a3ce3fd	Running	t3.micro	3/3 checks passed	View alarms +	eu-north-1b	ec2-13-48-184-238.eu...	13.48.184.23...

**Select an instance**

Screenshot of the AWS Elastic Beanstalk console showing the environment "Niraj24-env" successfully launched.

The interface includes:

- Left sidebar:** Shows the application "Niraj24" and its environment "Niraj24-env".
- Top bar:** Includes the AWS logo, Services, Search, and user information (Stockholm, Niraj017).
- Environment Overview:** Displays Health (Warning), Environment ID (e-vm4nnzhtmr), Domain (niraj24-env.eba-q5bn9xpg.eu-north-1.elasticbeanstalk.com), Application name (Niraj24), and Platform (Python 3.11 running on 64bit Amazon Linux 2023/4.1.3).
- Platform:** Shows Running version (–) and Platform state (Supported).
- Events tab:** Shows 12 events.
- Bottom navigation:** CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, Cookie preferences.

The browser window below shows the deployed application's landing page:

- Title:** Niraj24-env.eba-q5bn9xpg.eu-north-1.elasticbeanstalk.com
- Content:** A large green "Congratulations" banner with the message: "Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud". Below it says "This environment is launched with Elastic Beanstalk Python Platform".
- Right sidebar:** "What's Next?" section with links:
  - AWS Elastic Beanstalk overview
  - AWS Elastic Beanstalk concepts
  - Deploy a Django Application to AWS Elastic Beanstalk
  - Deploy a Flask Application to AWS Elastic Beanstalk
  - Customizing and Configuring a Python Container
  - Working with Logs

## Code Deployment using CodePipeline:

The screenshot shows the AWS Lambda console interface. At the top, there's a search bar with the text 'CodePipeline'. Below it, the left sidebar is titled 'Elastic Beanstalk' and includes sections for Applications, Environments, Change history, Application: Niraj24, Environment: Niraj24, Configuration, Events, Health, Logs, Monitoring, Alarms, Managed updates, and Tags. The main content area has a title 'Services (1)' and a sub-section 'Services'. It lists 'CodePipeline' with a description 'Release Software using Continuous Delivery'. To the right, there's a large panel for 'CodePipeline' with tabs for Actions, Upload and deploy, and Change version. A modal window titled 'Introducing resource search' is open, explaining how it enables cross-region resource search. The bottom of the screen shows standard AWS navigation links for CloudShell, Feedback, and cookie preferences.

The screenshot shows the AWS CodePipeline console. The left sidebar under 'Developer Tools' has a section for 'CodePipeline' with options for Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), Deploy (CodeDeploy), Pipeline (CodePipeline), and Settings. Under Pipeline, 'Getting started' and 'Pipelines' are listed, with 'Pipelines' being the active tab. The main content area shows the 'Pipelines' page with a title 'Introducing the new V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model. Learn more'. It features a search bar, a toolbar with Create pipeline, Notify, View history, Release change, and Delete pipeline buttons, and a table with columns for Name, Latest execution status, Latest source revisions, Latest execution started, and Most recent executions. The table displays the message 'No results' and 'There are no results to display.' The bottom of the screen shows the URL https://eu-north-1.console.aws.amazon.com/console/home?region=eu-north-1 and standard AWS navigation links.

Screenshot of the AWS CodePipeline 'Create new pipeline' wizard, Step 2 of 5: Add source stage.

The 'Source' provider is set to GitHub (Version 1). A success message indicates the action has been configured. A note states that GitHub (Version 1) is no longer recommended; instead, choose GitHub (Version 2) to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources.

Repository: Q\_2022NK/PLAB\_EXP2

Branch: Q\_main

Change detection options:

- GitHub webhooks (recommended)  
Use webhooks in GitHub to automatically start my pipeline when a change occurs.
- AWS CodePipeline  
Use AWS CodePipeline to check periodically for changes.

Buttons: Cancel, Previous, Next.

Screenshot of the AWS CodePipeline 'Create new pipeline' wizard, Step 4 of 5: Add deploy stage.

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

**Deploy**

Deploy provider: AWS Elastic Beanstalk

Region: Europe (Stockholm)

Input artifacts: SourceArtifact

Application name: Niraj24

Environment name: Niraj24-env

Configure automatic rollback on stage failure

Buttons: Cancel, Previous, Next.

aws Services Search [Alt+S]

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose pipeline settings Review Info Step 5 of 5

Step 2 Add source stage

Step 3 Add build stage

Step 4 Add deploy stage

Step 5 Review

**Step 1: Choose pipeline settings**

Pipeline settings

Pipeline name: Niraj\_Pipeline

Pipeline type: V2

Execution mode: QUEUED

Artifact location: A new Amazon S3 bucket will be created as the default artifact store for your pipeline

Service role name: AWSCodePipelineServiceRole-eu-north-1-Niraj\_Pipeline

**Variables**

Name	Default value	Description
No variables		

No variables defined at the pipeline level in this pipeline.

**Step 2: Add source stage**

Source action provider

Source action provider: GitHub (Version 1)

PollForSourceChanges: false

Repo: IPLAB\_EXP2

Owner: 2022NK

Branch: main

**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: Niraj24

EnvironmentName: Niraj24-env

Configure automatic rollback on stage failure: Disabled

Cancel Previous Create pipeline

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

**SUCCESS**  
Congratulations! The pipeline Niraj\_Pipeline has been created.

Developer Tools > CodePipeline > Pipelines > Niraj\_Pipeline

Niraj\_Pipeline  
Pipeline type: V2 Execution mode: QUEUED

**Source** Succeeded  
Pipeline execution ID: 08877347-55da-408f-a716-4b8a5a6cbb6e

Source  
GitHub (Version\_1) [View details](#)  
Succeeded - Just now  
508d20de [View details](#)

508d20de Source: styles.css

**Disable transition**

**Deploy** Succeeded  
Pipeline execution ID: 08877347-55da-408f-a716-4b8a5a6cbb6e

Deploy  
AWS Elastic Beanstalk [View details](#)  
Succeeded - Just now  
508d20de [View details](#)

508d20de Source: styles.css

**Start rollback**

**Home**  
Welcome to NK LIMITED



At NK Limited Steel Company, we are dedicated to providing high-quality steel products and services to meet the needs of our clients. With years of experience in the industry, our team of experts ensures that every product meets the highest standards of quality and reliability. Whether you're looking for structural steel, stainless steel, or custom solutions, we are here to support your projects with excellence and precision.

## Using S3 Bucket:

The screenshot shows the AWS Management Console search results for 'S3'. The search bar at the top has 'S3' typed into it. On the left, there's a sidebar with 'Services (8)' expanded, showing 'Features (39)', 'Resources New', 'Documentation (27,052)', 'Knowledge Articles (288)', 'Marketplace (1,893)', 'Blogs (1,428)', 'Events (26)', and 'Tutorials (12)'. The main content area displays 'Search results for 'S3'' under 'Services'. It lists three items: 'S3' (Scalable Storage in the Cloud), 'S3 Glacier' (Archive Storage in the Cloud), and 'AWS Snow Family' (Large Scale Data Transport). Below this, under 'Features', there are sections for 'Imports from S3' (DynamoDB feature) and 'Feature spotlight'. To the right, there's a sidebar with options like 'to default layout', '+ Add widgets', 'Create application', and 'Region' set to 'Stockholm'. The bottom of the screen shows the standard AWS footer with links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

The screenshot shows the 'Create bucket' wizard in the AWS Management Console. The title bar says 'Amazon S3 > Buckets > Create bucket'. The main section is titled 'General configuration'. It shows the 'AWS Region' as 'Europe (Stockholm) eu-north-1'. Under 'Bucket type', there are two options: 'General purpose' (selected) and 'Directory - New'. The 'Bucket name' field is filled with 'Niraj24'. Below it, a note says 'Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming'. There's also a 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button and a note about copied settings. The bottom of the screen shows the standard AWS footer with links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

**Object Ownership** [Info](#)

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

**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.

Object Ownership  
Bucket owner enforced

### Block Public Access settings for this bucket

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 this bucket and its 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 this bucket 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.

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

**Default encryption** [Info](#)

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)

**Server-side encryption with Amazon S3 managed keys (SSE-S3)**

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

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

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable

Enable

**Advanced settings**

After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

[Cancel](#) [Create bucket](#)

**Successfully created bucket "niraj24"**  
To upload files and folders, or to configure additional bucket settings, choose [View details](#).

[View details](#) [X](#)

Amazon S3 > Buckets

**Account snapshot - updated every 24 hours** [All AWS Regions](#)

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

[View Storage Lens dashboard](#)

[General purpose buckets](#) [Directory buckets](#)

**General purpose buckets (4) [Info](#) [All AWS Regions](#)**

Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
<a href="#">cf-templates-1lazs4t4no767-eu-north-1</a>	Europe (Stockholm) eu-north-1	<a href="#">View analyzer for eu-north-1</a>	August 22, 2024, 19:10:55 (UTC+05:30)
<a href="#">codepipeline-eu-north-1-420991496383</a>	Europe (Stockholm) eu-north-1	<a href="#">View analyzer for eu-north-1</a>	August 22, 2024, 19:53:39 (UTC+05:30)
<a href="#">elasticbeanstalk-eu-north-1-147997146975</a>	Europe (Stockholm) eu-north-1	<a href="#">View analyzer for eu-north-1</a>	August 22, 2024, 18:53:00 (UTC+05:30)
<a href="#">niraj24</a>	Europe (Stockholm) eu-north-1	<a href="#">View analyzer for eu-north-1</a>	August 22, 2024, 20:18:55 (UTC+05:30)

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

Amazon S3 > Buckets > niraj24 > Upload

### Upload Info

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

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

**Files and folders (1 Total, 3.1 KB)**

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

**Destination Info**

Destination  
s3://niraj24

▶ Destination details

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

**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	Succeeded	Failed
s3://niraj24	1 file, 3.1 KB (100.00%)	0 files, 0 B (0%)

**Files and folders** Configuration

**Files and folders (1 Total, 3.1 KB)**

Name		Folder	Type	Size	Status	Error
<input type="checkbox"/>	index.html	-	text/html	3.1 KB	<span style="color: green;">Succeeded</span>	-

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

Amazon S3 > Buckets > niraj24

**niraj24 Info**

**Objects** Info

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

**Actions** Copy S3 URI Copy URL Download Open Delete Create folder Upload

**Find objects by prefix**

Name	Type	Last modified	Size	Storage class
index.html	html	August 22, 2024, 20:24:20 (UTC+05:30)	3.1 KB	Standard

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

Amazon S3 > Buckets > niraj24 > index.html

### index.html Info

[Copy S3 URI](#) [Download](#) [Open](#) [Object actions](#)

[Properties](#) [Permissions](#) [Versions](#)

#### Object overview

Owner	S3 URI
f8f6a62f5e39c655bef5fa1089212b5faaf9875299519774d40b647b12045a5	<a href="s3://niraj24/index.html">s3://niraj24/index.html</a>
AWS Region	Amazon Resource Name (ARN)
Europe (Stockholm) eu-north-1	<a href="arn:aws:s3:::niraj24/index.html">arn:aws:s3:::niraj24/index.html</a>
Last modified	Entity tag (Etag)
August 22, 2024, 20:24:20 (UTC+05:30)	<a href="#">d49fc5123e55b943ed46b3a5c19e5d1b</a>
Size	Object URL
3.1 KB	<a href="https://niraj24.s3.eu-north-1.amazonaws.com/index.html">https://niraj24.s3.eu-north-1.amazonaws.com/index.html</a>
Type	
html	
Key	

[index.html](#)

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

Amazon S3 Services Search [Alt+S]

Amazon S3 > Buckets > niraj24

### niraj24 Info

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

#### Bucket overview

AWS Region	Amazon Resource Name (ARN)	Creation date
Europe (Stockholm) eu-north-1	<a href="arn:aws:s3:::niraj24">arn:aws:s3:::niraj24</a>	August 22, 2024, 20:18:55 (UTC+05:30)

#### Bucket Versioning

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

[Edit](#)

Bucket Versioning  
Disabled

Multi-factor authentication (MFA) delete  
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Multi-factor authentication (MFA) delete  
Disabled

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

#### Static website hosting

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

[Edit](#)

Static website hosting  
Disabled

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

AWS Services Search [Alt+S] Stockholm Niraj017

Amazon S3 > Buckets > niraj24 > Edit static website hosting

## Edit static website hosting [Info](#)

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

Static website hosting  
 Disable  
 Enable

Hosting type  
 Host a static website  
Use the bucket endpoint as the web address. [Learn more](#)  
 Redirect requests for an object  
Redirect requests to another bucket or domain. [Learn more](#)

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

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

Save changes

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

AWS Services Search [Alt+S] Stockholm Niraj017

Amazon S3 > Buckets > niraj24

## niraj24 [Info](#)

Objects Properties Permissions Metrics Management Access Points

### Permissions overview

Access finding  
Access findings are provided by IAM external access analyzers. Learn more about [How IAM analyzer findings work](#).  
View analyzer for eu-north-1

### 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**  
 On  
► Individual Block Public Access settings for this bucket

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

The screenshot shows a modal dialog titled "Edit Block public access (bucket settings)". Inside the dialog, there is a warning message in a yellow box: "⚠️ Updating the Block Public Access settings for this bucket will affect this bucket and all objects within. This may result in some objects becoming public." Below the warning, instructions say "To confirm the settings, enter *confirm* in the field." A text input field contains the word "confirm". At the bottom right are two buttons: "Cancel" and "Confirm", with "Confirm" being highlighted in orange.

Object Ownership	Info	Edit
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.		
Object Ownership		
Bucket owner enforced		
ACLs are disabled. All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.		

**Edit Object Ownership** [Info](#)

**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**

**Bucket owner preferred**  
If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

**Object writer**  
The object writer remains the object owner.

**If you want to enforce object ownership for new objects only, your bucket policy must specify that the bucket-owner-full-control canned ACL is required for object uploads.** [Learn more](#)

[Cancel](#) [Save changes](#)

**Amazon S3** > **Buckets** > **niraj24**

**niraj24** [Info](#)

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

**Objects (1) Info**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you

[Find objects by prefix](#)

<input checked="" type="checkbox"/>	Name	Type	Last modified	Size
<input checked="" type="checkbox"/>	<a href="#">index.html</a>	html	August 22, 2024, 20:24:20 (UTC+05:30)	3.1 KB

**Actions** [Upload](#)

Share with a presigned URL [Learn more](#)

Calculate total size

Copy

Move

Initiate restore

Query with S3 Select

Edit actions

Rename object

Edit storage class

Edit server-side encryption

Edit metadata

Edit tags

[Make public using ACL](#)

**Amazon S3** > **Buckets** > **niraj24** > **Make public**

**Make public** [Info](#)

The make public action enables public read access in the object access control list (ACL) settings. [Learn more](#).

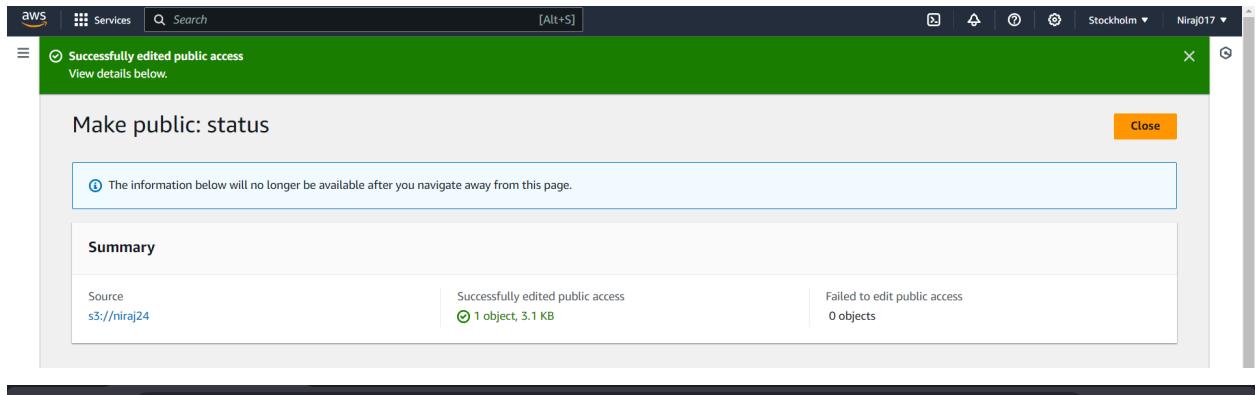
**When public read access is enabled and not blocked by Block Public Access settings, anyone in the world can access the specified objects.**

**Specified objects**

[Find objects by name](#)

Name	Type	Last modified	Size
<a href="#">index.html</a>	html	August 22, 2024, 20:24:20 (UTC+05:30)	3.1 KB

[Cancel](#) [Make public](#)



## Order Your Customized T-Shirt

T-Shirt Customization-

Tagline on the Shirt:

Color:

Size:

Quantity:

Delivery Date:

Delivery Details

Recipient's Name:

Address:

Email:

Phone Number:  Format: 1234567890

Additional Comments:

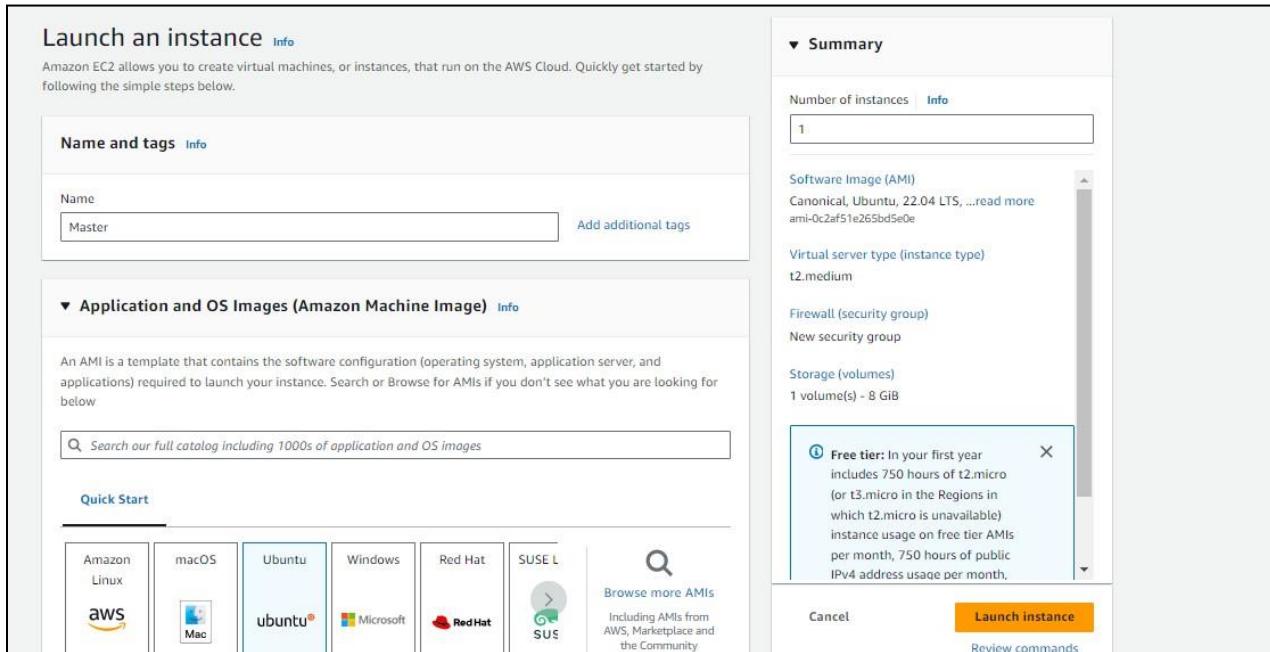
# ADVANCE DEVOPS EXP 3

Niraj S. Kothawade  
D15A - 24

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

## Step 1: Pre-requisites

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



1.2 Proceed with the following settings and create a new key pair as follows(use the same key pair for all the three nodes)

The screenshot shows the AWS Lambda 'Create Function' configuration interface. It includes sections for 'Instance type', 'Key pair (login)', and 'Network settings'.

**Instance type:** t2.medium (selected).  
Family: t2 - 2 vCPU, 4 GiB Memory. Current generation: true.  
On-Demand Linux base pricing: 0.0496 USD per Hour  
On-Demand Windows base pricing: 0.0676 USD per Hour  
On-Demand RHEL base pricing: 0.0784 USD per Hour  
On-Demand SUSE base pricing: 0.1496 USD per Hour

**Key pair (login):** two-tier-app-k8s (selected).  
Create new key pair

**Network settings:** Network: vpc-04007898e59a6979f  
Subnet: (Info)

## Create key pair

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

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

**Key pair type**

RSA  
RSA encrypted private and public key pair

ED25519  
ED25519 encrypted private and public key pair

**Private key file format**

.pem  
For use with OpenSSH

.ppk  
For use with PuTTY

**⚠ When prompted, store the private key in a secure and accessible location on**

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

Instances (1/3) <a href="#">Info</a>										
<span>Last updated less than a minute ago</span> <span></span> <span><a href="#">Connect</a></span> <span><a href="#">Instance state ▾</a></span> <span><a href="#">Actions ▾</a></span> <span><a href="#">Launch instances</a></span> <span></span>										
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/> <span>All states ▾</span>										
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IP		
Worker-2	i-0e3930ceb2d892d01	Running	t2.medium	2/2 checks passed	<a href="#">View alarms</a>	ap-south-1a	ec2-13-234-226-219.ap...	13.234.22		
Worker-1	i-0d16e01d1824e0e3a	Running	t2.medium	2/2 checks passed	<a href="#">View alarms</a>	ap-south-1a	ec2-65-0-104-95.ap-so...	65.0.104.		
Master	i-01ae3d388db90ad73	Running	t2.medium	2/2 checks passed	<a href="#">View alarms</a>	ap-south-1a	ec2-13-232-36-34.ap-s...	13.232.36		

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

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

Instance ID  
 i-0e3930ceb2d892d01 (Worker-2)

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

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

```
acer@TMP214-53 MINGW64 ~/Downloads
$ ssh -i "two-tier-app-k8s.pem" ubuntu@ec2-13-232-36-34.ap-south-1.compute.amazonaws.com
The authenticity of host 'ec2-13-232-36-34.ap-south-1.compute.amazonaws.com (13.232.36.34)' can't be established.
ED25519 key fingerprint is SHA256:uVGEO+FwYefj60j0ft70Sralv8NrzEi/IwxAtBY+EP.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-232-36-34.ap-south-1.compute.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1022-aws x86_64)

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

System information as of Wed Sep 11 14:07:10 UTC 2024

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

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

## Step 2: Prepare Nodes

### 2.1. Update the package manager on all nodes:

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

The screenshot shows a terminal window with the following text:

```
ubuntu@ip-172-31-22-29: ~
Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

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

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

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

Swap usage: 0% To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

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

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

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

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

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

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

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

//1 The list of available updates is more than a week old.
//1 To check for new updates run: sudo apt update

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

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

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

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

The screenshot shows a terminal window with the following text:

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

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

```
sudo swapoff -a
```

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

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

2.3. Load necessary kernel modules for networking and iptables:

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

```
overlay
```

```
br_netfilter
```

```
EOF
```

```
sudo modprobe overlay
```

```
sudo modprobe br_netfilter
```

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

2.4. Configure sysctl settings for Kubernetes networking:

```
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
```

```
net.bridge.bridge-nf-call-ip6tables = 1
```

```
net.bridge.bridge-nf-call-iptables = 1
```

```
EOF
```

```
sudo sysctl --system
```

```
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
overlay
br_netfilter
EOF
sudo modprobe overlay
sudo modprobe br_netfilter
overlay
br_netfilter
ubuntu@ip-172-31-22-29:~$ cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
EOF
sudo sysctl --system
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
# Applying /etc/sysctl.d/10-console-messages.conf ...
kernel.printk = 4 4 1 7
# Applying /etc/sysctl.d/10-ipv6-privacy.conf ...
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.default.use_tempaddr = 2
# Applying /etc/sysctl.d/10-kernel-hardening.conf ...
kernel.kptr_restrict = 1
```

### Step 3: Install Docker

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

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

```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y apt-transport-https ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu
$(lsb_release -cs) stable"
sudo apt-get update
sudo apt-get install -y docker-ce docker-ce-cli containerd.io
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Fetched 129 kB in 1s (241 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.81.0-1ubuntu1.17).
curl set to manually installed.
software-properties-common is already the newest version (0.99.22.9).
software-properties-common set to manually installed.
```

Configure Docker for Kubernetes:

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

```
sudo systemctl restart docker
```

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

## Step 4: Install kubeadm, kubelet, kubectl

Install Kubernetes tools on all nodes.

### 4.1. Add Kubernetes APT repository:

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

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

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

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

```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
```

### Step 5: Initialize the Kubernetes Cluster on Master Node

On the master node:

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

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

5.1. Set up kubectl on the master node:

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

```
ubuntu@ip-172-31-22-29:~$ sudo kubeadm config images pull  
sudo kubeadm init  
mkdir -p "$HOME"/.kube  
sudo cp -i /etc/kubernetes/admin.conf "$HOME"/.kube/config  
sudo chown "$(id -u):$(id -g)" "$HOME"/.kube/config  
  
# Network Plugin = calico  
kubectl apply -f https://raw.githubusercontent.com/projectcalico/calico/v3.26.0/manifests/calico.yaml  
  
kubeadm token create --print-join-command --v=5  
Found multiple CRI endpoints on the host. Please define which one do you wish to use by setting the 'criSocket' field in the kubeadm configuration file: unix:///var/run/containerd/containerd.sock, unix:///var/run/crio/crio.sock  
To see the stack trace of this error execute with --v=5 or higher  
Found multiple CRI endpoints on the host. Please define which one do you wish to use by setting the 'criSocket' field in the kubeadm configuration file: unix:///var/run/containerd/containerd.sock, unix:///var/run/crio/crio.sock
```

## Step 6: Install a Pod Network Add-on

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

### Install Flannel:

```
kubectl apply -f
```

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

```
ubuntu@ip-172-31-22-29:~$ kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml --validate=false  
E0913 15:35:04.261458 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.261902 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.263424 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.263795 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.265840 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
E0913 15:35:04.266524 19259 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 127.0.0.1:8080  
unable to recognize "https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml": Get "http://localhost:8080/api?timeout=32s": dial
```

## Step 7: Join Worker Nodes to the Cluster

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

```
sudo kubeadm join <master-ip>:6443 --token <token> --discovery-token-ca-cert-hash sha256:<hash>
```

```
clusterrolebinding.rbac.authorization.k8s.io/calico-cni-plugin created  
daemonset.apps/calico-node created  
deployment.apps/calico-kube-controllers created  
kubeadm join 172.31.62.216:6443 --token br7fe5.hq28adbm1mu17ky --discovery-token-ca-cert-hash sha256:2bc469a8d14fbebe879328d2b416fad  
32b29a8505d3f448b98703ffff3b014d9
```

## Step 8: Verify the Cluster

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

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

# ADVANCE DEVOPS EXP 4

**Niraj S. Kothawade**  
**D15A - 24**

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

## Step 1: Install Kubectl on Ubuntu

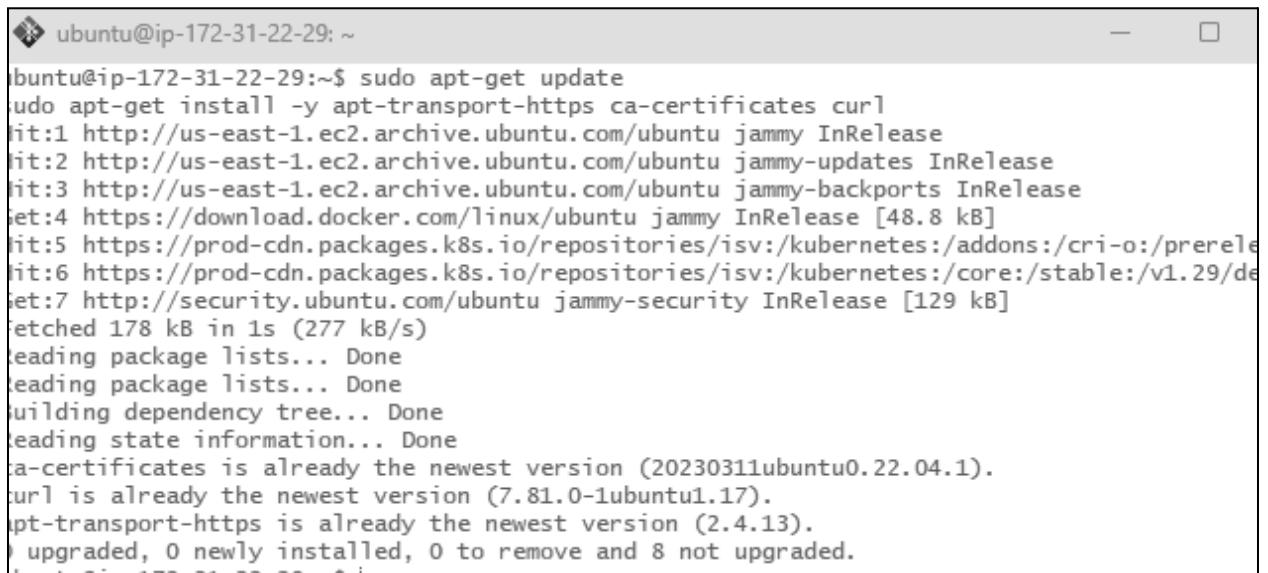
### 1.1 Add Kubernetes APT repository

First, add the Kubernetes repository to your system.

#### 1. Install prerequisites:

```
sudo apt-get update
```

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



```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
[sudo] password for ubuntu:
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Get:5 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/addons:/cri-o:/prerelease
Get:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.29/de
Get:7 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Fetched 178 kB in 1s (277 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
curl is already the newest version (7.81.0-1ubuntu1.17).
apt-transport-https is already the newest version (2.4.13).
0 upgraded, 0 newly installed, 0 to remove and 8 not upgraded.
```

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

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

<https://packages.cloud.google.com/apt/doc/apt-key.gpg>

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

### 3. Add the Kubernetes repository:

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

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

## 1.2 Install kubectl

Now install kubectl:

```
sudo apt-get update
```

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

```
ubuntu@ip-172-31-22-29:~$ sudo apt-get update
sudo apt-get install -y kubectl
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/addons:/cri-o:/prerelease:/main/deb InRelease
Ign:7 https://packages.cloud.google.com/apt kubernetes-focal InRelease
Err:8 https://packages.cloud.google.com/apt kubernetes-focal Release
  404  Not Found [IP: 172.253.62.138 443]
Reading package lists... Done
E: The repository 'https://apt.kubernetes.io kubernetes-focal Release' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
kubectl is already the newest version (1.29.0-1.1).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
```

Verify the installation(extra):

```
kubectl version --client
```

```
ubuntu@ip-172-31-22-29:~$ kubectl version --client
Client Version: v1.29.0
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
```

## **Step 2: Deploying Your Application on Kubernetes**

### **2.1 Set up Kubernetes Cluster**

1. If you haven't already set up a Kubernetes cluster (e.g., with kubeadm), use minikube or any managed Kubernetes service (like EKS, GKE, etc.) to get a cluster running.
2. Once your cluster is ready, verify the nodes:

```
kubectl get nodes
```

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

### **Step 3: Create the Deployment YAML file**

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

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

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

```
ubuntu@ip-172-31-45-227: ~          nginx-deployment.yaml
GNU nano 6.2
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.21.3
          ports:
            - containerPort: 80
```

#### Step 4: Create the Service YAML File

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

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

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

```
ubuntu@ip-172-31-45-227: ~          nginx-service.yaml *
GNU nano 6.2
apiVersion: v1
kind: Service
metadata:
  name: nginx-service
spec:
  selector:
    app: nginx
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
  type: LoadBalancer
```

## **Step 5:Apply the YAML Files**

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

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

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

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

## Describe the deployment(Extra)

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

## Step 6:Ensure Service is Running

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

```
kubectl get service
```

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

## Step 7:Forward the Service Port to Your Local Machine

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

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

```
kubectl port-forward service/<service-name> <local-port>:<service-port>
```

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

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

2. This means port forwarding is now active, and any traffic to localhost:8080 will be routed to the nginx-service on port 80.

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

## Step 8: Access the Application Locally

1. **Open a Web Browser:** Now open your web browser and go to the following URL:

`http://localhost:8080`

You should see the application (in this case, Nginx) that you have deployed running in the Kubernetes cluster, served locally via port 8080.

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



## ADVANCE DEVOPS EXP 5

Niraj Kothawade  
D15A - 24

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

### Installation for Windows:

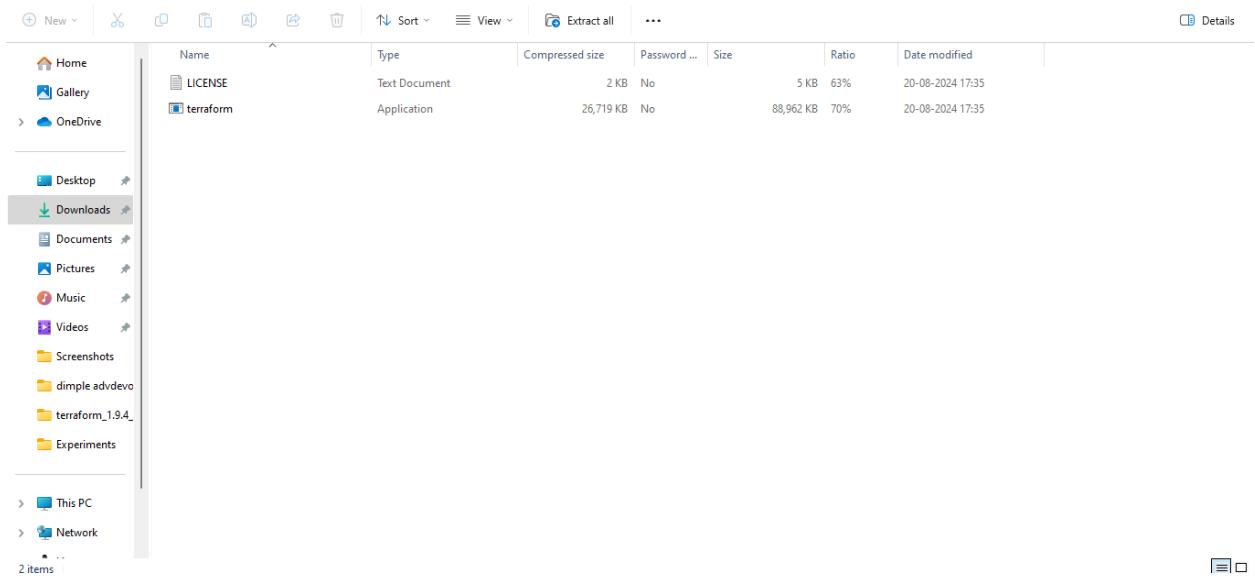
**Step 1:** Go to the website [terraform.io](https://developer.hashicorp.com/terraform/install) and install Terraform from there.. Select the AMD64 option for Windows and download Terraform.

The screenshot shows the Terraform installation page on developer.hashicorp.com/terraform/install. The top navigation bar includes links for Home, Install, Tutorials, Documentation, Registry, Try Cloud, and a search bar. The main content area is titled "Install Terraform". On the left, a sidebar lists "Operating Systems" with "macOS" selected, and "Release information" with "Next steps". The main content area has two sections: "macOS" and "Windows". The "macOS" section shows package manager instructions for Homebrew:

```
brew tap hashicorp/tap
brew install hashicorp/tap/terraform
```

The "Windows" section shows binary download options for 386 and AMD64 architectures, both at version 1.9.5. Each download link includes a "Download" button. A sidebar on the right provides "About Terraform" (defining configuration files), "Featured docs" (Introduction to Terraform, Configuration Language, Terraform CLI, HCP Terraform, Provider Use), and a "HCP Terraform" section.

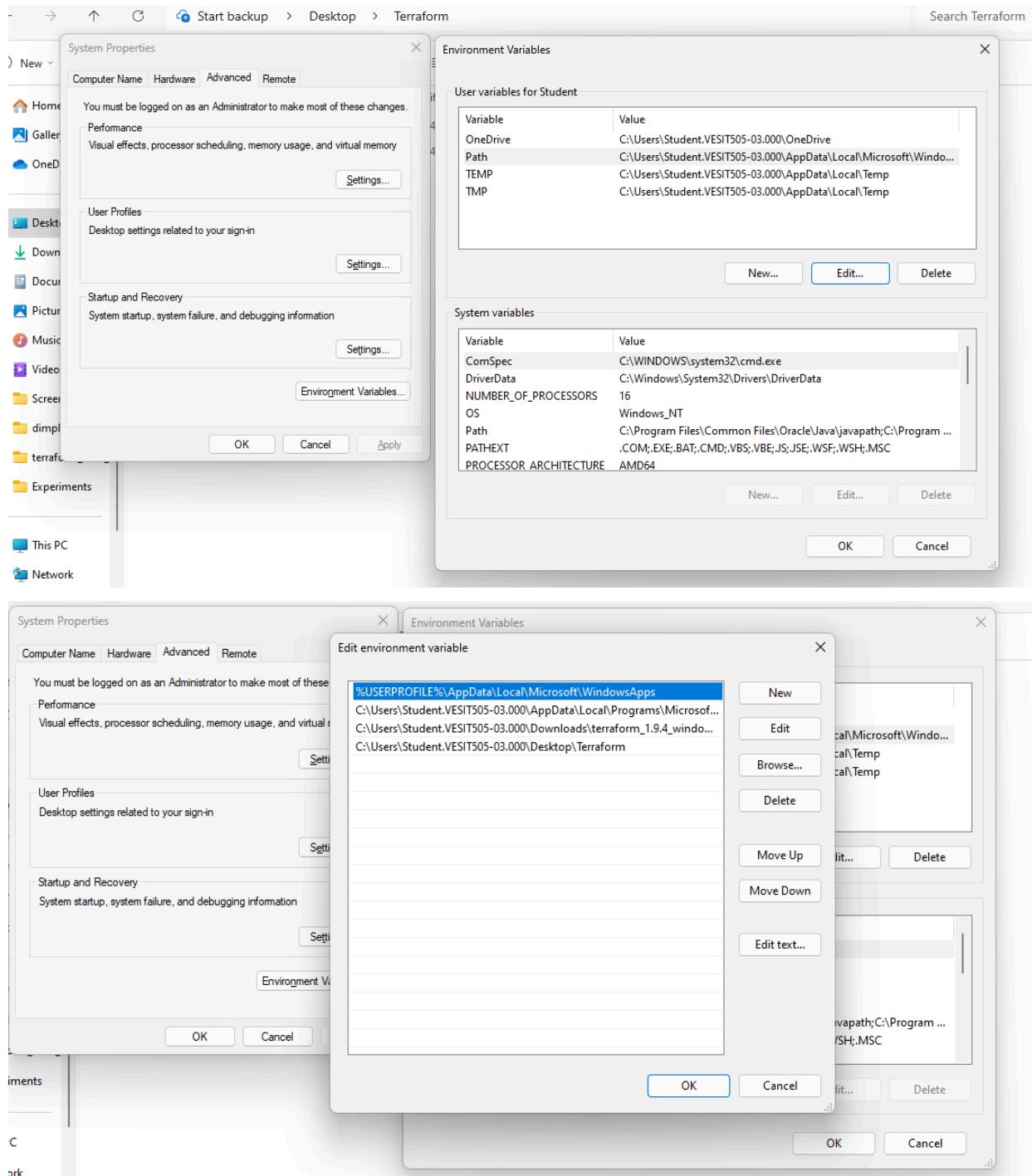
**Step 2:** Go to the zip file where Terraform is installed.



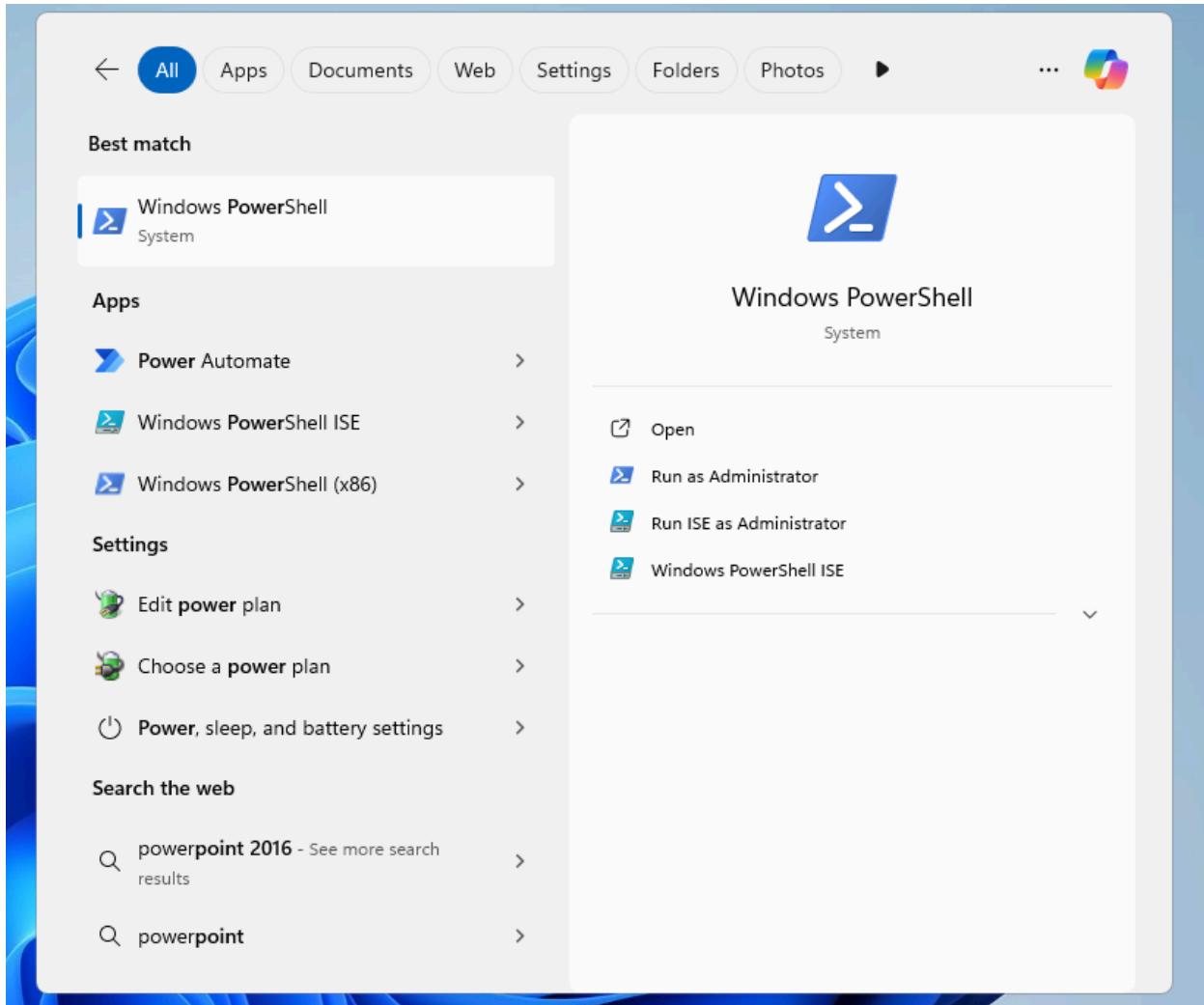
**Step 3:** Since the installed file is a zip file, create a new folder on desktop and copy the installed terraform application there.

Terraform	22-08-2024 14:24	File folder
VAISHNAVI	31-07-2024 14:07	File folder

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



**Step 5:** Now go to the folder where we have installed terraform and open Powershell inside it. After this type ‘terraform’ to make sure that terraform has been installed on the system. The command ‘terraform –version’ simply checks the current version of terraform that has been installed.



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

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

PS C:\Users\Student.VESIT505-03.000\Desktop\Terraform> terraform
Usage: terraform [global options] <subcommand> [args]

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

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

All other commands:
  console    Try Terraform expressions at an interactive command prompt
  fmt        Reformat your configuration in the standard style
  force-unlock Release a stuck lock on the current workspace
  get        Install or upgrade remote Terraform modules
  graph     Generate a Graphviz graph of the steps in an operation
  import    Associate existing infrastructure with a Terraform resource
  login     Obtain and save credentials for a remote host
  logout    Remove locally-stored credentials for a remote host

PS C:\Users\Student.VESIT505-03.000\Desktop\Terraform> terraform --version
Terraform v1.9.4
on windows_386

Your version of Terraform is out of date! The latest version
is 1.9.5. You can update by downloading from https://www.terraform.io/downloads.html
PS C:\Users\Student.VESIT505-03.000\Desktop\Terraform> |
```

# ADVANCE DEVOPS EXP 6

Niraj S. Kothawade  
D15A -24

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

## Part A:Creating docker image using terraform

Prerequisite:

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

### Step 1:Check Docker functionality

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

C:\Users\student>docker

Usage: docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

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

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

Check for the docker version with the following command.

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

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

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

Script:

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

```
}
```

```
"` docker.tf  X
` docker.tf
1  terraform {
2    required_providers {
3      docker = {
4        source  = "kreuzwerker/docker"
5        version = "2.21.0"
6      }
7    }
8  }
9
10 provider "docker" {
11   host = "npipe:///./pipe/docker_engine"
12 }
13
14 # Pull the image
15 resource "docker_image" "ubuntu" {
16   name = "ubuntu:latest"
17 }
18
19 # Create a container
20 resource "docker_container" "foo" {
21   image = docker_image.ubuntu.image_id
22   name  = "foo"
23   command = ["sleep", "3600"]
24 }
25 |
```

### Step 3: Execute Terraform Init command to initialize the resources

```
● PS C:\Users\Admin\TerraformScripts> cd Docker
● PS C:\Users\Admin\TerraformScripts\Docke> terraform init
Initializing the backend...
Initializing provider plugins...
  - Finding kreuzwerker/docker versions matching "2.21.0"...
  - Installing kreuzwerker/docker v2.21.0...
○ - Installed kreuzwerker/docker v2.21.0 (self-signed, key ID BD080C4571C6104C)
  Partner and community providers are signed by their developers.
  If you'd like to know more about provider signing, you can read about it here:
  https://www.terraform.io/docs/cli/plugins/signing.html
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.
```

**Terraform has been successfully initialized!**

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

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

## Step 4: Execute Terraform plan to see the available resources

```
PS C:\Users\Admin\TerraformScripts\Docker> terraform plan
```

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

```
+ create
```

```
Terraform will perform the following actions:
```

```
# docker_container.foo will be created
+ resource "docker_container" "foo" {
  + attach           = false
  + bridge          = (known after apply)
  + command         = [
    + "sleep",
    + "3600",
  ]
  + container_logs  = (known after apply)
  + entrypoint      = (known after apply)
  + env              = (known after apply)
  + exit_code        = (known after apply)
  + gateway          = (known after apply)
  + hostname         = (known after apply)
  + id               = (known after apply)
  + image             = (known after apply)
  + init              = (known after apply)
  + ip_address       = (known after apply)
  + ip_prefix_length = (known after apply)
  + ipc_mode         = (known after apply)
  + log_driver       = (known after apply)
  + logs              = false
  + must_run         = true
  + name              = "foo"
  + network_data     = (known after apply)
  + read_only         = false
  + remove_volumes   = true
  + restart           = "no"
  + rm                = false
}
```

```
+ runtime           = (known after apply)
+ security_opts     = (known after apply)
+ shm_size          = (known after apply)
+ start             = true
+ stdin_open        = false
+ stop_signal       = (known after apply)
+ stop_timeout      = (known after apply)
+ tty               = false

+ healthcheck (known after apply)

+ labels (known after apply)
}
```

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

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

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

```
● PS C:\Users\Admin\TerraformScripts\Docker> terraform apply

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

Terraform will perform the following actions:

# docker_container.foo will be created
+ resource "docker_container" "foo" {
    + attach          = false
    + bridge          = (known after apply)
    + command         = [
        + "sleep",
        + "3600",
    ]
    + container_logs = (known after apply)
    + entrypoint      = (known after apply)
    + env             = (known after apply)
    + exit_code       = (known after apply)
    + gateway         = (known after apply)
    + hostname        = (known after apply)
    + id              = (known after apply)
    + image           = (known after apply)
    + init            = (known after apply)
    + ip_address      = (known after apply)
    + ip_prefix_length = (known after apply)
    + ipc_mode        = (known after apply)
    + log_driver      = (known after apply)
    + logs            = false
    + must_run        = true
    + name            = "foo"
    + network_data    = (known after apply)
    + read_only       = false
}
```

```
+ remove_volumes  = true
+ restart         = "no"
+ rm              = false
+ runtime         = (known after apply)
+ security_opts   = (known after apply)
+ shm_size        = (known after apply)
+ start           = true
+ stdin_open      = false
+ stop_signal     = (known after apply)
+ stop_timeout    = (known after apply)
+ tty              = false

+ healthcheck (known after apply)

+ labels (known after apply)
}
```

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

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

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

Enter a value: yes

```

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

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

```

Docker images, Before Executing Apply step:

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

Docker images, After Executing Apply step:

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

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

```

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

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

Terraform will perform the following actions:

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

```

```

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

```

```

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

```

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

**Do you really want to destroy all resources?**

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

Enter a value: yes

```

docker_container.foo: Destroying... [id=01adf07e5918931fee9b90073726a03671037923dd92032ce0e15bbb764a6f24]
docker_container.foo: Destruction complete after 0s
docker_image.ubuntu: Destroying... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_image.ubuntu: Destruction complete after 1s

```

Destroy complete! Resources: 2 destroyed.

## Docker images After Executing Destroy step

```

● PS C:\Users\Admin\TerraformScripts\Docker> docker images
REPOSITORY          TAG      IMAGE ID      CREATED             SIZE

```

# ADVANCE DEVOPS EXP 7

**Niraj S. Kothawade**  
**D15A - 24**

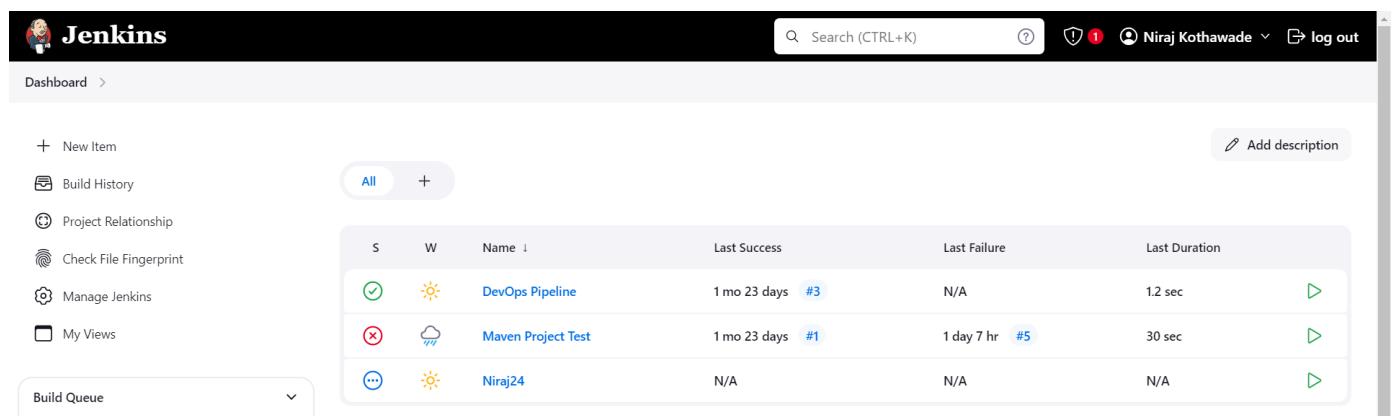
**Aim:** To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

## Integrating Jenkins with SonarQube:

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

## Steps to integrate Jenkins with SonarQube

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



The screenshot shows the Jenkins dashboard with the title "Jenkins". The top navigation bar includes a search field, user information for "Niraj Kothawade", and a "log out" button. Below the header, there's a "Dashboard" link and a "Build History" link. On the left, there are links for "Project Relationship", "Check File Fingerprint", "Manage Jenkins", and "My Views". A "Build Queue" dropdown is also present. The main content area displays a table of active builds:

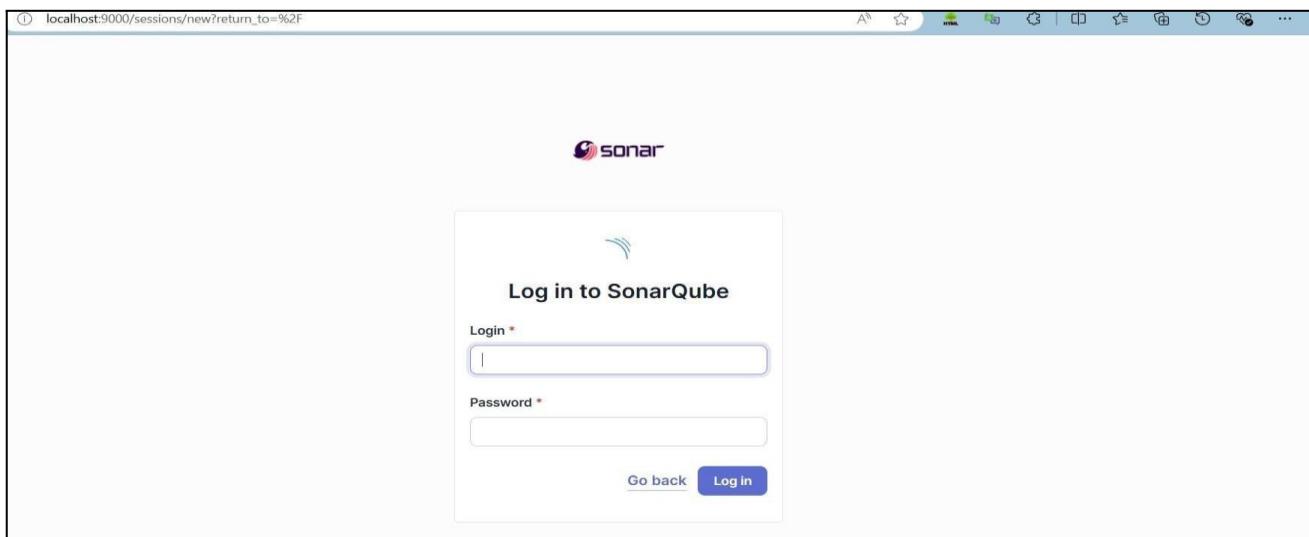
S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☀️	DevOps Pipeline	1 mo 23 days #3	N/A	1.2 sec
✗	🌧️	Maven Project Test	1 mo 23 days #1	1 day 7 hr #5	30 sec
...	☀️	Niraj24	N/A	N/A	N/A

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

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

```
PS C:\Windows\system32> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest  
Unable to find image 'sonarqube:latest' locally  
latest: Pulling from library/sonarqube  
7478e0ac0f23: Pull complete  
90a925ab929a: Pull complete  
7d9a34308537: Pull complete  
80338217a4ab: Pull complete  
1a5fd5c7e184: Pull complete  
7b87d6fa783d: Pull complete  
bd819c9b5ead: Pull complete  
4f4fb700ef54: Pull complete  
Digest: sha256:72e9fecc71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde  
Status: Downloaded newer image for sonarqube:latest  
5ab3928e5e27607e3661d129731e4e600a9019574c7dc2767aa9b3bfdaa941be
```

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



- Login to SonarQube using username admin and password admin.

**How do you want to create your project?**

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

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

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

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

Create a local project

- Create a manual project in SonarQube with the name sonarqube

**Create a local project**

**Project display name \***

**Project key \***

**Main branch name \***

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

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

**Set up project for Clean as You Code**

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

Choose the baseline for new code for this project

Use the global setting

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

Define a specific setting for this project

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

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

Setup the project and come back to Jenkins Dashboard.

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

The screenshot shows the Jenkins 'Manage Jenkins' interface under the 'Plugins' section. A search bar at the top contains the text 'sonarq'. Below the search bar, there are tabs for 'Updates' (25), 'Available plugins' (selected), 'Installed plugins', and 'Advanced settings'. The main area displays a table for the 'SonarQube Scanner' plugin, version 2.17.2. The table includes columns for 'Install' (button), 'Name ↓' (sorter), and 'Released' (date: 6 mo 29 days ago). Below the table, a description states: 'This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.' An 'Install' button is located in the top right corner of the plugin's card.

The screenshot shows the Jenkins 'Manage Jenkins' interface under the 'Plugins' section. A sidebar on the left lists 'Updates' (25), 'Available plugins' (selected), 'Installed plugins', and 'Advanced settings'. The main area is titled 'Download progress' and shows the status of the SonarQube Scanner plugin. It indicates 'Preparation' completed with three success items: 'Checking internet connectivity', 'Checking update center connectivity', and 'Success'. It also shows 'SonarQube Scanner' and 'Loading plugin extensions' both marked as 'Success'. At the bottom, there are links to 'Go back to the top page' and 'Restart Jenkins when installation is complete and no jobs are running'.

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

and enter the details.

Enter the Server Authentication token if needed.

In SonarQube installations: Under **Name** add <project name of sonarqube>, here we have named it as **adv\_devops\_7\_sonarqube**

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

## SonarQube servers

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

Environment variables

### SonarQube installations

List of SonarQube installations

#### Name

adv\_devops\_7\_sonarqube



#### Server URL

Default is http://localhost:9000

https://localhost:9000

#### Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

+ Add ▾

Advanced ▾

7. Search for SonarQube Scanner under Global Tool Configuration.

Choose the latest configuration and choose Install automatically.

## Dashboard > Manage Jenkins > Tools

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

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

The screenshot shows the 'SonarQube Scanner installations' configuration screen. It displays a list of installed scanners, including one named 'sonarqube\_exp7'. Below the list, there is a section for adding a new scanner. The 'Install automatically' checkbox is checked. A dropdown menu for 'Install from Maven Central' is open, showing the version 'SonarQube Scanner 6.1.0.4477'. There is also an 'Add Installer' button at the bottom.

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

adv\_devops\_exp7  
» Required field

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

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

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

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

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

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

**OK**

9. Choose this GitHub repository in Source Code Management.

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

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

The screenshot shows the 'Source Code Management' configuration page. The 'Git' option is selected. A 'Repository URL' field contains the value 'https://github.com/shazforiot/MSBuild\_firstproject.git'. The 'Credentials' dropdown is set to '- none -'. There is an 'Advanced' button and an 'Add Repository' button at the bottom.

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

The screenshot shows the Jenkins 'Configure' screen for a project. The 'Build Environment' section is active. A dropdown menu is open under 'Build Steps', listing options like 'Execute SonarQube Scanner', 'Execute Windows batch command', 'Execute shell', etc. At the bottom of the dropdown is an 'Add build step' button.

**Execute SonarQube Scanner**

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

**Path to project properties** ?  
[Empty input field]

**Analysis properties** ?  

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

**Additional arguments** ?  
[Empty input field]

**JVM Options** ?  
[Empty input field]

Then save

Status **adv\_devops\_exp7** Add description Disable Project

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

**SonarQube** Permalinks

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

11. Go to [http://localhost:9000/<user\\_name>/permissions](http://localhost:9000/<user_name>/permissions) and allow Execute Permissions to the Admin user

Configuration Security Projects System Marketplace

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

## ***IF CONSOLE OUTPUT FAILED:***

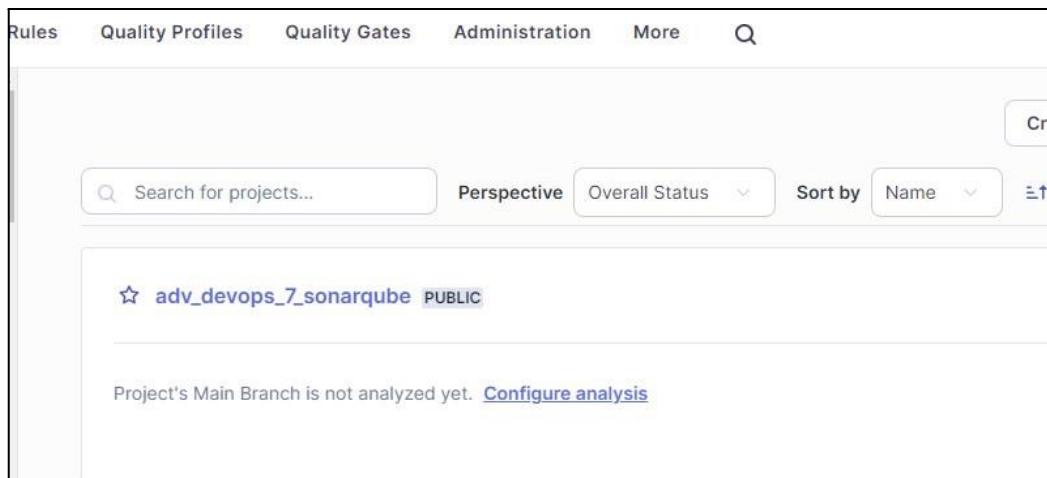
### **Step 1: Generate a New Authentication Token in SonarQube**

#### **1. Login to SonarQube:**

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

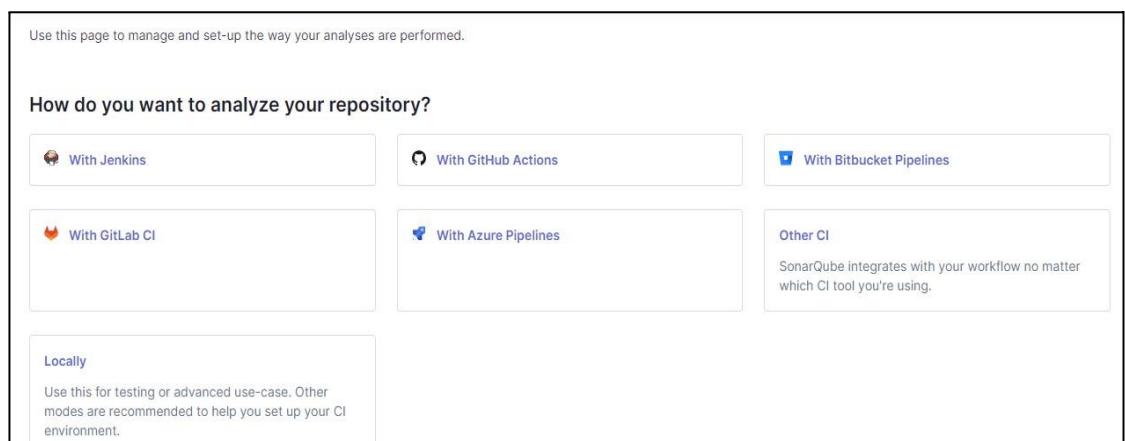
#### **2. Generate a New Token:**

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



The screenshot shows the SonarQube web interface. At the top, there is a navigation bar with tabs: Rules, Quality Profiles, Quality Gates, Administration, More, and a search icon. Below the navigation bar is a toolbar with a search bar labeled "Search for projects...", dropdown menus for "Perspective", "Overall Status", "Sort by", and "Name", and a sorting icon. The main content area displays a single project entry: "adv\_devops\_7\_sonarqube PUBLIC". Below the project name, a message states: "Project's Main Branch is not analyzed yet. [Configure analysis](#)".

- Click on **Locally**



The screenshot shows the "CI & Continuous Monitoring" section of the SonarQube configuration page. The heading is "How do you want to analyze your repository?". There are six options arranged in a grid: "With Jenkins", "With GitHub Actions", "With Bitbucket Pipelines", "With GitLab CI", "With Azure Pipelines", and "Other CI". The "Locally" option is highlighted with a border and contains the text: "Locally. Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI environment." To the right of the "Other CI" button, there is a note: "SonarQube integrates with your workflow no matter which CI tool you're using."

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

**1 Provide a token**

Generate a project token
Use existing token

Token name <small>?</small>	Expires in
"adv_devops_7.sonarqube"	1 year
<input type="button" value="Generate"/>	

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

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

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

**1 Provide a token**

```
"adv_devops_7.sonarqube": sqp_bfa5258ea4fd254f00c3d1d4e64205ebefcdd027 Delete
```

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

[Continue](#)

## Step 2: Update the Token in Jenkins

### 1. Go to Jenkins Dashboard:

- Open Jenkins and log in with your credentials.

The screenshot shows the Jenkins dashboard with the following details:

- Header:** Jenkins logo, Search bar, Notifications (1), User (Niraj Kothawade), Log out.
- Left sidebar:**
  - + New Item
  - Build History
  - Project Relationship
  - Check File Fingerprint
  - Manage Jenkins
  - My Views
- Dashboard Content:**
  - All** (selected) and **+** buttons.
  - A table with columns: S, W, Name, Last Success, Last Failure, Last Duration.
  - Jobs listed:
    - DevOps Pipeline**: Status green, Last Success 1 mo 23 days #3, Last Failure N/A, Last Duration 1.2 sec.
    - Maven Project Test**: Status red, Last Success 1 mo 23 days #1, Last Failure 1 day 7 hr #5, Last Duration 30 sec.
    - Niraj24**: Status blue, Last Success N/A, Last Failure N/A, Last Duration N/A.
- Bottom:** Build Queue button.

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

The screenshot shows the Jenkins 'Credentials' page under 'Manage Jenkins'. At the top, there's a breadcrumb navigation: Dashboard > Manage Jenkins > Credentials. Below the header, the title 'Credentials' is displayed. A table lists one credential entry:

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

Below the table, a section titled 'Stores scoped to Jenkins' is shown, containing a similar table:

P	Store ↓	Domains
File icon	User icon	System (global)

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

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

The screenshot shows the 'Add Credential' form in Jenkins. The fields are as follows:

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

At the bottom left is a blue 'Create' button.

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

The screenshot shows the 'Global credentials (unrestricted)' page under 'Manage Jenkins'. The page title is 'Global credentials (unrestricted)'. On the right, there's a blue '+ Add Credentials' button. The main content area displays a table of credentials:

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

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

SonarQube servers

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

Environment variables

SonarQube installations

List of SonarQube installations

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

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

Build Environment

Delete workspace before build starts

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

Add timestamps to the Console Output

Inspect build log for published build scans

Prepare SonarQube Scanner environment ?

Server authentication token

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

+ Add ▾

**Execute SonarQube Scanner**

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

**Path to project properties** ?

**Analysis properties** ?  
sonar.projectKey=adv\_devops\_7\_sonarqube  
sonar.host.url=http://localhost:9000  
-Dsonar.login=sqp\_568834b7b5e77a92843e4b3072e044643ce921c1  
sonar.sources=.

**Additional arguments** ?

**JVM Options** ?

## 12. Run the Jenkins build.

Dashboard > adv\_devops\_exp7 >

**Status**  **adv\_devops\_exp7**

 **SonarQube Quality Gate**

**adv\_devops\_7\_sonarqube**  **Passed**  
server-side processing:  **Success**

**Permalinks**

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

**Build History** 

Filter... /

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

## Check the console Output

The screenshot shows the Jenkins 'Console Output' page for build #6 of the 'adv\_devops\_exp7' project. The left sidebar includes links for Status, Changes, Console Output (which is selected), Edit Build Information, Delete build '#6', and Timings. The main content area displays the following log output:

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

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

The screenshot shows the SonarQube main dashboard for the 'adv\_devops\_7.sonarqube' project. The top navigation bar includes links for Overview, Issues, Security Hotspots, Measures, Code, Activity, Project Settings, and Project Information. The main content area shows the 'main' branch with a green 'Passed' status for the Quality Gate. It also displays the following metrics:

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

Below the metrics, there is a note: "The last analysis has warnings. See details".

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

# ADVANCE DEVOPS EXP 8

Niraj S. Kothawade  
D15A - 24

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

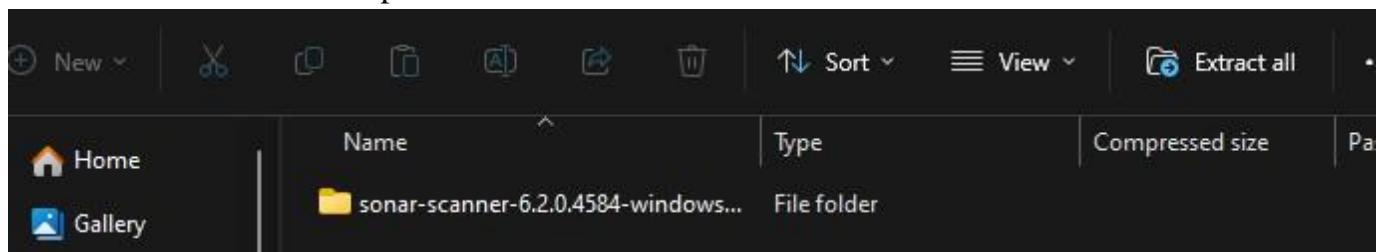
## Step 1: Download sonar scanner

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

The screenshot shows a web browser displaying the SonarScanner CLI documentation page. The URL in the address bar is <https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscan>. The page title is "SonarScanner CLI". On the left, there is a sidebar with navigation links for SonarQube, Docs 10.6, and various sections like "Analyzing source code", "Scanners", and "SonarScanner CLI". The main content area displays the "SonarScanner" section for version 6.1, released on 2024-06-27. It includes download links for Linux x64, Linux AArch64, Windows x64, macOS x64, macOS AArch64, Docker, and Any (Requires a pre-installed JVM). Below this, there are "Release notes" and a note about ARM support. At the bottom, there is a message about the SonarScanner not supporting ARM architecture yet.

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

Extract the downloaded zip file in a folder.



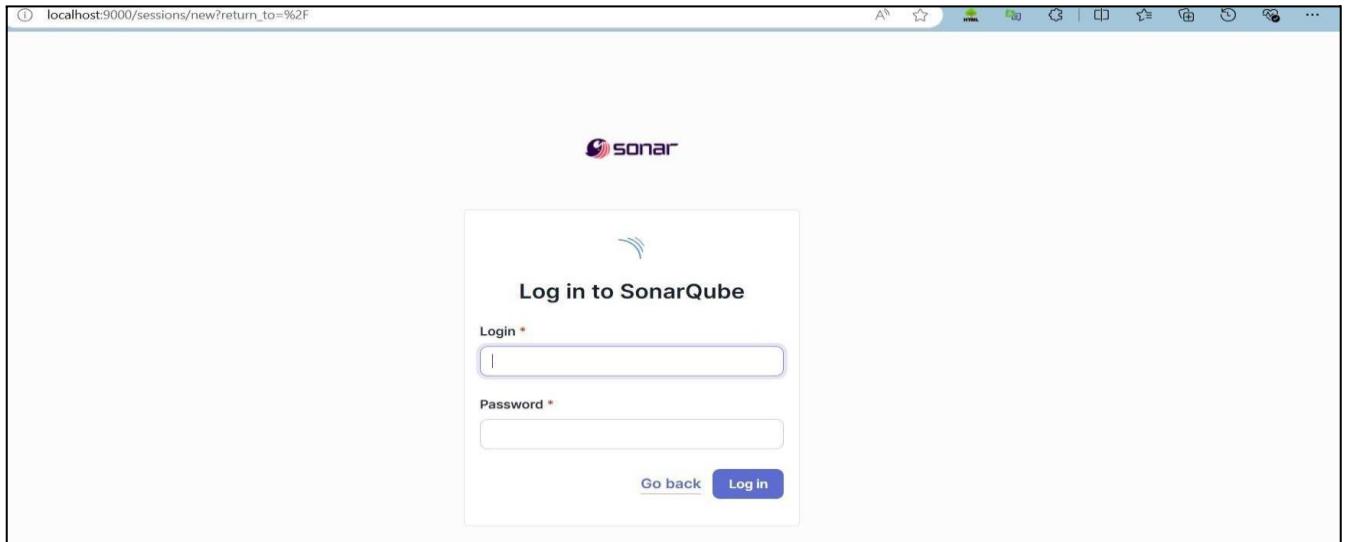
## 1. Install sonarqube image

Command: **docker pull**

**sonarqube**

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

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



SonarQube at localhost port 9000.

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

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More Q

How do you want to create your project?

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

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

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

Create a local project

4. Create a manual project in SonarQube with the name sonarqube

1 of 2

## Create a local project

**Project display name \***

**Project key \***

**Main branch name \***

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

[Cancel](#)

[Next](#)

2 of 2

## Set up project for Clean as You Code

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

Choose the baseline for new code for this project

Use the global setting

[Previous version](#)

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

Define a specific setting for this project

[Previous version](#)

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

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

The screenshot shows the Jenkins dashboard with the following details:

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

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

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

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

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

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

- Left sidebar:** Includes links for "Updates", "Available plugins" (selected), "Installed plugins", and "Advanced settings".
- Right panel:** **Download progress** section:
  - Preparation:** A list of steps: "• Checking internet connectivity", "• Checking update center connectivity", and "• Success".
  - SonarQube Scanner:** Status: "Success".
  - Loading plugin extensions:** Status: "Success".
  - Buttons:** "Go back to the top page" (with a note: "you can start using the installed plugins right away") and "Restart Jenkins when installation is complete and no jobs are running".

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

and enter the details.

Enter the Server Authentication token if needed.

In SonarQube installations: Under **Name** add <project name of sonarqube> for me  
**adv\_devops\_7\_sonarqube**

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



The screenshot shows the Jenkins configuration interface for SonarQube servers. It includes fields for Name (sonarqube), Server URL (http://localhost:9000), and a dropdown for Server authentication token (set to - none -). There are also 'Add' and 'Advanced' buttons.

Name	sonarqube
Server URL	Default is http://localhost:9000 http://localhost:9000
Server authentication token	- none - + Add Advanced

8. Search for SonarQube Scanner under Global Tool Configuration.

Choose the latest configuration and choose Install automatically.

### Dashboard > Manage Jenkins > Tools

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

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

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

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

- Name**: A text input field containing "sonarqube\_exp8".
- Install automatically**: A checkbox that is checked.
- Install from Maven Central**: A collapsed section header.
- Version**: A dropdown menu showing "SonarQube Scanner 6.2.0.4584".
- Add Installer**: A button to add more installers.

the “Install automatically” option.

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

New Item

Enter an item name  
adv\_devops\_exp8

Select an item type

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

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

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

OK

10. Under Pipeline script, enter the following:

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

}

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

Definition

Pipeline script

Script ?

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

Use Groovy Sandbox ?

[Pipeline Syntax](#)

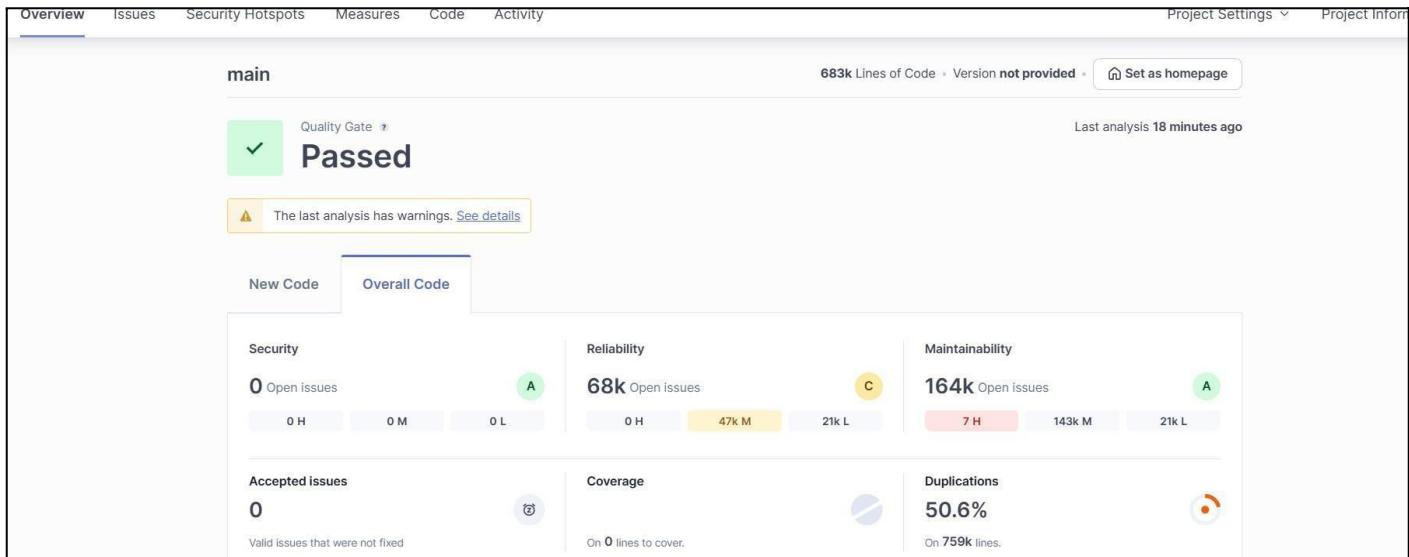
## 11. Build project

The screenshot shows the CircleCI pipeline interface for the project 'adv\_devops\_exp8'. On the left, there's a sidebar with various navigation options: Status, Changes, Build Now, Configure, Delete Pipeline, Full Stage View, SonarQube, Stages, Rename, and Pipeline Syntax. Below these are sections for Build History (with a trend dropdown) and a filter input. A specific build is highlighted: #9, dated Sep 18, 16:14, with 'No Changes'. The main area is titled 'Stage View' and displays two stages: 'Cloning the GitHub Repo' and 'SonarQube analysis'. The 'Cloning the GitHub Repo' stage took 3s. The 'SonarQube analysis' stage took 40s and failed, with a duration of 1s. The overall average stage time is 6min 2s.

## 12. Check console

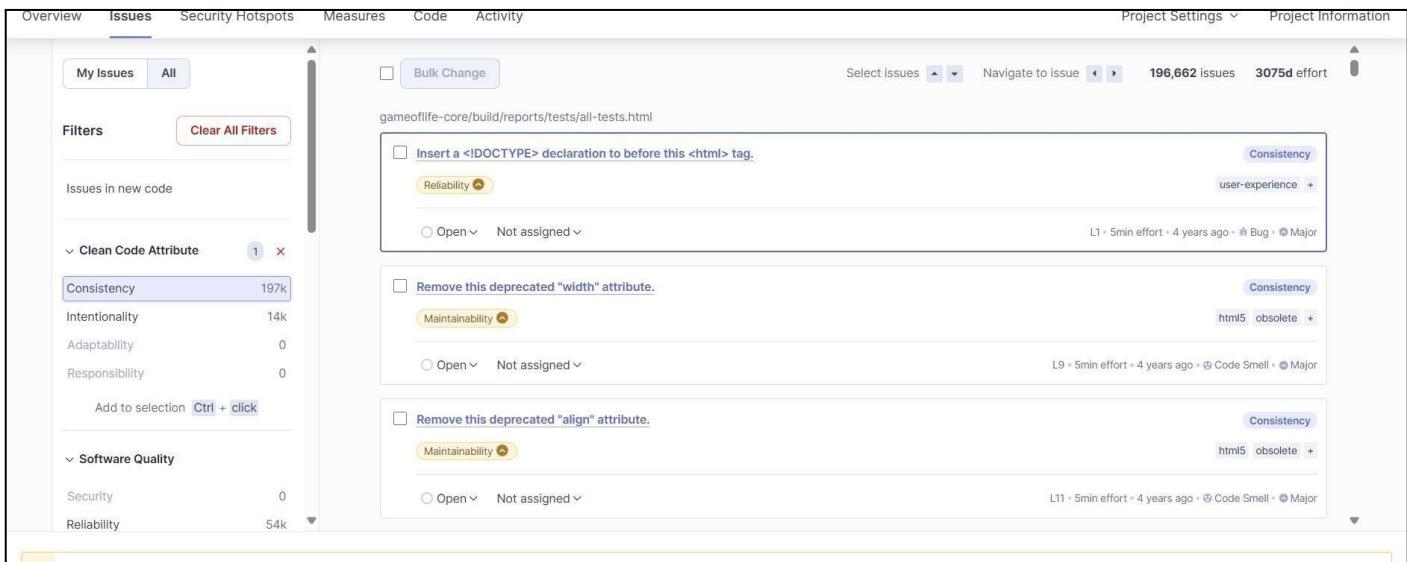
The screenshot shows the CircleCI pipeline interface for the project 'adv\_devops\_exp8'. The sidebar includes: Status, Changes, Console Output (which is selected), View as plain text, Edit Build Information, Delete build '#9', Timings, Git Build Data, Pipeline Overview, Pipeline Console, Replay, Pipeline Steps, Workspaces, and Previous Build. The main area is titled 'Console Output' and shows the log output for build #9. It starts with a note about skipping 4,246 KB of full log. The log itself consists of multiple warning messages from JMeter, all related to 'PropertyControlGui.html' files containing too many duplication references. The logs show repeated entries like '16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 512. Keep only the first 100 references.' and '16:19:49.751 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 248. Keep only the first 100 references.' These errors occur across various lines of code, such as 664, 913, 152, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 5810, 5811, 5812, 5813, 5814, 5815, 5816, 5817, 5818, 5819, 5820, 5821, 5822, 5823, 5824, 5825, 5826, 5827, 5828, 5829, 5830, 5831, 5832, 5833, 5834, 5835, 5836, 5837, 5838, 5839, 5840, 5841, 5842, 5843, 5844, 5845, 5846, 5847, 5848, 5849, 5850, 5851, 5852, 5853, 5854, 5855, 5856, 5857, 5858, 5859, 5860, 5861, 5862, 5863, 5864, 5865, 5866, 5867, 5868, 5869, 5870, 5871, 5872, 5873, 5874, 5875, 5876, 5877, 5878, 5879, 5880, 5881, 5882, 5883, 5884, 5885, 5886, 5887, 5888, 5889, 58810, 58811, 58812, 58813, 58814, 58815, 58816, 58817, 58818, 58819, 58820, 58821, 58822, 58823, 58824, 58825, 58826, 58827, 58828, 58829, 58830, 58831, 58832, 58833, 58834, 58835, 58836, 58837, 58838, 58839, 58840, 58841, 58842, 58843, 58844, 58845, 58846, 58847, 58848, 58849, 58850, 58851, 58852, 58853, 58854, 58855, 58856, 58857, 58858, 58859, 58860, 58861, 58862, 58863, 58864, 58865, 58866, 58867, 58868, 58869, 58870, 58871, 58872, 58873, 58874, 58875, 58876, 58877, 58878, 58879, 58880, 58881, 58882, 58883, 58884, 58885, 58886, 58887, 58888, 58889, 588810, 588811, 588812, 588813, 588814, 588815, 588816, 588817, 588818, 588819, 588820, 588821, 588822, 588823, 588824, 588825, 588826, 588827, 588828, 588829, 588830, 588831, 588832, 588833, 588834, 588835, 588836, 588837, 588838, 588839, 588840, 588841, 588842, 588843, 588844, 588845, 588846, 588847, 588848, 588849, 588850, 588851, 588852, 588853, 588854, 588855, 588856, 588857, 588858, 588859, 588860, 588861, 588862, 588863, 588864, 588865, 588866, 588867, 588868, 588869, 588870, 588871, 588872, 588873, 588874, 588875, 588876, 588877, 588878, 588879, 588880, 588881, 588882, 588883, 588884, 588885, 588886, 588887, 588888, 588889, 5888810, 5888811, 5888812, 5888813, 5888814, 5888815, 5888816, 5888817, 5888818, 5888819, 5888820, 5888821, 5888822, 5888823, 5888824, 5888825, 5888826, 5888827, 5888828, 5888829, 5888830, 5888831, 5888832, 5888833, 5888834, 5888835, 5888836, 5888837, 5888838, 5888839, 5888840, 5888841, 5888842, 5888843, 5888844, 5888845, 5888846, 5888847, 5888848, 5888849, 5888850, 5888851, 5888852, 5888853, 5888854, 5888855, 5888856, 5888857, 5888858, 5888859, 5888860, 5888861, 5888862, 5888863, 5888864, 5888865, 5888866, 5888867, 5888868, 5888869, 5888870, 5888871, 5888872, 5888873, 5888874, 5888875, 5888876, 5888877, 5888878, 5888879, 5888880, 5888881, 5888882, 5888883, 5888884, 5888885, 5888886, 5888887, 5888888, 5888889, 58888810, 58888811, 58888812, 58888813, 58888814, 58888815, 58888816, 58888817, 58888818, 58888819, 58888820, 58888821, 58888822, 58888823, 58888824, 58888825, 58888826, 58888827, 58888828, 58888829, 58888830, 58888831, 58888832, 58888833, 58888834, 58888835, 58888836, 58888837, 58888838, 58888839, 58888840, 58888841, 58888842, 58888843, 58888844, 58888845, 58888846, 58888847, 58888848, 58888849, 58888850, 58888851, 58888852, 58888853, 58888854, 58888855, 58888856, 58888857, 58888858, 58888859, 58888860, 58888861, 58888862, 58888863, 58888864, 58888865, 58888866, 58888867, 58888868, 58888869, 58888870, 58888871, 58888872, 58888873, 58888874, 58888875, 58888876, 58888877, 58888878, 58888879, 58888880, 58888881, 58888882, 58888883, 58888884, 58888885, 58888886, 58888887, 58888888, 58888889, 588888810, 588888811, 588888812, 588888813, 588888814, 588888815, 588888816, 588888817, 588888818, 588888819, 588888820, 588888821, 588888822, 588888823, 588888824, 588888825, 588888826, 588888827, 588888828, 588888829, 588888830, 588888831, 588888832, 588888833, 588888834, 588888835, 588888836, 588888837, 588888838, 588888839, 588888840, 588888841, 588888842, 588888843, 588888844, 588888845, 588888846, 588888847, 588888848, 588888849, 588888850, 588888851, 588888852, 588888853, 588888854, 588888855, 588888856, 588888857, 588888858, 588888859, 588888860, 588888861, 588888862, 588888863, 588888864, 588888865, 588888866, 588888867, 588888868, 588888869, 588888870, 588888871, 588888872, 588888873, 588888874, 588888875, 588888876, 588888877, 588888878, 588888879, 588888880, 588888881, 588888882, 588888883, 588888884, 588888885, 588888886, 588888887, 588888888, 588888889, 5888888810, 5888888811, 5888888812, 5888888813, 5888888814, 5888888815, 5888888816, 5888888817, 5888888818, 5888888819, 5888888820, 5888888821, 5888888822, 5888888823, 5888888824, 5888888825, 5888888826, 5888888827, 5888888828, 5888888829, 5888888830, 5888888831, 5888888832, 5888888833, 5888888834, 5888888835, 5888888836, 5888888837, 5888888838, 5888888839, 5888888840, 5888888841, 5888888842, 5888888843, 5888888844, 5888888845, 5888888846, 5888888847, 5888888848, 5888888849, 5888888850, 5888888851, 5888888852, 5888888853, 5888888854, 5888888855, 5888888856, 5888888857, 5888888858, 5888888859, 5888888860, 5888888861, 5888888862, 5888888863, 5888888864, 5888888865, 5888888866, 5888888867, 5888888868, 5888888869, 5888888870, 5888888871, 5888888872, 5888888873, 5888888874, 5888888875, 5888888876, 5888888877, 5888888878, 5888888879, 5888888880, 5888888881, 5888888882, 5888888883, 5888888884, 5888888885, 5888888886, 5888888887, 5888888888, 5888888889, 58888888810, 58888888811, 58888888812, 58888888813, 58888888814, 58888888815, 58888888816, 58888888817, 58888888818, 58888888819, 58888888820, 58888888821, 58888888822, 58888888823, 58888888824, 58888888825, 58888888826, 58888888827, 58888888828, 58888888829, 58888888830, 58888888831, 58888888832, 58888888833, 58888888834, 58888888835, 58888888836, 58888888837, 58888888838, 58888888839, 58888888840, 58888888841, 58888888842, 58888888843, 58888888844, 58888888845, 58888888846, 58888888847, 58888888848, 58888888849, 58888888850, 58888888851, 58888888852, 58888888853, 58888888854, 58888888855, 58888888856, 58888888857, 58888888858, 58888888859, 58888888860, 58888888861, 58888888862, 58888888863, 58888888864, 58888888865, 58888888866, 58888888867, 58888888868, 58888888869, 58888888870, 58888888871, 58888888872, 58888888873, 58888888874, 58888888875, 58888888876, 58888888877, 58888888878, 58888888879, 58888888880, 58888888881, 58888888882, 58888888883, 58888888884, 58888888885, 58888888886, 58888888887, 58888888888, 58888888889, 588888888810, 588888888811, 588888888812, 588888888813, 588888888814, 588888888815, 588888888816, 588888888817, 588888888818, 588888888819, 588888888820, 588888888821, 588888888822, 588888888823, 588888888824, 588888888825, 588888888826, 588888888827, 588888888828, 588888888829, 588888888830, 588888888831, 588888888832, 588888888833, 588888888834, 588888888835, 588888888836, 588888888837, 588888888838, 588888888839, 588888888840, 588888888841, 588888888842, 588888888843, 588888888844, 588888888845, 588888888846, 588888888847, 588888888848, 588888888849, 588888888850, 588888888851, 588888888852, 588888888853, 588888888854, 588888888855, 588888888856, 588888888857, 588888888858, 588888888859, 588888888860, 588888888861, 588888888862, 588888888863, 588888888864, 588888888865, 588888888866, 588888888867, 588888888868, 588888888869, 588888888870, 588888888871, 588888888872, 588888888873, 588888888874, 588888888875, 588888888876, 588888888877, 588888888878, 588888888879, 588888888880, 588888888881, 588888888882, 588888888883, 588888888884, 588888888885, 588888888886, 588888888887, 588888888888, 588888888889, 5888888888810, 5888888888811, 5888888888812, 5888888888813, 5888888888814, 5888888888815, 5888888888816, 5888888888817, 5888888888818, 5888888888819, 5888888888820, 5888888888821, 5888888888822, 5888888888823, 5888888888824, 5888888888825, 5888888888826, 5888888888827, 5888888888828, 5888888888829, 5888888888830, 5888888888831, 5888888888832, 5888888888833, 5888888888834, 5888888888835, 5888888888836, 5888888888837, 5888888888838, 5888888888839, 5888888888840, 5888888888841, 5888888888842, 5888888888843, 5888888888844, 5888888888845, 5888888888846, 5888888888847, 5888888888848, 5888888888849, 5888888888850, 5888888888851, 5888888888852, 5888888888853, 5888888888854, 5888888888855, 5888888888856, 5888888888857, 5888888888858, 5888888888859, 5888888888860, 5888888888861, 5888888888862, 5888888888863, 5888888888864, 5888888888865, 5888888888866, 5888888888867, 5888888888868, 5888888888869, 5888888888870, 5888888888871, 5888888888872, 5888888888873, 5888888888874, 5888888888875, 5888888888876, 5888888888877, 5888888888878, 5888888888879, 5888888888880, 5888888888881, 5888888888882, 5888888888883, 5888888888884, 5888888888885, 5888888888886, 5888888888887, 5888888888888, 5888888888889, 58888888888810, 58888888888811, 58888888888812, 58888888888813, 58888888888814, 58888888888815, 58888888888816, 58888888888817, 58888888888818, 58888888888819, 5888888888

## 13. Now, check the project in SonarQube



## 14. Code Problems

### • Consistency



# Intentionality

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

**My Issues** All

**Bulk Change**

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

**Filters** Clear All Filters

Issues in new code

✓ Clean Code Attribute 1 1 X

Consistency 197k

Intentionality 14k

Adaptability 0

Responsibility 0

Add to selection Ctrl + click

✗ Software Quality

Security 0

Reliability 14k

gameoflife-acceptance-tests/Dockerfile

Use a specific version tag for the image. Intentionality

Maintainability No tags +

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

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

Maintainability No tags +

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

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

Maintainability No tags +

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

## Bugs

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

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

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

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

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

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

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

## Code Smells

gameoflife-acceptance-tests/Dockerfile

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

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

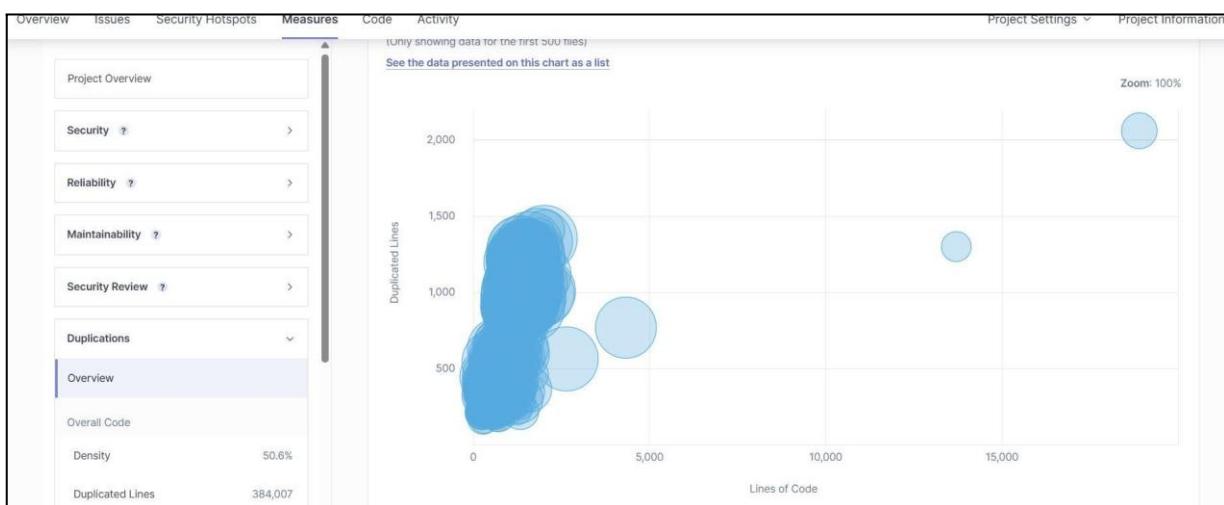
Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. Intentionality  
Maintainability No tags +

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

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

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

## Duplications



## Cyclomatic Complexities

The screenshot shows the SonarQube interface for a project named "gameoflife". The top navigation bar includes links for Overview, Issues, Security Hotspots, Measures, Code, Activity, Project Settings, and Project Information. The "Measures" tab is currently selected. On the left, a sidebar lists various metrics: Security, Reliability, Maintainability, Security Review, Duplications, Size, Complexity, and Cyclomatic Complexity (which is highlighted in blue). The main content area displays the "Cyclomatic Complexity" report, which shows a total of 1,112 complexities. The report lists several components and their complexity counts:

Component	Complexity Count
gameoflife-acceptance-tests	—
gameoflife-build	—
gameoflife-core	18
gameoflife-deploy	—
gameoflife-web	1,094
pom.xml	—

At the bottom of the report, it says "6 of 6 shown".

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

# Advanced DevOps Exp-9

Niraj S. Kothawade

D15A – 24

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

## Procedure:-

### Step 1: Create an EC2 Instance and name it as nagios-host

The screenshot shows the AWS EC2 Instances page. The left sidebar includes links for EC2 Dashboard, EC2 Global View, Events, Console-to-Code Preview, and Instances. The main area displays a table of instances with columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability. Two instances are listed: 'aws-cloud9-W...' (terminated, t2.micro, status -) and 'nagios\_host' (running, t2.micro, initializing). A 'Launch instances' button is at the top right.

### Step 2: Under the security groups, click on edit inbound rules and set as shown in the figure below

The screenshot shows the AWS Security Groups page. The left sidebar includes links for EC2, Services, Cloud9, and IAM. The main area shows the 'Edit inbound rules' section for a security group named 'sg-01ef0d8aa2a9e64cc3'. It lists seven inbound rules with details like Type, Protocol, Port range, Source, and Description. Most rules have '0.0.0.0/0' as the source. A 'Delete' button is present for each rule.

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-08a541db6665fe82fe	Custom TCP	TCP	5666	Custom	0.0.0.0/0
sgr-03c975f755e85f40c	All ICMP - IPv6	IPv6 ICMP	All	Custom	-/0
sgr-01babcc816f0a0d75	HTTP	TCP	80	Custom	-/0
sgr-07e2ebc18ff0e4246	SSH	TCP	22	Custom	0.0.0.0/0
sgr-051a509fc38ce37d	HTTPS	TCP	443	Custom	0.0.0.0/0
sgr-0a675a9e058ee37f0	All ICMP - IPv4	ICMP	All	Custom	0.0.0.0/0
sgr-07420d0scb92b59fd	All traffic	All	All	Custom	0.0.0.0/0

### Step 3: Now, run the following commands -

```
sudo su
```

```
sudo yum update
```

```
sudo yum install httpd php
```

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

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

```
[ec2-user@ip-172-31-16-211 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:04:23 ago on Sun Sep 29 04:05:53 2024.
Dependencies resolved.
```

Package	Architecture	Version
Installing:		
httpd	x86_64	2.4.62-1.amzn2023
php8.3	x86_64	8.3.10-1.amzn2023.0.1
Installing dependencies:		
apr	x86_64	1.7.2-2.amzn2023.0.2
apr-util	x86_64	1.6.3-1.amzn2023.0.1

```
[ec2-user@ip-172-31-16-211 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:04:54 ago on Sun Sep 29 04:05:53 2024.
Package glibc-2.34-52.amzn2023.0.11.x86_64 is already installed.
Package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Dependencies resolved.
```

Package	Architecture	Version
Installing:		
gcc	x86_64	11.4.1-2.amzn2023.0.2
Installing dependencies:		
annobin-docs	noarch	10.93-1.amzn2023.0.1
annobin-plugin-gcc	x86_64	10.93-1.amzn2023.0.1
c++	x86_64	11.4.1-2.amzn2023.0.2

```
[ec2-user@ip-172-31-16-211 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:05:21 ago on Sun Sep 29 04:05:53 2024.
Dependencies resolved.
```

Package	Architecture	Version
Installing:		
gd	x86_64	2.3.3-5.amzn2023.0.3
gd-devel	x86_64	2.3.3-5.amzn2023.0.3

#### **Step 4: Create a new nagios user with its password.**

```
sudo adduser -m nagios  
sudo passwd nagios sudo  
groupadd nagcmd  
sudo usermod -a -G nagcmd nagios  
sudo usermod -a -G nagcmd apache
```

```
[ec2-user@ip-172-31-16-211 ~]$ sudo adduser -m nagios  
[ec2-user@ip-172-31-16-211 ~]$ sudo passwd nagios  
Changing password for user nagios.  
New password:  
BAD PASSWORD: The password is shorter than 8 characters  
Retype new password:  
passwd: all authentication tokens updated successfully.
```

```
[ec2-user@ip-172-31-16-211 ~]$ sudo groupadd nagcmd  
[ec2-user@ip-172-31-16-211 ~]$ sudo usermod -a -G nagcmd nagios  
[ec2-user@ip-172-31-16-211 ~]$ sudo usermod -a -G nagcmd apache
```

#### **Step 5: Now, run the following commands -**

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

```
http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz wget  
http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz tar zxvf nagios-  
4.0.8.tar.gz
```

```
[ec2-user@ip-172-31-16-211 ~]$ mkdir ~/downloads  
[ec2-user@ip-172-31-16-211 ~]$ cd ~/downloads  
[ec2-user@ip-172-31-16-211 downloads]$ █
```

```
[ec2-user@ip-172-31-16-211 downloads]$ wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz  
--2024-09-29 04:17:09-- 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]  
--2024-09-29 04:17:10-- http://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz
```

```
[ec2-user@ip-172-31-16-211 downloads]$ wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz  
--2024-09-29 04:17:25-- 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%[=====] 2.54M  1.68MB/s   in 1.5s  
2024-09-29 04:17:27 (1.68 MB/s) - 'nagios-plugins-2.0.3.tar.gz' saved [2659772/2659772]
```

```
[ec2-user@ip-172-31-16-211 downloads]$ tar zxvf nagios-4.0.8.tar.gz
nagios-4.0.8/
nagios-4.0.8/.gitignore
nagios-4.0.8/Changelog
nagios-4.0.8/INSTALLING
nagios-4.0.8/LEGAL
nagios-4.0.8/LICENSE
nagios-4.0.8/Makefile.in
nagios-4.0.8/README
nagios-4.0.8/README.asciidoc
nagios-4.0.8/THANKS
nagios-4.0.8/UPGRADING
nagios-4.0.8/base/
nagios-4.0.8/base/.gitignore
nagios-4.0.8/base/Makefile.in
nagios-4.0.8/base/nagios.cfg
nagios-4.0.8/base/nagios.cfg.bak
nagios-4.0.8/base/nagios.cfg.old
[ec2-user@ip-172-31-16-211 downloads]$ cd nagios-4.0.8
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$
```

**Step 6:** Now to run the configuration script run the following command.

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

```
[ec2-user@ip-172-31-16-211 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 C99... none needed
```

**Step 7: Now, to compile the source code run the following command - make all**

sudo make install sudo

**make install-init**

`sudo make install-config`

```
sudo make install-commandmode
```

```
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ make all
cd ./base && make
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/base'
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o broker.o broker.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nebmods.o nebmods.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ..//common/shared.o ..//common/shared.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nerd.o nerd.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o query-handler.o query-handler.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
In function 'get_wproc_list',
  inlined from 'get_worker' at workers.c:224:12:
workers.c:209:17: warning: '%s' directive argument is null [-Wformat-overflow=]
  209 |         log_debug_info(DEBUG_CHECKS, 1, "Found specialized worker(s) for '%s'", (slash && *slash != '/') ? slash : cmd_name);
    |         ^
  ~~~~~
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o checks.o checks.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o config.o config.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o commands.o commands.c
commands.c: In function 'process_passive_service_check':
commands.c:2247:19: warning: assignment discards 'const' qualifier from pointer target type [-Wdiscarded-qualifiers]
  2247 |         cr.source = command_worker.source_name;
    |         ^
commands.c: In function 'process_passive_host_check':
```

```
c2-user@ip-172-31-16-211 nagios-4.0.8]$ ./base/..//common/shared.c:24: f
collect2: error: ld returned 1 exit status
make[1]: *** [Makefile:177: archivejson.cgi] Error 1
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-
make: *** [Makefile:72: all] Error 2
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ █
```

```
c2-user@ip-172-31-16-211 nagios-4.0.8$ ./base/..//common/shared.c:24: first undefined r
collect2: error: ld returned 1 exit status
make[1]: *** [Makefile:177: archivejson.cgi] Error 1
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
make: *** [Makefile:72: all] Error 2
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ █
```

```
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ sudo make install
cd ./base && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/base'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -m 774 -o nagios -g nagios nagiostats /usr/local/nagios/bin
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
make strip-post-install
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/base'
/usr/bin/strip /usr/local/nagios/bin/nagios
/usr/bin/strip /usr/local/nagios/bin/nagiostats
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/sbin
for file in *.cgi; do \
    /usr/bin/install -c -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin; \
done
/usr/bin/install: cannot stat '*.cgi': No such file or directory
make[2]: *** [Makefile:205: install-basic] Error 1
```

```
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /etc/rc.d/init.d
/usr/bin/install -c -m 755 -o root -g root daemon-init /etc/rc.d/init.d/nagios

*** Init script installed ***

[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ █
```

```
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ sudo make install-config
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cgi /usr/local/nagios/etc/cgi.cgi
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg

*** Config files installed ***

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

```
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ █
```

```
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ █
```

## **Step 8: Edit the config file and change the email address. sudo nano**

/usr/local/nagios/etc/objects/contacts.cfg

```
define contact{
    contact_name          nagiosadmin      ; Short name of user
    use                   generic-contact   ; Inherit default values from generic-contact template (defined above)
    alias                Nagios Admin     ; Full name of user
    email                2022.niraj.kothawade@ves.ac.in ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
```

## **Step 9: Now run the following commands –**

sudo make install-webconf

sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin sudo  
service httpd restart

cd ~/downloads

tar zxvf nagios-plugins-2.0.3.tar.gz

```
[ec2-user@ip-172-31-16-211 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 ***
```

```
[ec2-user@ip-172-31-16-211 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-172-31-16-211 nagios-4.0.8]$ 
```

```
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ 
```

```
[ec2-user@ip-172-31-16-211 nagios-4.0.8]$ cd ~/downloads
[ec2-user@ip-172-31-16-211 downloads]$ tar zxvf nagios-plugins-2.0.3.tar.gz
nagios-plugins-2.0.3/
nagios-plugins-2.0.3/perlmods/
nagios-plugins-2.0.3/perlmods/Config-Tiny-2.14.tar.gz
nagios-plugins-2.0.3/perlmods/parent-0.226.tar.gz
nagios-plugins-2.0.3/perlmods/Test-Simple-0.98.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile.in
nagios-plugins-2.0.3/perlmods/version-0.9903.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile.am
nagios-plugins-2.0.3/perlmods/Module-Runtime-0.013.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Metadata-1.000014.tar.gz
nagios-plugins-2.0.3/perlmods/Params-Validate-1.08.tar.gz
nagios-plugins-2.0.3/perlmods/Class-Accessor-0.34.tar.gz
nagios-plugins-2.0.3/perlmods/Try-Tiny-0.18.tar.gz
```

## Step 10: Compile and install plugins

```
cd nagios-plugins-2.0.3
```

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

```
sudo make install
```

```
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ ./configure --with-nagios-user=nagios --with-nagios-group=nagios
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether to disable maintainer-specific portions of Makefiles... yes
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
checking for gcc... gcc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
```

```
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ sudo make install
```

```
make  install-exec-hook
make[3]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.0.3/plugins'
cd /usr/local/nagios/libexec && \
for i in check_ftp check_imap check_ntp check_pop check_udp check_clamd ; do rm -f $i; ln -s ./check_$i $i; done
if [ -x check_ldap ] ; then rm -f check_ldaps ; ln -s check_ldap check_ldaps ; fi
make[3]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.0.3/plugins'
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.0.3/plugins'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.0.3/plugins'
Making install in plugins-scripts
make[1]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.0.3/plugins-scripts'
make[2]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.0.3/plugins-scripts'
test -z "/usr/local/nagios/libexec" || /usr/bin/mkdir -p "/usr/local/nagios/libexec"
/usr/bin/install -c -o nagios -g nagios check_breeze check_disk_smb check_flexlm check_ircd
tus check_ifoperstatus check_mailq check_file_age utils.sh utils.pm '/usr/local/nagios/libexec'
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.0.3/plugins-scripts'
```

## Step 11: To start nagios run the following commands –

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

```
sudo chkconfig nagios
```

on Verify using the following command

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

```
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ sudo chkconfig --add nagios
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ sudo chkconfig nagios on
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ █
```

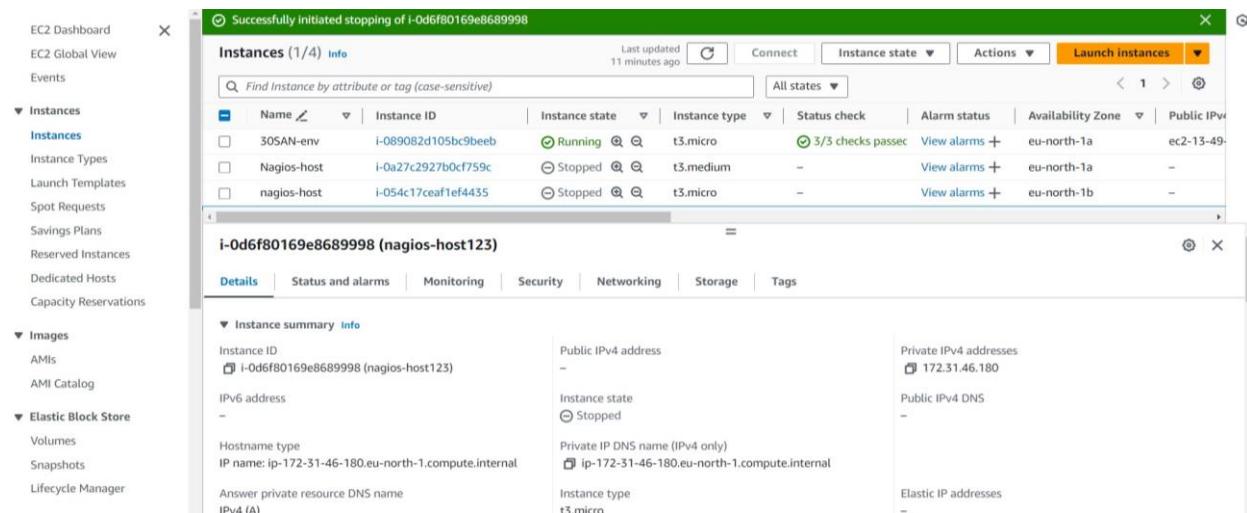
```
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ sudo chkconfig --add nagios
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ sudo chkconfig nagios on
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.0.8
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 08-12-2014
License: GPL

Website: http://www.nagios.org
Reading configuration data...
Error in configuration file '/usr/local/nagios/etc/nagios.cfg' - Line 452 (Check result path '/usr/local/nagios/var/spool/checkresults' is not a valid directory
    Error processing main config file!
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ █
```

```
[ec2-user@ip-172-31-16-211 nagios-plugins-2.0.3]$ sudo service nagios start
Starting nagios (via systemctl): [ OK ]
```

## Step 12: Go to EC2 instance and copy the public IP address of the instance



**Step 13: Now visit <http://<your IP address>/nagios> Enter correct credentials and then you will see this page**

The screenshot shows the Nagios Core 4.4.6 web interface. At the top, a browser window displays a 'Sign in' dialog box with the URL <http://51.20.118.97/nagios>. The dialog box contains fields for 'Username' (nagiosadmin) and 'Password' (redacted), along with 'Sign in' and 'Cancel' buttons. Below the dialog, the main Nagios dashboard is visible. The dashboard features the Nagios logo and the text 'Nagios® Core™ Version 4.4.6 April 28, 2020 Check for updates'. A blue banner at the bottom left of the dashboard area says 'A new version of Nagios Core is available! Visit [nagios.org](http://nagios.org) to download Nagios 4.5.5.' On the left side, there is a sidebar with links for General (Home, Documentation), Current Status (Tactical Overview, Map (Legacy), Hosts, Services, Host Groups, Summary, Grid, Service Groups, Summary, Grid), Problems (Services (Unhandled), Hosts (Unhandled), Network Outages), Reports (Availability, Trends (Legacy), Alerts, History, Summary, Histogram (Legacy)), Notifications (Event Log), and System (Comments, Downtime, Process Info, Performance Info, Scheduling Queue, Configuration). The main content area includes sections for 'Get Started' (with a bulleted list of steps), 'Latest News' (empty), 'Quick Links' (with a bulleted list of links), and 'Don't Miss...' (empty). A 'Page Tour' link is located on the right edge of the dashboard.

# Advance DevOps Exp 10

Niraj S. Kothawade  
D15A - 24

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

## Procedure:-

Check if the nagios service is running by executing following command

```
ubuntu@ip-172-31-89-161:~$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-09-28 16:08:58 UTC; 1min 2s ago
     Docs: https://www.nagios.org/documentation
 Process: 15743 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Process: 15753 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 15764 (nagios)
   Tasks: 6 (limit: 1130)
  Memory: 2.4M (peak: 3.2M)
    CPU: 29ms
   CGroup: /system.slice/nagios.service
           ├─15764 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           ├─15765 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─15766 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─15767 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─15768 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           └─15769 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 28 16:08:58 ip-172-31-89-161 nagios[15764]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Sep 28 16:08:58 ip-172-31-89-161 nagios[15764]: qh: core query handler registered
Sep 28 16:08:58 ip-172-31-89-161 nagios[15764]: qh: echo service query handler registered
Sep 28 16:08:58 ip-172-31-89-161 nagios[15764]: qh: help for the query handler registered
Sep 28 16:08:58 ip-172-31-89-161 nagios[15764]: wproc: Successfully registered manager as @wproc with query handler
Sep 28 16:08:58 ip-172-31-89-161 nagios[15764]: wproc: Registry request: name=Core Worker 15765;pid=15765
Sep 28 16:08:58 ip-172-31-89-161 nagios[15764]: wproc: Registry request: name=Core Worker 15766;pid=15766
Sep 28 16:08:58 ip-172-31-89-161 nagios[15764]: wproc: Registry request: name=Core Worker 15767;pid=15767
```

sudo systemctl status nagios

Now, create a new EC2 instance on AWS

Instances (2) <a href="#">Info</a>		Last updated <a href="#">C</a>	Connect	Instance state <a href="#">▼</a>	Actions <a href="#">▼</a>	<a href="#">Launch instances</a> <a href="#">▼</a>		
<a href="#">Find Instance by attribute or tag (case-sensitive)</a>				All states <a href="#">▼</a>				
<input type="checkbox"/>	Name <a href="#">▼</a>	Instance ID	Instance state <a href="#">▼</a>	Instance type <a href="#">▼</a>	Status check	Alarm status	Availability Zone <a href="#">▼</a>	P
<input type="checkbox"/>	nagios-host	i-09e8ea019f24f4be2	<span>Running</span> <a href="#">@</a> <a href="#">Q</a>	t2.micro	<span>2/2 checks passed</span> <a href="#">View alarms</a> +	<a href="#">View alarms</a> +	us-east-1c	e
<input type="checkbox"/>	linux-client	i-0ad38836f030e3784	<span>Running</span> <a href="#">@</a> <a href="#">Q</a>	t2.micro	<span>Initializing</span> <a href="#">View alarms</a> +	<a href="#">View alarms</a> +	us-east-1c	e

Now perform the following commands on nagios-host EC2 instance.

On the server, run this command

```
ubuntu@ip-172-31-89-161:~$ ps -ef | grep nagios
nagios 15764 1 0 16:08 ? 00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios 15765 15764 0 16:08 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 15766 15764 0 16:08 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 15767 15764 0 16:08 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 15768 15764 0 16:08 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 15769 15764 0 16:08 ? 00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ubuntu 15957 1342 0 16:13 pts/0 00:00:00 grep --color=auto nagios
ubuntu@ip-172-31-89-161:~$
```

ps -ef | grep nagios

Become a root user and create 2 folders

sudo su

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

```
ubuntu@ip-172-31-89-161:~$ sudo su  
mkdir /usr/local/nagios/etc/objects/monitorhosts  
mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts  
root@ip-172-31-89-161:/home/ubuntu#
```

Copy localhost.cfg file to the mentioned location

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

```
root@ip-172-31-89-161:/usr/local/nagios/etc/objects# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts  
cp: cannot create regular file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts': No such file or directory  
root@ip-172-31-89-161:/usr/local/nagios/etc/objects# sudo mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts  
root@ip-172-31-89-161:/usr/local/nagios/etc/objects# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts  
root@ip-172-31-89-161:/usr/local/nagios/etc/objects#
```

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

Open the nano editor for localhost.cfg file and make these changes. Add the Ip address of the linux-client for the address field.

```
nano
```

```
GNU nano 7.2                                     /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/localhost.cfg  
#####  
#  
# HOST DEFINITION  
#  
#####  
  
# Define a host for the local machine  
  
define host {  
  
    use          linux-server ; Name of host template  
    ; This host definition is used as a base for other host definitions  
    ; in (or inherited by) other host definitions.  
    host_name    linuxserver  
    alias        linuxserver  
    address      52.207.253.18  
}  
  
#####  
#  
# HOST GROUP DEFINITION  
  
^G Help      ^O Write Out     ^W Where Is      ^K Cut      ^T Exit  
^X Exit      ^R Read File     ^\ Replace      ^U Paste     ^J Jump  
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/localhost.cfg
```

Note - Here replace hostname with linuxserver

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

Add the following line to the nagios.cfg file

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

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

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

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

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

cfg\_dir=/usr/local/nagios/etc/objects/monitorhosts/

After making the changes in nagios.cfg file now check validate the file by typing the following command in the terminal.

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

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

Total Warnings: 0
Total Errors: 0

Things look okay - No serious problems were detected during the pre-flight check
root@ip-172-31-89-161:/usr/local/nagios/etc/objects/monitorhosts/linuxhosts#
```

Now restart the service by using this command

```

root@ip-172-31-89-161:/usr/local/nagios/etc/objects/monitorhosts/linuxhosts# service nagios restart
root@ip-172-31-89-161:/usr/local/nagios/etc/objects/monitorhosts/linuxhosts# systemctl status nagios
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-09-28 17:36:35 UTC; 19s ago
     Docs: https://www.nagios.org/documentation
  Process: 1870 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
  Process: 1872 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 1874 (nagios)
    Tasks: 8 (limit: 1130)
   Memory: 3.0M (peak: 3.2M)
      CPU: 24ms
     CGroupl: /system.slice/nagios.service
           |-1874 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           |-1875 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           |-1876 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           |-1877 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           |-1878 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           |-1879 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           |-1880 /usr/local/nagios/libexec/check_ping -H 52.207.253.18 -w 3000.0,80% -c 5000.0,100% -p 5
           └─1881 /usr/bin/ping -n -U -w 30 -c 5 52.207.253.18

Sep 28 17:36:35 ip-172-31-89-161 nagios[1874]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Sep 28 17:36:35 ip-172-31-89-161 nagios[1874]: qh: core query handler registered
Sep 28 17:36:35 ip-172-31-89-161 nagios[1874]: qh: echo service query handler registered
Sep 28 17:36:35 ip-172-31-89-161 nagios[1874]: qh: help for the query handler registered
Sep 28 17:36:35 ip-172-31-89-161 nagios[1874]: wproc: Successfully registered manager as @wproc with query handler
Sep 28 17:36:35 ip-172-31-89-161 nagios[1874]: wproc: Registry request: name=Core Worker 1875;pid=1875
lines 1-26

```

service nagios restart

Now using this command update the apt repository of ubuntu (linux-client),  
install gcc, nagios-nrpe-server and nagios-plugin

sudo apt update -y

sudo apt install gcc -y

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

Now open nrpe.cfg file and add the ip address of the nagios host as shown. To  
open the nrpe.cfg file copy this command.

```

# _supported.
#
# Note: The daemon only does rudimentary checking on the
# address. I would highly recommend adding entries to your
# file to allow only the specified host to connect if you
# are running this daemon on.
#
# NOTE: This option is ignored if NRPE is running in
#       daemon mode.
allowed_hosts=127.0.0.1,54.167.169.0

#
# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE
# to specify arguments to commands that are executed
# if the daemon was configured with the --enable-command
# option.

```

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

Now restart nrpe server by using this command  
sudo systemctl restart nagios-nrpe-server

Now, check nagios dashboard, you should see linuxserver up and running, if not

The screenshot shows the Nagios interface with the following sections:

- Current Network Status:** Last Updated: 5h Sep 28 18:47:41 UTC 2024. Logged in as nagiosadmin.
- Host Status Totals:** Up: 2, Down: 0, Unreachable: 0, Pending: 0. All Problems: 0, All Types: 2.
- Service Status Totals:** Ok: 12, Warning: 0, Unknown: 0, Critical: 4, Pending: 0. All Problems: 4, All Types: 16.
- Host Status Details For All Host Groups:** Shows two hosts: linuxserver and localhost, both marked as UP.
- Reports:** Availability, Trends (Legacy), Alerts, History, Summary, Histogram (Legacy), Notifications, Event Log.
- System:** Comments, Downtime, Process Info, Performance Info, Scheduling Queue, Configuration.

check security groups of the EC2 instances



# Adv. DevOps Exp. 11

**Niraj S. Kothawade**

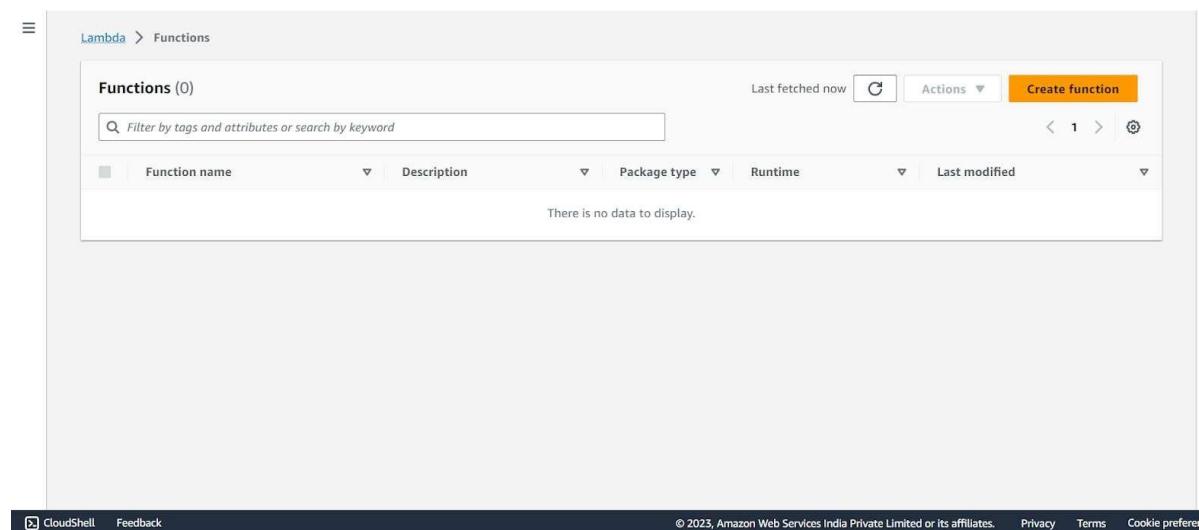
**D15A - 24**

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

## Steps to create an AWS Lambda function

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

Be mindful of where you create your functions since Lambda is region-dependent.



The screenshot shows the AWS Lambda Functions page. At the top, there is a breadcrumb navigation: Lambda > Functions. Below the breadcrumb, a header bar includes a 'Create function' button. A search bar labeled 'Filter by tags and attributes or search by keyword' is present. Underneath the search bar is a table with columns: Function name, Description, Package type, Runtime, and Last modified. The table displays the message 'There is no data to display.' At the bottom of the page, there are links for CloudShell, Feedback, Privacy, Terms, and 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.  
 ▼ G

Architecture Info  
Choose the instruction set architecture you want for your function code.  
 x86\_64  
 arm64

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

Lambda > Functions > Create function Info

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

Author from scratch  
Start with a simple Hello World example.

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

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

#### Basic information

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

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/app1> © 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.  
 ▼ G

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.

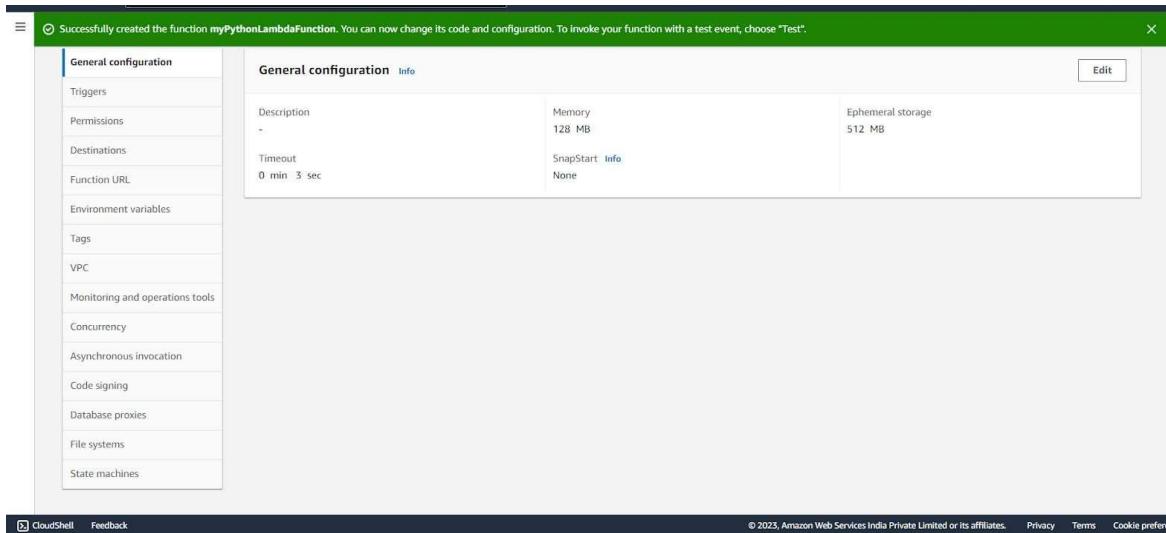
The screenshot shows the AWS Lambda Functions console. A green banner at the top indicates "Successfully created the function myPythonLambdaFunction. You can now change its code and configuration. To invoke your function with a test event, choose 'Test'." Below the banner, the function name "myPythonLambdaFunction" is displayed. The "Function overview" section includes a thumbnail of the function, a "Layers" button, and buttons for "+ Add trigger" and "+ Add destination". On the right, there is a "Description" field, "Last modified" (15 seconds ago), "Function ARN" (arn:aws:lambda:ap-south-1:447953971928:function:myPythonLambdaFunction), and a "Function URL" link. At the bottom of the overview section, 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:

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

This screenshot is identical to the one above, but the "Configuration" tab is now selected at the bottom of the navigation bar. The rest of the interface and code editor content remain the same.

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

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



The screenshot shows the "Edit basic settings" page for the function. The navigation bar includes the AWS logo, Services, a search bar, and a keyboard shortcut [Alt+S]. The breadcrumb trail is: Lambda > Functions > myPythonLambdaFunction > Edit basic settings. The title "Edit basic settings" is centered above the configuration form.

**Basic settings**

Description - optional
<input type="text"/>

**Memory** [Info](#)  
Your function is allocated CPU proportional to the memory configured.  
 MB  
Set memory to between 128 MB and 10240 MB.

**Ephemeral storage** [Info](#)  
You can configure up to 10 GB of ephemeral storage (/tmp) for your function. [View pricing](#)  
 MB  
Set ephemeral storage (/tmp) to between 512 MB and 10240 MB.

**SnapStart** [Info](#)  
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 [SnapStart compatibility considerations](#).  
  
Supported runtimes: Java 11, Java 17.

**Timeout**  
 min  sec

**Execution role**

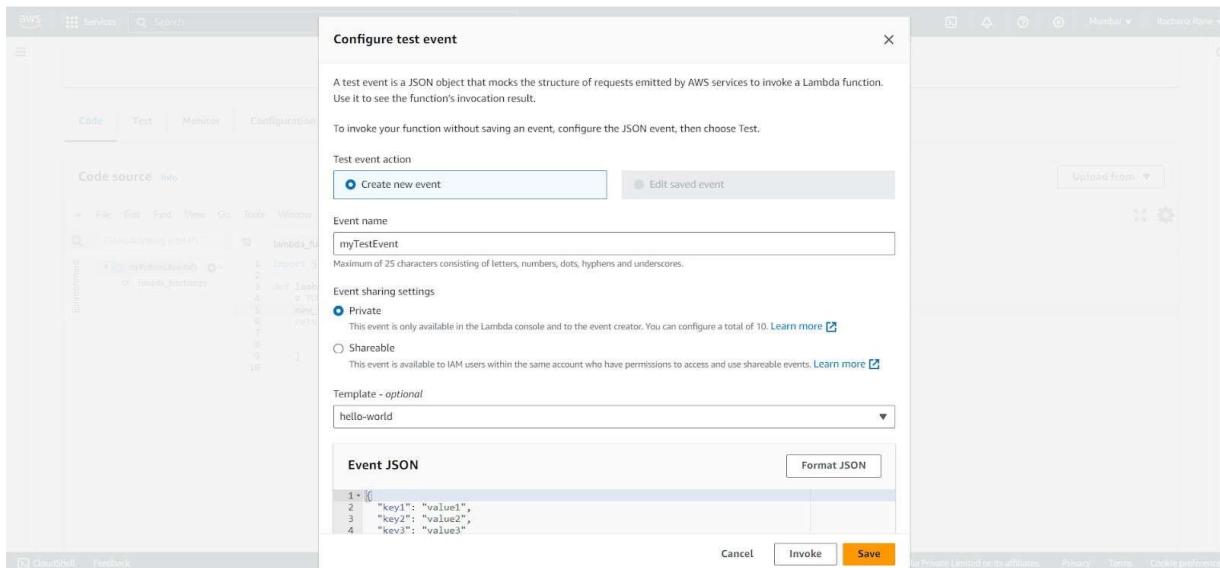
CloudShell Feedback

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.

The screenshot shows the AWS Lambda function configuration interface. At the top, there are tabs for Code, Test, Monitor, Configuration, Aliases, and Versions. The Code tab is selected. Below the tabs is a toolbar with File, Edit, Find, View, Go, Tools, Window, a dropdown for Test, Deploy, and a status message 'Changes not deployed'. On the left, there's a sidebar for Environment and a file browser showing a folder 'myPythonLambdaFn' containing a file 'lambda\_function.py'. The code editor displays the following Python code:

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

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



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

The screenshot shows the AWS Lambda Test interface. At the top, a green banner displays the message: "The test event myTestEvent was successfully saved." Below this, the main interface has tabs for "File", "Edit", "Find", "View", "Go", "Tools", "Window", "Test" (which is selected), "Deploy", and "Changes not deployed".

The left sidebar shows the environment structure: "myPythonLambdaFn" (selected) and "lambda\_function.py". Under "Execution results", there is a section for "Test Event Name: myTestEvent". The "Response" field contains the following JSON:

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

The "Function Logs" section shows the request and response details:

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

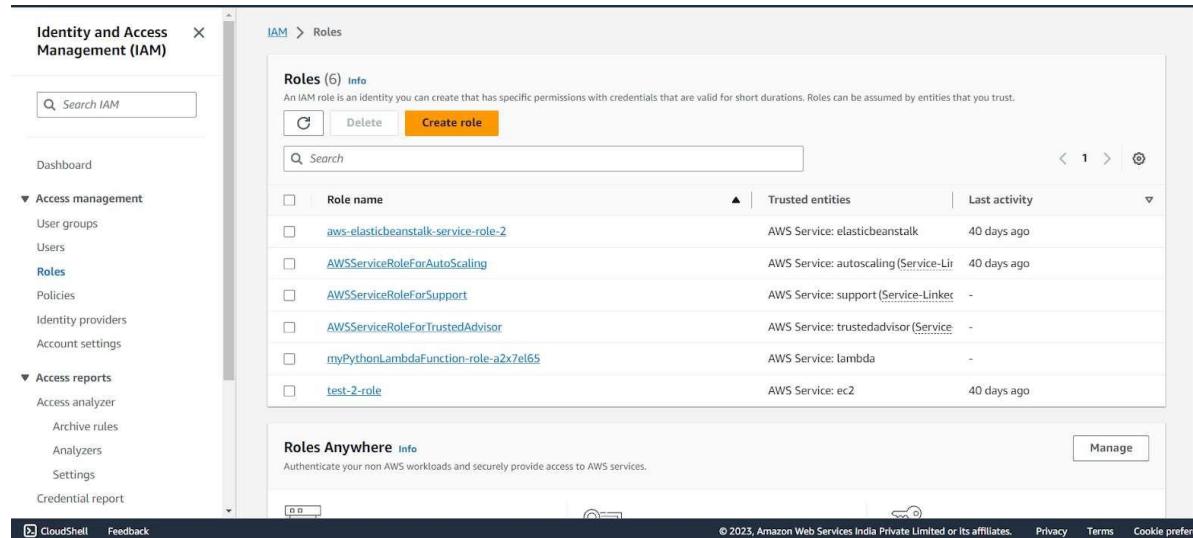
At the bottom, there are links for "CloudShell", "Feedback", and copyright information: "© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences".

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

# Adv. DevOps Exp. 12

Niraj S. Kothawade  
D15A - 24

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

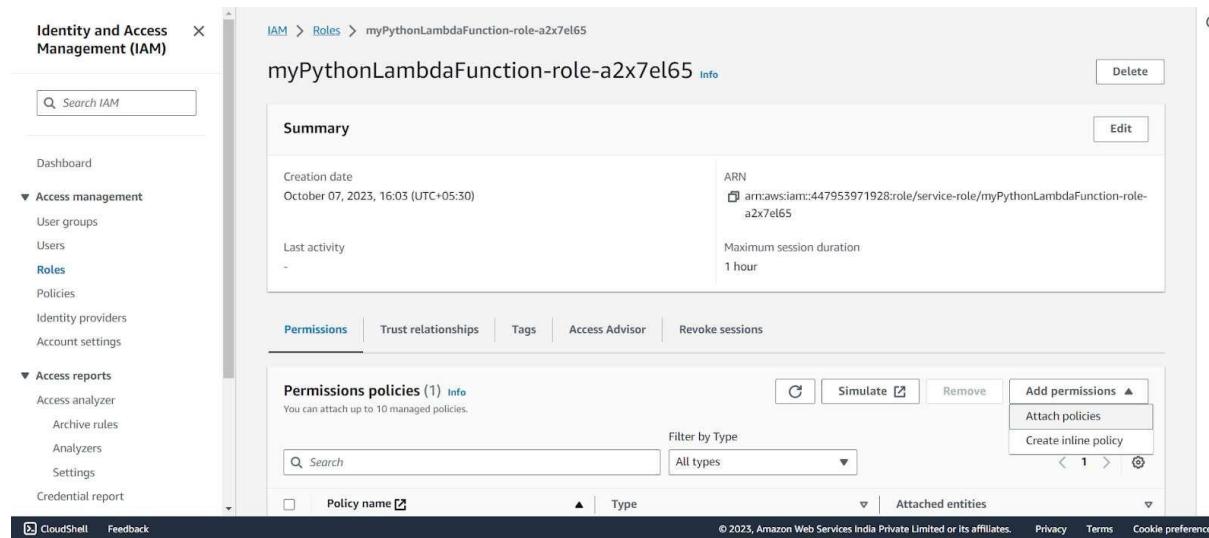


The screenshot shows the AWS IAM Roles page. On the left, there's a navigation sidebar with options like Dashboard, Access management (User groups, Users, Roles, Policies, Identity providers, Account settings), Access reports (Archive rules, Analyzers, Settings, Credential report), and CloudShell/Feedback.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

Role name	Trusted entities	Last activity
aws-elasticbeanstalk-service-role-2	AWS Service: elasticbeanstalk	40 days ago
AWSServiceRoleForAutoScaling	AWS Service: autoscaling (Service-Linked)	40 days ago
AWSServiceRoleForSupport	AWS Service: support (Service-Linked)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linked)	-
myPythonLambdaFunction-role-a2x7el65	AWS Service: lambda	-
test-2-role	AWS Service: ec2	40 days ago

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

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



The screenshot shows the details of the "myPythonLambdaFunction-role-a2x7el65" role. The left sidebar is identical to the previous screenshot. The main content has a "Summary" section with creation date (October 07, 2023, 16:03 (UTC+05:30)), ARN (arn:aws:iam::447953971928:role/service-role/myPythonLambdaFunction-role-a2x7el65), last activity (no activity), and maximum session duration (1 hour). Below is a "Permissions" tab with a "Permissions policies (1) Info" section. It shows one policy attached: "S3-ReadOnly". There are buttons for "Add permissions" (with "Attach policies" and "Create inline policy" options), "Simulate", and "Remove". A search bar and filter dropdown are also present.

S3-ReadOnly

IAM > Roles > myPythonLambdaFunction-role-a2x7el65 > Add permissions

Attach policy to myPythonLambdaFunction-role-a2x7el65

▶ Current permissions policies (1)

Other permissions policies (882)

Filter by Type: All types, 1 match

Policy name	Type	Description
AmazonS3ReadOnlyAccess	AWS managed	Provides read only access to all bucket...

Cancel Add permissions

## CloudWatchFull

IAM > Roles > myPythonLambdaFunction-role-a2x7el65 > Add permissions

Attach policy to myPythonLambdaFunction-role-a2x7el65

▶ Current permissions policies (2)

Other permissions policies (881)

Filter by Type: All types, 2 matches

Policy name	Type	Description
CloudWatchFullAccess	AWS managed	Provides full access to CloudWatch.
CloudWatchFullAccessV2	AWS managed	Provides full access to CloudWatch.

Cancel Add permissions

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

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles**
- Policies
- Identity providers
- Account settings

Access reports

- Access analyzer
- Archive rules
- Analyzers
- Settings
- Credential report

Policy was successfully attached to role.

Last activity: - Maximum session duration: 1 hour

Permissions | Trust relationships | Tags | Access Advisor | Revoke sessions

Permissions policies (3) info

You can attach up to 10 managed policies.

Filter by Type: All types

Policy name	Type	Attached entities
AmazonS3ReadOnlyAccess	AWS managed	1
AWSLambdaBasicExecutionRole-c4946a...	Customer managed	1
CloudWatchFullAccess	AWS managed	1

Permissions boundary (not set)

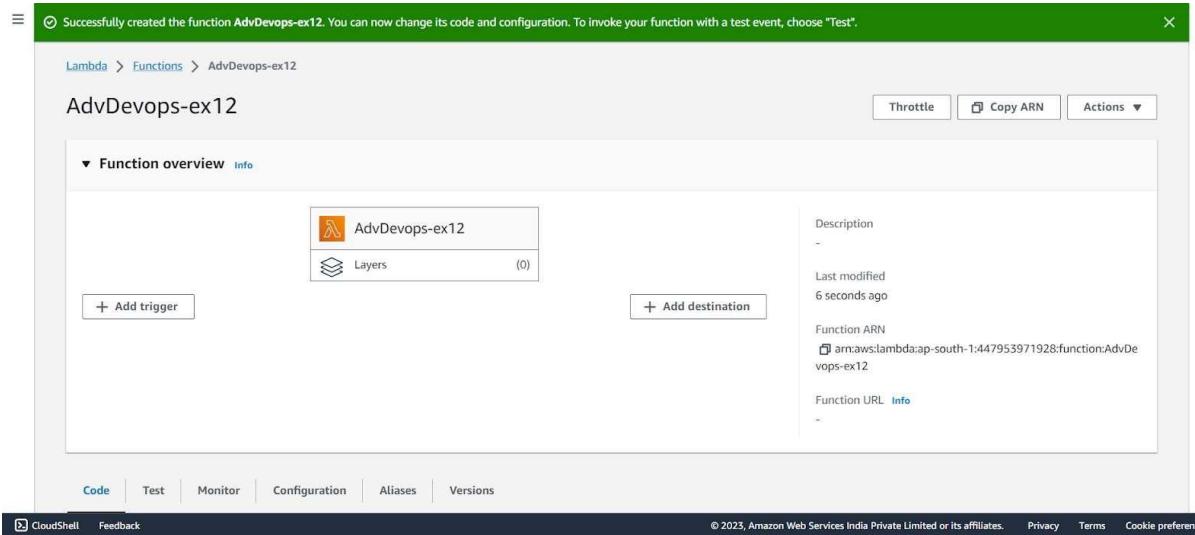
### Step 3: Open up AWS Lambda and create a new Python function.

The screenshot shows the 'Create function' wizard in the AWS Lambda console. The 'Basic information' section is visible, containing fields for 'Function name' (set to 'AdvDevops-ex12'), 'Runtime' (set to 'Python 3.11'), and 'Architecture' (set to 'x86\_64'). Other options like 'Container image' and 'Blueprints' are also shown. The bottom of the screen includes standard AWS navigation links like CloudShell, Feedback, and a footer with copyright and privacy information.

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

The screenshot shows the 'Execution role' configuration step in the 'Create function' wizard. It shows the 'Use an existing role' option selected, with the role 'service-role/myPythonLambdaFunction-role-a2x7el65' chosen from a dropdown. The 'Advanced settings' section is partially visible at the bottom. The bottom of the screen includes standard AWS navigation links like CloudShell, Feedback, and a footer with copyright and privacy information.

Step 4: The function is up and running.



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

```
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_unquoted = urllib.parse.unquote_plus(key, encoding='utf-8')
11    message = f'An file has been added with key {key} to the bucket {bucket_name}'
12    print(message)
13    response = s3_client.get_object(Bucket=bucket_name, Key=key)
14    contents = response['Body'].read().decode()
15    contents_json = json.loads(contents)
16
17    print("These are the Contents of the File: \n", contents_json)
18
19
```

The screenshot shows the AWS Lambda code editor. The left sidebar shows the environment variables and the "lambda\_function" tab. The main editor area contains the provided Python code. The status bar at the bottom right shows "18:5 Python Spaces: 4". The footer includes standard AWS links.

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

```
{ "Records": [ { "eventVersion": "1.0", "eventSource": "aws:s3", "awsRegion": "us-east-1", "s3": { "bucket": { "name": "advdevops-ex12-test" }, "object": { "key": "testfile.txt" } } } ] }
```

Buttons at the bottom of the modal include "Format JSON", "Cancel", "Invoke", and "Save".

And Save it.

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

The screenshot shows the AWS S3 Buckets page. At the top, there's an 'Account snapshot' section with a link to 'View Storage Lens dashboard'. Below it, a table lists three buckets:

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

At the bottom of the page, there are links for CloudShell, Feedback, and a footer with copyright information.

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

The screenshot shows the 'Create bucket' page. In the 'General configuration' section, the bucket name is set to 'AdvDevopsxp12' and the AWS Region is set to 'Asia Pacific (Mumbai) ap-south-1'. In the 'Object Ownership' section, it says '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.'

At the bottom of the page, there are links for CloudShell, Feedback, and a footer with copyright information.

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

The screenshot shows the 'Event notifications' section with a table for event notifications. It also includes sections for 'Amazon EventBridge' and 'Transfer acceleration'.

Name	Event types	Filters	Destination type	Destination
No event notifications				

Below the table, there's a note: 'Choose [Create event notification](#) to be notified when a specific event occurs.' A 'Create event notification' button is shown.

In the 'Amazon EventBridge' section, it says 'Send notifications to Amazon EventBridge for all events in this bucket' and has a dropdown set to 'Off'.

In the 'Transfer acceleration' section, it says 'Use an accelerated endpoint for faster data transfers.' and has a dropdown set to 'Disabled'.

At the bottom of the page, there are links for CloudShell, Feedback, and a footer with copyright information.

Click on Create Event Notification.

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

The screenshot shows the 'General configuration' section of the AWS S3 console. The 'Event name' field contains 'S3putrequest'. Under 'Event types', the 'Put' checkbox is checked, while 'Post' is unchecked. Other options like 'All object create events' and 'Object creation' are also visible.

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

The screenshot shows the 'Destination' configuration page. The 'Lambda function' option is selected. In the 'Specify Lambda function' section, 'Choose from your Lambda functions' is selected. The 'Lambda function' dropdown menu shows 'AdvDevops-ex12'. At the bottom, there are 'Cancel' and 'Save changes' buttons.

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

The screenshot shows the AWS Lambda Function Overview page for the function 'AdvDevops-ex12'. In the 'Triggers' section, there is one entry for 'S3'. Below the triggers, there are buttons for '+ Add destination' and '+ Add trigger'. On the right side, there are sections for 'Description', 'Last modified' (1 minute ago), 'Function ARN' (arn:aws:lambda:ap-south-1:447953971928:function:AdvDevops-ex12), and 'Function URL' (Info). At the bottom, there are tabs for 'Code' (selected), 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. A 'Code source' dropdown is also visible.

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

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

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 area is titled 'Upload' with a 'Info' link. A large dashed box allows users to 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.' Below this, a table lists 'Files and folders (1 Total, 89.0 B)'. The table has columns for Name, Folder, Type, and Size. One item, 'dummy.json', is listed with a size of 89.0 B and type application/json. There are 'Remove', 'Add files', and 'Add folder' buttons at the top of the table. A search bar labeled 'Find by name' is present above the table. The 'Destination' section shows the target bucket as 's3://advopssexp12'. At the bottom, there are links for 'CloudShell' and 'Feedback', and a copyright notice: '© 2023, Amazon Web Services India Private Limited or its affiliates'.

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

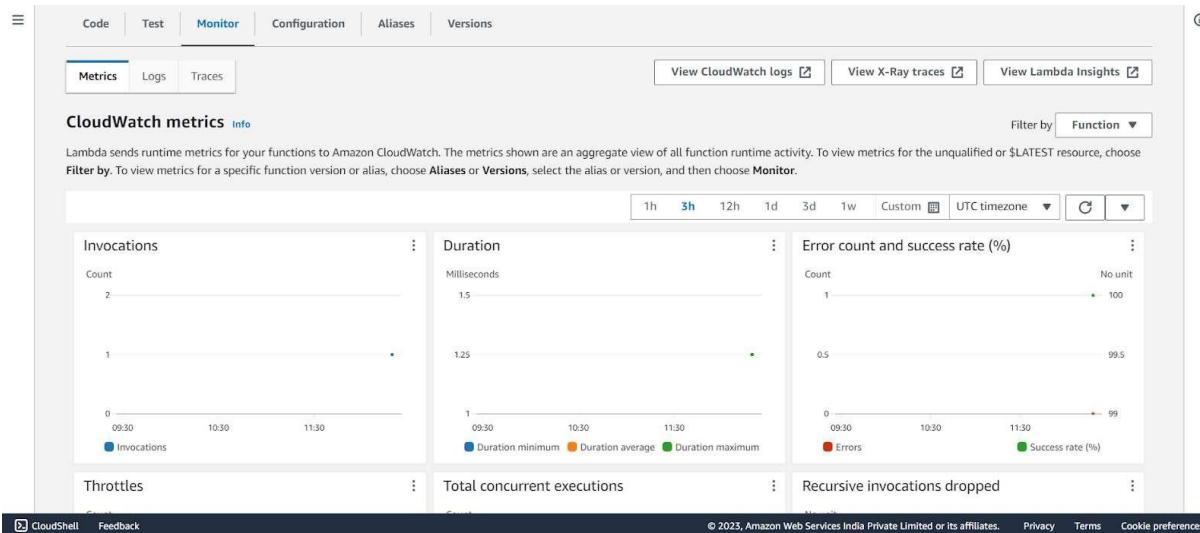
The screenshot shows the Lambda function configuration interface. The 'Event JSON' tab is selected, displaying a large JSON code block. The code is a test event for a Lambda function, structured as follows:

```

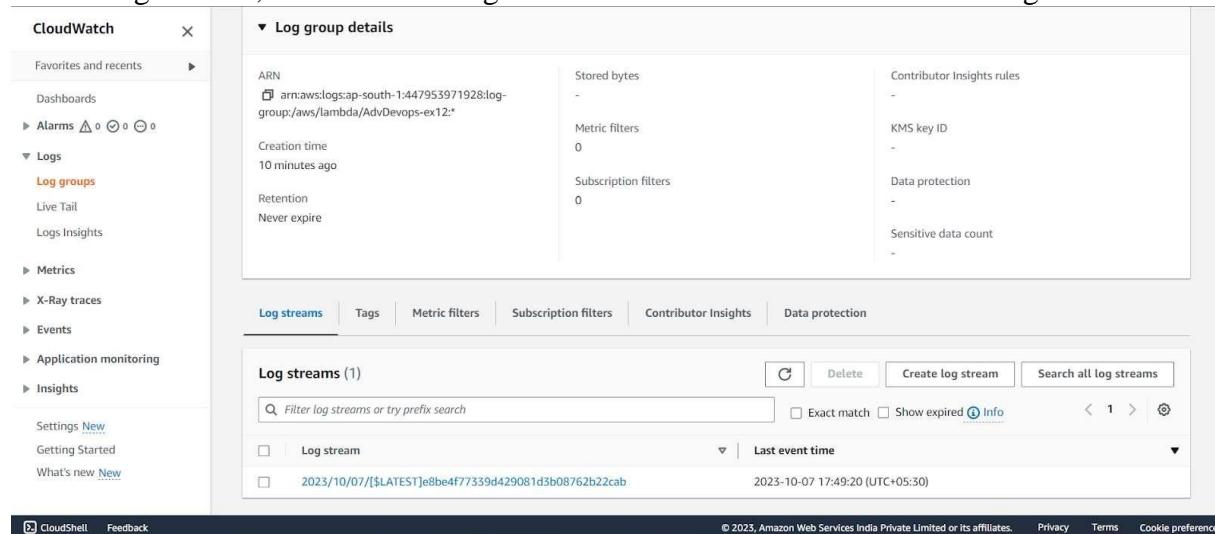
10     "principalId": "EXAMPLE"
11   },
12   "requestParameters": {
13     "sourceIPAddress": "127.0.0.1"
14   },
15   "responseElements": {
16     "x-amz-request-id": "EXAMPLE123456789",
17     "x-amz-id-2": "EXAMPLE123/5678abcdefghijklambdaisawesome/mnopqrstuvwxyzABCDEFGHIJKLMN"
18   },
19   "s3": {
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 }

```

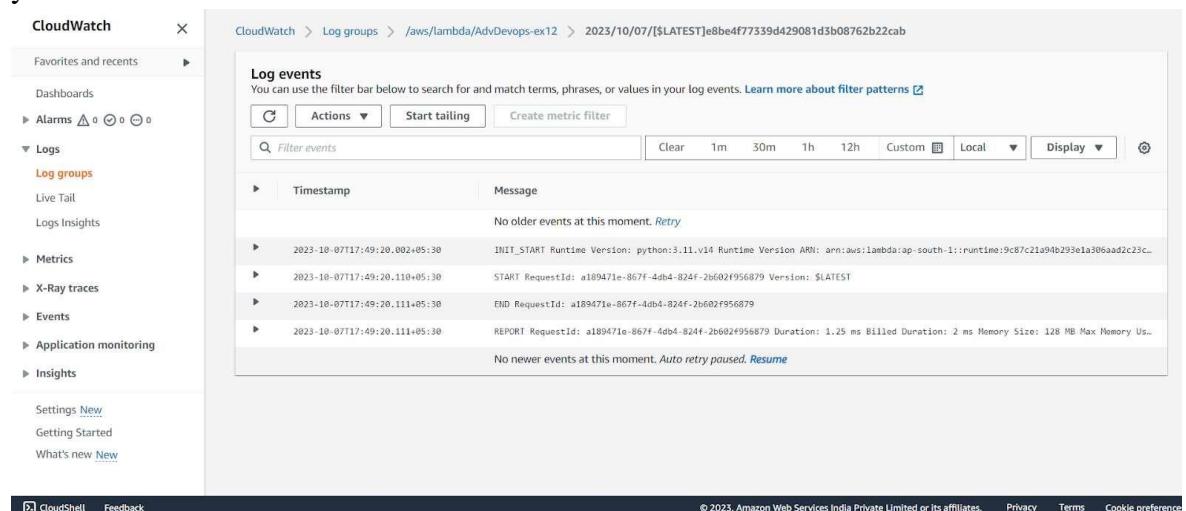
Step 16: Go back to your Lambda function , Refresh it and check the Monitor tab.



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.