

Don Bosco Institute of Technology(DBIT),Mumbai
Department of Information Technology



Lab Journal

On

I.T/T.E./Sem V/ITL504: Advanced DevOPS Lab

By

24 Swasti Jain

Academic Year: Nov,2022 Index

Sr no.	Name of the Experiment	Page no.	Date
1	To understand the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration	2-7	23/07/2022
2	To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.	8-12	12/08/2022
3	To understand the Kubernetes Cluster Architecture, install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms	13-20	06/08/2022

4	To install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy Your First Kubernetes Application		13/08/2022
5	To understand terraform lifecycle, core concepts/terminologies and install it on a Linux Machine	21	20/08/2022
6	To Build, change, and destroy AWS / GCP /Microsoft Azure/ DigitalOcean infrastructure Using Terraform	21-28	27/08/2022
7	To understand the Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.	29-39	03/09/2022
8	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		10/09/2022
9	To Understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine	40-52	17/09/2022
10	To perform Port, Service monitoring, Windows/Linux server monitoring using Nagios	53-67	24/09/2022

11	To understand AWS Lambda, its workflow, and various functions and create your first Lambda functions using Python / Java / Nodejs	68-70	01/10/2022
12	To create a Lambda function that will log “An Image has been added” once add you an object to a specific bucket in S3	71-77	08/10/2022

Experiment no. 1

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

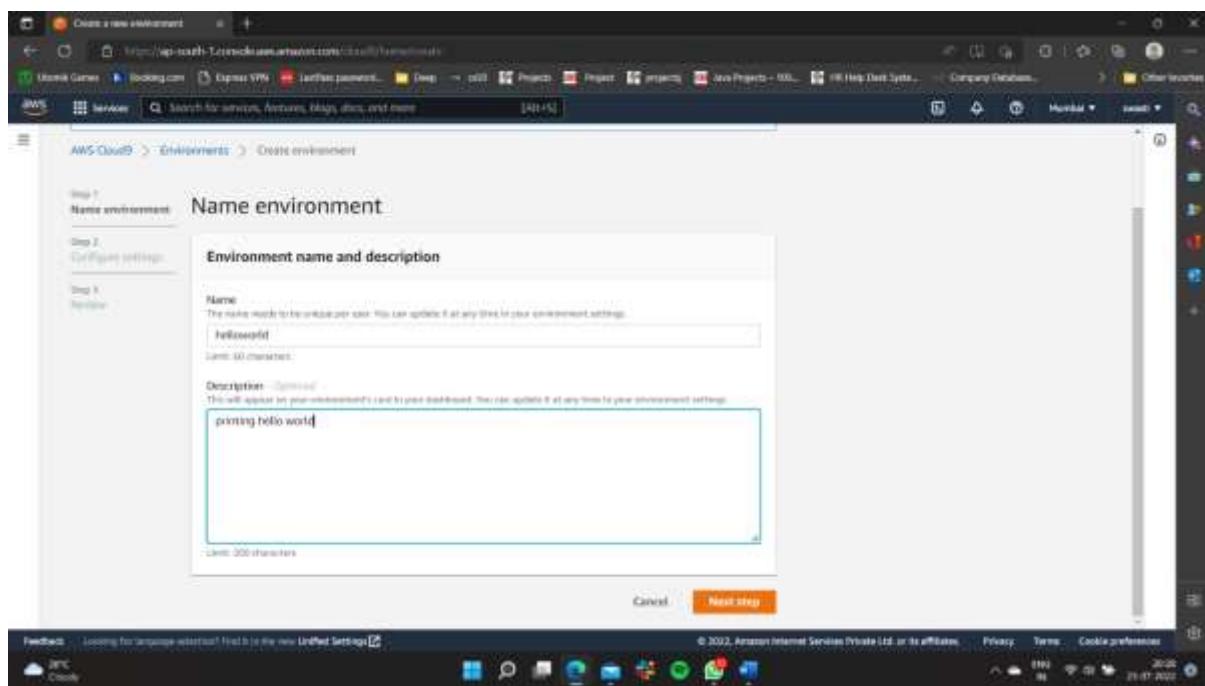
Procedure:

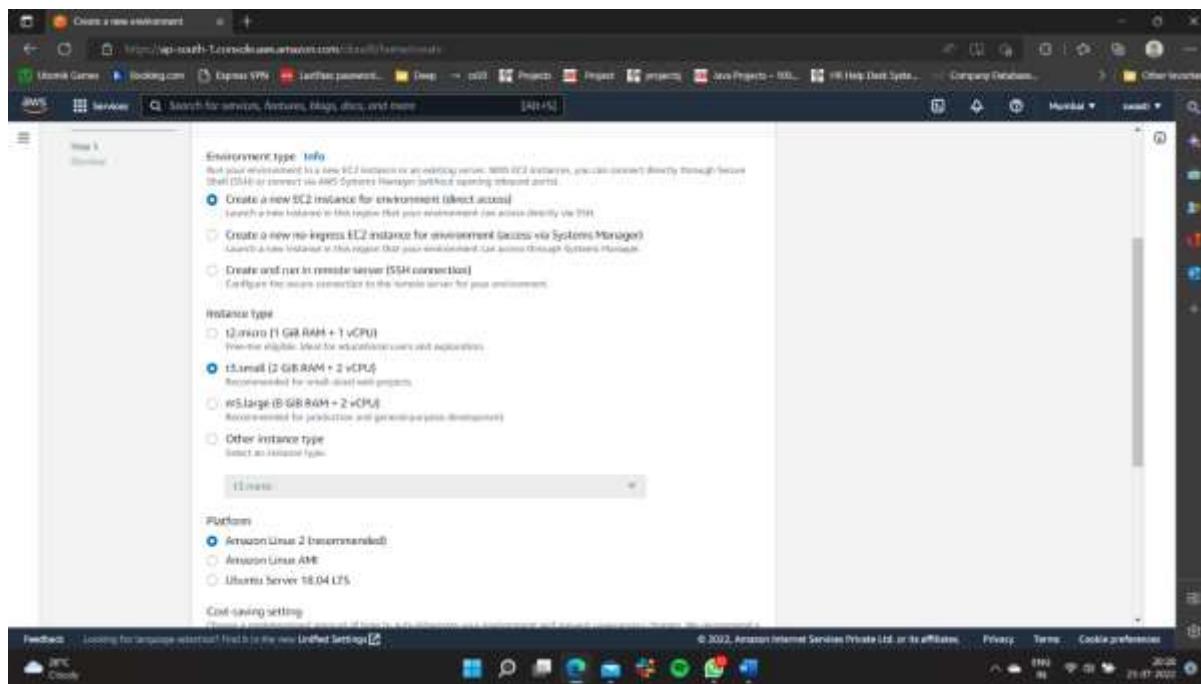
Setup and launch the cloud9 IDE:

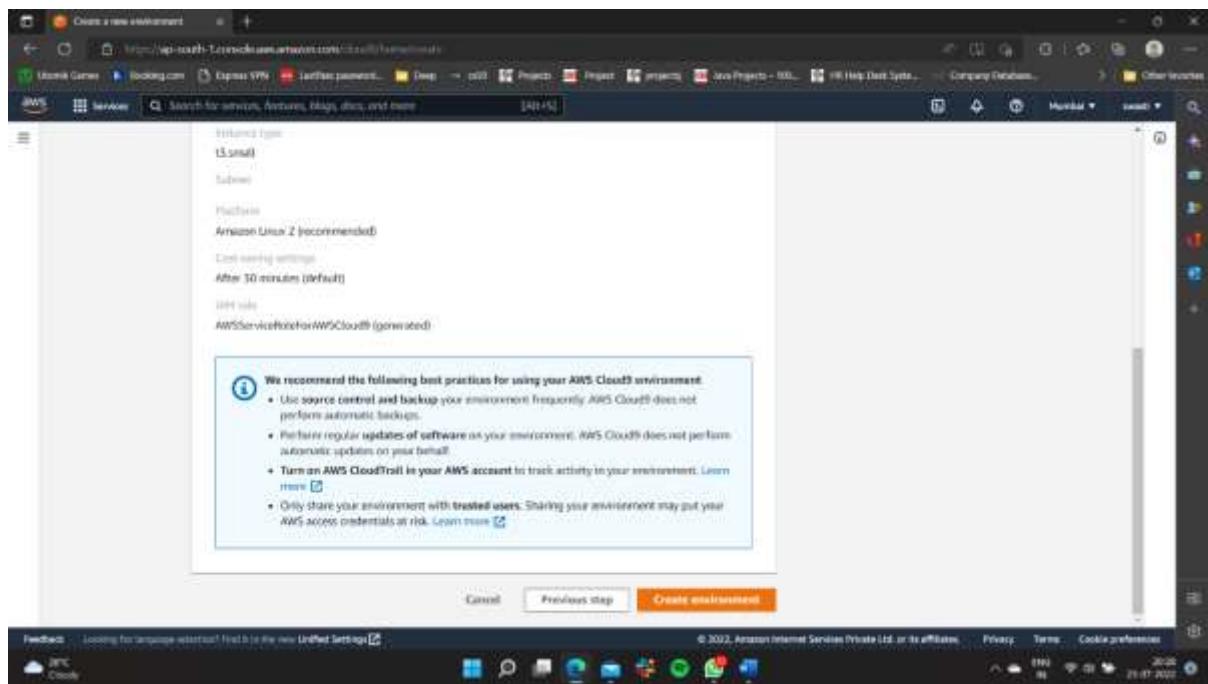
- Create an account on AWS
- After creating an account search for cloud9 in the search bar of aws and click on cloud9
- After that you will land on the cloud9 tool page read all the instruction and click on create environment
- Fill all the details asked and proceed remember select the t2.micro instance since its free
- After reviewing, click on create environment. This part of experiment is complete

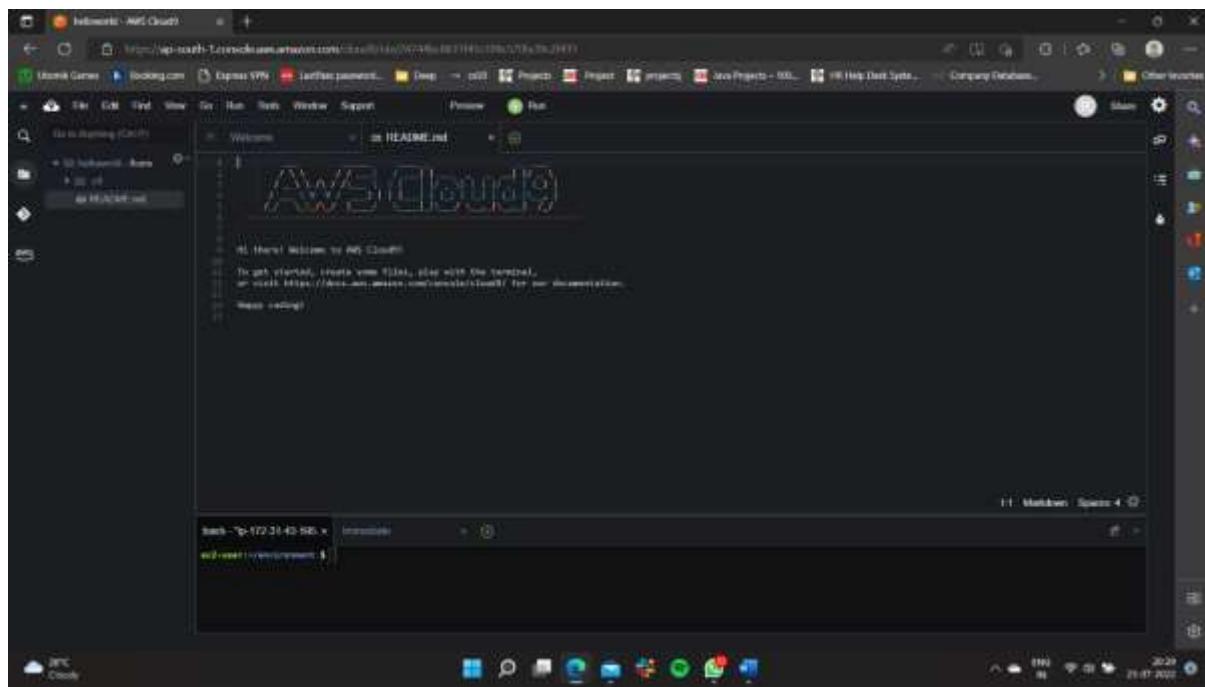
Output :

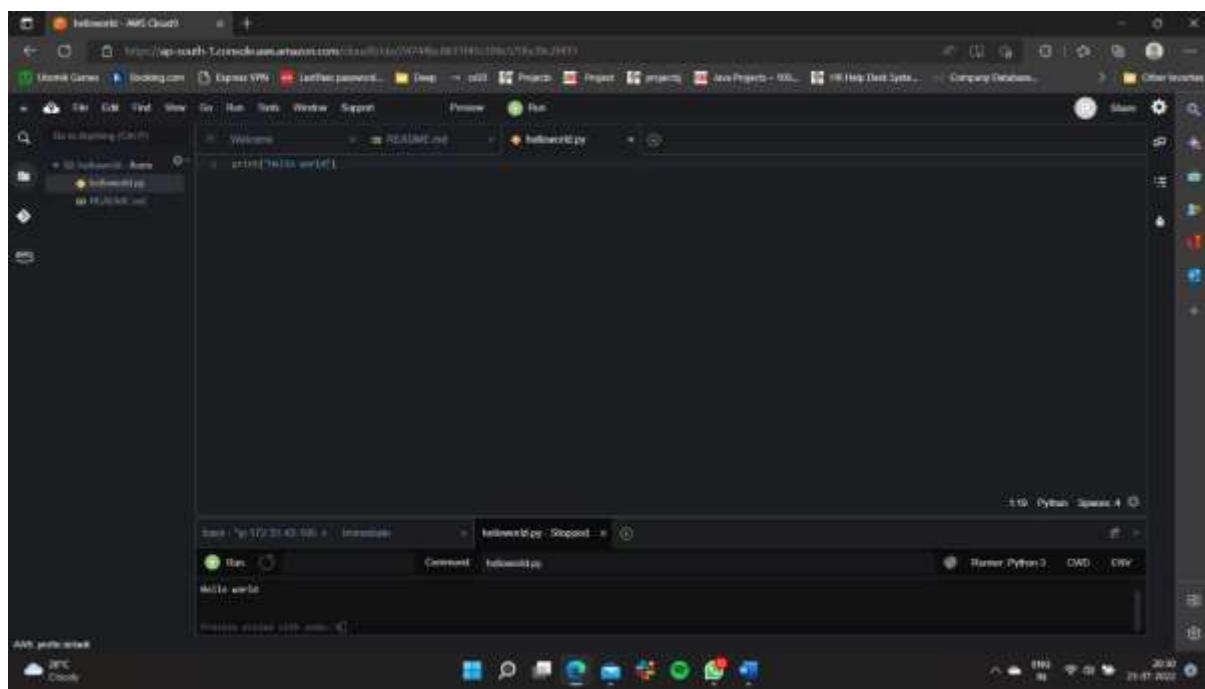
CREATING ENVIRONMENT:

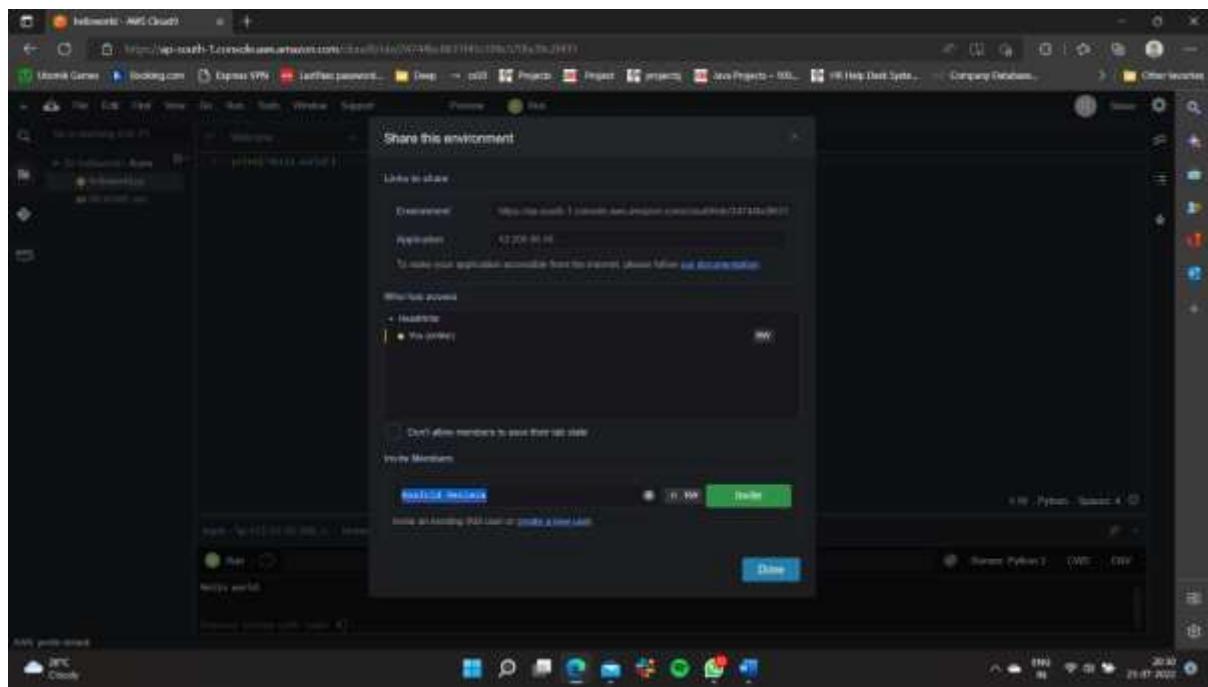












Stopping Instances

The screenshot shows the AWS EC2 Management Console. On the left, a sidebar navigation menu includes: New EC2 Experience, EC2 dashboard, EC2 Global View, Events, Tags, Launch, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (selected), AMIs (selected), AMI Catalog, and Elastic Block Store. A feedback message at the bottom left says "Feedback: Looking for campaign information? Find it or file a new [Untested Settings](#) issue." The main content area displays a success message: "Successfully stopped i-00a70ec64ebaa24a5" and "Successfully stopped i-07174a860ec1aa8a9". Below this is a table titled "Instances (1/2) - 1" showing two instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 address
aws-cloud9-pa...	i-00a70ec64ebaa24a5	Stopping	t3.small	2/2 checks passed	No alarms	ap-south-1a	ec2-45...
aws-cloud9-ho...	i-07174a860ec1aa8a9	Stopping	t3.small	Initializing	No alarms	ap-south-1a	ec2-43...

Details for instance i-07174a860ec1aa8a9 are shown in the modal:

Details	Security	Networking	Storage	Status checks	Monitoring	Tags
Instance summary: i-07174a860ec1aa8a9 (aws-cloud9-helloworld-24744bc8631f41c399c570fa39c2f431)						
Instance ID: i-07174a860ec1aa8a9 (aws-cloud9-helloworld-24744bc8631f41c399c570fa39c2f431)	Public IPv4 address: 43.205.95.45 [open address]	Private IPv4 address: 172.31.43.195				
IPv6 address: -	Instance state: Stopping	Public IPv6 DNS: ec2-43-205-95-45.ap-south-1.compute.amazonaws.com [open address]				

The screenshot shows the AWS Cloud9 interface. On the left, there's a sidebar with options like 'Your environments', 'Shared with you' (which is selected and highlighted in orange), and 'Account environments'. Below that is a 'How-to guide'. The main area is titled 'Shared with you (1)'. It shows a single environment entry for 'Nidhi Sarda'. The details for this environment are as follows:

- Type: EC2
- Permissions: Read-write
- Description: first collaboration
- Owner: Am: amazonelem:059057543376:root

At the top of this section, there are buttons for 'Open IDE' (with a dropdown arrow), 'View details', 'Edit', 'Delete', and 'Create environment'. Below the environment details is a small navigation bar with icons for back, forward, and search.

At the bottom of the screenshot, the Windows taskbar is visible, showing various pinned icons and the system clock indicating it's 10:21 PM on 21-07-2022.

Experiment no. 2

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.

Steps:

Advanced DevOps

NAME :Swasti Jain

CLASS : TE IT

ROLL NO : 24

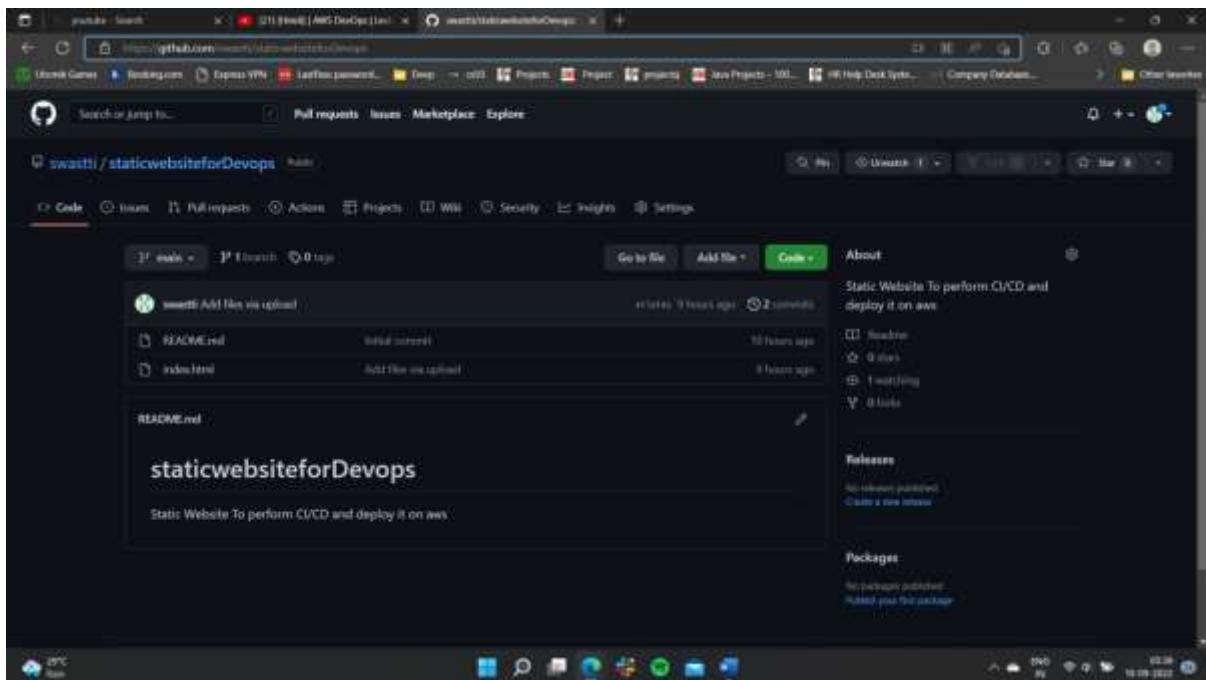
Exp 2

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

Steps:

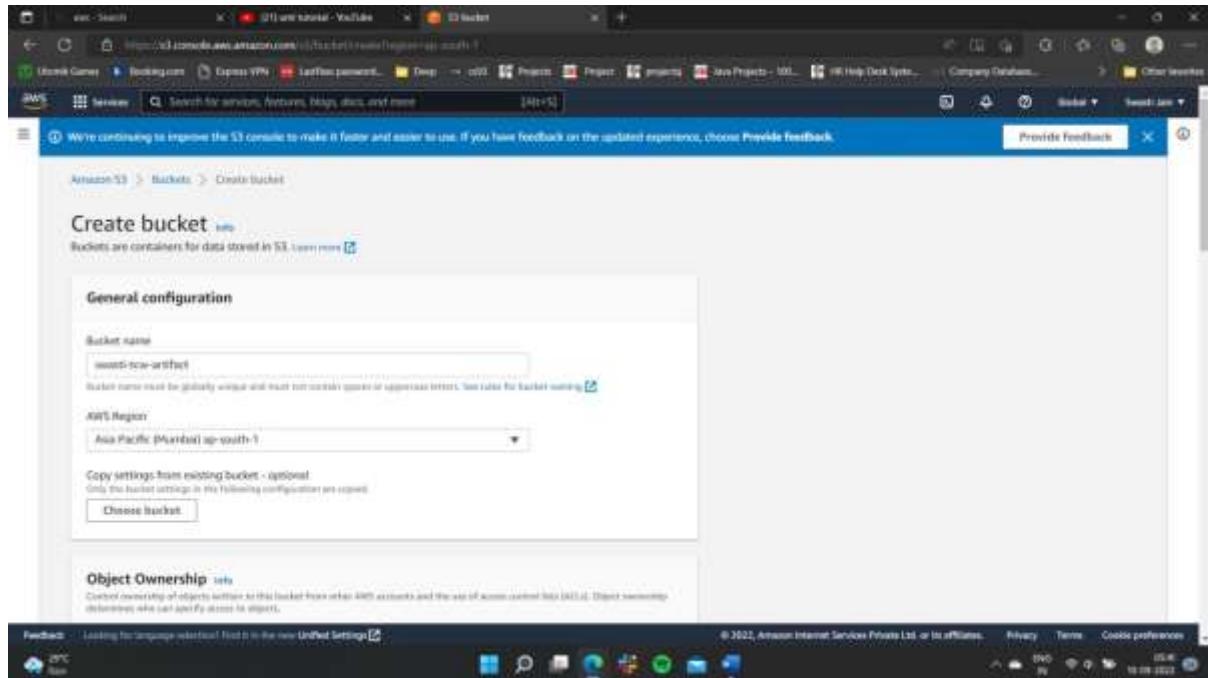
Select a github repository from which you want to deploy the source code

Reference link for Github:[swasthi/staticwebsiteforDevops: Static Website To perform CI/CD and deploy it on aws \(github.com\)](https://github.com/swasthi/staticwebsiteforDevops)

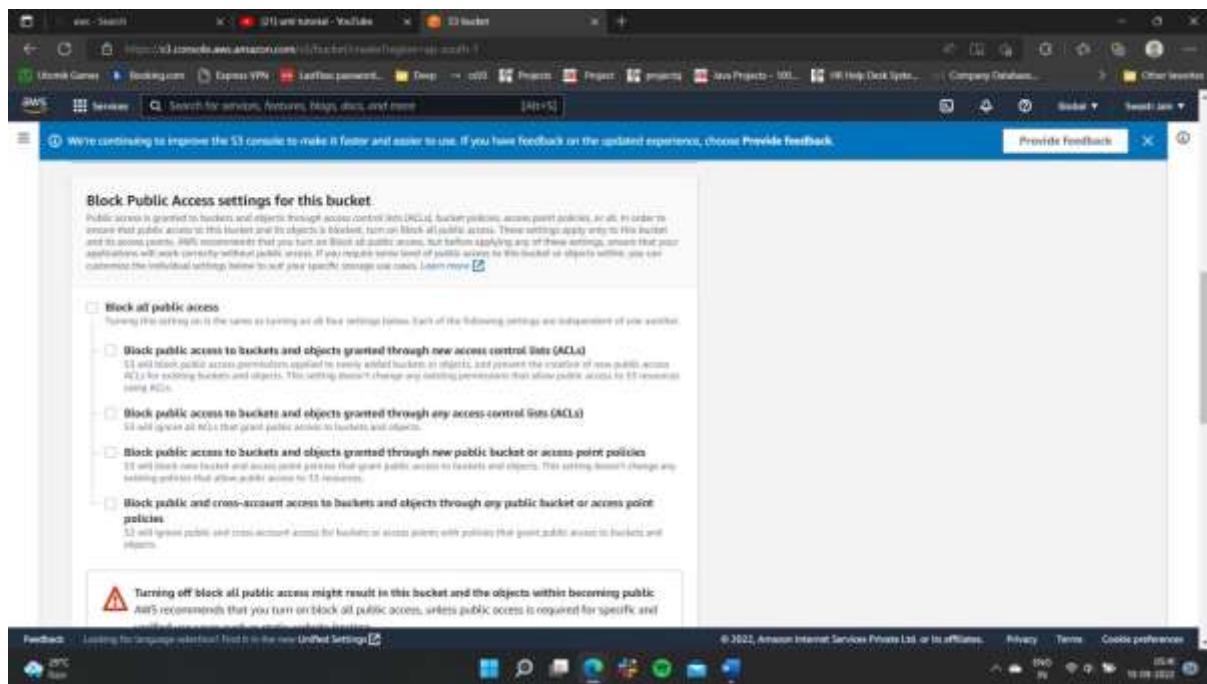


Go to aws console and search for s3 service . Here we have to create 2 buckets one for storing artifacts and the other for deployed code .

Now we start creating the first bucket:



Now we will name the bucket as swasti-tcw-artifacts .



now the only change you have to make is to disable the block all public access and crete bucket and the first bucket will be ready.

We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.

Turning off block all public access might result in this bucket and the objects within becoming public.

Amazon S3 recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Bucket Versioning

Versioning is a feature 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 faulty uploaded user actions and application failures, learn more.

Bucket Versioning

Disable
 Enable

Tags (0) - optional

Add tags or other criteria by tagging your bucket. Learn more.

No tags associated with this bucket.

Feedback Looking for language identical? Find it in the new Unified Settings.

Default encryption

Automatically encrypt new objects stored in this bucket. Learn more.

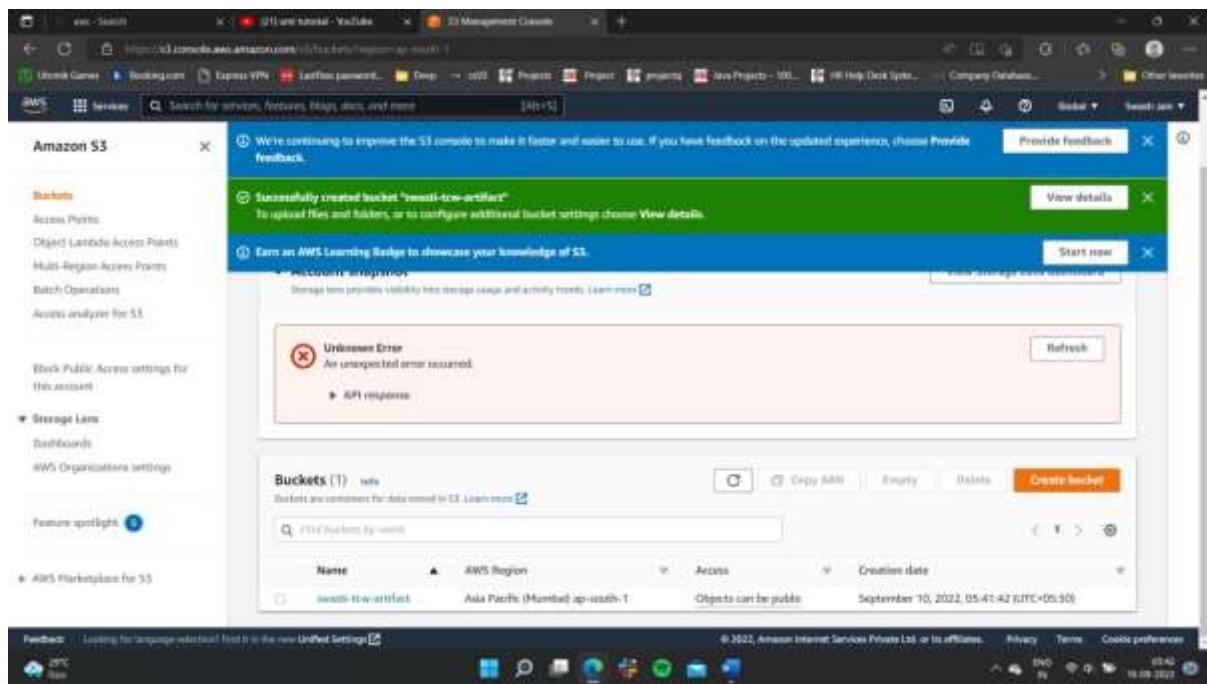
Server-side encryption

Disable
 Enable

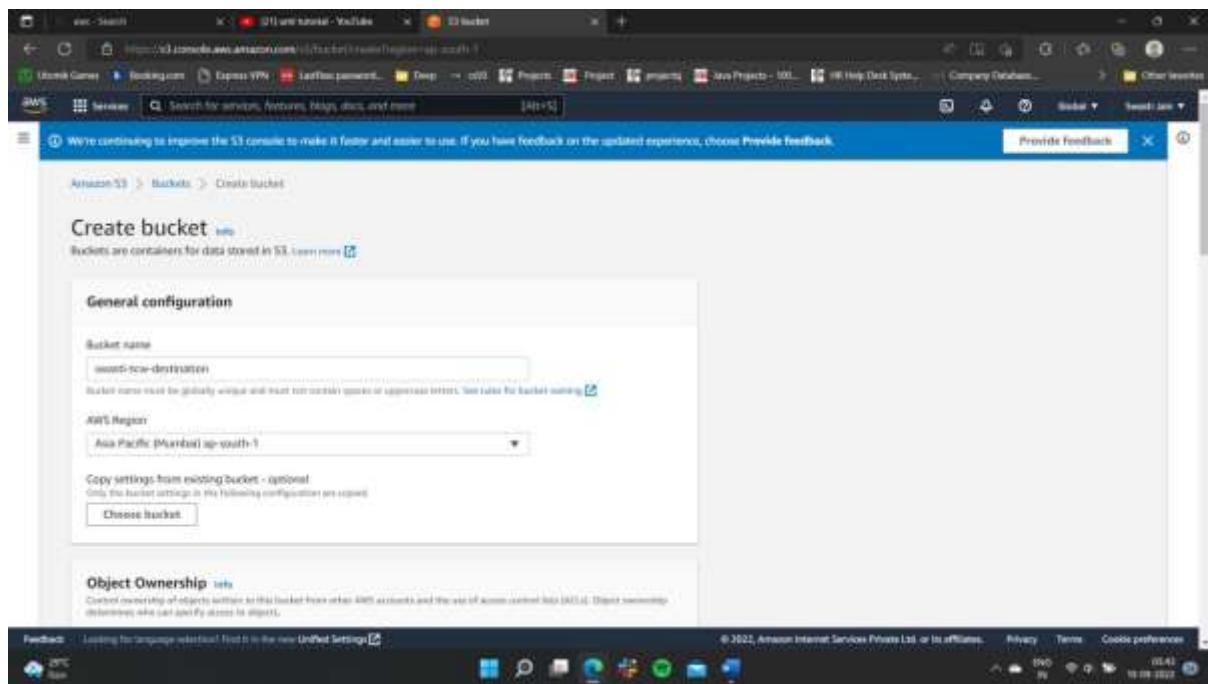
Advanced settings

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

[Create bucket](#)



Now in the similar manner we have to create one more bucket for the destination



We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, server-side encryption, or off. 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 buckets applying only one of these settings, allow that your applications will work correctly without public access. If you require bucket-level or object-level public access to the 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 off all other 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)**
- 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**
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**

Turning off block all public access might result in this bucket and the objects within becoming public

AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Bucket Versioning

Versioning is a feature of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, review, and restore previous versions of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from unintended user actions and application failures. Learn more.

Bucket Versioning

Disable

Enable

Tags (0) - optional

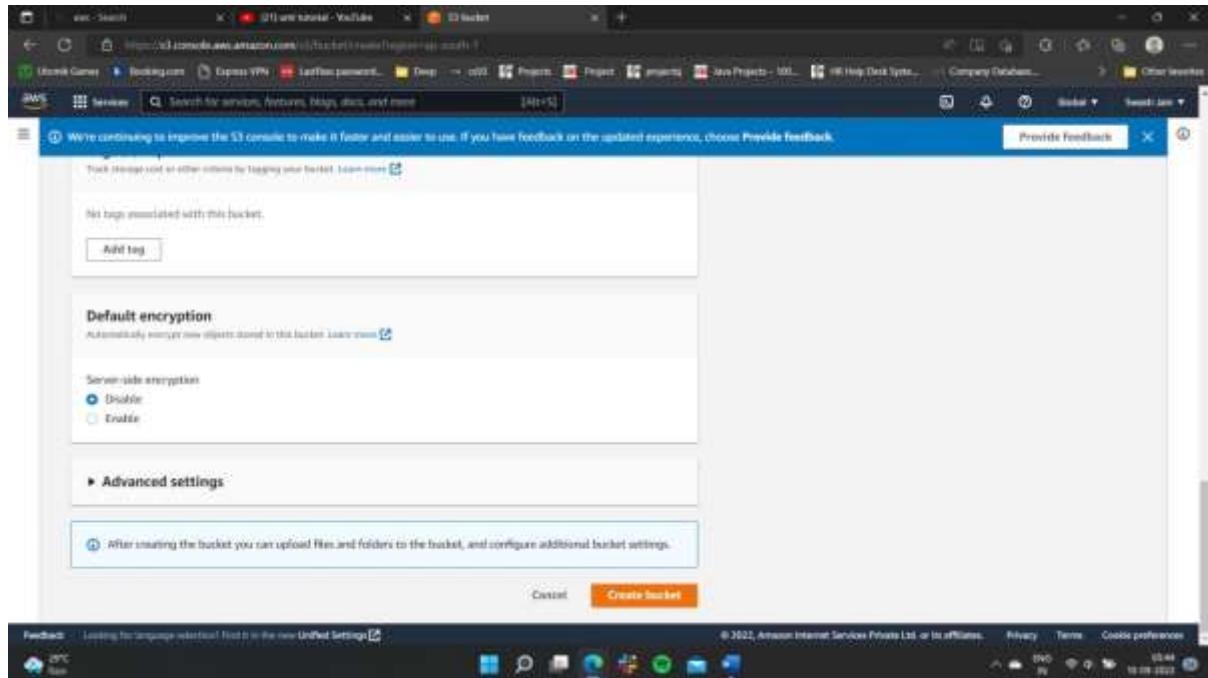
Bucket tags can be used for helping your S3 buckets. Learn more.

No tags associated with this bucket.

Add tags

Feedback: Looking for language identical? Find in the new Unified Settings. © 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookies preferences 19:44 96 10-09-2022

Here the only change for destination bucket is that you have to enable bucket versioning.
Now both our buckets are ready.



Now we will edit the static website hosting.

The screenshot shows the AWS S3 console interface. At the top, there are several tabs and a search bar. On the left, a sidebar lists various S3-related services and features. The main content area displays a success message: "Successfully created bucket 'aws-tutorial-destination'". It also encourages users to "Earn an AWS Learning Badge to showcase your knowledge of S3." Below this, an "Account snapshot" provides an overview of storage usage and activity trends. The central part of the screen shows a table of existing buckets:

Name	AWS Region	Access	Creation date
aws-tutorial-artifact	Asia Pacific (Mumbai) ap-south-1	Objects can be public	September 10, 2022, 06:41:42 (UTC+05:30)
aws-tutorial-destination	Asia Pacific (Mumbai) ap-south-1	Objects can be public	September 10, 2022, 05:44:18 (UTC+05:30)

At the bottom of the page, there is a feedback link, a footer with copyright information, and a standard Windows taskbar.

Bucket overview

AWS Region: Asia Pacific (Mumbai) ap-south-1 | Amazon Resource Name (ARN): arn:aws:s3:::swasti-tcw-destination | Creation date: September 10, 2022, 05:44:18 (UTC+05:30)

Bucket Versioning

Versioning is a feature 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

Edit static website hosting

Static website hosting

Use this feature to host a website or redirect requests. Learn more.

Static website hosting

Disable Enable

Hosting type

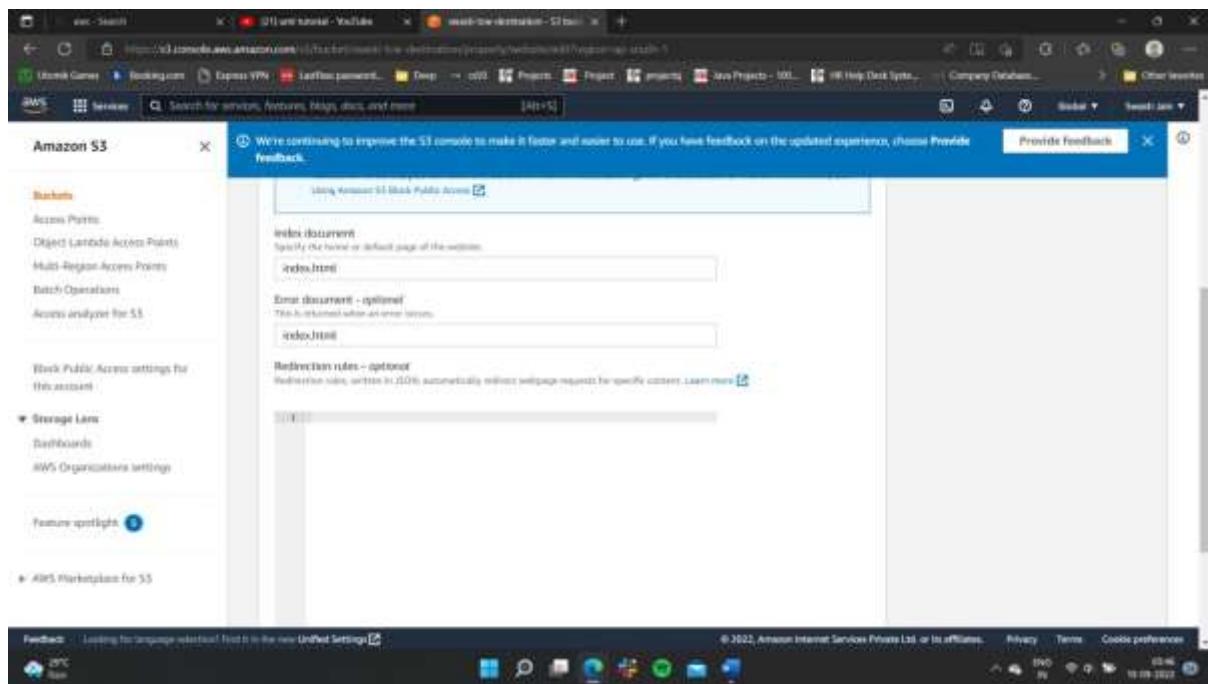
Host a static website
Set the bucket endpoint as the website address. Learn more.

Redirect requests for an object
Redirect requests to another bucket or domain. Learn more.

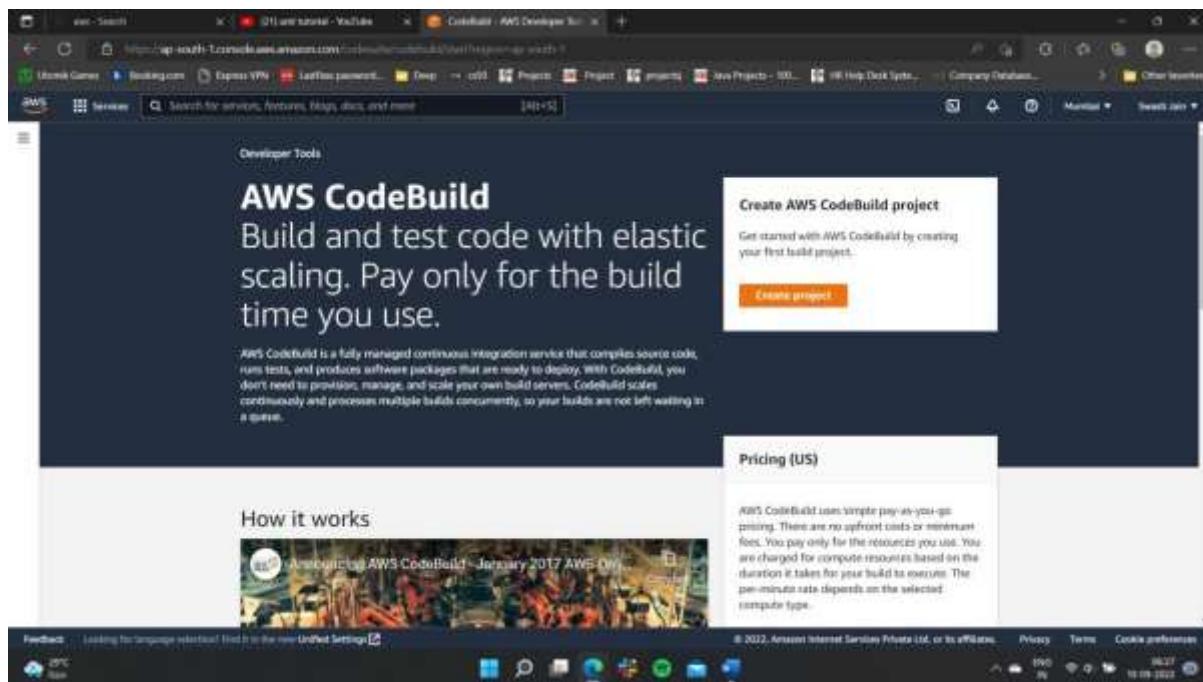
Note

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



Now we will create a pipeline from github to S3.



The screenshot displays two side-by-side browser windows. The left window shows the 'Create build project' configuration page with a 'Project name' of 'codebuild'. The right window shows the 'Processing OAuth request' dialog from GitHub, prompting the user to 'Choose Confirm to connect CodeBuild to your GitHub account.'

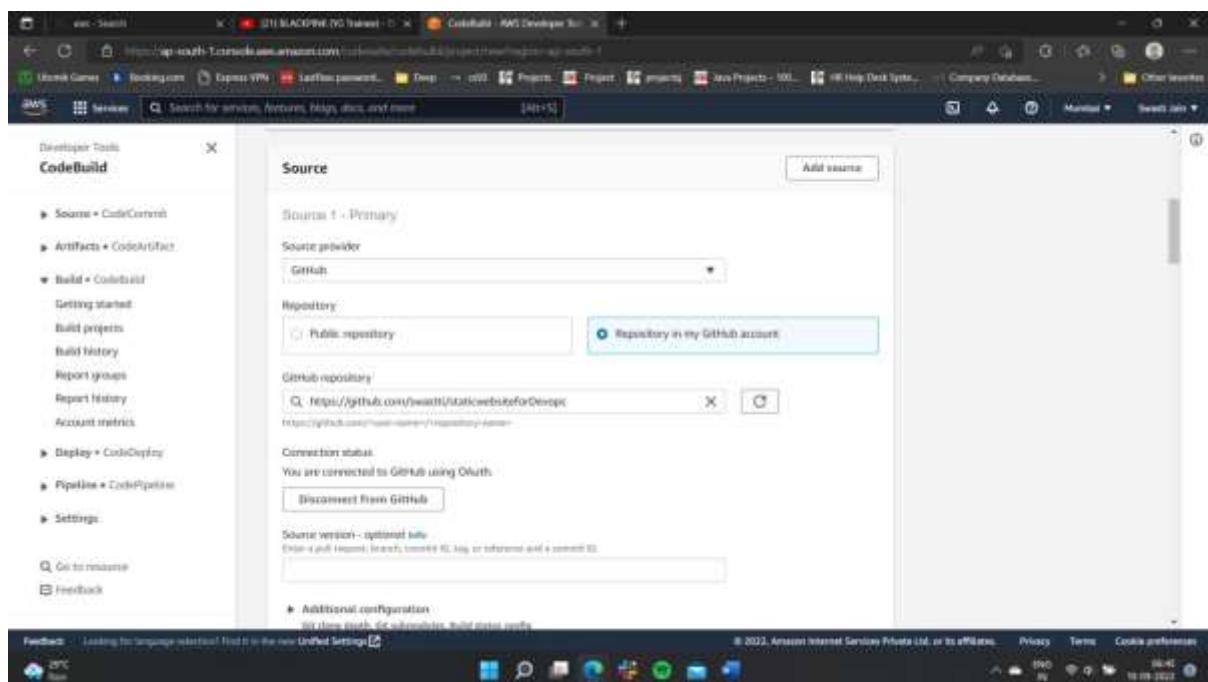
Left Window: Create build project

- Project configuration:**
 - Project name:** codebuild
 - Description - optional:** codebuild
 - Build badge - optional:** Enable build badge
 - Enable concurrent build limit - optional:** Restrict number of concurrent builds this project can start
- Additional configuration:** tags

Right Window: Processing OAuth request

Choose Confirm to connect CodeBuild to your GitHub account.

Buttons: Cancel, Confirm

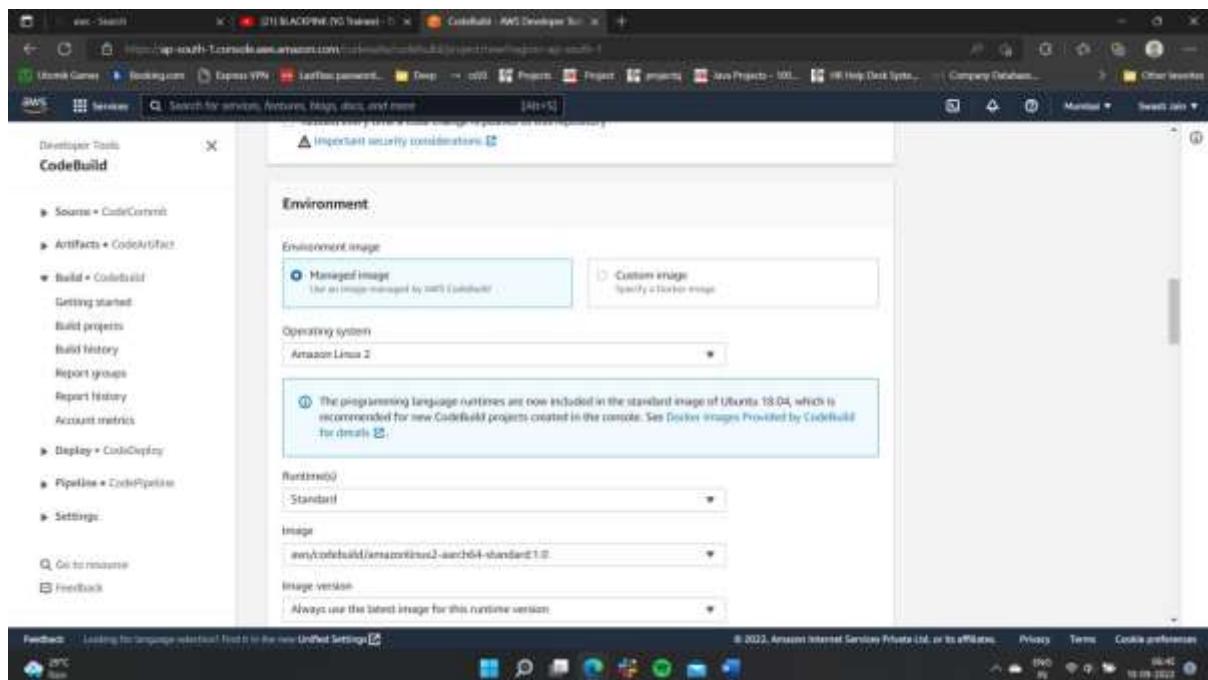


The screenshot shows the AWS CodeBuild console with the 'Source' configuration page. The left sidebar lists navigation options: Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), Display (CodeDeploy), Pipeline (CodePipeline), and Settings. Under 'Build', 'Getting started', 'Build projects', 'Build history', 'Report groups', 'Report History', and 'Account metrics' are listed. Below these are 'Display' (CodeDeploy), 'Pipeline' (CodePipeline), and 'Settings'. At the bottom of the sidebar are links for 'Go to resource' and 'Feedback'.

The main panel is titled 'Source' and contains the following fields:

- Source type:** Primary
- Source provider:** GitHub
- Repository:** Public repository (disabled) or Repository in my GitHub account (selected)
- Github repository:** https://github.com/swarnit/staticwebsiteforDevops
- Connection status:** You are connected to GitHub using OAuth. (with a 'Disconnect from GitHub' button)
- Source version - optional info:** Enter a pull request, branch, commit ID, tag or reference and a comment ID.
- Additional configuration:** A note about using GitHub Actions or AWS Lambda code as build triggers.

Feedback at the bottom: Looking for language identical to this in the new Unified Settings.

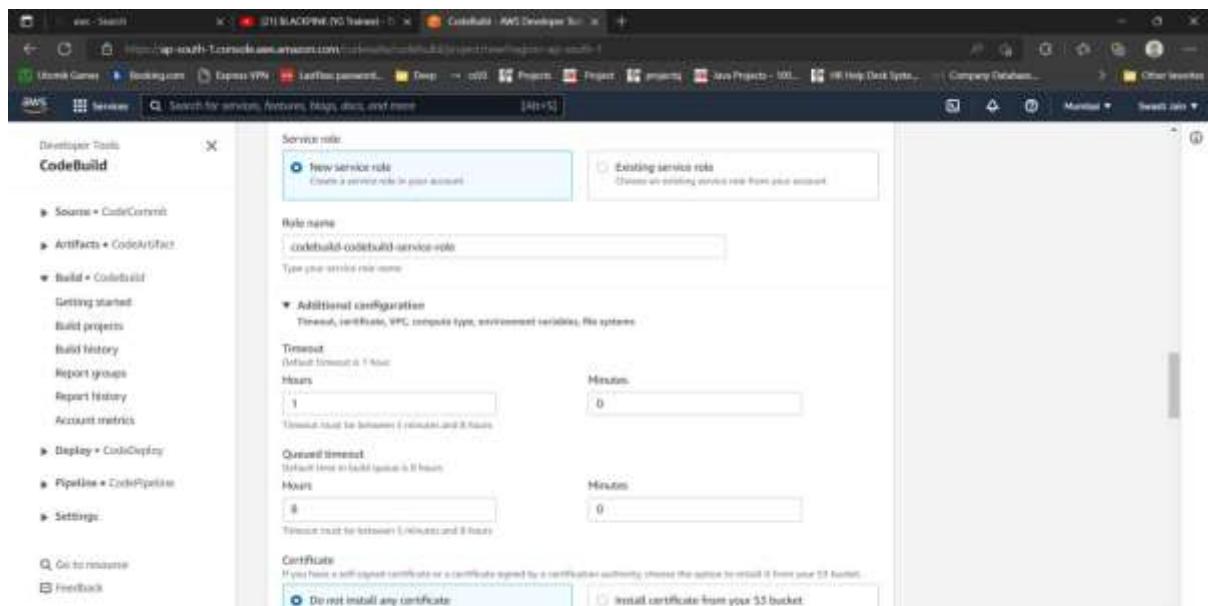


The screenshot shows the AWS CodeBuild console with the 'Environment' configuration page. The left sidebar is identical to the previous screenshot. The main panel is titled 'Environment' and contains the following fields:

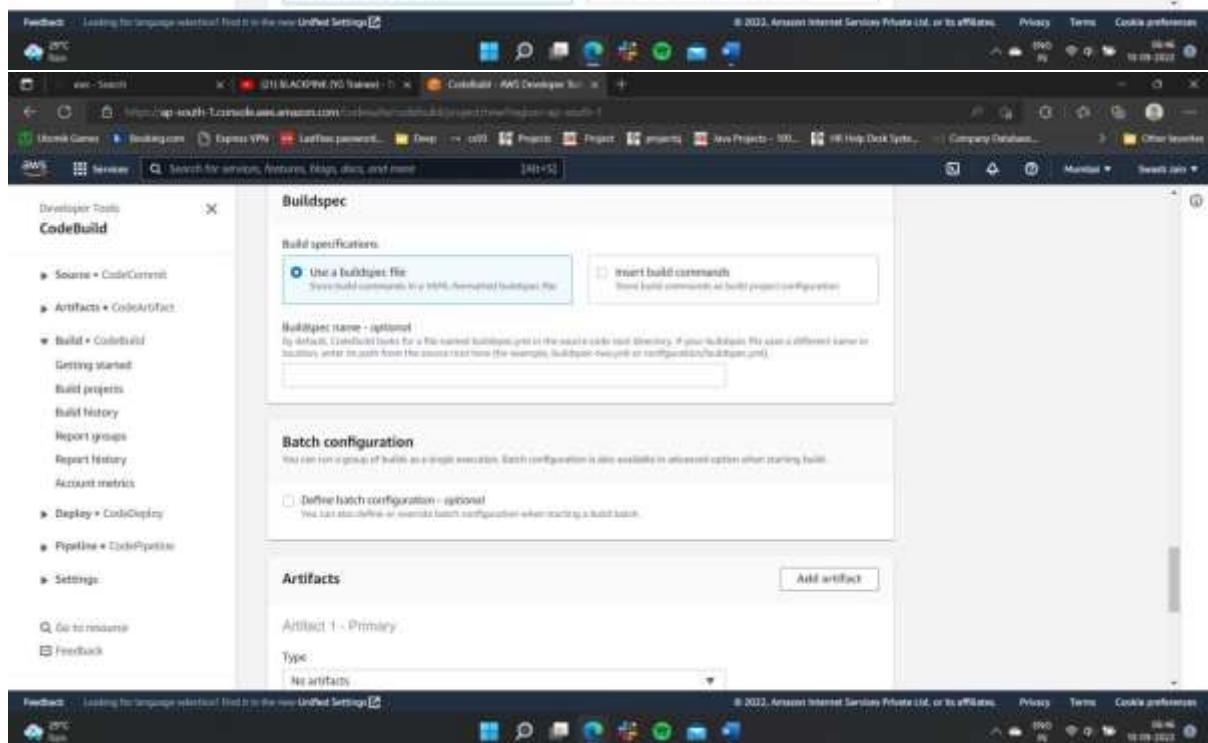
- Environment image:** Managed image (selected) or Custom image (disabled)
- Operating system:** Amazon Linux 2
- Runtimes:** Standard
- Image:** aws/codebuild/amazonlinux2-x64-standard:1.0
- Image version:** Always use the latest image for this runtime version.

A note in the environment section states: "The programming language runtimes are now included in the standard image of Ubuntu 18.04, which is recommended for new CodeBuild projects created in the console. See Docker images provided by CodeBuild for details."

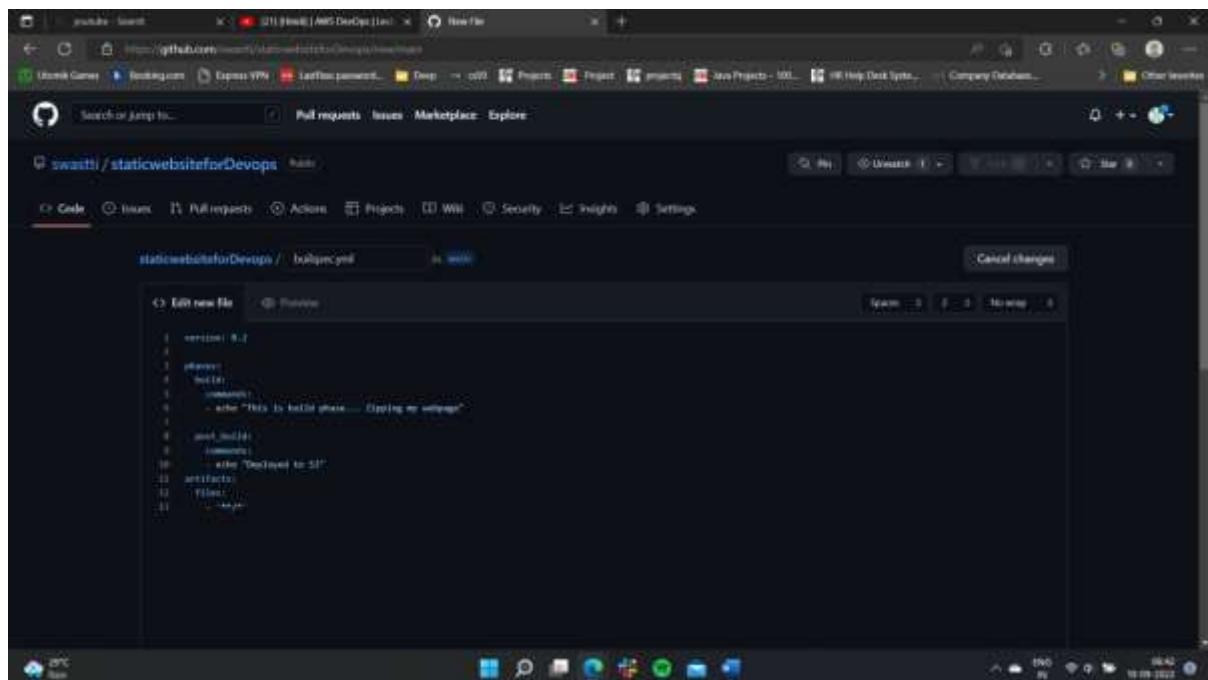
Feedback at the bottom: Looking for language identical to this in the new Unified Settings.



The screenshot shows the AWS CodeBuild service role configuration page. On the left, a sidebar menu lists various services and features under 'Developer Tools' and 'CodeBuild'. The main content area is titled 'Service role' and contains two tabs: 'New service role' (selected) and 'Existing service role'. A 'Role name' input field is set to 'codebuild-codebuild-service-role'. Below it, 'Additional configuration' includes 'Timeout' (set to 1 hour), 'Queued timeout' (set to 0 hours), and a 'Certificate' section where 'Do not install any certificate' is selected.



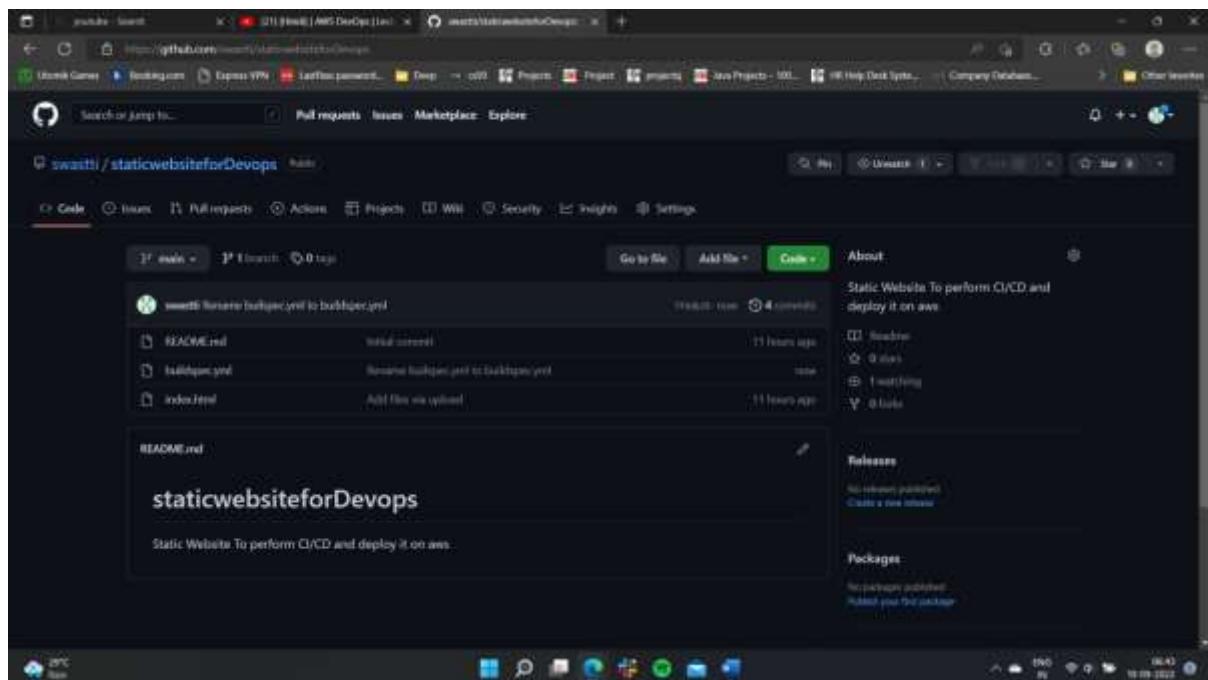
The screenshot shows the AWS CodeBuild buildspec configuration page. The sidebar menu is identical to the previous page. The main content area is titled 'Buildspec' and contains sections for 'Build specifications' (using a buildspec file or inserting build commands), 'Batch configuration' (defining batch configurations), and 'Artifacts' (specifying artifact types). The 'Artifacts' section shows 'Artifact 1 - Primary' with a dropdown menu showing 'No artifacts'.



A screenshot of a GitHub repository page for 'staticwebsiteforDevops'. The user is viewing the 'Code' tab, specifically the 'buildspec.yml' file. The file content is as follows:

```
version: 0.2
phases:
  install:
    commands:
      - echo "This is build phase.... skipping no artifacts"
  post_build:
    commands:
      - echo "Deployed to S3"
artifacts:
  files:
    - index.html
```

The GitHub interface shows the file has been committed 4 times ago by the user 'swasthi'. Below the code editor, there is a preview of the repository's contents, including a README.md file and a static website.



A screenshot of the same GitHub repository page, but now showing the 'About' section. It describes the repository as a 'Static Website To perform CI/CD and deploy it on aws'. The 'About' section includes a 'Readme' link, a 'Issues' link, and a 'Releases' section which states 'No releases published'. There is also a 'Packages' section with a note 'No packages published'.

The screenshot shows the AWS CodeBuild 'Create build project' configuration page. On the left, a sidebar menu lists various services: Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), Display (CodeDeploy), Pipeline (CodePipeline), and Settings. The 'Build' section is expanded, showing options like Getting started, Build projects, Build history, Report groups, Report History, and Account metrics. Under 'Build projects', 'Build project' is selected. The main content area is titled 'Logs' and contains sections for CloudWatch and S3. Under CloudWatch, there is a checked checkbox for 'CloudWatch logs - optional' with the note 'Choosing this option will enable build output logs via CloudWatch'. Below it is a 'Group name' input field. Under S3, there is an unchecked checkbox for 'S3 (log - optional)' with the note 'Choosing this option will enable build output logs via S3'. At the bottom of the configuration panel are 'Cancel' and 'Create build project' buttons. A green banner at the top right says 'Project created' and 'You have successfully created the following project: codebuild'. Below the banner are tabs for Configuration, Build history, Batch history, Build details, Build triggers, and Metrics. The Build history tab is active, showing a table with columns: Build run, Status, Build number, Source version, Submitter, Duration, and Correlated. The table currently has one row: 'Build run 1, Status Success, Build number 1, Source version master, Submitter [redacted], Duration 00:00:00, Correlated [redacted]'. Action buttons for this row include Stop build, View artifacts, View logs, Delete build, and Retry build.

Now our project is ready with us.

Now go to pipeline getting started and then here you have to create a pipeline. Steps for creating pipeline

The screenshot shows two instances of the AWS CodePipeline console side-by-side. Both instances are titled 'CodePipeline - AWS Developer'.

Left Instance:

- The title bar says 'Developer Tools > CodePipeline'.
- The sidebar on the left lists: Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), Deploy (CodeDeploy), Pipeline (CodePipeline), Getting started, Pipelines, and Settings.
- The main content area displays the 'AWS CodePipeline' landing page with the heading 'visualize and automate the different stages of your software release process'. It includes a 'Create AWS CodePipeline pipeline' button and a 'How it works' section with a screenshot of a browser showing a CI/CD pipeline.
- A 'Pricing (US)' table shows the cost for an 'Entitled Pipeline': \$1/month.
- Feedback links at the bottom include 'Looking for language identical? Find it in the new Unified Settings' and 'Feedback'.

Right Instance:

- The title bar says 'Developer Tools > CodePipeline > Pipelines > Create new pipeline'.
- The sidebar on the left is identical to the left instance.
- The main content area is titled 'Add source stage'.
- The 'Source' section shows a configuration for a 'GitHub (Version 1)' provider, with a message: 'You have successfully configured the action with the provider.'
- A warning message in a box states: 'The GitHub (Version 1) action is not recommended. The selected action uses OAuth app to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub (Version 2) action to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources. Learn more.'
- Feedback links at the bottom include 'Looking for language identical? Find it in the new Unified Settings' and 'Feedback'.

The screenshots illustrate the configuration of an AWS CodePipeline pipeline.

Screenshot 1: GitHub Action Configuration

This screenshot shows the configuration of a GitHub action. A success message states: "You have successfully configured the action with the provider". A note indicates: "The GitHub (Version 1) action is not recommended. The selected action uses OAuth apps to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub Version 2 action to access your repository by creating a connection. Connectors use GitHub Apps to manage authentication and can be shared with other resources. Learn more".

Screenshot 2: Pipeline Settings

This screenshot shows the "Choose pipeline settings" step. It includes fields for "Repository" (awsmtu/statedatabasefordevops), "Branch" (main), and "Change detection options". Two options are available: "GitHub webhooks (recommended)" and "AWS CodePipeline".

Screenshot 3: Build Stage Configuration

This screenshot shows the "Build - optional" configuration step. It includes fields for "Build provider" (AWS CodeBuild), "Region" (Asia Pacific (Mumbai)), "Project name" (awsmtu), and "Environment variables - optional". It also shows the "Build type" section with "Single build" selected (Triggers a single build).

The image consists of three vertically stacked screenshots from the AWS CodePipeline console, illustrating the step-by-step creation of a new pipeline.

Screenshot 1: Step 1 - Add deploy stage

This screenshot shows the "Add deploy stage" configuration screen. The "Deploy provider" dropdown is set to "Amazon S3". The "Region" dropdown is set to "Asia Pacific (Mumbai)". The "Bucket" input field contains "swasti-tru-destination". The "Deployment path" input field is empty. A checked checkbox says "Extract file before deploy". Below the main form, there is a link to "Additional configuration". At the bottom right are buttons for "Cancel", "Previous", "Skip deploy stage", and "Next".

Screenshot 2: Step 3: Add build stage

This screenshot shows the "Step 3: Add build stage" configuration screen. It displays a "Build action provider" section with "AWS CodeBuild" selected. The "ProjectName" input field contains "codebuild". Below the main form, there is a link to "Step 4: Add deploy stage". At the bottom right are buttons for "Cancel", "Previous", and "Create pipeline".

Screenshot 3: Step 4: Add deploy stage

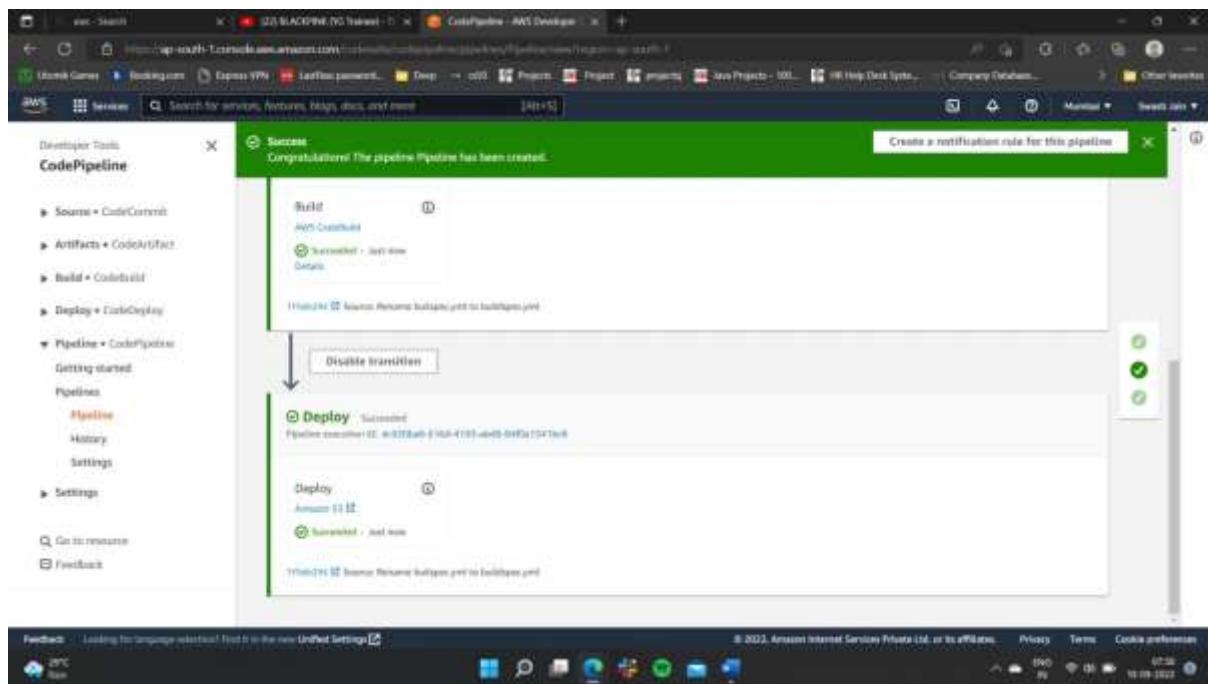
This screenshot shows the "Step 4: Add deploy stage" configuration screen. It displays a "Deploy action provider" section with "Amazon S3" selected. The "Extract" dropdown is set to "true". The "BucketName" input field contains "swasti-tru-destination". Below the main form, there is a link to "Step 3: Add build stage". At the bottom right are buttons for "Cancel", "Previous", and "Create pipeline".

Screenshot 1: AWS CodePipeline Pipeline Creation - Step 1: Source

Screenshot 2: AWS CodePipeline Pipeline Creation - Step 2: Build

Screenshot 3: AWS CodePipeline Pipeline Creation - Step 3: Deploy

The screenshots show the sequential steps of creating a new AWS CodePipeline pipeline. Each step adds a new stage: Source, Build, and Deploy. The pipeline is currently empty, with no transitions between stages.



The screenshot shows the AWS S3 console interface. On the left, a sidebar menu for 'Amazon S3' includes options like Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, Access analyzer for S3, Block Public Access settings for this account, Storage Lens, Dashboards, AWS Organizations settings, Feature spotlight, and AWS Marketplace for S3. The main content area displays a 'Follow security best practices for S3.' banner and the path 'Amazon S3 > Buckets > codepipeline-ap-south-1-866101200226 > Pipeline/'. Below this, there are tabs for 'Objects' (selected) and 'Properties'. A 'Copy S3 URI' button is visible. The 'Objects' section shows a table with two items:

Name	Type	Last modified	Size	Storage class
buildArtif/	Folder			
SourceArtif/	Folder			

SourceArti/

Name	Type	Last modified	Size	Storage class
infrastructure.zip	zip	September 10, 2022, 07:46:44 (UTC+05:30)	1.2 KB	Standard

BuildArtif/

Name	Type	Last modified	Size	Storage class
NTfyApp.zip	zip	September 10, 2022, 07:49:22 (UTC+05:30)	1.3 KB	Standard

The screenshots show the AWS S3 console interface for managing buckets and objects.

Screenshot 1: Objects List for Bucket 'swasti-tcw-destination'

This screenshot shows the 'Objects' tab for the 'swasti-tcw-destination' bucket. It lists three objects:

Name	Type	Last modified	Size	Storage class
buildings.json	JSON	September 10, 2022, 07:49:52 (UTC+05:30)	191.0 B	Standard
index.html	HTML	September 10, 2022, 07:49:52 (UTC+05:30)	1.2 KB	Standard
README.md	MD	September 10, 2022, 07:49:52 (UTC+05:30)	75.0 B	Standard

Screenshot 2: Bucket Policy for Bucket 'swasti-tcw-destination'

This screenshot shows the 'Bucket policy' section for the 'swasti-tcw-destination' bucket. The policy document is as follows:

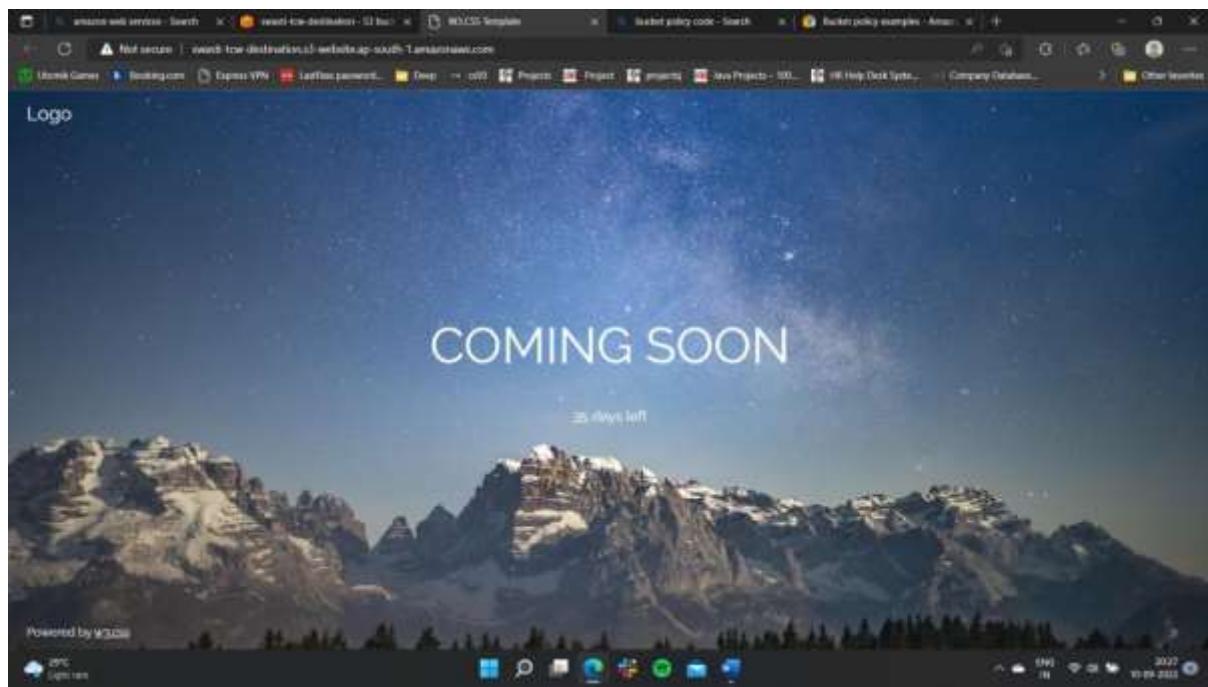
```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicReadGetObject",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::swasti-tcw-destination/*"
        }
    ]
}

```

Screenshot 3: Object Ownership for Object 'index.html'

This screenshot shows the 'Object Ownership' section for the 'index.html' object. It indicates that the object was written to the bucket from another AWS account and that access control is managed by AWS IAM.



[W3.CSS Template \(swasti-tcw-destination.s3-website.ap-south-1.amazonaws.com\)](http://swasti-tcw-destination.s3-website.ap-south-1.amazonaws.com)

Experiment no. 5

Name : Swasti Jain

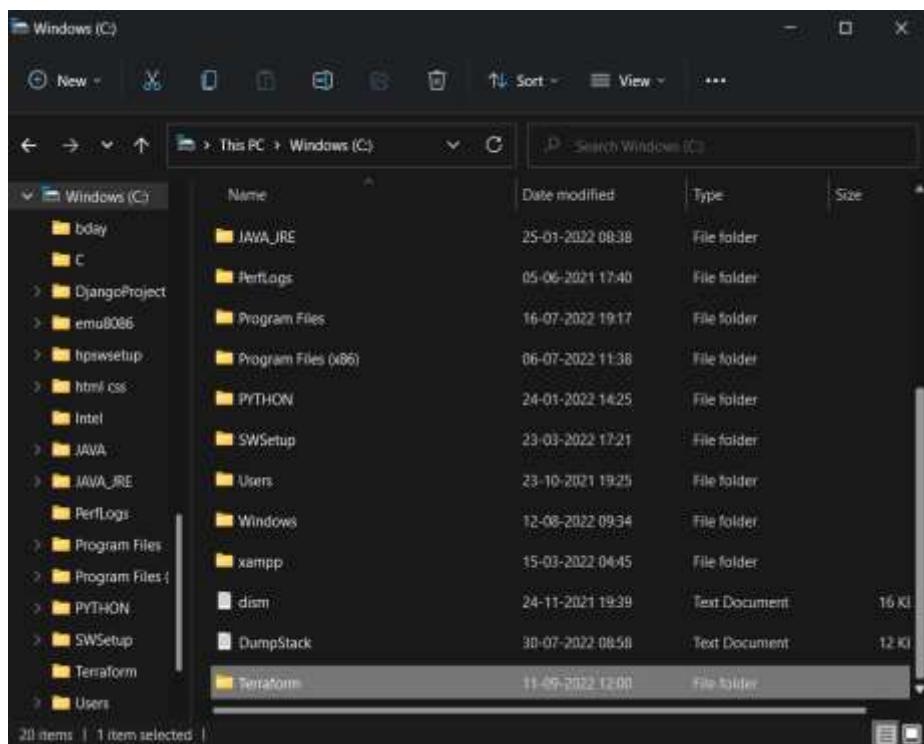
Roll No: 24

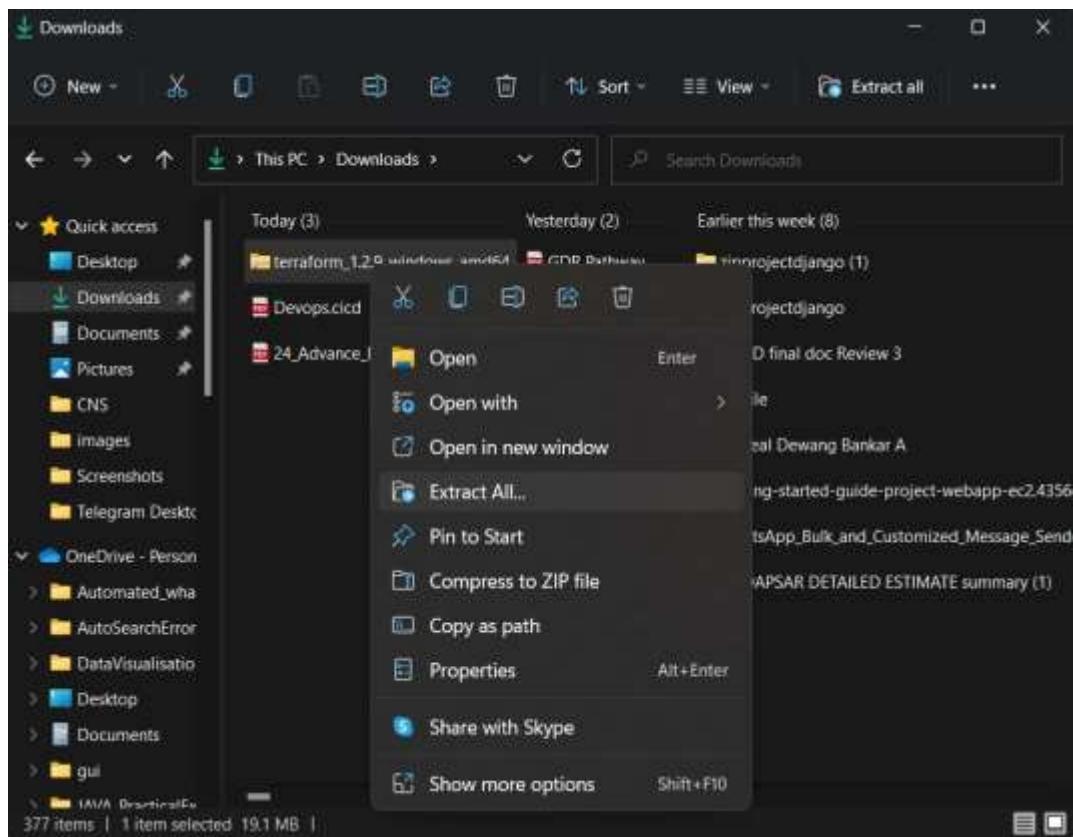
Aim: To understand terraform lifecycle, core concepts/terminologies and install it on any Machine. After that create a folder and a file with extension .tf. Mention the provider and resource in the .tf file as shown below .Before writing the provider get the access key and secret key from aws .Also you will have to get ami from aws

Procedure:

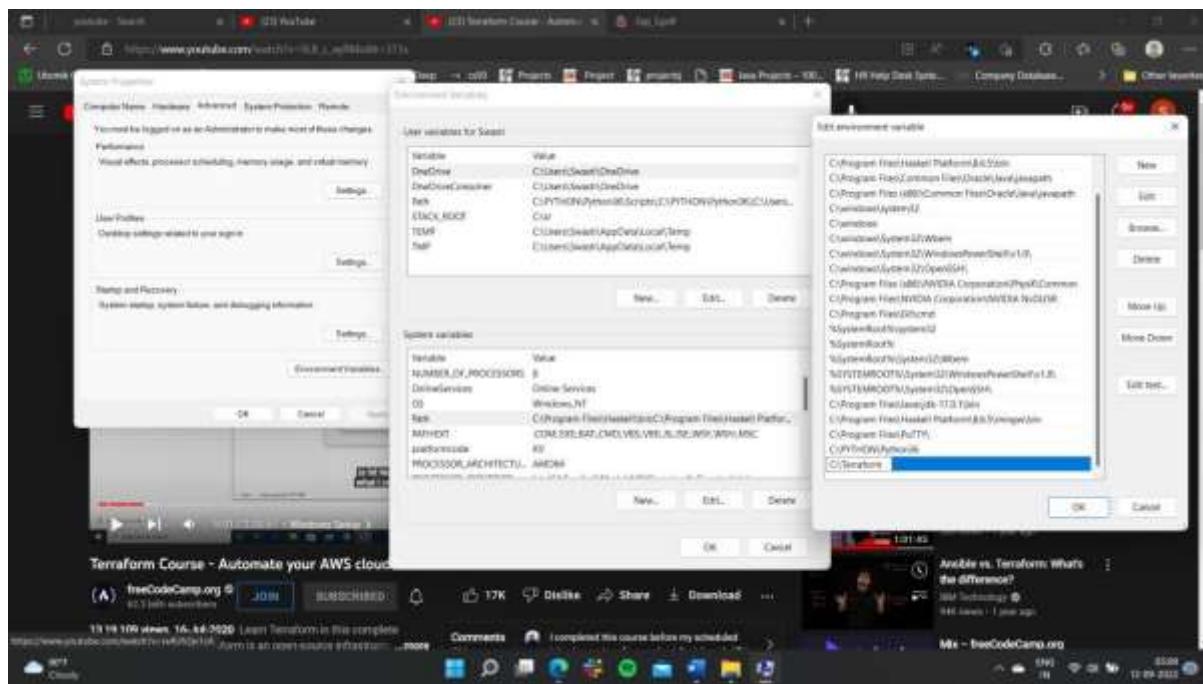
- 1) Install Terraform package and install it on your machine Link for terraform installation:
:https://releases.hashicorp.com/terraform/1.2.7/terraform_1.2.7_windows_amd64.zip
- 2) Make a folder named terraform and extract all the downloaded packages over there.

The screenshot shows a web browser window with the Terraform website open. The URL in the address bar is <https://www.terraform.io/downloads>. The page content includes the Terraform logo and the text "Terraform 1.29". Below this, there is a download link for "Windows" labeled "200 MB 64-bit". At the bottom of the page, there is a button with the text "Sign up for Terraform Cloud".



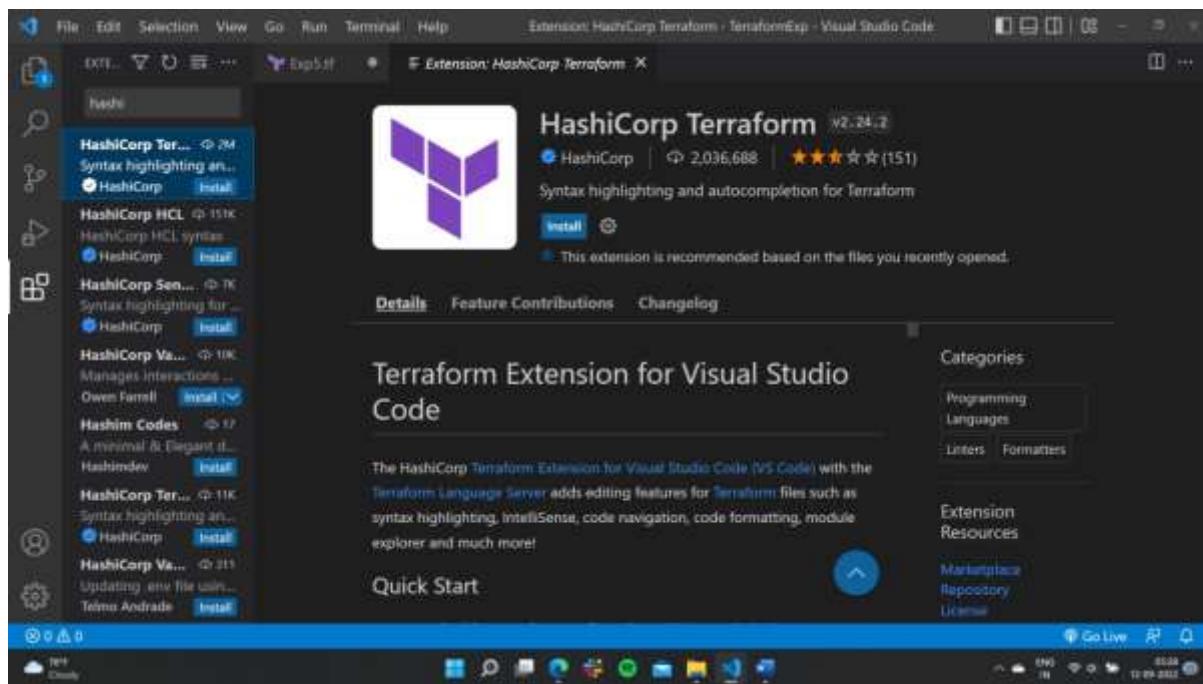


3)Now go to the environment variables and add path of the terraform folder

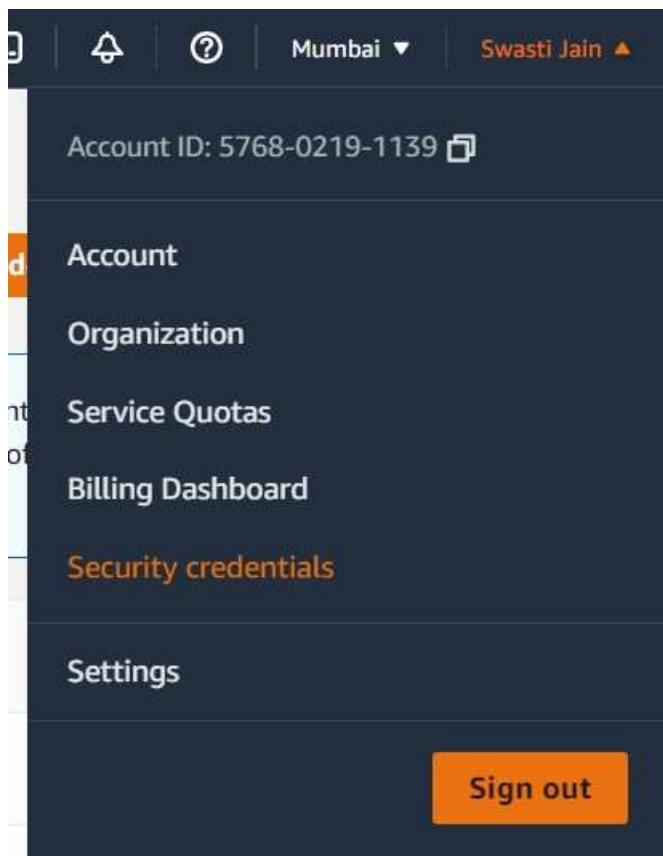


We have successfully configured terraform on our system.

4) Install the terraform extension in VScode



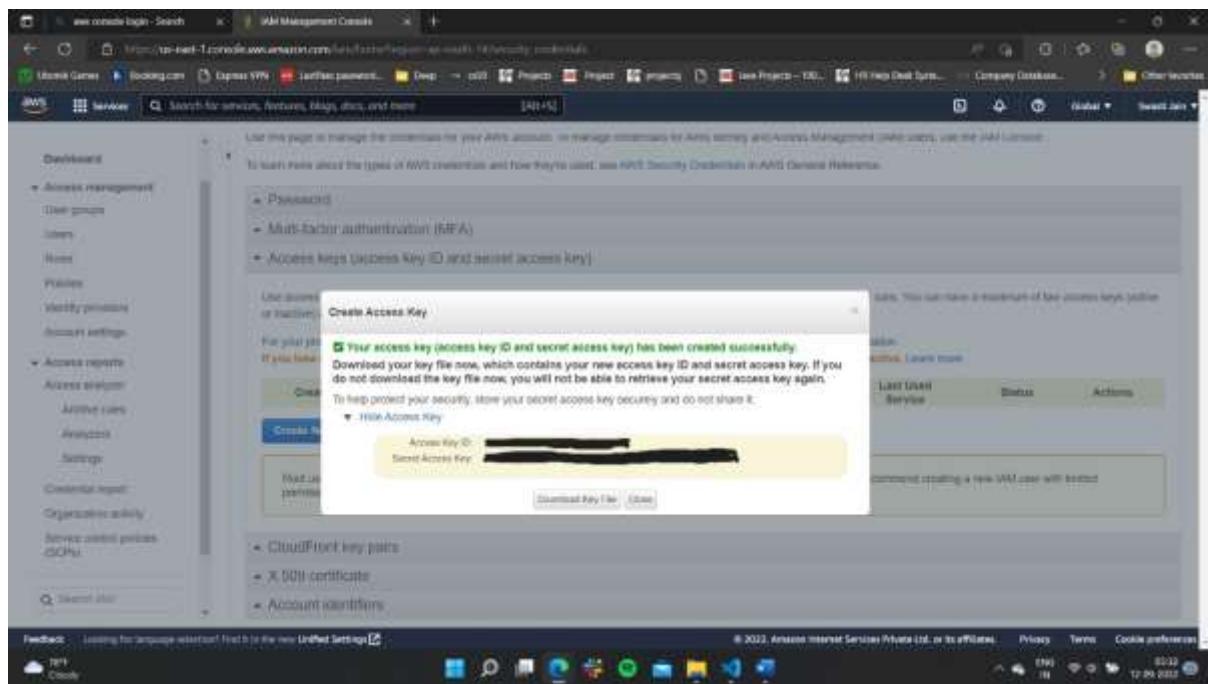
5) Open aws account, Go to Your account and click on security credentials



- 6) Click on Access key if you have a key you can use that only if you don't have a key you have to create a new access key:

The screenshot shows the AWS IAM Management Console. The left sidebar has a tree view with nodes like Dashboard, Access management, User groups, Users, Roles, Policies, Identity providers, Account settings, Access reports, Access analyzer, Analytics, Settings, Credential report, Organization activity, Service control policies (SCPs), and a search bar. The main content area is titled 'Access Keys' and contains a message about managing AWS credentials. It lists three types of credentials: Passwords, Multi-factor authentication (MFA), and Access Keys (access key ID and secret access key). A note says: 'Use access keys to make programmatic calls to AWS from the AWS CLI, Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time.' Below this is another note: 'For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation. If you lose or forget your secret key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.' A large blue button labeled 'Create New Access Key' is prominently displayed. A callout box over this button states: 'Root user access keys provide unrestricted access to your entire AWS account. If you need long-term access keys, we recommend creating a new IAM user with limited permissions and generating access keys for that user instead.' At the bottom of the page, there are tabs for CloudFront key pairs, X.509 certificate, and Account identifiers.

7)Create new access key and also download the root key folder

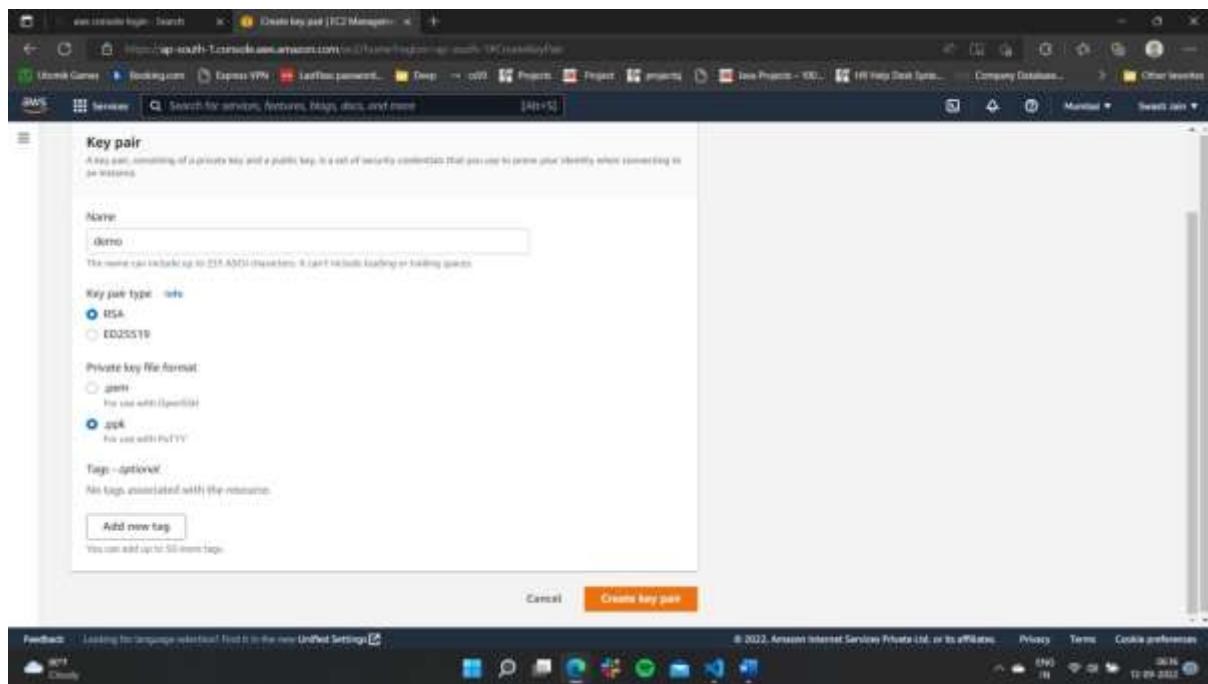


8) Create new folder named terraform and also create .tf file, Now add the code with the access key and secret key

```
Exp5.tf
File Edit Selection View Go Run Terminal Help
Exp5 - Terraform (1) - Visual Studio Code
provider "aws" {
  region = "ap-south-1"
  access_key =
  secret_key =
}

resource "aws_lambda" "my-first-server" {
  arn = "arn:aws:lambda:ap-south-1:608230297729700"
  instance_type="t1.micro"
```

9) Add the key pair name



10) Now we have to initilize terraform

The screenshot shows a Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** Exp5.tf - TerraformExp - Visual Studio Code.
- Left Sidebar:** Shows a tree view of the file structure: Exp5.tf > ... > resource aws_instance by-latest-server. Below it are icons for Find, Replace, and other code navigation.
- Terminal Tab:** The terminal tab is active, showing the command-line interface for Terraform. The output is as follows:

```
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, either
commands will detect it and remind you to do so if necessary.PS C:\Users\Swasti\OneDrive\Desktop\TerraformExp> terraform plan
No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your
configuration and found no differences, so no changes are
needed.
PS C:\Users\Swasti\OneDrive\Desktop\TerraformExp>
```
- Bottom Status Bar:** Shows file information (In 11, Col 1), encoding (Spaces: 4, UTF-8, CRLF), Terraform, and a Go Live button. It also displays system status like battery level (50%), signal strength, and time (12:09 2022).

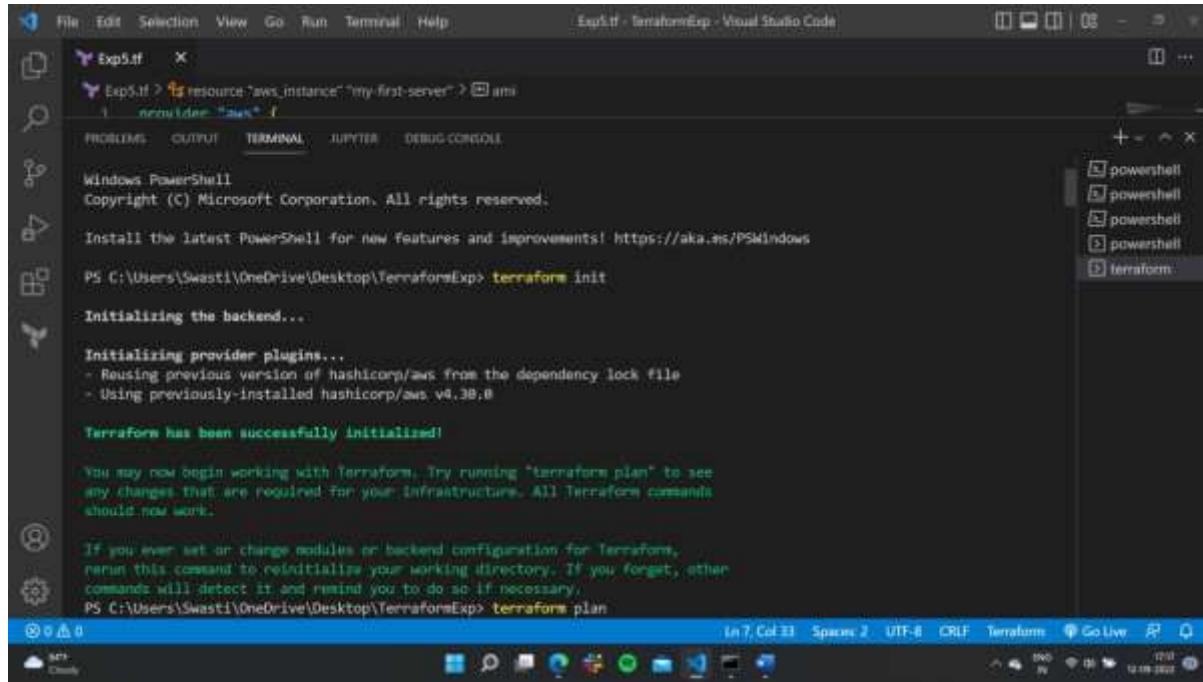
Experiment 6 : Terraform

Name : Swasti Jain

Roll No: 24

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

To initialize open cmd and type terraform init



The screenshot shows a Visual Studio Code interface with a terminal window open. The terminal tab is selected at the top. The command `terraform init` is being run in a Windows PowerShell environment. The output shows the initialization process, including the download of provider plugins for AWS. A message at the end indicates successful initialization and provides instructions for running `terraform plan`.

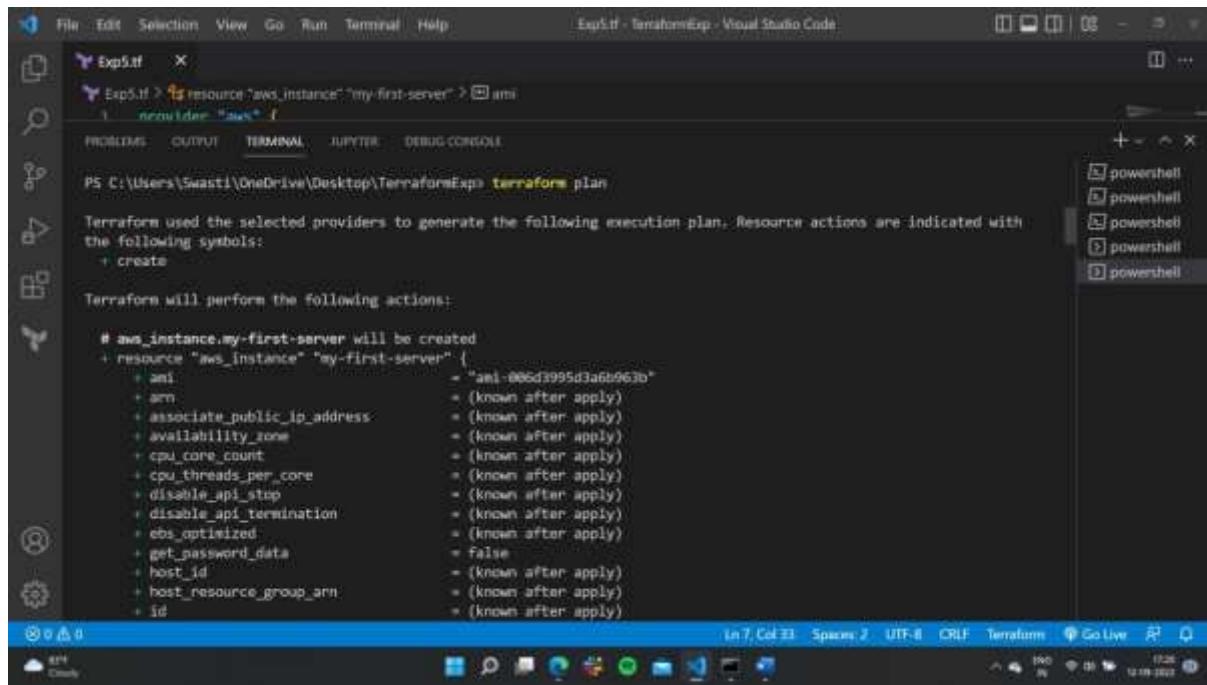
```
PS C:\Users\Swasti\OneDrive\Desktop\TerraformExp> terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v4.38.0

Terraform has been successfully initialized!

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

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

If you want to first see what changes will appear write the command `terraform plan`



The screenshot shows a Visual Studio Code interface with a terminal window open. The terminal title is "Exp5.tf - TerraformExp - Visual Studio Code". The command "terraform plan" has been run, and the output shows the following:

```
PS C:\Users\Swasti\OneDrive\Desktop\TerraformExp> 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:

# aws_instance.my-first-server will be created
+ resource "aws_instance" "my-first-server" {
    ami = "ami-086d3995d3a6b963b"
    + arm
    + associate_public_ip_address
    + availability_zone
    + cpu_core_count
    + cpu_threads_per_core
    + disable_api_stomp
    + disable_api_termination
    + ebs_optimized
    + get_password_data
    + host_id
    + host_resource_group_arn
    + id
}
```

To build a infrastructure using terraform you'll have to use the command `terraform apply`

The screenshot shows a Visual Studio Code interface with a dark theme. The left sidebar contains icons for file operations like Open, Save, Find, and Refresh. The main editor area displays a Terraform configuration file named `Exp5.tf`. The code defines a single AWS instance resource:

```
resource "aws_instance" "my-first-server" {
  ami           = "ami-0a2a1a2a1a2a1a2a1"
  instance_type = "t2.micro"
  key_name      = "aws"
}

provider "aws" {
  region = "us-east-1"
}
```

Below the code, the terminal pane shows the output of the `terraform plan` command:

```
Plan: 1 to add, # to change, # to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.

PS C:\Users\Swasti\Desktop\TerraformExp> terraform apply

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

The status bar at the bottom indicates the terminal has 7 lines, 33 columns, and is using UTF-8 encoding.

To apply any changes to terraform we will write the changes we want to apply in the .tf file run the terraform apply command

The screenshot shows a Visual Studio Code window titled "Exp5.tf - TerraformExp - Visual Studio Code". The code editor displays a Terraform configuration file (Exp5.tf) with the following content:

```
resource "aws_instance" "my-first-server" {
  ami           = "ami-006d3995d3a6b963b"
  ami_type      = "(known after apply)"
  associate_public_ip_address = "(known after apply)"
  availability_zone = "(known after apply)"
  cpu_core_count = "(known after apply)"
  cpu_threads_per_core = "(known after apply)"
  disable_api_stop = "(known after apply)"
  disable_api_termination = "(known after apply)"
  ebs_optimized = "(known after apply)"
  get_password_data = false
  host_id       = "(known after apply)"
```

The terminal tab shows the command "terraform apply" being run, and the output indicates the execution plan for creating an AWS instance named "my-first-server". The status bar at the bottom shows the file has 7 lines, 33 columns, and is saved.

The screenshot shows a Visual Studio Code interface with a dark theme. On the left is a file tree with a single item: 'Exp5.tf'. The main editor area contains Terraform configuration code:

```
resource "aws_instance" "my-first-server" {
  ami           = "ami-0e7a2dd1f9ca05"
  instance_type = "t2.micro"
}

Plan: 1 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

aws_instance.my-first-server: Creating...
aws_instance.my-first-server: Still creating... [10s elapsed]
aws_instance.my-first-server: Still creating... [20s elapsed]
aws_instance.my-first-server: Still creating... [30s elapsed]
aws_instance.my-first-server: Creation complete after 32s [id=i-08dcb7e2dd1f9ca05]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PS: C:\Users\Swastik\OneDrive\Desktop\TerraformExp>
```

The terminal tab at the bottom shows the command line environment with the prompt 'PS: C:\Users\Swastik\OneDrive\Desktop\TerraformExp>'. The status bar at the bottom right indicates the file has 7 lines, 33 columns, and is saved.

ss of running instance in aws

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like 'New EC2 Experience', 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Tags', 'Limits', 'Instances' (selected), 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', 'Capacity Reservations', 'Images' (selected), 'AMIs', and 'Elastic Block Store'. The main content area has a search bar with 'instance state = running' and a 'Clear Filters' button. A table lists one instance: Name - i-00d1b7e2aff94005, Instance ID - i-00d1b7e2aff94005, Instance State - running, Status check - 12 mins, Alarm status - 2/2 checks passed, Availability Zone - ap-south-1b, and Public IP/V4 - [REDACTED]. Below the table is a modal window titled 'Select an instance' with a single entry: 'i-00d1b7e2aff94005'. At the bottom, there's a feedback link, a footer with copyright information and links to Privacy, Terms, and Cookies, and a system tray with icons for battery, signal, volume, and time.

Now we have to terminate and destroy the connection. To destroy a terraform infrastructure we will use the command `terraform destroy`

```

File Edit Selection View Go Run Terminal Help Exp5.tf - TerraformExp - Visual Studio Code
PROBLEMS OUTPUT TERMINAL JUPYTER DEBUG CONSOLE
PS C:\Users\Swasti\OneDrive\Desktop\TerraformExp> terraform destroy
aws_instance.my-first-server: Refreshing state... [id=i-09dcb7e2dd1f9ca05]
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:

# aws_instance.my-first-server will be destroyed
resource "aws_instance" "my-first-server" {
  ami                               = "ami-086d3995d3a6b963b" -> null
  ami_id                            = "arn:aws:ec2:ap-south-1:576882191139:instance/i-09dcb7e2dd1f9ca05" -> n
  ...
  associate_public_ip_address       = true -> null
  availability_zone                 = "ap-south-1b" -> null
  cpu_core_count                    = 1 -> null
  cpu_threads_per_core             = 1 -> null
  disable_api_stop                 = false -> null
  disable_api_termination          = false -> null
  ebs_optimized                     = false -> null
  get_password_data                = false -> null
  hibernation                       = false -> null
}

```

```

File Edit Selection View Go Run Terminal Help Exp5.tf - TerraformExp - Visual Studio Code
PROBLEMS OUTPUT TERMINAL JUPYTER DEBUG CONSOLE
...
- volume_id           = "vvol-075a30ff27b257a82" -> null
- volume_size         = 8 -> null
- volume_type         = "gp2" -> null
}

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

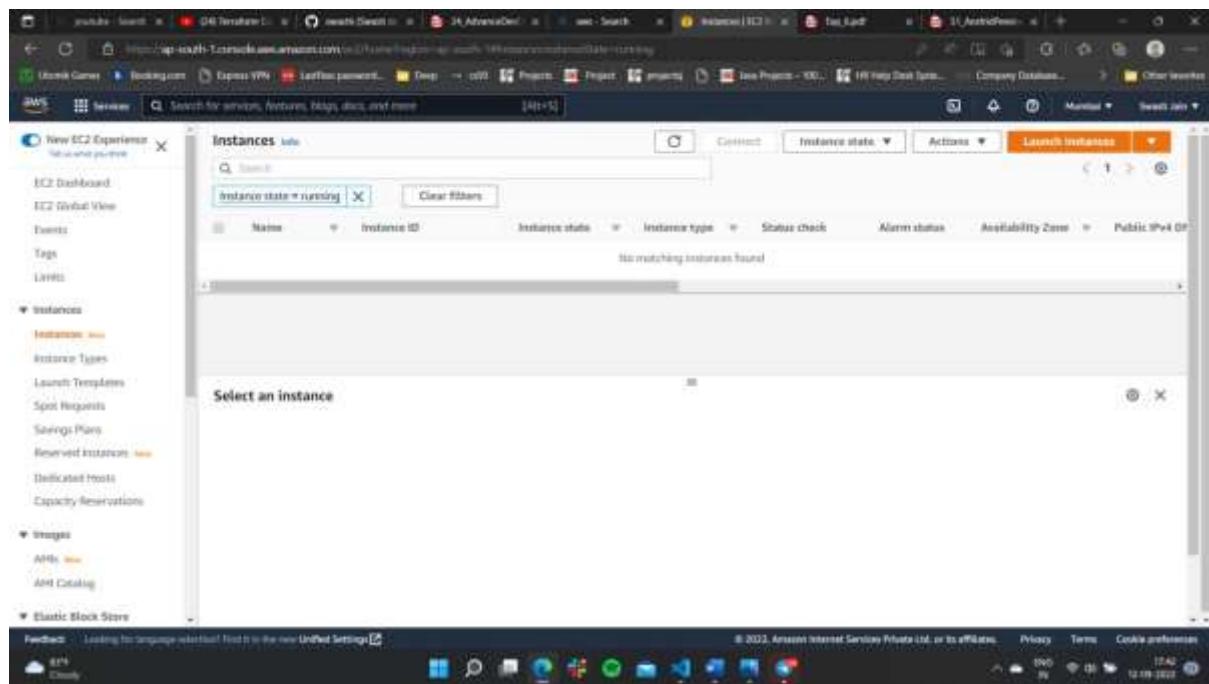
Do you really want to destroy all resources?
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

aws_instance.my-first-server: Destroying... [id=i-09dcb7e2dd1f9ca05]
aws_instance.my-first-server: Still destroying... [id=i-09dcb7e2dd1f9ca05, 10s elapsed]
aws_instance.my-first-server: Still destroying... [id=i-09dcb7e2dd1f9ca05, 20s elapsed]
aws_instance.my-first-server: Still destroying... [id=i-09dcb7e2dd1f9ca05, 30s elapsed]
aws_instance.my-first-server: Still destroying... [id=i-09dcb7e2dd1f9ca05, 40s elapsed]
aws_instance.my-first-server: Destruction complete after 50s

Destroy completed! Resources: 1 destroyed.
PS C:\Users\Swasti\OneDrive\Desktop\TerraformExp>

```



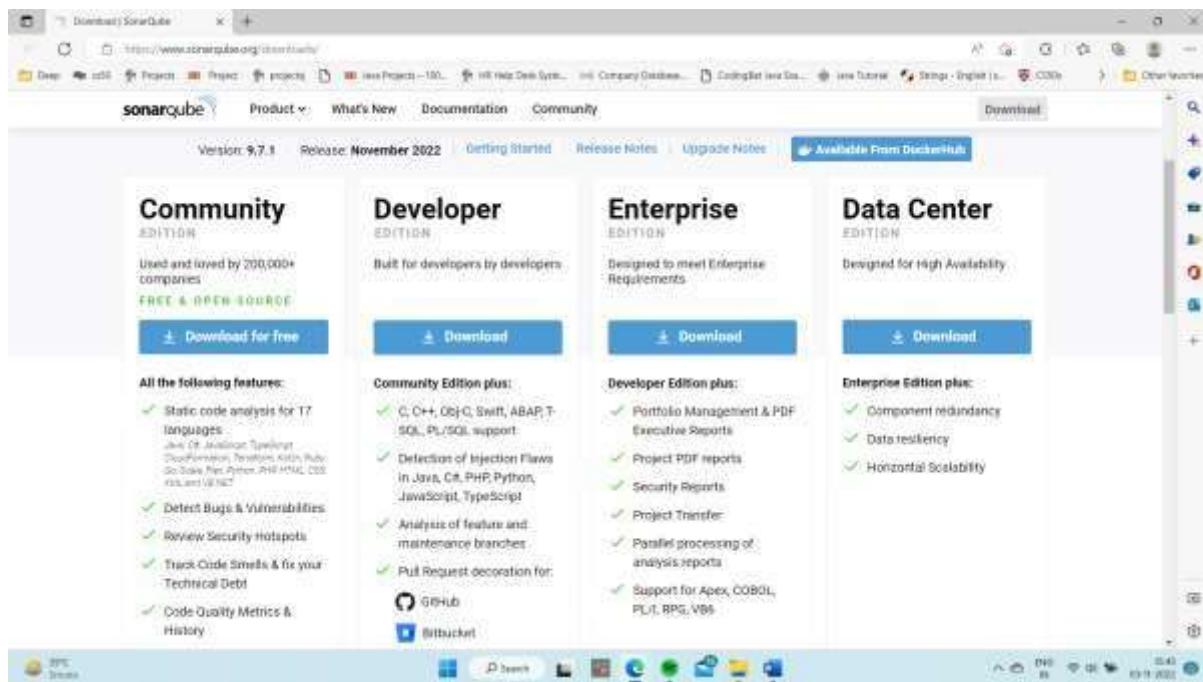
Experiment no. 7

Aim: Perform static analysis using sonarqube and show the analysis.

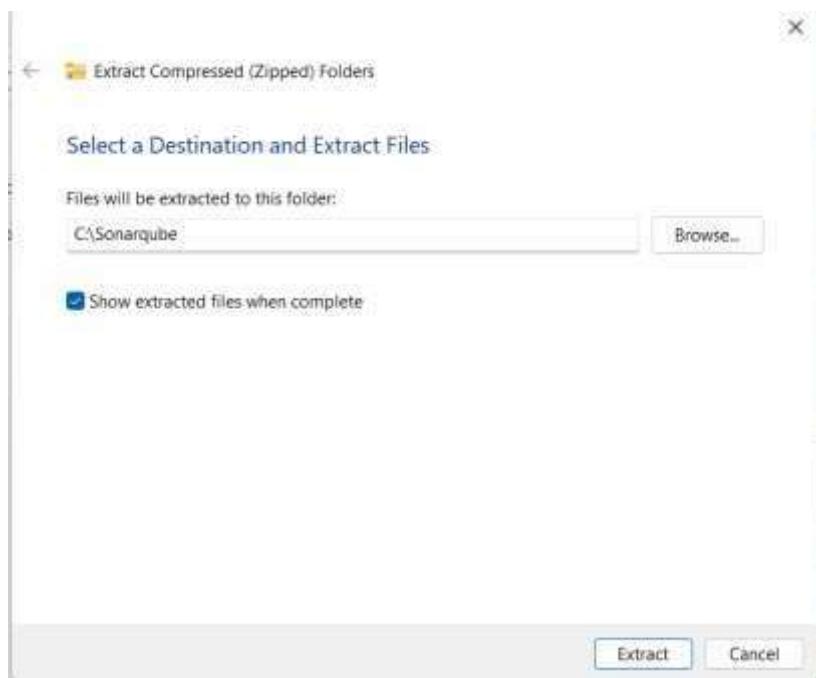
Steps:

Aim: To understand the Static Analysis SAST process and show the analysis of any web based application

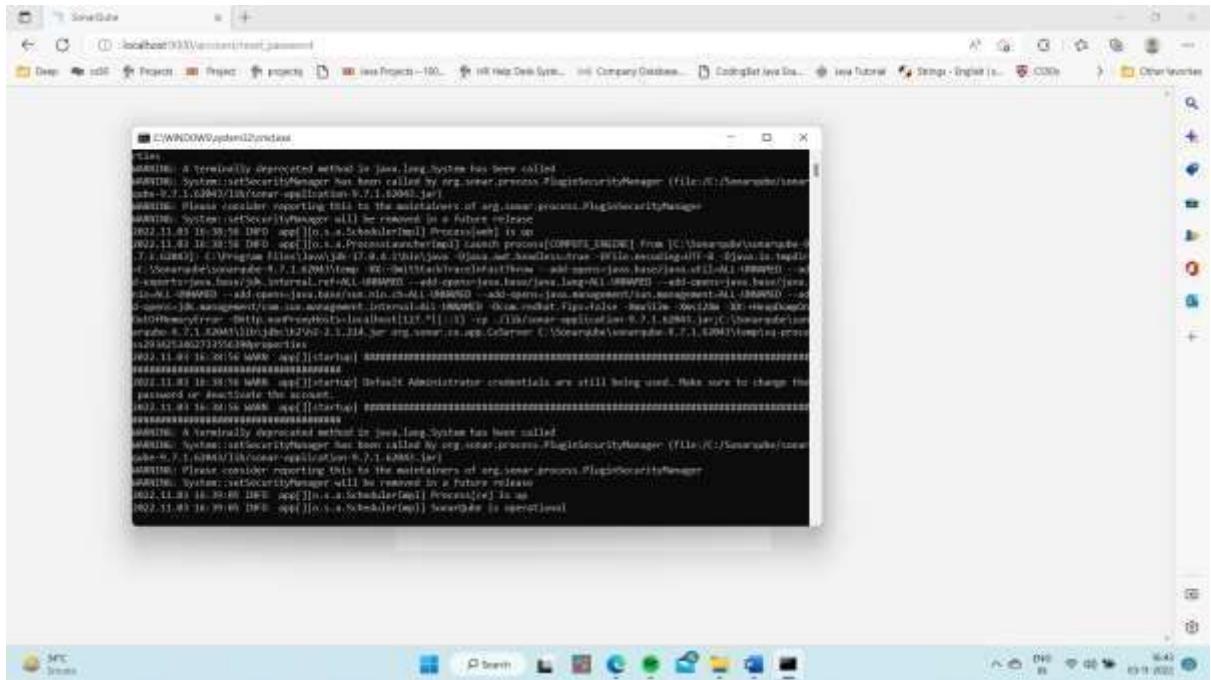
Download the SonarQube Community Edition zip file.



Download and install Java 11 on your system. 3. As a non-rootuser, unzip it, let's say in C:\sonarqube or /opt/sonarqube.



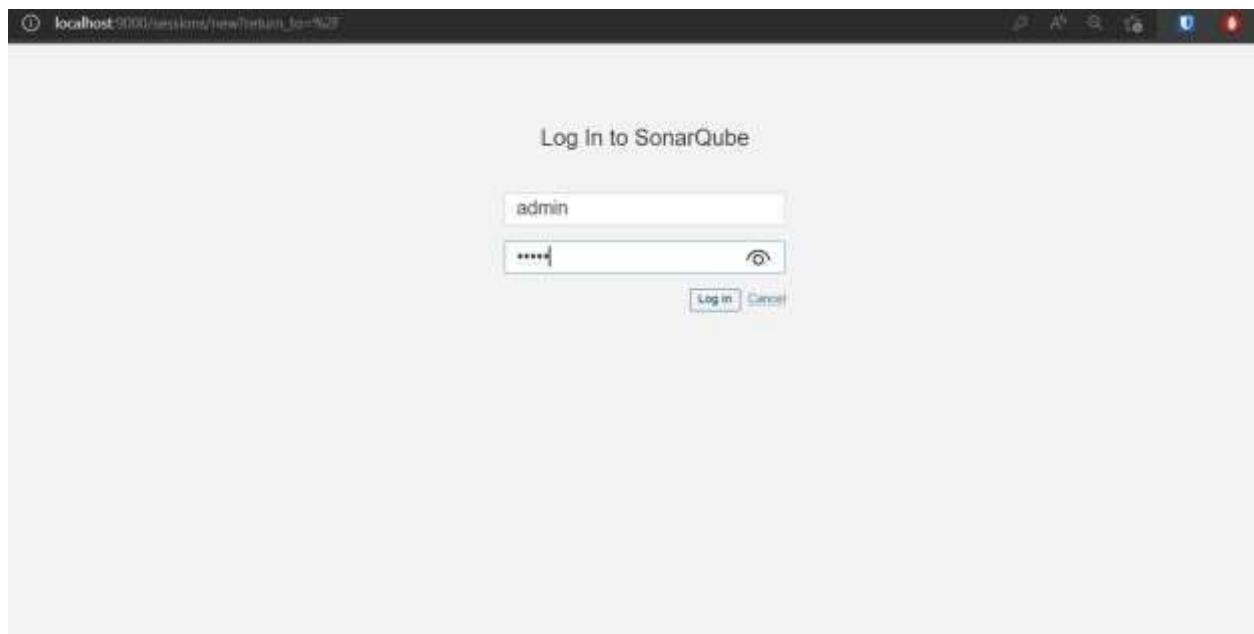
As a non-root user, start the SonarQube Server:



```

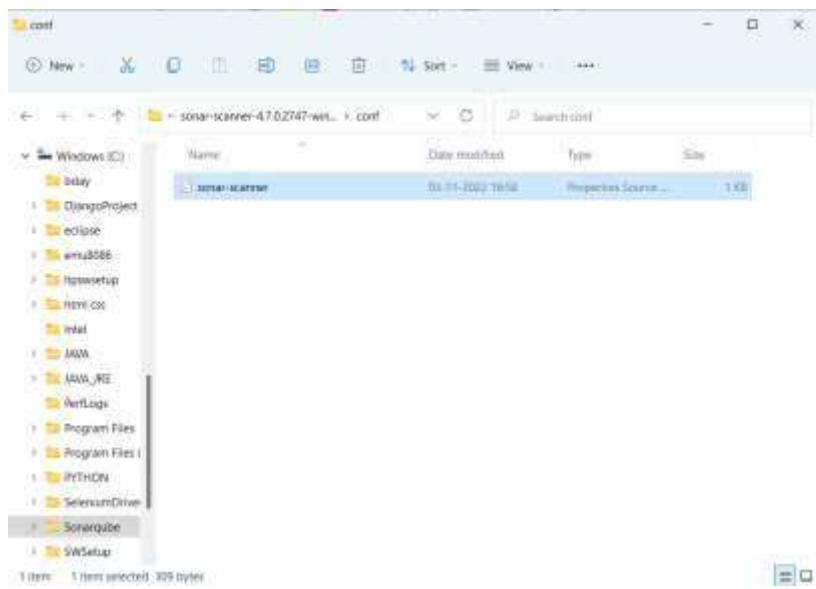
2022-11-09 18:38:58 DEBUG [app][localhost:9000][SchedulerImpl] Process[SaveGate] is up
2022-11-09 18:38:58 DEBUG [app][localhost:9000][ProcessLauncherService] launch process [CORPUS_ENGINE]. From [C:\Users\mudit\Downloads\corpus-engine-0.7.1.0.0001] to [C:\Users\mudit\Downloads\corpus-engine-0.7.1.0.0001]. C:\Users\mudit\Downloads\corpus-engine-0.7.1.0.0001\java -jar -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\Users\mudit\Downloads\corpus-engine-0.7.1.0.0001\ -XX:+UseConcMarkSweepGC -XX:+UseG1GC -XX:+G1NewSizeCalculator -XX:MaxGCPauseMillis=5000 -XX:InitialHeapSize=16M -XX:MaxHeapSize=16M -XX:MinHeapSize=1M -XX:SoftRefLRUPolicyMSPerMB=1000 -XX:ParallelGCThreads=2 -XX:+AlwaysPreTouch -XX:+DoFullCollectionOnGCT -XX:+DisableExplicitGC -XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+PrintGCTimeStamps -XX:+UseG1GC
2022-11-09 18:38:58 DEBUG [app][localhost:9000][ProcessLauncherService] process [CORPUS_ENGINE] is up
2022-11-09 18:38:58 DEBUG [app][localhost:9000][SchedulerImpl] Process[SaveGate] is up
2022-11-09 18:38:58 DEBUG [app][localhost:9000][ProcessLauncherService] process [CORPUS_ENGINE] has been called by org.wesir.process.PluginSecurityManager. (File: C:\Users\mudit\Downloads\corpus-engine-0.7.1.0.0001\src\main\java\org\wesir\process\PluginSecurityManager.java)
2022-11-09 18:38:58 DEBUG [app][localhost:9000][ProcessLauncherService] Please consider reporting this to the maintainers of org.wesir process [PluginSecurityManager].
2022-11-09 18:38:58 DEBUG [app][localhost:9000][ProcessLauncherService] SaveGate will be removed in a future release
2022-11-09 18:38:58 DEBUG [app][localhost:9000][SchedulerImpl] Process[SaveGate] is up
2022-11-09 18:38:58 DEBUG [app][localhost:9000][SchedulerImpl] SaveGate is operational
  
```

Once your instance is up and running, Log in to <http://localhost:9000> using System Administrator credentials

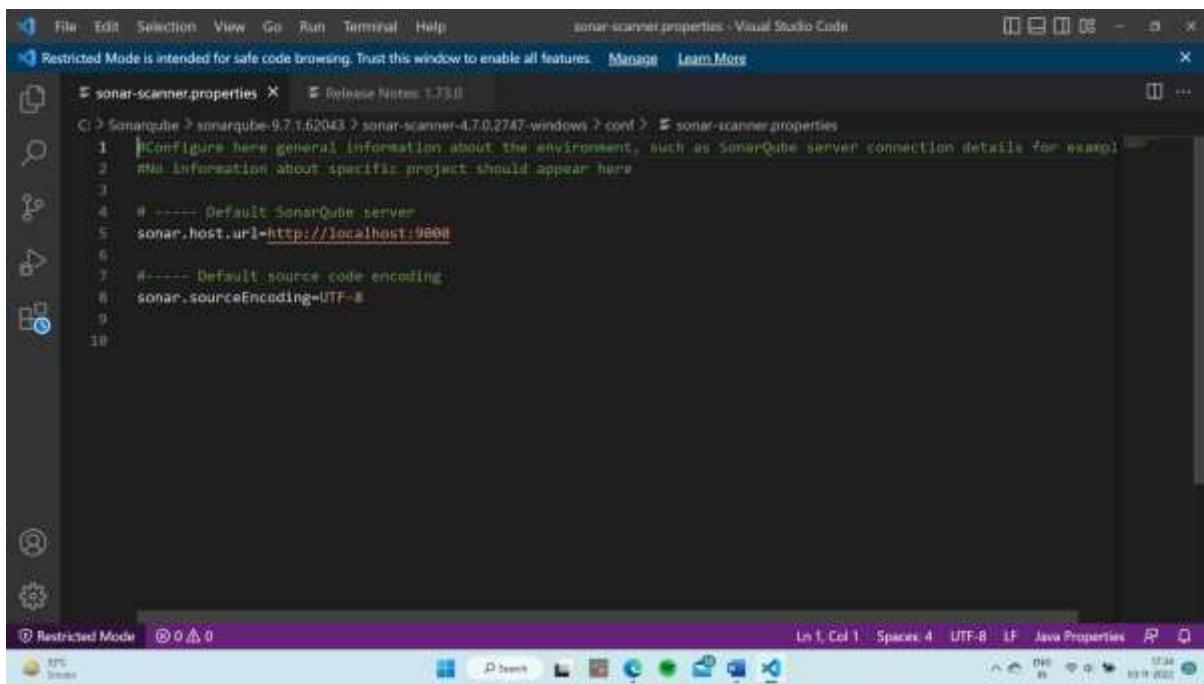


Install sonar scanner from the official link and extract it to the same folder where sonarqube has been installed <https://docs.sonarqube.org/latest/analysis/scan/sonarscanner>

Conf>sonar-scanner

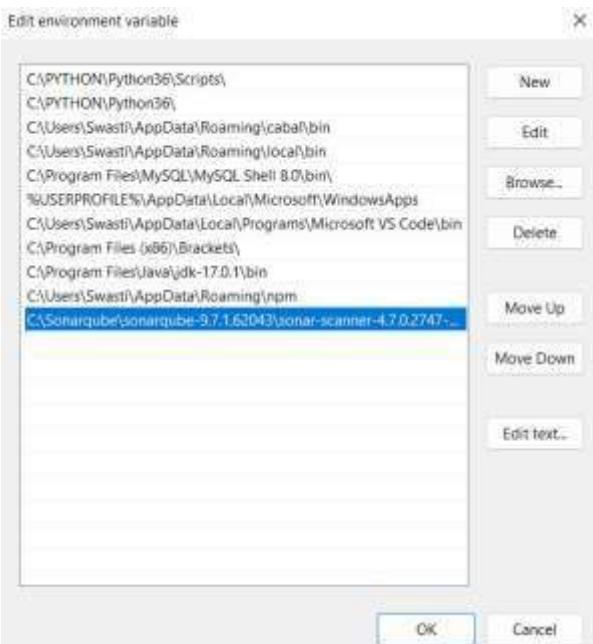


Update the global settings to point to your SonarQube server by editing



```
sonar-scanner.properties - Visual Studio Code
File Edit Selection View Go Run Terminal Help
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
sonar-scanner.properties X Release Notes: 1.75.0
SonarQube > sonarqube-9.7.1.62043 > sonar-scanner-4.7.0.2747-windows > conf > sonar-scanner.properties
1 #Configure here general information about the environment, such as SonarQube server connection details for example
2 #All information about specific project should appear here
3
4 #----- Default SonarQube server
5 sonar.host.url=http://localhost:9000
6
7 #----- Default source code encoding
8 sonar.sourceEncoding=UTF-8
9
10
```

Add the bin path of sonarscanner into the environment variables of your system



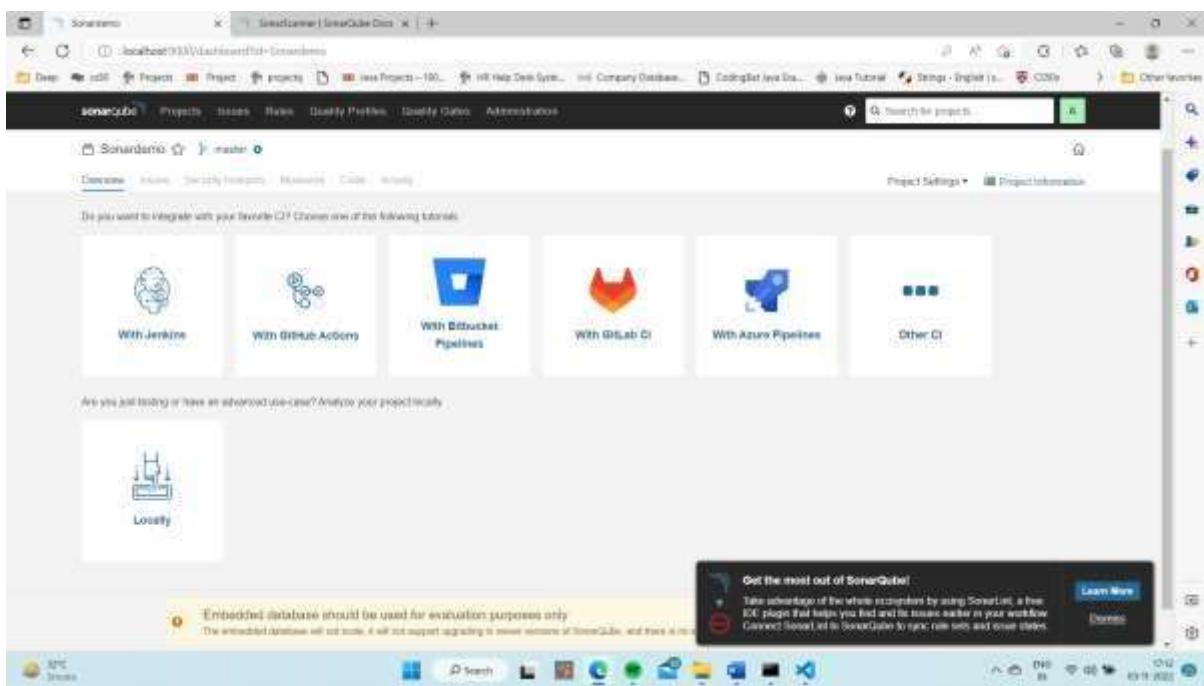
Verify your installation by opening a new shell and executing the command `sonar-scanner -h`

```
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

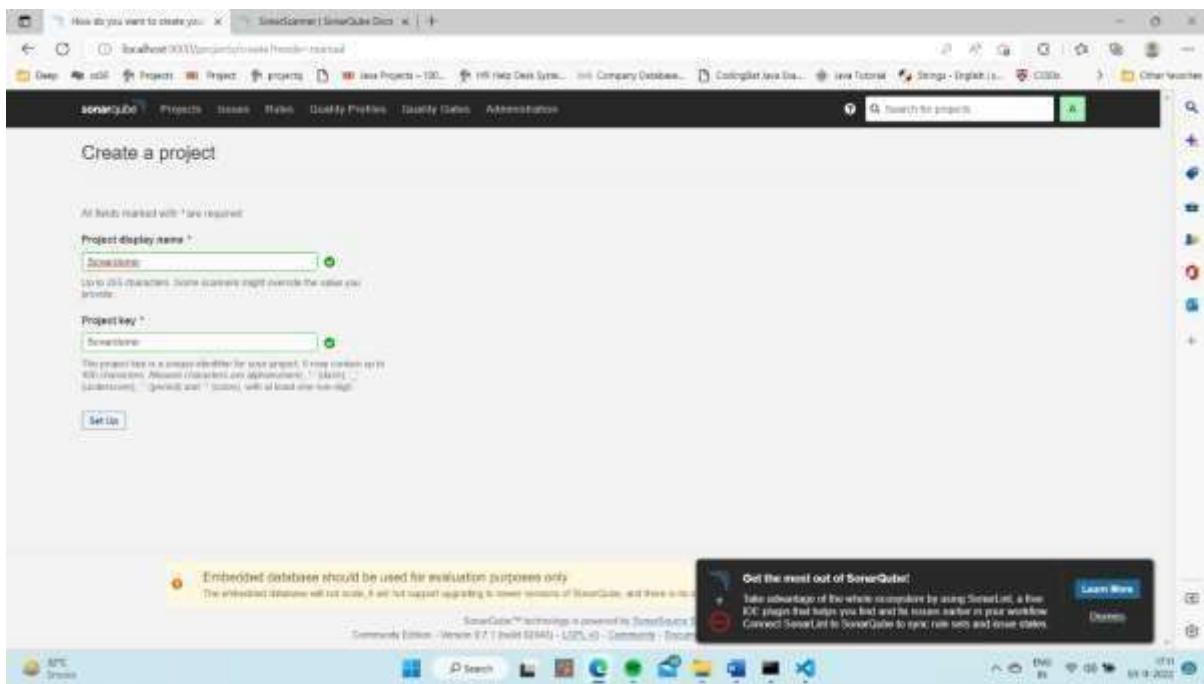
C:\Users\Swasti>sonar-scanner -h
INFO:
INFO: usage: sonar-scanner [options]
INFO:
INFO: Options:
INFO: -D,<arg>           Define property
INFO: -h,-help             Display help information
INFO: -v,-version          Display version information
INFO: -X,-debug            Produce execution debug output

C:\Users\Swasti>
```

- . Now that our sonar-scanner is installed, we will create a project in our sonarqube that is logged in through <http://localhost:9000/projects/create>.

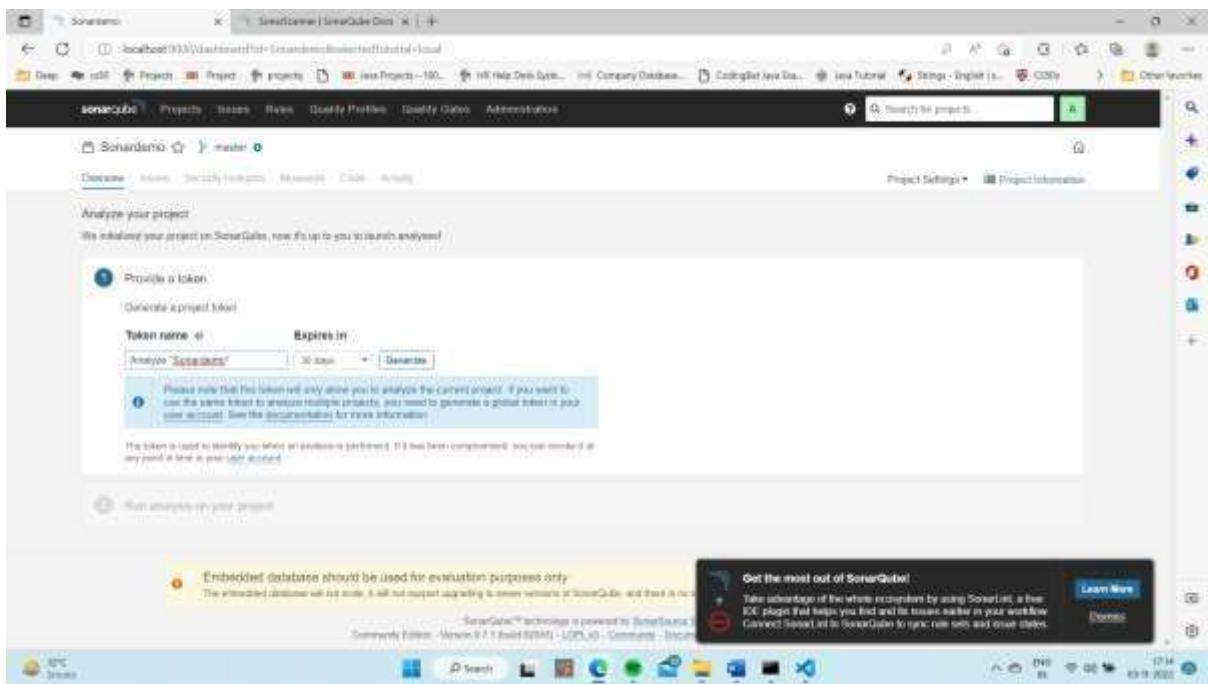


Click on manually and create the project by giving in the values for the asked fields



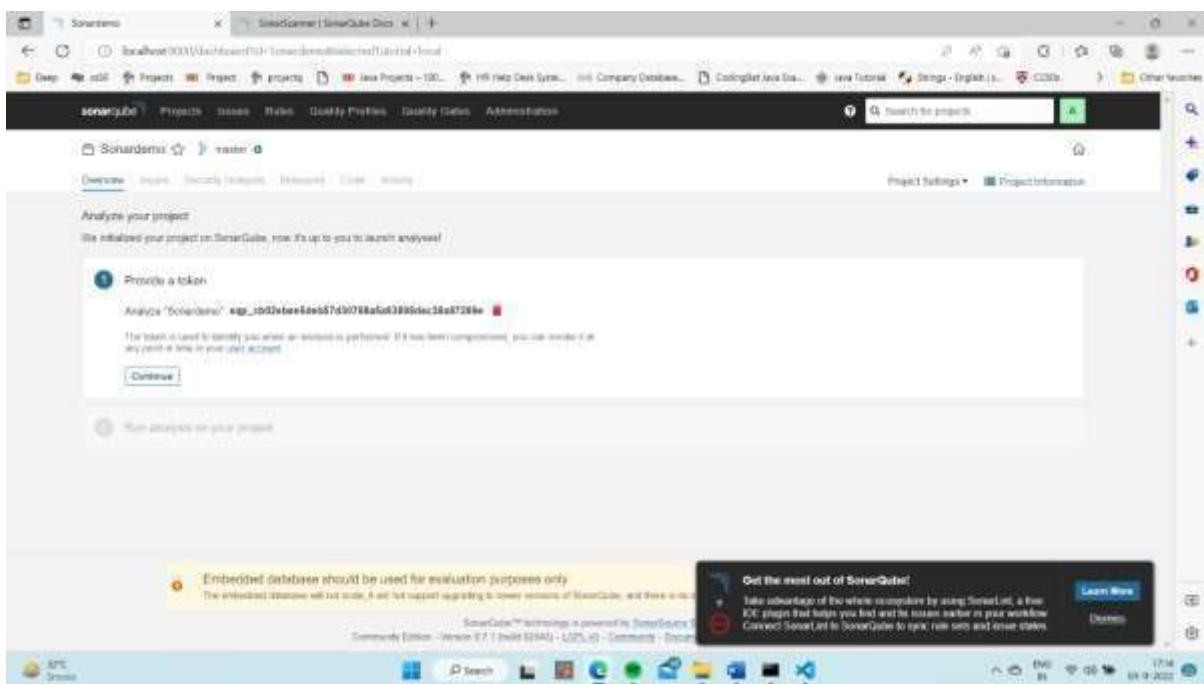
It will look something like this after clicking set up, now in order to analyze the code, click on locally.

Click on generate a token

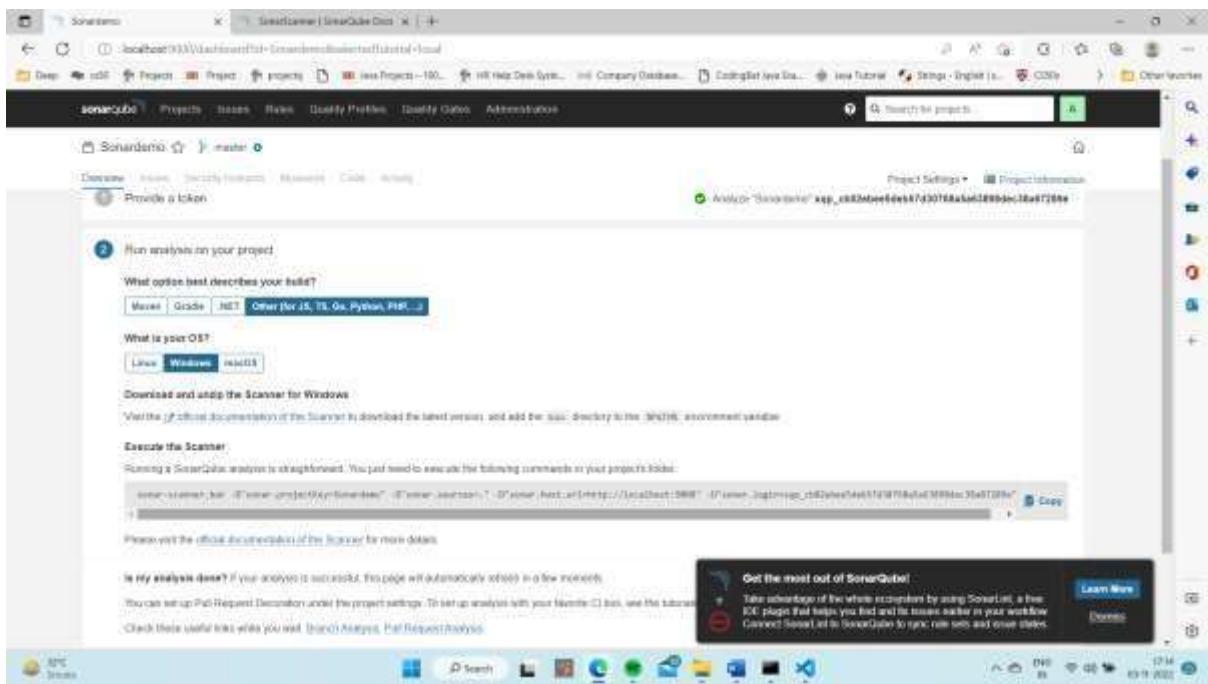


The following output will be received

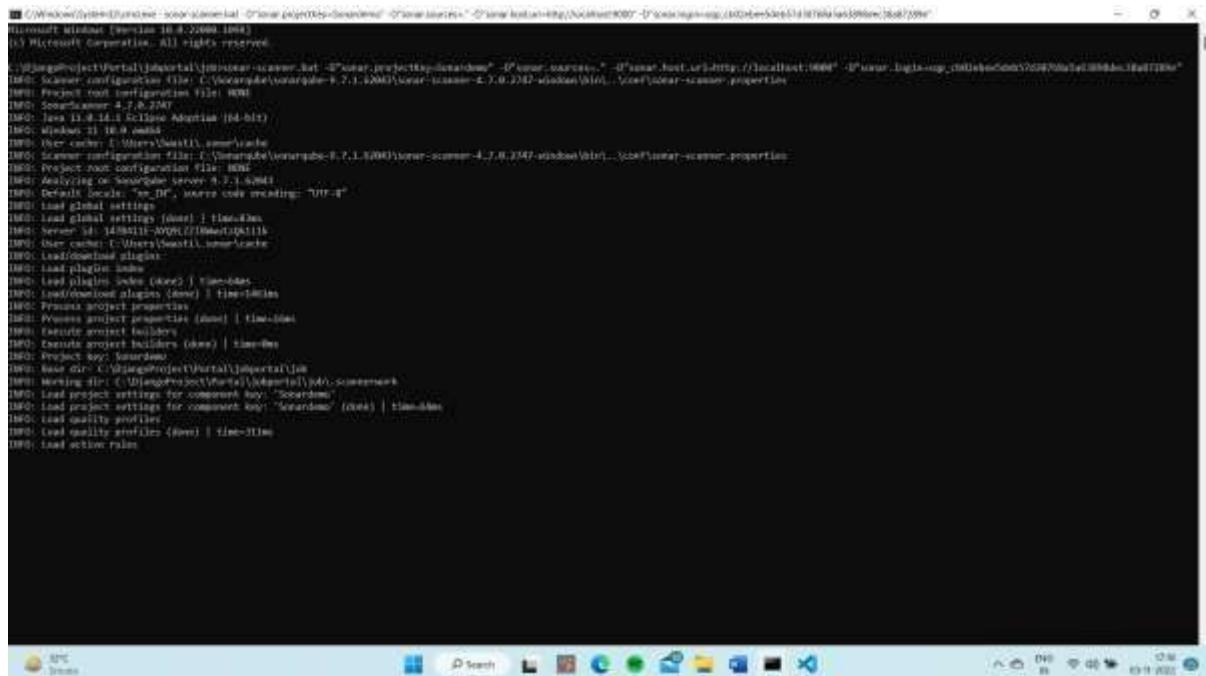
After the token is copied and clicked on continue, then choose your project type



In this case it falls under other



Now go the root of your project directory and run the command given by the sonar scanner

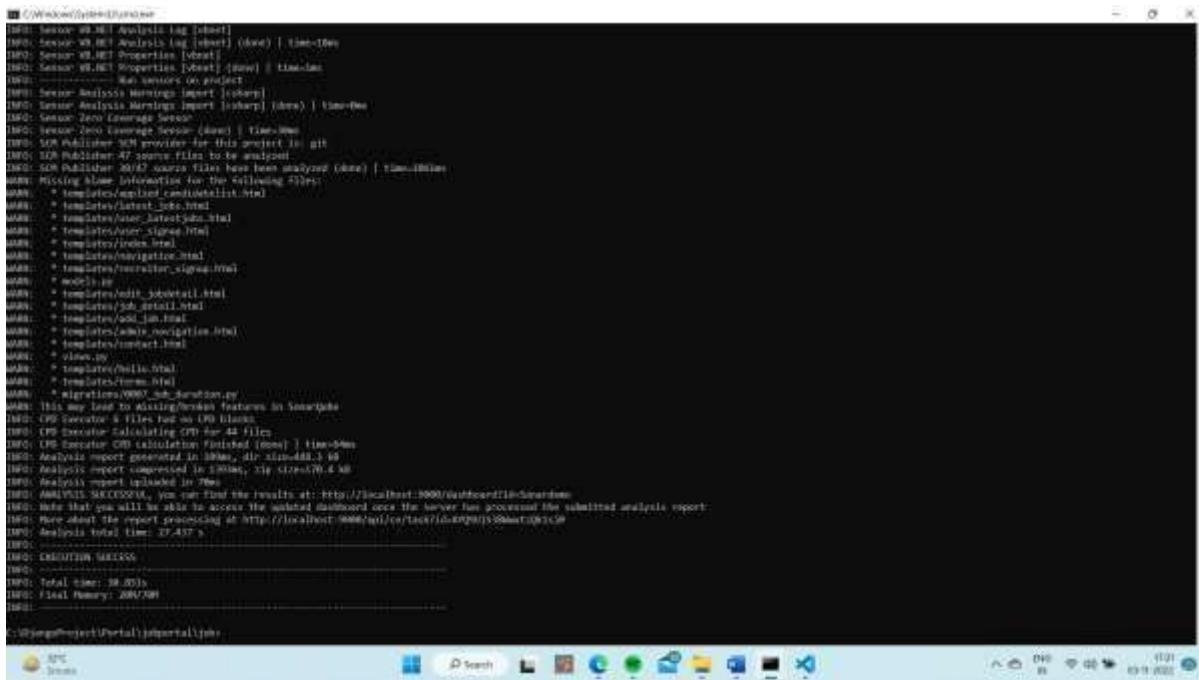


```
C:\Windows\system32\cmd.exe -c "clear-project -C 'clear-scan.bat' -O 'clear-project>ScansDir' -G 'clear-scan>log'" &> C:\Windows\system32\cmd.log 2>> C:\Windows\system32\cmd.log 3>> C:\Windows\system32\cmd.log
Microsoft Windows [version 10.0, build 10040]
Copyright (c) Microsoft Corporation. All rights reserved.

C:\Windows\system32\cmd.exe -c "clear-project -C 'clear-scan.bat' -O 'clear-project>ScansDir' -G 'clear-scan>log'" &> C:\Windows\system32\cmd.log 2>> C:\Windows\system32\cmd.log 3>> C:\Windows\system32\cmd.log
clear-project -C 'clear-scan.bat' -O 'clear-project>ScansDir' -G 'clear-scan>log'
[1]: clear-scan -S 'clear-project>ScansDir'
[2]: clear-project -C 'clear-scan.bat' -O 'clear-project>ScansDir' -G 'clear-scan>log'

09:58:41:6846 INFO: Scanner configuration file: C:\Users\UmarIb\OneDrive\UmarIb\clear-project\8.7.1.0203\clear-scanner-4.7.0.2047-windows\bin\..\conf\clear-scanner.properties
09:58:41:6846 INFO: Project root configuration file: none
09:58:41:6846 INFO: Default scale: "m_Dr", source code encoding: "UTF-8"
09:58:41:6846 INFO: Load global settings
09:58:41:6846 INFO: Load global settings (done) | time=0ms
09:58:41:6846 INFO: Server id: 1498511-WIN-2T0W6RQH113
09:58:41:6846 INFO: User cache: C:\Users\UmarIb\OneDrive\UmarIb\clear-project\8.7.1.0203\clear-scanner-4.7.0.2047-windows\bin\..\cache
09:58:41:6846 INFO: Load/desired plugins:
09:58:41:6846 INFO: Load plugin index
09:58:41:6846 INFO: Load plugin index (done) | time=0ms
09:58:41:6846 INFO: Load/desired profiles (done) | time=0ms
09:58:41:6846 INFO: Project project properties (done) | time=0ms
09:58:41:6846 INFO: Execute engine builders
09:58:41:6846 INFO: Execute project builders (done) | time=0ms
09:58:41:6846 INFO: Project key: SourceDir
09:58:41:6846 INFO: Base dir: C:\Windows\system32\cmd\clear-project.bat
09:58:41:6846 INFO: Working dir: C:\Windows\system32\cmd\clear-project\8.7.1.0203\clear-scanner
09:58:41:6846 INFO: Load project settings for component key: "Scanner"
09:58:41:6846 INFO: Load default settings for component key: "Scanner" (done) | time=0ms
09:58:41:6846 INFO: Load quality profiles (done) | time=0ms
09:58:41:6846 INFO: Load action rules
09:58:41:6846 INFO: Scan completed successfully in 00:00:02
```

Once the scanning process is done, you'll see the following output



```

C:\Windows\system32\cmd.exe
1800: Session 99 MET Analysis [Import] (done) | time=0ms
1800: Session 99 MET Analysis [Import] (done) | time=0ms
1800: Session 99,87 Properties [Import] (done) | time=0ms
1800: Session 99,87 Properties [Import] (done) | time=0ms
1800: ..... Has variants on project
1800: Session Analysis Warnings Import [Import] (done) | time=0ms
1800: Session-Zero Coverage Session
1800: Session-Zero Coverage Session (done) | time=0ms
1800: SCM Publisher SCM provider for this project is git
1800: SCM Publisher AT source files to be analyzed
1800: SCM Publisher SCM analysis files have been analyzed (done) | time=0ms
1800: Missing class information for the following files:
1800:   * com.sipcymetamod.listablelist.html
1800:   * tempDeploy/list.html
1800:   * templates/user_index.html
1800:   * templates/index.html
1800:   * templates/navigation.html
1800:   * templates/monitor_index.html
1800:   * models.xml
1800:   * templates/edit_index.html
1800:   * templates/edit_detail.html
1800:   * templates/add_index.html
1800:   * templates/admin_navigation.html
1800:   * templates/contact.html
1800:   * views.py
1800:   * templates/help.html
1800:   * templates/terms.html
1800:   * migrations/0001_initial.py
1800:   * migrations/0002_alter_features_in_Software
1800: CSV Executive & Files has no CSV blank
1800: CSV Executive calculating CSV for 44 files
1800: CSV Executive CSV calculation finished (done) | time=0ms
1800: Analysis report generated in 180ms, dir size=448.3 KB
1800: Analysis report compressed in 180ms, zip size=76.4 KB
1800: Analysis report uploaded in 7ms
1800: ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard#dashboard
1800: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
1800: You can download the report processing at http://localhost:9000/api/task/download/159400000000000000
1800: Analysis total time: 27.217 s
1800:
1800: EXECUTION SUCCESS
1800:
1800: Total time: 39.05s
1800: Final Memory: 208479K
1800:

C:\MyJavaProject\Portal\src\main\java\com\...

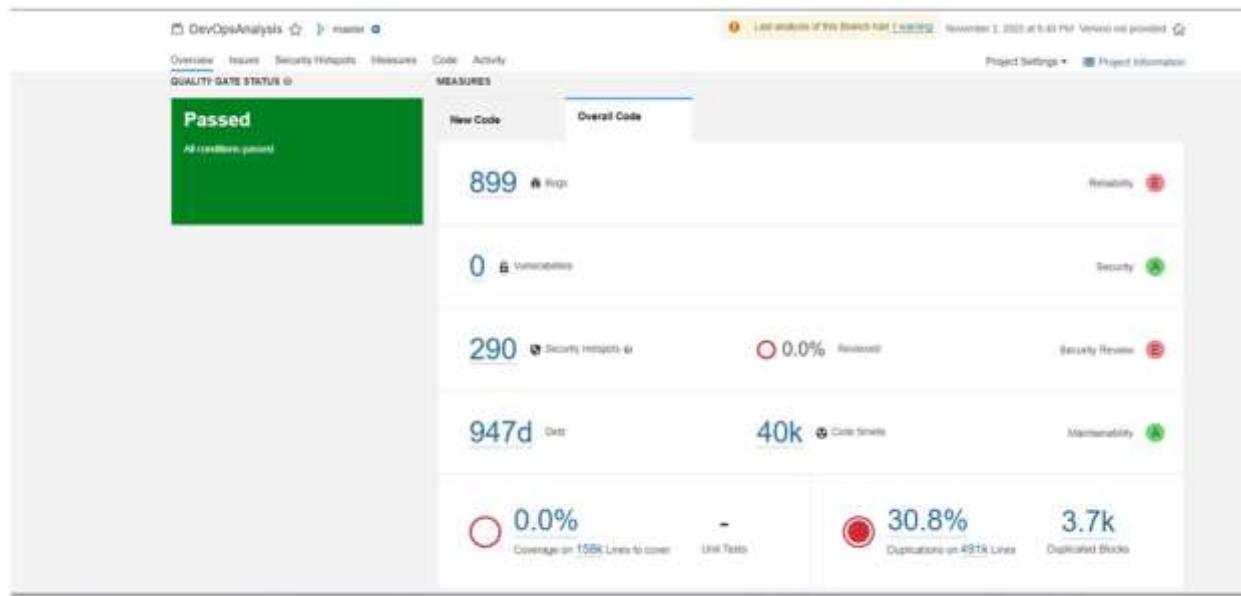
```

Now we can see the results in our sonarqube dashboard under our project

The screenshot shows the SonarQube web interface for a project named "Sonardemo". The main dashboard displays the following key metrics:

- Overall Status:** Passed (Green)
- Overall Code Quality:** 84% (Yellow)
- Security:** 0 vulnerabilities (Green)
- Security Review:** 9 issues (Red)
- Dependencies:** 25 (Yellow)
- Time Spent:** 2d 5h (Grey)

The interface includes a navigation bar with links like "Sonardemo", "Projects", "Issues", "Rules", "Quality Profiles", "Quality Gates", and "Administration". A search bar at the top right allows searching for projects. On the right side, there is a sidebar with various icons for project management and reporting.



Conclusion: The analysis is performed and displayed in sonarqube successfully.

Experiment no. 9

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

1. Install and Configure Nagios Core

Step 1: Update the system packages.

Sudo apt update and sudo apt upgrade:

Activities Terminal Nov 3 19:21

```
root@swasti-VirtualBox: /home/swasti
swasti@swasti-VirtualBox:~$ sudo su
[sudo] password for swasti:
root@swasti-VirtualBox:/home/swasti# apt get update
E: Invalid operation get
root@swasti-VirtualBox:/home/swasti# apt update
Hit:2 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease
Ign:1 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:4 https://pkg.jenkins.io/debian-stable binary/ Release
Hit:5 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:7 http://security.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
13 packages can be upgraded. Run 'apt list --upgradable' to see them.
W: http://pkg.jenkins.io/debian-stable/binary/Release.gpg: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
root@swasti-VirtualBox:/home/swasti# upgrade
upgrade: command not found
root@swasti-VirtualBox:/home/swasti# apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
Try Ubuntu Pro beta with a free personal subscription on up to 5 machines.
Learn more at https://ubuntu.com/pro
The following packages have been kept back:
  distro-info-data grub-efi-amd64-bin grub-efi-amd64-signed
```

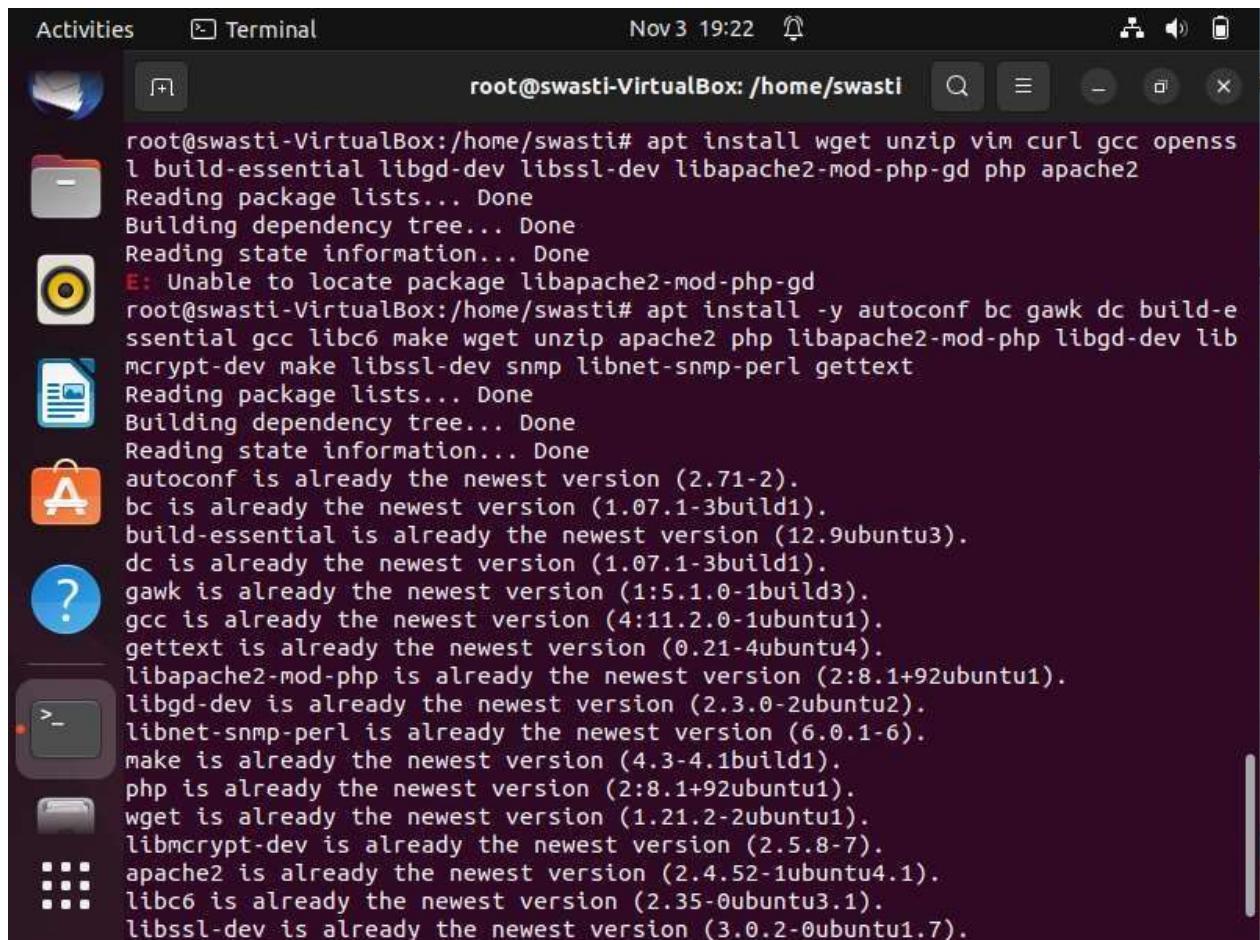
Activities Terminal Oct 28 18:05

```
swasti@swasti-VirtualBox:~$ sudo apt update && sudo apt upgrade -y
[sudo] password for swasti:
Hit:1 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
20 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libflashrom1 libftdi1-2
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  linux-headers-5.15.0-52 linux-headers-5.15.0-52-generic
  linux-image-5.15.0-52-generic linux-modules-5.15.0-52-generic
  linux-modules-extra-5.15.0-52-generic systemd-hwe-hwdb
The following packages have been kept back:
  grub-efi-amd64-bin grub-efi-amd64-signed
The following packages will be upgraded:
  alsu-ucm-conf dbus dbus-user-session gdb libcurl3-gnutls libcurl4
  libdbus-1-3 libudev1 linux-generic-hwe-22.04
  linux-headers-generic-hwe-22.04 linux-image-generic-hwe-22.04 snapd tzdata
  udev xserver-common xserver-xephyr xserver-xorg-core xserver-xorg-legacy
18 upgraded, 6 newly installed, 0 to remove and 2 not upgraded.
```

Activities Terminal Oct 11 20:08

```
lavenaglavena-VirtualBox:~$ sudo apt update
[sudo] password for lavena:
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:2 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:3 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [40.5 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [77.6 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease [308 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 DEP-11 Metadata [2,468 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [277 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [391 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [944 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu focal-backports/main amd64 DEP-11 Metadata [7,996 kB]
Get:12 http://in.archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [36.5 kB]
Fetched 1,164 kB in 6s (188 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
88 packages can be upgraded. Run 'apt list --upgradable' to see them.
lavenaglavena-VirtualBox:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
  linux-headers-5.15.0-58-generic linux-hwe-5.15-headers-5.15.0-58
  linux-image-5.15.0-58-generic linux-modules-5.15.0-58-generic
  linux-modules-extra-5.15.0-58-generic
The following packages will be upgraded:
  bind9-dnsutils bind9-host bind9-libs dejavu-dup firefox fonts-opensymbol
  ghostscript ghostscript-x gir1.2-gdkpixbuf-2.0 gir1.2-javascriptcoregtk-4.0
  gir1.2-notify-0.7 gir1.2-webkit2-4.0 intel-microcode lsc-dhcp-client
```

Step 2: Install all the required packages



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "root@swasti-VirtualBox: /home/swasti". The terminal content shows the following command being run and its output:

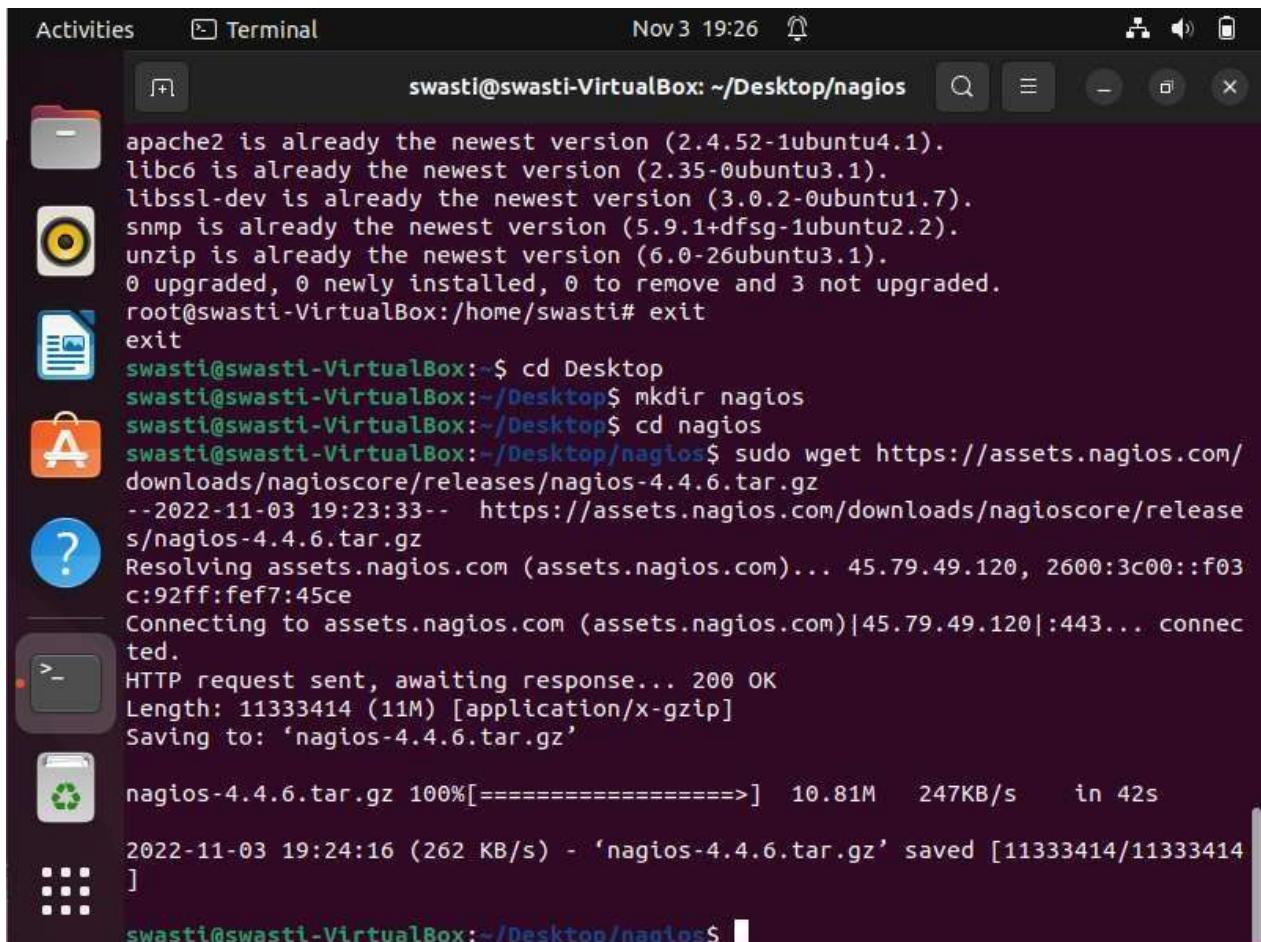
```
root@swasti-VirtualBox:/home/swasti# apt install wget unzip vim curl gcc openssl build-essential libgd-dev libssl-dev libapache2-mod-php7.0 php apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package libapache2-mod-php7.0
root@swasti-VirtualBox:/home/swasti# apt install -y autoconf bc gawk dc build-essential gcc libc6 make wget unzip apache2 php libapache2-mod-php7.0 libgd-dev libmcrypt-dev make libssl-dev snmp libnet-snmp-perl gettext
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
autoconf is already the newest version (2.71-2).
bc is already the newest version (1.07.1-3build1).
build-essential is already the newest version (12.9ubuntu3).
dc is already the newest version (1.07.1-3build1).
gawk is already the newest version (1:5.1.0-1build3).
gcc is already the newest version (4:11.2.0-1ubuntu1).
gettext is already the newest version (0.21-4ubuntu4).
libapache2-mod-php7.0 is already the newest version (2:8.1+92ubuntu1).
libgd-dev is already the newest version (2.3.0-2ubuntu2).
libnet-snmp-perl is already the newest version (6.0.1-6).
make is already the newest version (4.3-4.1build1).
php is already the newest version (2:8.1+92ubuntu1).
wget is already the newest version (1.21.2-2ubuntu1).
libmcrypt-dev is already the newest version (2.5.8-7).
apache2 is already the newest version (2.4.52-1ubuntu4.1).
libc6 is already the newest version (2.35-0ubuntu3.1).
libssl-dev is already the newest version (3.0.2-0ubuntu1.7).
```

Activities Terminal Oct 28 18:19

swasti@swasti-VirtualBox: ~

```
Thunderbird Mail
swasti@swasti-VirtualBox:~$ sudo apt install -y autoconf bc gawk dc build-essential
gcc libc6 make wget unzip apache2 php libapache2-mod-php libgd-dev libmcrypt-dev
libssl-dev snmp libnet-snmp-perl gettext
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
bc is already the newest version (1.07.1-3build1).
bc set to manually installed.
dc is already the newest version (1.07.1-3build1).
dc set to manually installed.
wget is already the newest version (1.21.2-2ubuntu1).
wget set to manually installed.
libc6 is already the newest version (2.35-0ubuntu3.1).
libc6 set to manually installed.
unzip is already the newest version (6.0-26ubuntu3.1).
unzip set to manually installed.
The following packages were automatically installed and are no longer required:
  libflashrom1 libfdt1-2 linux-headers-5.15.0-25
  linux-headers-5.15.0-25-generic linux-image-5.15.0-25-generic
  linux-modules-5.15.0-25-generic linux-modules-extra-5.15.0-25-generic
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils automake autotools-dev binutils
  binutils-common binutils-x86-64-linux-gnu dpkg-dev fakeroot g++ g++-11
  gcc-11 libalgorithm-merge-perl libapache2-mod-php8.1 libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libasan6 libatomic1 libbinutils
  libbrotli-dev libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev
  libctf-nobfd0 libctf0 libdeflate-dev libdpkg-perl libexpat1-dev libfakeroot
```

Step 3: Download Nagios Core Setup files



The screenshot shows a terminal window titled "Terminal" with the command-line interface. The terminal window has a dark background and light-colored text. At the top, it shows the date and time: "Nov 3 19:26". The user's session is listed as "swasti@swasti-VirtualBox: ~/Desktop/nagios". The terminal window contains the following text:

```
apache2 is already the newest version (2.4.52-1ubuntu4.1).
libc6 is already the newest version (2.35-0ubuntu3.1).
libssl-dev is already the newest version (3.0.2-0ubuntu1.7).
snmp is already the newest version (5.9.1+dfsg-1ubuntu2.2).
unzip is already the newest version (6.0-26ubuntu3.1).
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
root@swasti-VirtualBox:/home/swasti# exit
exit
swasti@swasti-VirtualBox:~$ cd Desktop
swasti@swasti-VirtualBox:~/Desktop$ mkdir nagios
swasti@swasti-VirtualBox:~/Desktop$ cd nagios
swasti@swasti-VirtualBox:~/Desktop/nagios$ sudo wget https://assets.nagios.com/
downloads/nagioscore/releases/nagios-4.4.6.tar.gz
--2022-11-03 19:23:33-- https://assets.nagios.com/downloads/nagioscore/releas
es/nagios-4.4.6.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00::f03
c:92ff:fef7:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 11333414 (11M) [application/x-gzip]
Saving to: 'nagios-4.4.6.tar.gz'

nagios-4.4.6.tar.gz 100%[=====] 10.81M 247KB/s in 42s
2022-11-03 19:24:16 (262 KB/s) - 'nagios-4.4.6.tar.gz' saved [11333414/11333414]
```

```
swasti@swasti-VirtualBox:~/Desktop/nagios$ ll
total 11076
drwxrwxr-x  2 swasti swasti    4096 Nov  3 19:23 ./
drwxr-xr-x 10 swasti swasti    4096 Nov  3 19:23 ../
-rw-r--r--  1 root   root   11333414 May  5 2020 nagios-4.4.6.tar.gz
swasti@swasti-VirtualBox:~/Desktop/nagios$
```

Step 4: Extract the downloaded files.

```
swasti@swasti-VirtualBox:~/Desktop/nagios$ sudo tar -xvf nagios-4.4.6.tar.gz
nagios-4.4.6/
nagios-4.4.6/.gitignore
nagios-4.4.6/.travis.yml
```

The screenshot shows a Linux desktop environment with a terminal window and a file manager window.

The terminal window at the top shows the command:

```
swasti@swasti-VirtualBox:~/Desktop/nagios$ sudo tar -xvf nagios-4.4.6.tar.gz
```

The output of the command is:

```
nagios-4.4.6/
nagios-4.4.6/.gitignore
nagios-4.4.6/.travis.yml
```

The file manager window below shows the contents of the extracted directory:

```
nagios-4.4.6/worker/ping/.gitignore
nagios-4.4.6/worker/ping/Makefile.in
nagios-4.4.6/worker/ping/worker-ping.c
nagios-4.4.6/xdata/
nagios-4.4.6/xdata/.gitignore
nagios-4.4.6/xdata/Makefile.in
nagios-4.4.6/xdata/xcddefault.c
nagios-4.4.6/xdata/xcddefault.h
nagios-4.4.6/xdata/xodtemplate.c
nagios-4.4.6/xdata/xodtemplate.h
nagios-4.4.6/xdata/xpddefault.c
nagios-4.4.6/xdata/xpddefault.h
nagios-4.4.6/xdata/xrddefault.c
nagios-4.4.6/xdata/xrddefault.h
nagios-4.4.6/xdata/xsddefault.c
nagios-4.4.6/xdata/xsddefault.h
swasti@swasti-VirtualBox:~/Desktop/nagios$ cd nagios-4.4.6/
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ ll
total 616
drwxrwxr-x 20 root root 4096 Apr 29 2020 .
drwxrwxr-x 3 swasti swasti 4096 Nov 3 19:29 ..
-rw-rw-r-- 1 root root 346 Apr 29 2020 acllocal.m4*
drwxrwxr-x 2 root root 4096 Apr 29 2020 autoconf-macros/
drwxrwxr-x 2 root root 4096 Apr 29 2020 base/
drwxrwxr-x 2 root root 4096 Apr 29 2020 cgi/
-rw-rw-r-- 1 root root 32590 Apr 29 2020 Changelog
drwxrwxr-x 2 root root 4096 Apr 29 2020 common/
-rw-rw-r-- 1 root root 43765 Apr 29 2020 config.guess*
-rw-rw-r-- 1 root root 36345 Apr 29 2020 config.sub*
```

```
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo ./configure --with
-httpd-conf=/etc/apache2/sites-enabled
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking whether make sets $(MAKE)... yes

swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo make all
cd ./base && make
make[1]: Entering directory '/home/swasti/Desktop/nagios/nagios-4.4.6/base'
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
nagios.c: In function 'main':
nagios.c:611:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
  611 |             asprintf(&mac->x[MACRO_PROCESSSTARTTIME], "%llu
", (unsigned long long)program_start);
               ^
nagios.c:841:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
```

```
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo make install-group
-users
make: *** No rule to make target 'install-group-users'. Stop.
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo make install-group
s-users
Group nagios already exists
User nagios already exists
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo usermod -a -G nagi
os www-data
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo make install
cd ./base && make install
make[1]: Entering directory '/home/swasti/Desktop/nagios/nagios-4.4.6/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagiosstats /usr/local/nagios/
bin
make[1]: Leaving directory '/home/swasti/Desktop/nagios/nagios-4.4.6/base'
cd ./cgi && make install
make[1]: Entering directory '/home/swasti/Desktop/nagios/nagios-4.4.6/cgi'
make install-basic
make[2]: Entering directory '/home/swasti/Desktop/nagios/nagios-4.4.6/cgi'
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo make install-init
[sudo] password for swasti:
Sorry, try again.
[sudo] password for swasti:
/usr/bin/install -c -m 755 -d -o root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd
/system/nagios.service
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo make install-comm
ndmode
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$
```

Activities Terminal Nov 3 21:41

```
swasti@swasti-VirtualBox: ~/Desktop/nagios/nagios-4.4.6$ sudo make install-confi
g
/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.cfg /usr/lo
cal/nagios/etc/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /u
sr/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 ***

swasti@swasti-VirtualBox: ~/Desktop/nagios/nagios-4.4.6$ sudo make install-webco
nf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/apache2/sites-enabled/
nagios.conf
if [ 0 -eq 1 ]; then \
    ln -s /etc/apache2/sites-enabled/nagios.conf /etc/apache2/sites-enabled
/nagios.conf; \
fi

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

swasti@swasti-VirtualBox: ~/Desktop/nagios/nagios-4.4.6$
```

```
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo a2enmod rewrite cgi  
Enabling module rewrite.  
Enabling module cgi.  
To activate the new configuration, you need to run:  
    systemctl restart apache2  
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$
```

```
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6$
```

Activities Terminal Nov 3 21:50

```
swasti@swasti-VirtualBox: ~/Desktop/nagios
```

New password:
Re-type new password:
Adding password for user nagiosadmin
swasti@swasti-VirtualBox:~/Desktop/nagios/nagios-4.4.6\$ cd ..
swasti@swasti-VirtualBox:~/Desktop/nagios\$ sudo apt install monitoring-plugins
nagios-nrpe-plugin -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
libdbi1 libmysqlclient21 libpq5 libradcli4 liburiparser1
monitoring-plugins-basic monitoring-plugins-common
monitoring-plugins-standard mysql-common python3-gpg python3-samba
python3-tdb rpcbind samba-common samba-common-bin samba-dsdb-modules
smbclient
Suggested packages:
icinga2 nagios-plugins-contrib fping postfix | sendmail-bin
| exim4-daemon-heavy | exim4-daemon-light qstat heimdal-clients
python3-markdown python3-dnspython cifs-utils
The following NEW packages will be installed:
libdbi1 libmysqlclient21 libpq5 libradcli4 liburiparser1 monitoring-plugins
monitoring-plugins-basic monitoring-plugins-common
monitoring-plugins-standard mysql-common nagios-nrpe-plugin python3-gpg
python3-samba python3-tdb rpcbind samba-common samba-common-bin
samba-dsdb-modules smbclient
0 upgraded, 19 newly installed, 0 to remove and 3 not upgraded.
Need to get 6,872 kB of archives.
After this operation, 39.2 MB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu jessie/main amd64 rpcbind amd64 1.2.6-

```
swasti@swasti-VirtualBox:~/Desktop/nagios$ cd /usr/local/nagios/etc/
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ ll
total 84
drwxrwxr-x 3 nagios nagios 4096 Nov  3 21:48 .
drwxr-xr-x 8 root  root  4096 Nov  3 21:41 ..
-rw-rw-r-- 1 nagios nagios 13710 Nov  3 21:41 cgi.cfg
-rw-r--r-- 1 root   root   50 Nov  3 21:49 htpasswd.users
-rw-rw-r-- 1 nagios nagios 45843 Nov  3 21:41 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Nov  3 21:41 objects/
-rw-rw---- 1 nagios nagios 1312 Nov  3 21:41 resource.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ vim nagios.cfg
Command 'vim' not found, but can be installed with:
sudo apt install vim          # version 2:8.2.3995-1ubuntu2.1, or
sudo apt install vim-tiny      # version 2:8.2.3995-1ubuntu2.1
sudo apt install neovim        # version 0.6.1-3
sudo apt install vim-athena    # version 2:8.2.3995-1ubuntu2
sudo apt install vim-gtk3       # version 2:8.2.3995-1ubuntu2
sudo apt install vim-nox        # version 2:8.2.3995-1ubuntu2
swasti@swasti-VirtualBox:/usr/local/nagios/etc$
```

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ sudo vi nagios.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc$
```

The screenshot shows a Linux desktop environment with a terminal window and a file viewer window.

Terminal Window:

```
Activities Terminal Nov 3 21:59
swasti@swasti-VirtualBox: /usr/local/nagios/etc
```

File Viewer Window:

```
# You can specify individual object config files as shown below:
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg

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

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

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

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

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

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

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ sudo mkdir servers printers switches routers
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ ll
total 100
drwxrwxr-x 7 nagios nagios 4096 Nov  3 22:00 .
drwxr-xr-x 8 root  root  4096 Nov  3 21:41 ..
-rw-rw-r-- 1 nagios nagios 13710 Nov  3 21:41 cgi.cfg
-rw-r--r-- 1 root  root   50 Nov  3 21:49 htpasswd.users
-rw-rw-r-- 1 nagios nagios 45839 Nov  3 21:59 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Nov  3 21:41 objects/
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 printers/
-rw-rw---- 1 nagios nagios 1312 Nov  3 21:41 resource.cfg
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 routers/
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 servers/
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 switches/
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ 
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ sudo vi resource.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ cd objects/
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ ll
total 60
drwxrwxr-x 2 nagios nagios 4096 Nov  3 21:41 .
drwxrwxr-x 7 nagios nagios 4096 Nov  3 22:06 ..
-rw-rw-r-- 1 nagios nagios 6747 Nov  3 21:41 commands.cfg
-rw-rw-r-- 1 nagios nagios 1797 Nov  3 21:41 contacts.cfg
-rw-rw-r-- 1 nagios nagios 4777 Nov  3 21:41 localhost.cfg
-rw-rw-r-- 1 nagios nagios 3001 Nov  3 21:41 printer.cfg
-rw-rw-r-- 1 nagios nagios 3484 Nov  3 21:41 switch.cfg
-rw-rw-r-- 1 nagios nagios 12533 Nov  3 21:41 templates.cfg
-rw-rw-r-- 1 nagios nagios 3512 Nov  3 21:41 timeperiods.cfg
-rw-rw-r-- 1 nagios nagios 4074 Nov  3 21:41 windows.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo contacts.cfg
```

Activities Terminal Nov 3 22:09

```
swasti@swasti-VirtualBox: /usr/local/nagios/etc/objects
#
#      You don't need to keep these definitions in a separate file from your
#      other object definitions.  This has been done just to make things
#      easier to understand.
#
#####
#
# CONTACTS
#
#####
#
# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the
# 'generic-contact' template which is defined elsewhere.
define contact {
    contact_name          nagiosadmin           ; Short name of user
    use                   generic-contact        ; Inherit default values fr
    om generic-contact template (defined above)
    alias                Nagios Admin          ; Full name of user
    email                swastijain716@gmail.com ; <<***** CHANGE THIS TO YO
UR EMAIL ADDRESS *****
```

swasti@swasti-VirtualBox: /usr/local/nagios/etc/objects\$ sudo vi commands.cfg

Activities Terminal Nov 3 22:11

```
swasti@swasti-VirtualBox: /usr/local/nagios/etc/objects
#
#       that you can reference in host, service, and contact definitions.
#
#       You don't need to keep commands in a separate file from your other
#       object definitions. This has been done just to make things easier to
#       understand.
#
#####
#
# SAMPLE NOTIFICATION COMMANDS
#
# These are some example notification commands. They may or may not work on
# your system without modification. As an example, some systems will require
# you to use "/usr/bin/mailx" instead of "/usr/bin/mail" in the commands below.
#
#####
#
define command{
    command_name check_nrpe
    command_line $USER1$/check_nrpe -H $HOSTADDRESS$ -c $ARG1$}
```

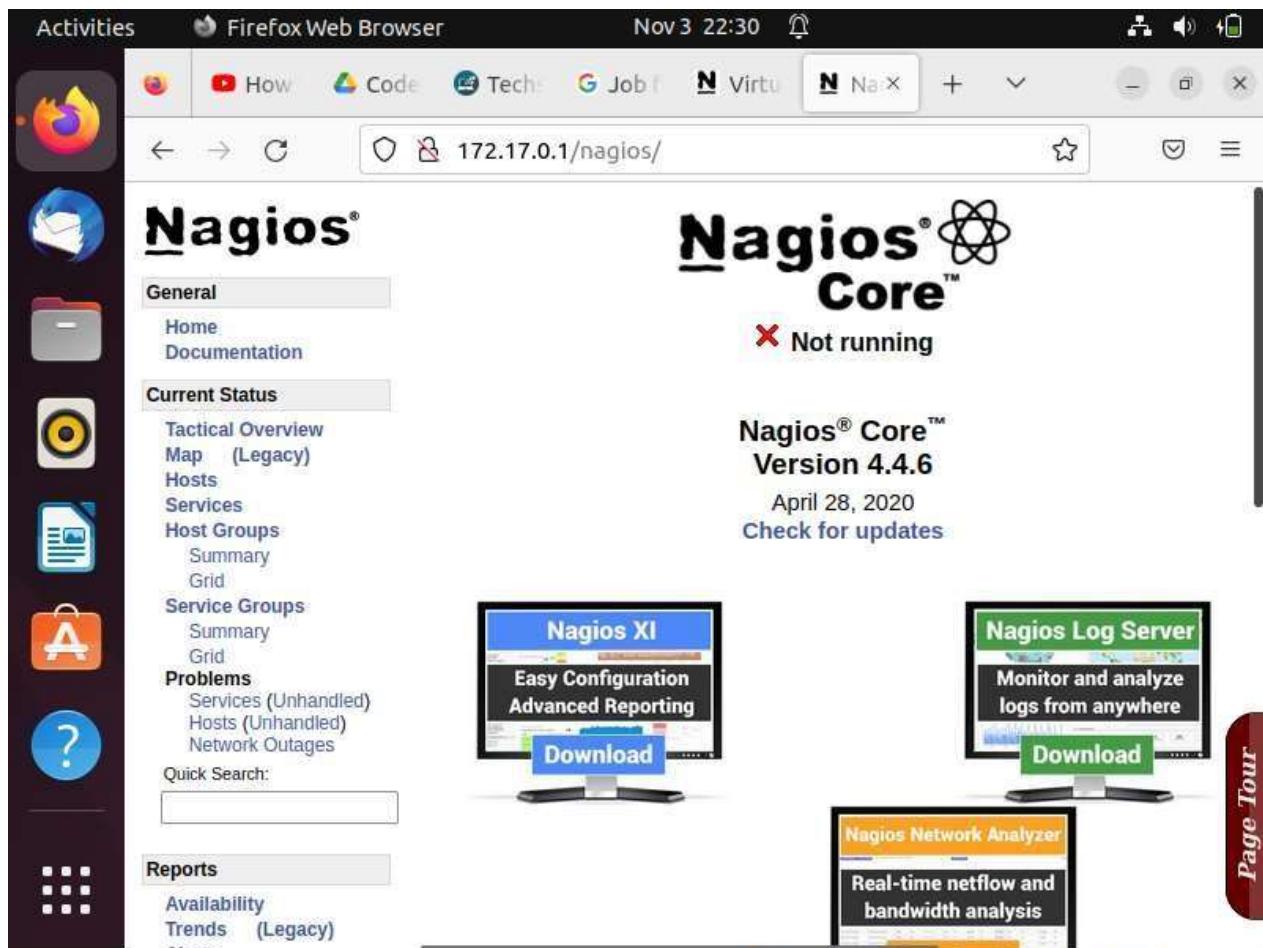
```
swasti@swasti-VirtualBox:~$ sudo ufw status
Status: inactive
status: inactive
swasti@swasti-VirtualBox:~$ sudo systemctl restart apache2
swasti@swasti-VirtualBox:~$ sudo systemctl restart nagios
Job for nagios.service failed because the control process exited with error code.
See "systemctl status nagios.service" and "journalctl -xeu nagios.service" for
details.
swasti@swasti-VirtualBox:~$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
swasti@swasti-VirtualBox:~$ sudo systemctl enable nagios
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /lib/systemd/system/nagios.service.
```

```
swasti@swasti-VirtualBox:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor pres>
   Active: active (running) since Thu 2022-11-03 22:15:57 IST; 6min ago
     Docs: https://httpd.apache.org/docs/2.4/
 Process: 21001 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/>
 Main PID: 21005 (apache2)
    Tasks: 6 (limit: 5793)
   Memory: 10.1M
      CPU: 60ms
     CGroup: /system.slice/apache2.service
             ├─21005 /usr/sbin/apache2 -k start
             ├─21006 /usr/sbin/apache2 -k start
             ├─21007 /usr/sbin/apache2 -k start
             ├─21008 /usr/sbin/apache2 -k start
             ├─21009 /usr/sbin/apache2 -k start
             └─21010 /usr/sbin/apache2 -k start

Nov 03 22:15:57 swasti-VirtualBox systemd[1]: Starting The Apache HTTP Server.>
Nov 03 22:15:57 swasti-VirtualBox apachectl[21004]: AH00558: apache2: Could no>
Nov 03 22:15:57 swasti-VirtualBox systemd[1]: Started The Apache HTTP Server.
...skipping...
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor pres>
   Active: active (running) since Thu 2022-11-03 22:15:57 IST; 6min ago
     Docs: https://httpd.apache.org/docs/2.4/
```

```
swasti@swasti-VirtualBox:~$ sudo systemctl status nagios
* nagios.service - Nagios Core 4.4.6
  Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor prese>
  Active: failed (Result: exit-code) since Thu 2022-11-03 22:16:01 IST; 6mi>
    Docs: https://www.nagios.org/documentation
 Process: 21015 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios>
 Process: 21016 ExecStopPost=/usr/bin/rm -f /usr/local/nagios/var/rw/nagios>
   CPU: 5ms

Nov 03 22:16:01 swasti-VirtualBox nagios[21015]: ***> One or more problems was>
Nov 03 22:16:01 swasti-VirtualBox nagios[21015]:     Check your configuration>
Nov 03 22:16:01 swasti-VirtualBox nagios[21015]:     directives and data defi>
Nov 03 22:16:01 swasti-VirtualBox nagios[21015]:     version of Nagios, you s>
Nov 03 22:16:01 swasti-VirtualBox nagios[21015]:     may have been removed or>
Nov 03 22:16:01 swasti-VirtualBox nagios[21015]:     the HTML documentation r>
Nov 03 22:16:01 swasti-VirtualBox nagios[21015]:     'Whats New' section to f>
Nov 03 22:16:01 swasti-VirtualBox systemd[1]: nagios.service: Control process >
Nov 03 22:16:01 swasti-VirtualBox systemd[1]: nagios.service: Failed with resu>
Nov 03 22:16:01 swasti-VirtualBox systemd[1]: Failed to start Nagios Core 4.4.>
lines 1-18/18 (END)
```



Successfully Installed and Configured Nagios Plugins

Reference: <https://www.vultr.com/de/docs/install-nagios-on-ubuntu-20->

04/?lang=de&utm_source=performancemax-
apac&utm_medium=paidmedia&obility_id=16876059738&utm_adgroup=&utm_campaign=&ut
m_term
=&utm_content=&gclid=Cj0KCQjwyOuYBhCGARIsAIdGQRO6LONSLKjtHIQs1smDa4yHgeTmA1me
VkPAToKqKrekrNzvsNbFiAaAjUMEALw_wcB

Experiment no. 10

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

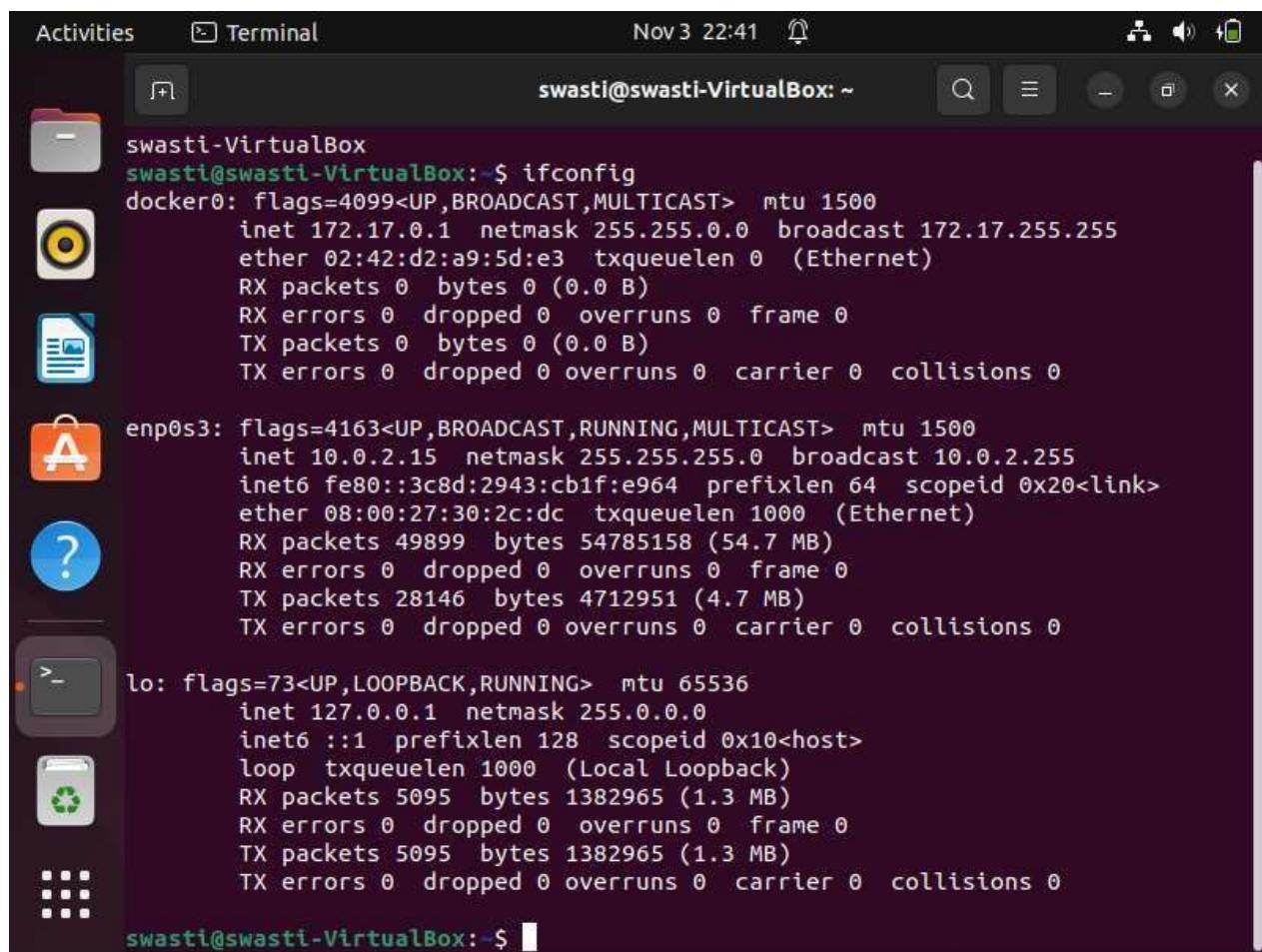
Prerequisite: Install Nagios Server

Used server:

Nagios Server: 172.12.0.1

Nagios Client: 172.12.0.1

Step 1 : Check Hostname and IP of Linux Host



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "swasti@swasti-VirtualBox: ~". The terminal displays the output of the "ifconfig" command, listing network interfaces and their configurations. The interfaces shown are docker0, enp0s3, and lo.

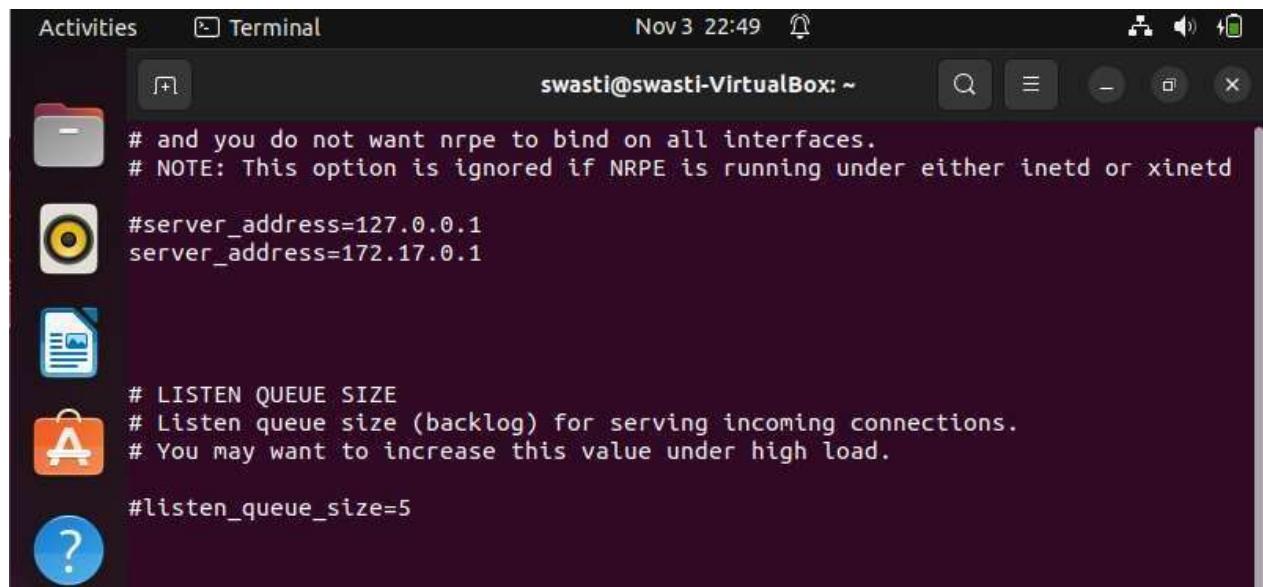
```
swasti@swasti-VirtualBox:~$ ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
                ether 02:42:d2:a9:5d:e3 txqueuelen 0 (Ethernet)
                RX packets 0 bytes 0 (0.0 B)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 0 bytes 0 (0.0 B)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fe80::3c8d:2943:cb1f:e964 prefixlen 64 scopeid 0x20<link>
                ether 08:00:27:30:2c:dc txqueuelen 1000 (Ethernet)
                RX packets 49899 bytes 54785158 (54.7 MB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 28146 bytes 4712951 (4.7 MB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
                loop txqueuelen 1000 (Local Loopback)
                RX packets 5095 bytes 1382965 (1.3 MB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 5095 bytes 1382965 (1.3 MB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

swasti@swasti-VirtualBox:~$
```

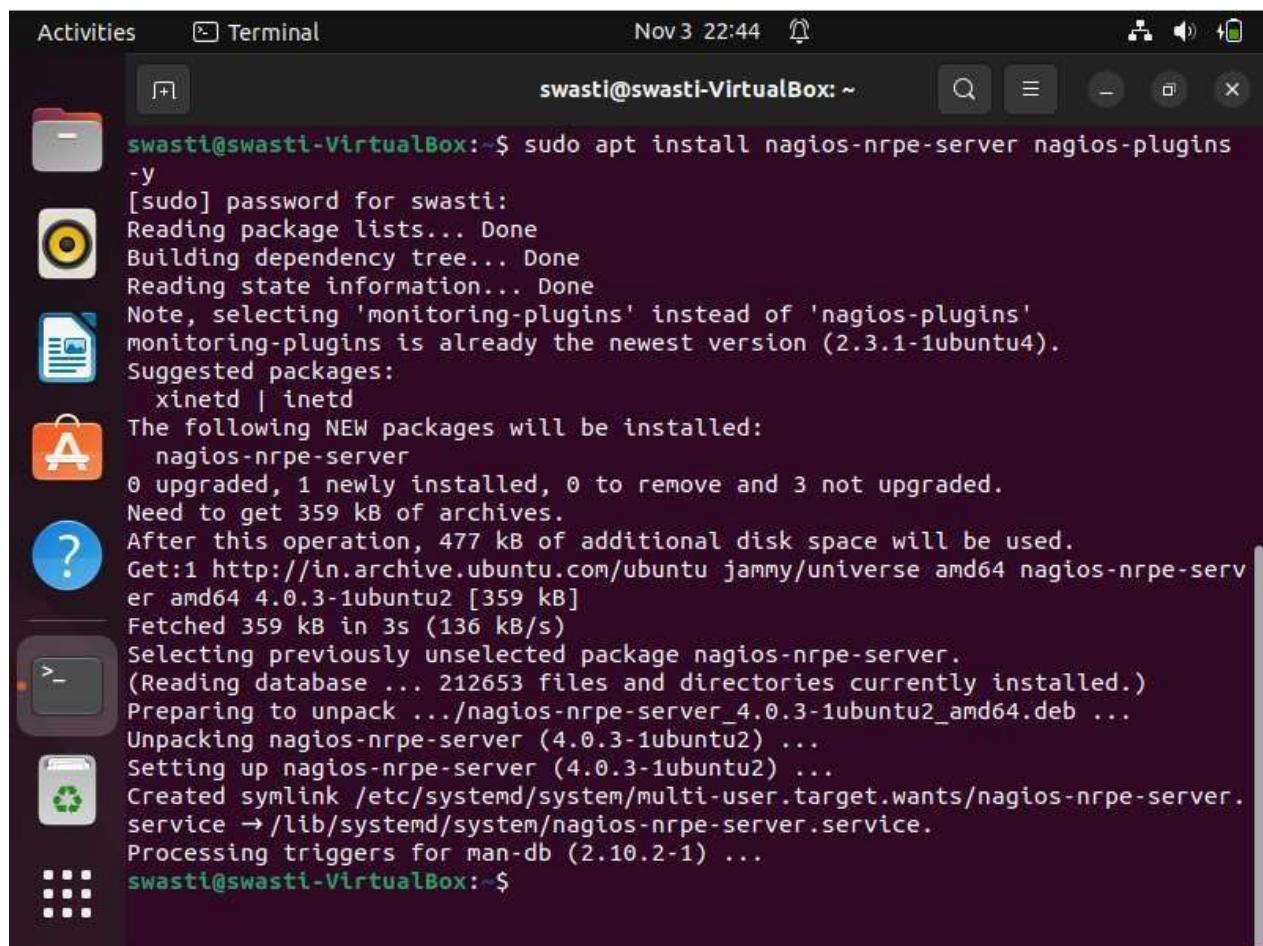
Step



The screenshot shows a terminal window titled "Terminal" with the command-line interface "swasti@swasti-VirtualBox: ~". The window displays configuration files for NRPE and Nagios. The configuration includes setting the server address to 127.0.0.1 and 172.17.0.1, and specifying a listen queue size of 5. The terminal also contains comments about NRPE's behavior under different runlevels.

```
# and you do not want nrpe to bind on all interfaces.  
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd  
  
#server_address=127.0.0.1  
server_address=172.17.0.1  
  
# LISTEN QUEUE SIZE  
# Listen queue size (backlog) for serving incoming connections.  
# You may want to increase this value under high load.  
  
#listen_queue_size=5
```

Step 2: Install NRPE and Nagios Plugins



The screenshot shows a terminal window on an Ubuntu desktop. The terminal title is "swasti@swasti-VirtualBox: ~". The command run is "sudo apt install nagios-nrpe-server nagios-plugins -y". The output shows the package manager selecting 'monitoring-plugins' instead of 'nagios-plugins' because it is already the newest version (2.3.1-1ubuntu4). It lists suggested packages like xinetd and inetd, shows the number of packages to be installed (1), and the amount of disk space required (359 kB). It then proceeds to download the package from http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 nagios-nrpe-server amd64 4.0.3-1ubuntu2 [359 kB]. After unpacking, it creates a symlink /etc/systemd/system/multi-user.target.wants/nagios-nrpe-server.service → /lib/systemd/system/nagios-nrpe-server.service. Finally, it processes triggers for man-db.

```
swasti@swasti-VirtualBox:~$ sudo apt install nagios-nrpe-server nagios-plugins -y
[sudo] password for swasti:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'monitoring-plugins' instead of 'nagios-plugins'
monitoring-plugins is already the newest version (2.3.1-1ubuntu4).
Suggested packages:
  xinetd | inetd
The following NEW packages will be installed:
  nagios-nrpe-server
0 upgraded, 1 newly installed, 0 to remove and 3 not upgraded.
Need to get 359 kB of archives.
After this operation, 477 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 nagios-nrpe-server amd64 4.0.3-1ubuntu2 [359 kB]
Fetched 359 kB in 3s (136 kB/s)
Selecting previously unselected package nagios-nrpe-server.
(Reading database ... 212653 files and directories currently installed.)
Preparing to unpack .../nagios-nrpe-server_4.0.3-1ubuntu2_amd64.deb ...
Unpacking nagios-nrpe-server (4.0.3-1ubuntu2) ...
Setting up nagios-nrpe-server (4.0.3-1ubuntu2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/nagios-nrpe-server.service → /lib/systemd/system/nagios-nrpe-server.service.
Processing triggers for man-db (2.10.2-1) ...
swasti@swasti-VirtualBox:~$
```

Step 3: Edit /etc/nagios/nrpe.cfg file to configure the NRPE agent using command

```
$ sudo vim /etc/nagios/nrpe.cfg
```

Activities Terminal Nov 3 22:46

```
swasti@swasti-VirtualBox: ~
#####
#
# Sample NRPE Config File
#
# Notes:
#
# This is a sample configuration file for the NRPE daemon. It needs to be
# located on the remote host that is running the NRPE daemon, not the host
# from which the check_nrpe client is being executed.
#
#####
#
# LOG FACILITY
# The syslog facility that should be used for logging purposes.
log_facility=daemon

#
# LOG FILE
# If a log file is specified in this option, nrpe will write to
# that file instead of using syslog.
#log_file=/var/log/nrpe.log

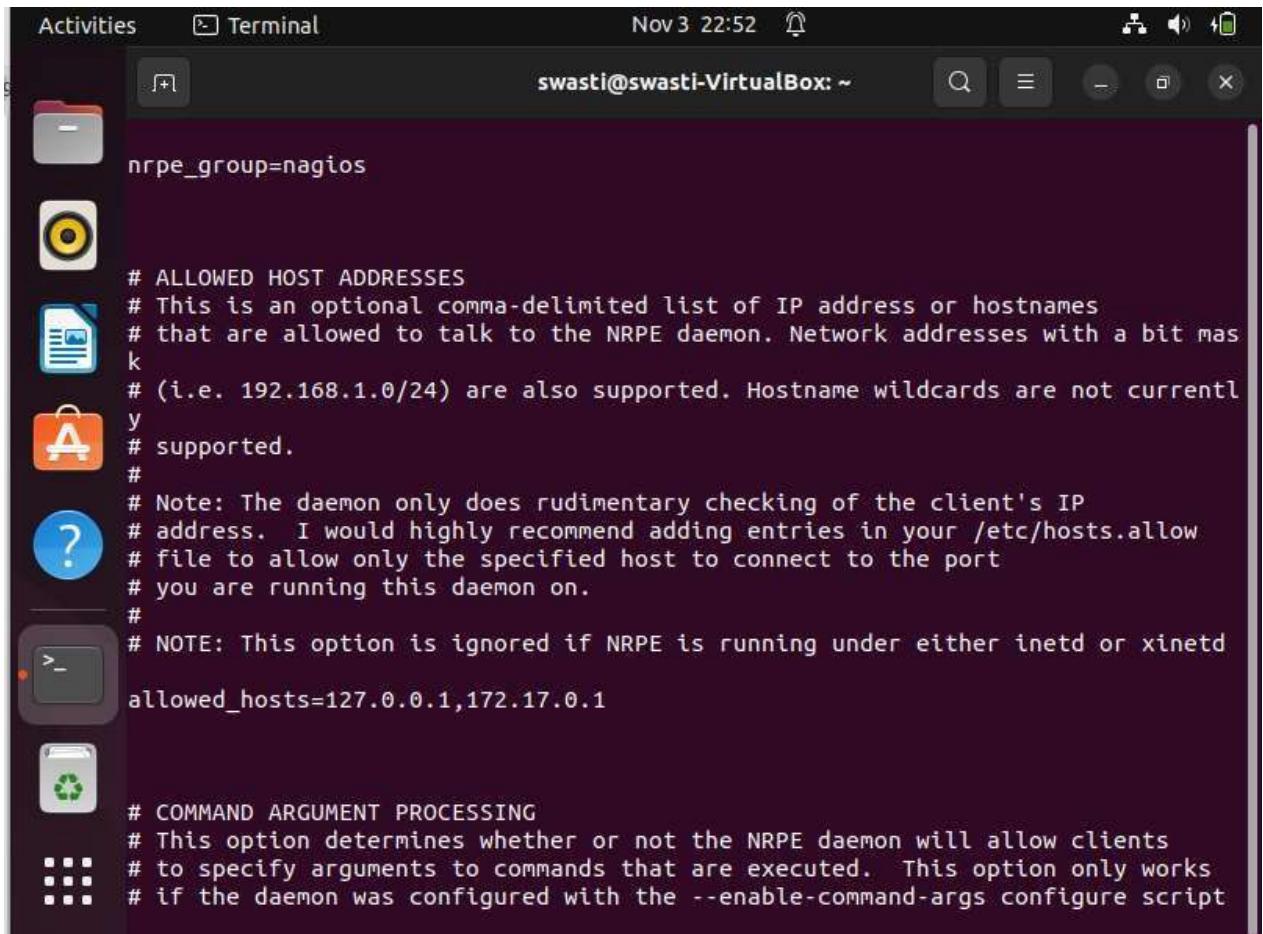
#####
"/etc/nagios/nrpe.cfg" 382 lines, 13016 bytes
```

4: Find and edit line 62 – add server address
(Press I to insert)

Step 5: Find and edit line 106: add allowed host

Step

6: Press Esc to exit Insert mode and type :x and press enter to save and exit the file



```
nrpe_group=nagios

# ALLOWED HOST ADDRESSES
# This is an optional comma-delimited list of IP address or hostnames
# that are allowed to talk to the NRPE daemon. Network addresses with a bit mas
k
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not currentl
y
# supported.
#
# Note: The daemon only does rudimentary checking of the client's IP
# address. I would highly recommend adding entries in your /etc/hosts.allow
# file to allow only the specified host to connect to the port
# you are running this daemon on.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd
allowed_hosts=127.0.0.1,172.17.0.1

# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE daemon will allow clients
# to specify arguments to commands that are executed. This option only works
# if the daemon was configured with the --enable-command-args configure script
```

Step 7: Restart and enable NRPE services to make the changes active

```
swasti@swasti-VirtualBox:~$ sudo systemctl restart nagios-nrpe-server
swasti@swasti-VirtualBox:~$ sudo systemctl enable nagios-nrpe-server
Synchronizing state of nagios-nrpe-server.service with SysV service script with
/lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable nagios-nrpe-server
swasti@swasti-VirtualBox:~$
```

8: Check UFW Firewall status and enable UFW

```
swasti@swasti-VirtualBox:~$ sudo ufw status  
Status: inactive  
swasti@swasti-VirtualBox:~$
```

Step 9: By default, Nagios NRPE agent listens on port 5666. You need to open 5666 on your Firewall

```
swasti@swasti-VirtualBox:~$ sudo ufw allow 5666/tcp  
Rules updated  
Rules updated (v6)  
swasti@swasti-VirtualBox:~$
```

10: Verify that the port has been allowed. Also don't forget to allow port ssh Firewall to remote access.

```
swasti@swasti-VirtualBox:~$ sudo ufw status  
Status: inactive  
swasti@swasti-VirtualBox:~$ sudo ufw allow 5666/tcp  
Rules updated  
Rules updated (v6)  
swasti@swasti-VirtualBox:~$ sudo ufw status  
Status: inactive  
swasti@swasti-VirtualBox:~$ sudo ufw allow 5666/tcp  
Skipping adding existing rule  
Skipping adding existing rule (v6)  
swasti@swasti-VirtualBox:~$ sudo ufw status  
Status: inactive  
swasti@swasti-VirtualBox:~$ sudo ufw allow ssh  
Rules updated  
Rules updated (v6)  
swasti@swasti-VirtualBox:~$
```

We have successfully installed and configured NRPE agent on Remote Linux Host.

Step 11: Go to /usr/local/nagios/etc/ and use command ll
Step

```
swasti@swasti-VirtualBox:~$ cd /usr/local/nagios/etc
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ ll
total 100
drwxrwxr-x 7 nagios nagios 4096 Nov  3 22:06 .
drwxr-xr-x 8 root  root  4096 Nov  3 21:41 ..
-rw-rw-r-- 1 nagios nagios 13710 Nov  3 21:41 cgi.cfg
-rw-r--r-- 1 root  root   50 Nov  3 21:49 htpasswd.users
-rw-rw-r-- 1 nagios nagios 45839 Nov  3 21:59 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Nov  3 22:12 objects/
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 printers/
-rw-rw---- 1 nagios nagios 1313 Nov  3 22:06 resource.cfg
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 routers/
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 servers/
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 switches/
swasti@swasti-VirtualBox:/usr/local/nagios/etc$
```

12: If there is no ‘servers’ folder, then create one to store all server host configuration. Using command

```
$ sudo vim nagios.cfg
```

Activities Terminal Nov 3 22:58

```
swasti@swasti-VirtualBox: /usr/local/nagios/etc
```

```
#####
#
# NAGIOS.CFG - Sample Main Config File for Nagios 4.4.6
#
# Read the documentation for more information on this configuration
# file. I've provided some comments here, but things may not be so
# clear without further explanation.
#
#####
#
# LOG FILE
# This is the main log file where service and host events are logged
# for historical purposes. This should be the first option specified
# in the config file!!!
#
log_file=/usr/local/nagios/var/nagios.log

>-
#
# OBJECT CONFIGURATION FILE(S)
# These are the object configuration files in which you define hosts,
# host groups, contacts, contact groups, services, etc.
# You can split your object definitions across several config files
# if you wish (as shown below), or keep them all in a single config file.
#
# You can specify individual object config files as shown below:
"nagios.cfg" 1378 lines, 45839 bytes
```

Step 13: add the following lines

Activities Terminal Nov 3 23:23

```
swasti@swasti-VirtualBox: /usr/local/nagios/etc
```

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

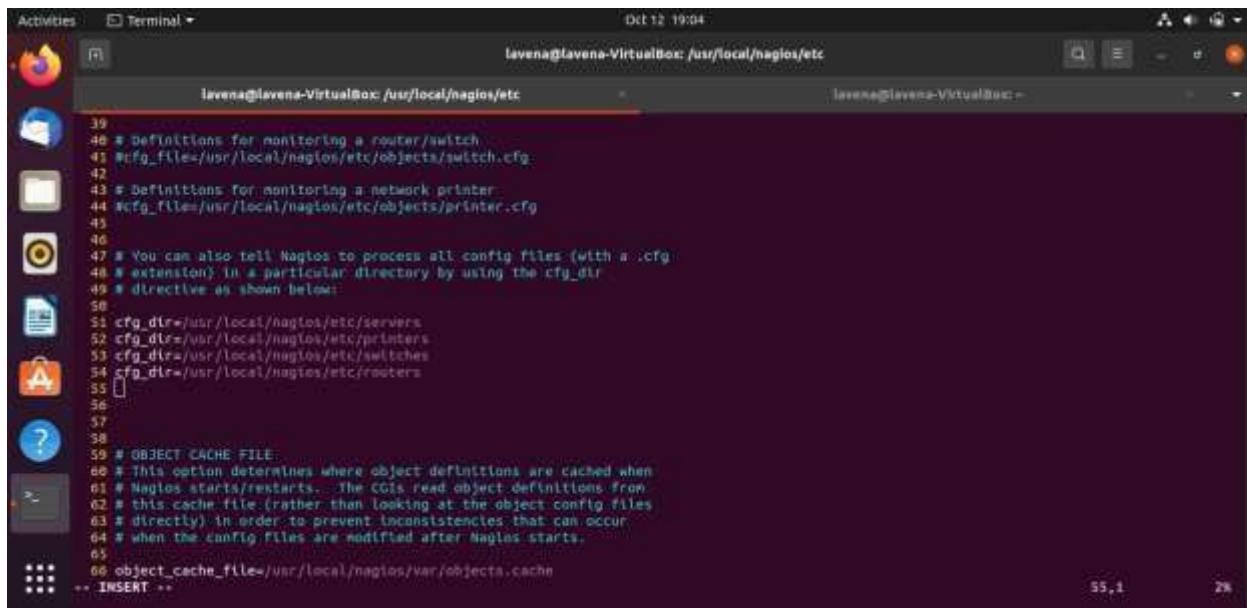
# Definitions for monitoring a router/switch
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg

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

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

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

# OBJECT CACHE FILE
# This option determines where object definitions are cached when
# Nagios starts/restarts. The CGIs read object definitions from
```



A screenshot of a Linux desktop environment (Ubuntu) showing a terminal window. The terminal title is "Terminal" and the command entered is "vi nagios.cfg". The file content shows configuration directives for Nagios, including sections for switches, printers, and routers, along with an object cache file definition.

```
39 # Definitions for monitoring a router/switch
40 #cfg_file=/usr/local/nagios/etc/objects/switch.cfg
41
42 # Definitions for monitoring a network printer
43 #cfg_file=/usr/local/nagios/etc/objects/printer.cfg
44
45
46 # You can also tell Nagios to process all config files (with a .cfg
47 # extension) in a particular directory by using the cfg_dir
48 # directive as shown below:
49
50 cfg_dir=/usr/local/nagios/etc/servers
51 cfg_dir=/usr/local/nagios/etc/printers
52 cfg_dir=/usr/local/nagios/etc/switches
53 cfg_dir=/usr/local/nagios/etc/routers
54
55
56
57
58
59 # OBJECT CACHE FILE
60 # This option determines where object definitions are cached when
61 # Nagios starts/restarts. The coils read object definitions from
62 # this cache file (rather than looking at the objects config files
63 # directly) in order to prevent inconsistencies that can occur
64 # when the config files are modified after Nagios starts.
65
66 object_cache_file=/var/local/nagios/var/objects.cache
-- INSERT --
```

14: Create directory of server

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ sudo vi nagios.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ ll
total 100
drwxrwxr-x 7 nagios nagios 4096 Nov  3 23:23 /
drwxr-xr-x 8 root   root   4096 Nov  3 21:41 .
-rw-rw-r-- 1 nagios nagios 13710 Nov  3 21:41 cgi.cfg
-rw-r--r-- 1 root   root   50 Nov  3 21:49 htpasswd.users
-rw-rw-r-- 1 nagios nagios 45839 Nov  3 21:59 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Nov  3 22:12 objects/
drwxr-xr-x 2 root   root   4096 Nov  3 22:00 printers/
-rw-rw---- 1 nagios nagios 1313 Nov  3 22:06 resource.cfg
drwxr-xr-x 2 root   root   4096 Nov  3 22:00 routers/
drwxrwxr-x 2 nagios nagios 4096 Nov  3 23:17 servers/
drwxr-xr-x 2 root   root   4096 Nov  3 22:00 switches/
```

Activities Terminal Oct 12 19:07

```
lavena@lavena-VirtualBox: /usr/local/nagios/etc
lavena@lavena-VirtualBox: $ sudo ufw allow ssh
Rule added
Rule added (v6)
lavena@lavena-VirtualBox: $ cd /usr/local/nagios/etc
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ ll
total 84
drwxrwxr-x 3 nagios nagios 4096 Oct 11 20:31 .
drwxr-xr-x 9 root root 4096 Oct 11 20:37 ../
-rw-rw-r-- 1 nagios nagios 13710 Oct 11 20:38 cgi.cfg
-rw-rw-r-- 1 root root 44 Oct 11 20:31 httppasswd.users
-rw-rw-r-- 1 nagios nagios 45843 Oct 11 20:30 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Oct 11 20:38 objects/
-rw-rw-r-- 1 nagios nagios 1312 Oct 11 20:36 resource.cfg
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ sudo vim nagios.cfg
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ sudo mkdir servers printers switches routers
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ ll
total 108
drwxrwxr-x 7 nagios nagios 4096 Oct 12 19:07 .
drwxr-xr-x 9 root root 4096 Oct 11 20:37 ../
-rw-rw-r-- 1 nagios nagios 13710 Oct 11 20:30 cgi.cfg
-rw-r--r-- 1 root root 44 Oct 11 20:31 httppasswd.users
-rw-rw-r-- 1 nagios nagios 45839 Oct 12 19:05 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Oct 11 20:38 objects/
drwxr-xr-x 2 root root 4096 Oct 12 19:07 printers/
-rw-rw-r-- 1 nagios nagios 1312 Oct 11 20:36 resource.cfg
drwxr-xr-x 2 root root 4096 Oct 12 19:07 routers/
drwxr-xr-x 2 root root 4096 Oct 12 19:07 servers/
drwxr-xr-x 2 root root 4096 Oct 12 19:07 switches/
lavena@lavena-VirtualBox:/usr/local/nagios/etc$
```

Activities Terminal Oct 12 19:08

```
lavena@lavena-VirtualBox: /usr/local/nagios/etc
lavena@lavena-VirtualBox: $ sudo ufw allow ssh
Rule added
Rule added (v6)
lavena@lavena-VirtualBox: $ cd /usr/local/nagios/etc
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ ll
total 84
drwxrwxr-x 3 nagios nagios 4096 Oct 11 20:31 .
drwxr-xr-x 9 root root 4096 Oct 11 20:37 ../
-rw-rw-r-- 1 nagios nagios 13710 Oct 11 20:38 cgi.cfg
-rw-r--r-- 1 root root 44 Oct 11 20:31 httppasswd.users
-rw-rw-r-- 1 nagios nagios 45843 Oct 11 20:30 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Oct 11 20:38 objects/
-rw-rw-r-- 1 nagios nagios 1312 Oct 11 20:36 resource.cfg
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ sudo vim nagios.cfg
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ sudo mkdir servers printers switches routers
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ ll
total 108
drwxrwxr-x 7 nagios nagios 4096 Oct 12 19:07 .
drwxr-xr-x 9 root root 4096 Oct 11 20:37 ../
-rw-rw-r-- 1 nagios nagios 13710 Oct 11 20:30 cgi.cfg
-rw-r--r-- 1 root root 44 Oct 11 20:31 httppasswd.users
-rw-rw-r-- 1 nagios nagios 45839 Oct 12 19:05 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Oct 11 20:38 objects/
drwxr-xr-x 2 root root 4096 Oct 12 19:07 printers/
-rw-rw-r-- 1 nagios nagios 1312 Oct 11 20:36 resource.cfg
drwxr-xr-x 2 root root 4096 Oct 12 19:07 routers/
drwxr-xr-x 2 root root 4096 Oct 12 19:07 servers/
drwxr-xr-x 2 root root 4096 Oct 12 19:07 switches/
lavena@lavena-VirtualBox:/usr/local/nagios/etc$ sudo vim resource.cfg
```

Activities Terminal Nov 3 23:26

```
swasti@swasti-VirtualBox: /usr/local/nagios/etc
swasti@swasti-VirtualBox: /usr/local/n... x swasti@swasti-VirtualBox: ~ x v

# $USERx$ macros, instead of modifying a lot of command definitions.
#
# The CGIs will not attempt to read the contents of resource files, so
# you can set restrictive permissions (600 or 660) on them.
#
# Nagios supports up to 256 $USERx$ macros ($USER1$ through $USER256$)
#
# Resource files may also be used to store configuration directives for
# external data sources like MySQL...
#
#####
# Sets $USER1$ to be the path to the plugins
#
$USER1$=/usr/lib/nagios/plugins
# Sets $USER2$ to be the path to event handlers
#$USER2$=/usr/local/nagios/libexec/eventhandlers
#
# Store some usernames and passwords (hidden from the CGIs)
#$USER3$=someuser
#$USER4$=somepassword
```

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ sudo vi resource.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ ll
total 100
drwxrwxr-x 7 nagios nagios 4096 Nov  3 23:26 .
drwxr-xr-x 8 root   root   4096 Nov  3 21:41 ..
-rw-rw-r-- 1 nagios nagios 13710 Nov  3 21:41 cgi.cfg
-rw-r--r-- 1 root   root   50 Nov  3 21:49 htpasswd.users
-rw-rw-r-- 1 nagios nagios 45839 Nov  3 21:59 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Nov  3 22:12 objects/
drwxr-xr-x 2 root   root   4096 Nov  3 22:00 printers/
-rw-rw---- 1 nagios nagios 1313 Nov  3 22:06 resource.cfg
drwxr-xr-x 2 root   root   4096 Nov  3 22:00 routers/
drwxrwxr-x 2 nagios nagios 4096 Nov  3 23:17 servers/
drwxr-xr-x 2 root   root   4096 Nov  3 22:00 switches/
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ cd objects/
```

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ cd objects/
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ ll
total 60
drwxrwxr-x 2 nagios nagios 4096 Nov  3 22:12 .
drwxrwxr-x 7 nagios nagios 4096 Nov  3 23:26 ..
-rw-rw-r-- 1 nagios nagios 6848 Nov  3 22:12 commands.cfg
-rw-rw-r-- 1 nagios nagios 1805 Nov  3 22:09 contacts.cfg
-rw-rw-r-- 1 nagios nagios 4777 Nov  3 21:41 localhost.cfg
-rw-rw-r-- 1 nagios nagios 3001 Nov  3 21:41 printer.cfg
-rw-rw-r-- 1 nagios nagios 3484 Nov  3 21:41 switch.cfg
-rw-rw-r-- 1 nagios nagios 12533 Nov  3 21:41 templates.cfg
-rw-rw-r-- 1 nagios nagios 3512 Nov  3 21:41 timeperiods.cfg
-rw-rw-r-- 1 nagios nagios 4074 Nov  3 21:41 windows.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo vi contacts.cfg
```

Activities Terminal Nov 3 23:28

```
swasti@swasti-VirtualBox: /usr/local/nagios/etc/objects
swasti@swasti-VirtualBox: /usr/local/n... ~
#####
#
# CONTACTS
#
#####
# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the
# 'generic-contact' template which is defined elsewhere.

define contact {
    contact_name          nagiosadmin           ; Short name of user
    use                   generic-contact        ; Inherit default values fr
    om generic-contact template (defined above)
    alias                Nagios Admin          ; Full name of user

    email                swastijain716@gmail.com ; <<***** CHANGE THIS TO YO
    UR EMAIL ADDRESS *****
}
```

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo ufw status
Status: inactive
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo ufw allow apache
Skipping adding existing rule
Skipping adding existing rule (v6)
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo ufw status
Status: inactive
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo ufw reload
Firewall not enabled (skipping reload)
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$
```

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo systemctl restart apache2
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo systemctl restart nagios
Job for nagios.service failed because the control process exited with error code.
See "systemctl status nagios.service" and "journalctl -xeu nagios.service" for details.
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo systemctl enable nagios
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$
```

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor pres>
   Active: active (running) since Thu 2022-11-03 23:30:29 IST; 1min 40s ago
     Docs: https://httpd.apache.org/docs/2.4/
 Main PID: 22456 (apache2)
    Tasks: 6 (limit: 5793)
   Memory: 10.4M
      CPU: 44ms
     CGroup: /system.slice/apache2.service
             └─22456 /usr/sbin/apache2 -k start
                  ├─22457 /usr/sbin/apache2 -k start
                  ├─22458 /usr/sbin/apache2 -k start
                  ├─22459 /usr/sbin/apache2 -k start
                  ├─22460 /usr/sbin/apache2 -k start
                  └─22461 /usr/sbin/apache2 -k start

Nov 03 23:30:29 swasti-VirtualBox systemd[1]: Starting The Apache HTTP Server.>
Nov 03 23:30:29 swasti-VirtualBox apachectl[22455]: AH00558: apache2: Could no>
Nov 03 23:30:29 swasti-VirtualBox systemd[1]: Started The Apache HTTP Server.

swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$
```

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc/objects$ sudo systemctl status nagios
* nagios.service - Nagios Core 4.4.6
  Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: disabled)
  Active: failed (Result: exit-code) since Thu 2022-11-03 23:30:36 IST; 59s ago
    Docs: https://www.nagios.org/documentation
           CPU: 4ms

Nov 03 23:30:36 swasti-VirtualBox nagios[22466]: ***> One or more problems was>
Nov 03 23:30:36 swasti-VirtualBox nagios[22466]: Check your configuration>
Nov 03 23:30:36 swasti-VirtualBox nagios[22466]: directives and data defini>
Nov 03 23:30:36 swasti-VirtualBox nagios[22466]: version of Nagios, you s>
Nov 03 23:30:36 swasti-VirtualBox nagios[22466]: may have been removed or>
Nov 03 23:30:36 swasti-VirtualBox nagios[22466]: the HTML documentation r>
Nov 03 23:30:36 swasti-VirtualBox nagios[22466]: 'Whats New' section to f>
Nov 03 23:30:36 swasti-VirtualBox systemd[1]: nagios.service: Control process >
Nov 03 23:30:36 swasti-VirtualBox systemd[1]: nagios.service: Failed with resu>
Nov 03 23:30:36 swasti-VirtualBox systemd[1]: Failed to start Nagios Core 4.4.>
```

Step 15: Change permission on servers folder

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ sudo chmod 775 servers/
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ sudo chown nagios:nagios servers/
swasti@swasti-VirtualBox:/usr/local/nagios/etc$ ll
total 100
drwxrwxr-x 7 nagios nagios 4096 Nov  3 23:00 .
drwxr-xr-x 8 root  root  4096 Nov  3 21:41 ..
-rw-rw-r-- 1 nagios nagios 13710 Nov  3 21:41 cgi.cfg
-rw-r--r-- 1 root  root   50 Nov  3 21:49 htpasswd.users
-rw-rw-r-- 1 nagios nagios 45839 Nov  3 21:59 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Nov  3 22:12 objects/
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 printers/
-rw-rw---- 1 nagios nagios 1313 Nov  3 22:06 resource.cfg
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 routers/
drwxrwxr-x 2 nagios nagios 4096 Nov  3 22:00 servers/
drwxr-xr-x 2 root  root  4096 Nov  3 22:00 switches/
```

cd

Step 16: Go to servers directory and create a new configuration called “HG211.cfg” for define Remote Linux Host. Open and add the following content:-

```
#Replace :
# host_name = Your-Hostname
# alias = Your-Alias
# address = Your-IP address of host
define
host{
```

```
use          linux-server
host_name      swasti-
VirtualBox alias
SlaveDNS address
172.12.0.1 }
define service{
use          local-service host_name
lavena-VirtualBox service_description      Root
/ Partition check_command
check_nrpe!check_disk
}
define service{
use          local-service host_name
lavena-VirtualBox service_description      /mnt
Partition check_command
check_nrpe!check_mnt_disk
}
define service{
use          local-service host_name
lavena-VirtualBox service_description
Current Users check_command
check_nrpe!check_users
}
define service{
use          local-service host_name
lavena-VirtualBox service_description      Total
Processes check_command
check_nrpe!check_total_procs
}
define service{
use          local-service host_name
lavena-VirtualBox service_description
Current Load check_command
check_nrpe!check_load }
```

Activities Terminal Nov 3 23:38

```
swasti@swasti-VirtualBox: /usr/local/nagios/etc/servers
```

```
#Replace
# host_name = Your-Hostname
# alias = Your-Alias
# address = Your-IP address of host

define host{
use           linux-server
host_name     swasti-VirtualBox
alias         SlaveDNS
address       172.17.0.1
}

define service{
use           local-service
host_name     swasti-VirtualBox
service_description Root / Partition
check_command  check_nrpe!check_disk
}

define service{
use           local-service
host_name     swasti-VirtualBox
service_description /mnt Partition
check_command  check_nrpe!check_mnt_disk
}

:x
```

Activities Terminal Oct 12 20:34

```
lavena@lavena-VirtualBox: /usr/local/nagios/etc/servers
```

```
drwxrwxr-x 2 nagios nagios 4096 Oct 12 20:09 objects/
drwxr-xr-x 2 root   root   4096 Oct 12 19:07 printers/
-rw-rw-r--  1 nagios nagios 1344 Oct 12 19:12 resource.cfg
-rw-r----- 1 root   nagios 12288 Oct 12 19:09 .resource.cfg.swp
drwxr-xr-x 2 root   root   4096 Oct 12 19:07 routers/
drwxr-xr-x 2 root   root   4096 Oct 12 19:07 servers/
drwxr-xr-x 2 root   root   4096 Oct 12 19:07 switches/
lavena@lavena-VirtualBox: /usr/local/nagios/etc$ sudo chmod 775 servers
lavena@lavena-VirtualBox: /usr/local/nagios/etc$ sudo chown nagios:nagios servers
lavena@lavena-VirtualBox: /usr/local/nagios/etc$ sudo chown nagios:nagios servers/
lavena@lavena-VirtualBox: /usr/local/nagios/etc$ ll
total 112
drwxrwxr-x 7 nagios nagios 4096 Oct 12 19:12 .
drwxr-xr-x 9 root   root   4096 Oct 11 20:37 ..
-rw-rw-r-- 1 nagios nagios 13710 Oct 11 20:30 cgt.cfg
-rw-r----- 1 root   root   44 Oct 11 20:30 htpasswd.users
-rw-rw-r-- 1 nagios nagios 45839 Oct 12 19:05 nagios.cfg
drwxrwxr-x 2 nagios nagios 4096 Oct 12 20:09 objects/
drwxr-xr-x 2 root   root   4096 Oct 12 19:07 printers/
drwxrwxr-x 2 nagios nagios 4096 Oct 12 19:07 servers/
drwxr-xr-x 2 root   root   4096 Oct 12 19:07 switches/
lavena@lavena-VirtualBox: /usr/local/nagios/etc$ cd servers/
lavena@lavena-VirtualBox: /usr/local/nagios/etc/servers$ ll
total 8
drwxrwxr-x 2 nagios nagios 4096 Oct 12 19:07 .
drwxrwxr-x 7 nagios nagios 4096 Oct 12 19:12 ..
lavena@lavena-VirtualBox: /usr/local/nagios/etc/servers$ sudo touch HG211.cfg
lavena@lavena-VirtualBox: /usr/local/nagios/etc/servers$ sudo vim HG211.cfg
lavena@lavena-VirtualBox: /usr/local/nagios/etc/servers$
```

Step 17: Change permission on HG211.cfg

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$ sudo vi HG211.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$ ll
total 16
drwxrwxr-x 2 nagios nagios 4096 Nov  3 23:38 .
drwxrwxr-x 7 nagios nagios 4096 Nov  3 23:26 ../
-rw-r--r-- 1 root   root   1349 Nov  3 23:38 HG211.cfg
-rw-rw-r-- 1 nagios nagios 1352 Nov  3 23:17 swasti-VirtualBox.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$ sudo chown nagios:nagios HG211.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$ sudo chmod 664 HG211.cfg
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$
```

Step 18: Verify Nagios configuration file for any errors in Nagios server

Total warnings: 0

Total Errors: 0

Step 19: Restart Nagios server

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$ sudo systemctl restart
nagios
Job for nagios.service failed because the control process exited with error cod
e.
See "systemctl status nagios.service" and "journalctl -xeu nagios.service" for
details.
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$ cd /
swasti@swasti-VirtualBox:$ cd /
swasti@swasti-VirtualBox:$ sudo systemctl restart nagios
Job for nagios.service failed because the control process exited with error cod
e.
See "systemctl status nagios.service" and "journalctl -xeu nagios.service" for
details.
swasti@swasti-VirtualBox:$ sudo ufw status
Status: inactive
swasti@swasti-VirtualBox:$ sudo ufw allow 5666/tcp
Skipping adding existing rule
Skipping adding existing rule (v6)
swasti@swasti-VirtualBox:$
```

Step 20: Allow port 5666 on your Firewall

```
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$ sudo systemctl restart nagios
Job for nagios.service failed because the control process exited with error code.
See "systemctl status nagios.service" and "journalctl -xeu nagios.service" for details.
swasti@swasti-VirtualBox:/usr/local/nagios/etc/servers$ cd /
swasti@swasti-VirtualBox:$ sudo systemctl restart nagios
Job for nagios.service failed because the control process exited with error code.
See "systemctl status nagios.service" and "journalctl -xeu nagios.service" for details.
swasti@swasti-VirtualBox:$ sudo ufw status
Status: inactive
swasti@swasti-VirtualBox:$ sudo ufw allow 5666/tcp
Skipping adding existing rule
Skipping adding existing rule (v6)
swasti@swasti-VirtualBox:$ █
```

Step 21: If faced error while viewing web interface, use command

```
$ sudo htpasswd /usr/local/nagios/etc/htpasswd.users nagiosadmin nagiosadmin
```

Use nagiosadmin as your username and password while logging in web interface.

Step 22: Check Nagios web interface, Host under Current Status.

Activities Firefox Web Browser Nov 3 23:45

How to Me Add-Linux- Code-in-th Nagios: 1x

172.17.0.1/nagios/

Nagios®

General

- Home
- Documentation

Current Status

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

Reports

- Availability
- Trends (Legacy)

Start monitoring your infrastructure

- Change the look and feel of Nagios
- Extend Nagios with hundreds of addons
- Get support
- Get training
- Get certified

Nagios Library (tutorials and docs)

- Nagios Labs (development blog)
- Nagios Exchange (plugins and addons)
- Nagios Support (tech support)
- Nagios.com (company)
- Nagios.org (project)

Latest News

- Nagios Update: XI 5.6.6
- Nagios Update: XI 5.6.5
- Nagios Update: XI 5.6.4
- More news...

Don't Miss...

- Monitoring Log Data with Nagios - in one central location.
- Can Nagios monitor netflow? - Yes! variety of flow data. Learn More
- Nagios XI 5 Available Now! - Easier Download Today!

Copyright © 2010-2020 Nagios Core Development Team and Community Contributors. Copyright © 1999-2005 on contributors.

Nagios Core is licensed under the GNU General Public License and is provided AS IS with NO WARRANTY, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. Nagios, Nagios Core and the Nagios logo are registered servicemarks owned by Nagios Enterprises, LLC. Use of the Nagios marks

MONITORED BY
Nagios®
NAGIOS CORE

SOURCEFOR

Page Tour

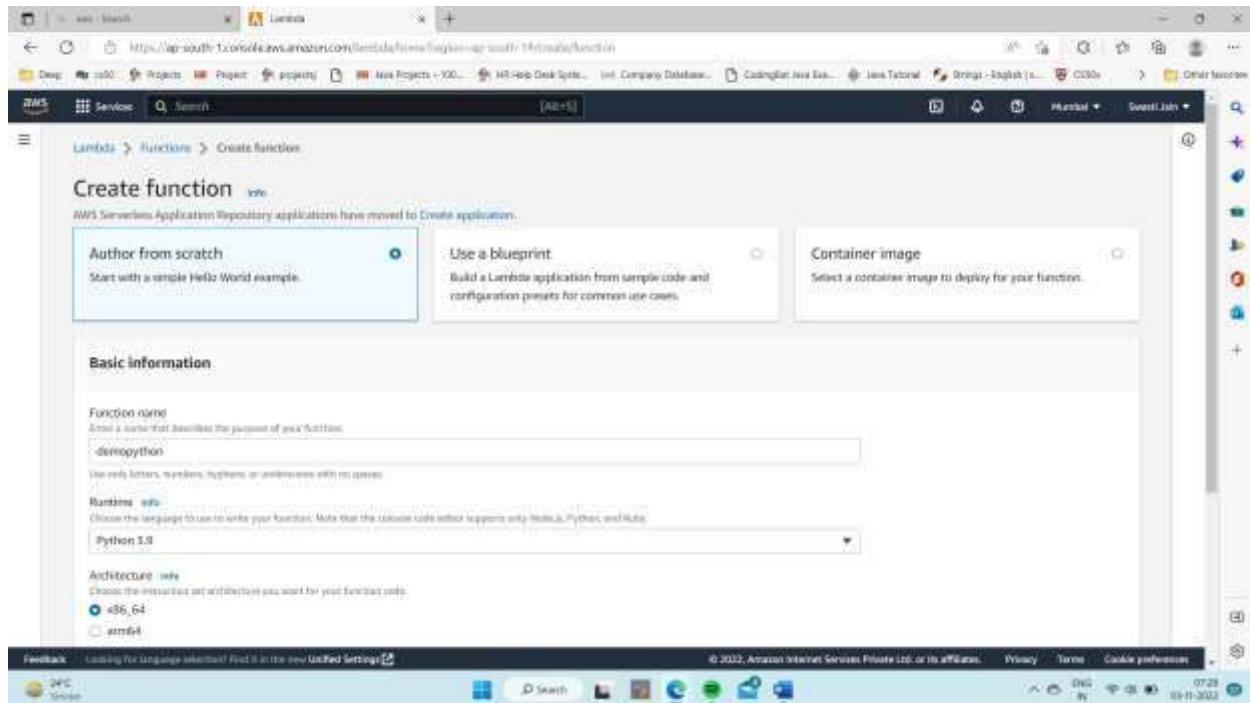
Reference: <https://www.youtube.com/watch?v=HOqA1zVEWSk>

Experiment no. 11

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

Steps:

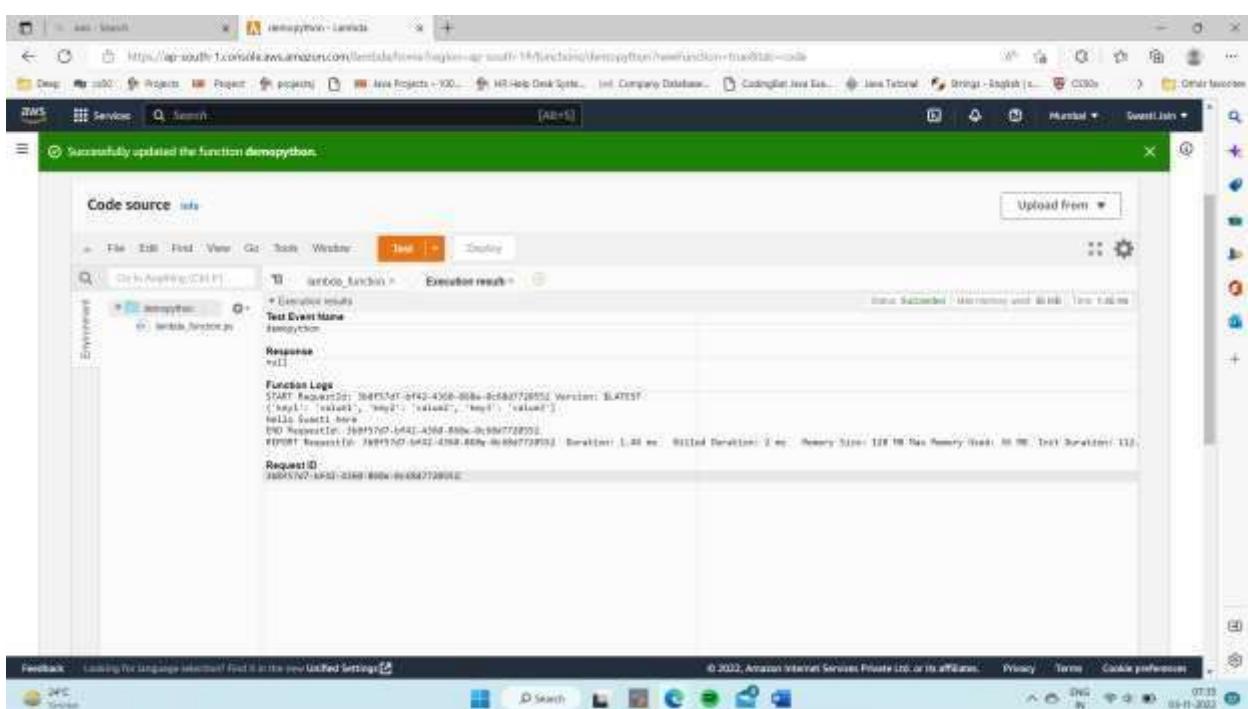
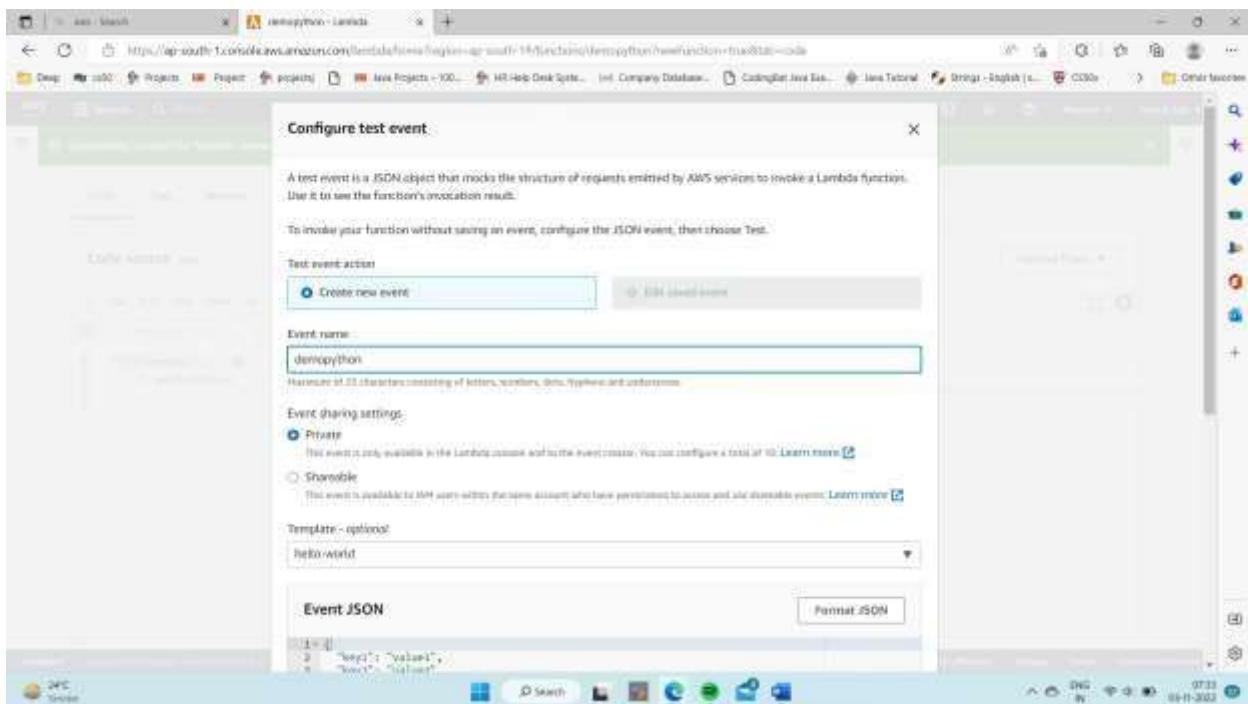
To understand AWS Lambda, its workflow, and various functions and create your first Lambda functions using Python / Java / Nodejs

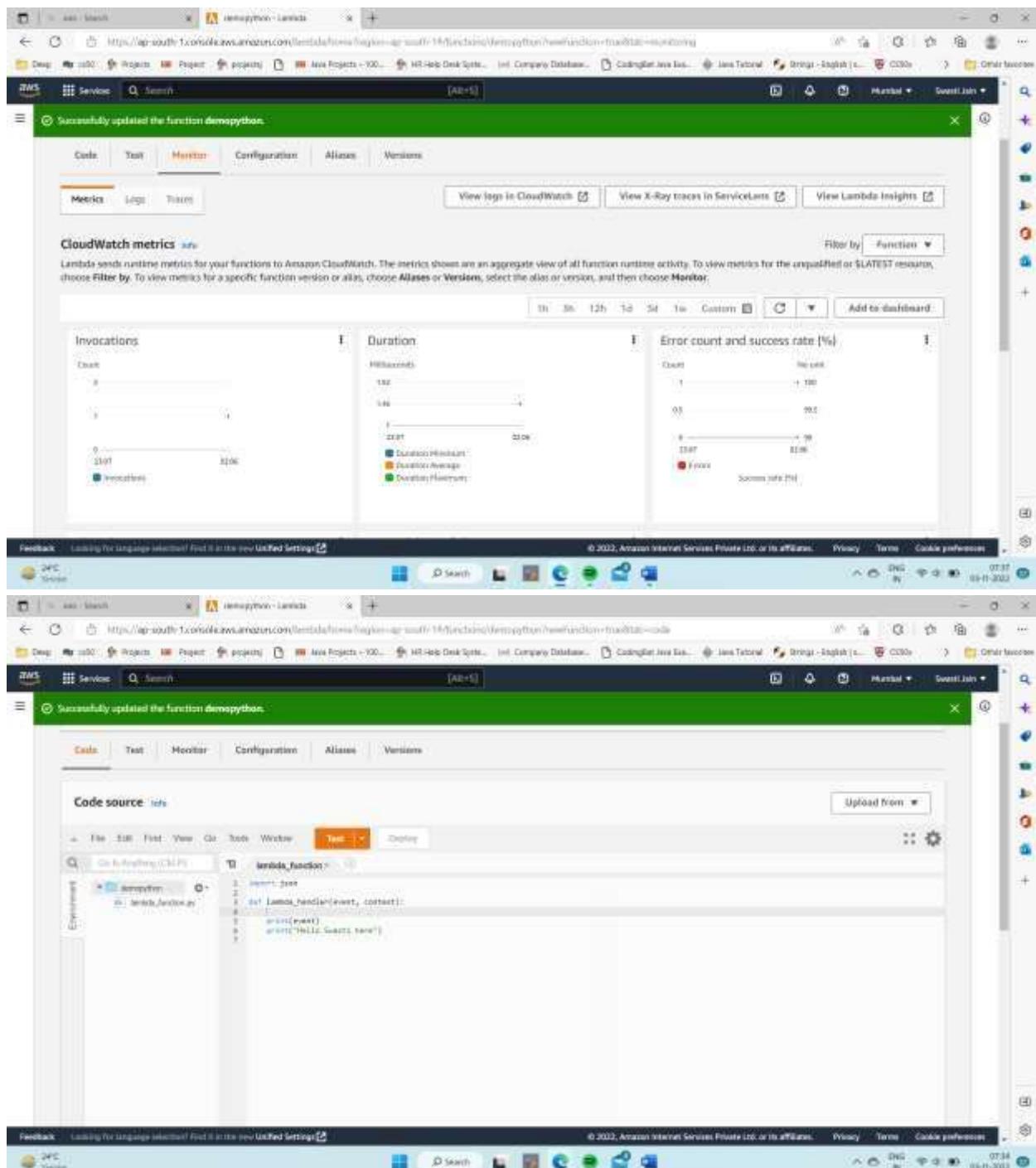


The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with tabs like 'Service', 'Search', and 'Actions'. Below it, a green banner says 'Successfully created the function demopython. You can now change its code and configuration. To invoke your function with a test event, choose "Test".' The main area is titled 'demopython' and contains a 'Function overview' section. It shows a thumbnail for the function, a 'Layers' button, and buttons for '+ Add trigger' and '+ Add destination'. On the right, there's a 'Description' field (empty), 'Last modified' (7 seconds ago), 'Function ARN' (arn:aws:lambda:ap-south-1:57690210110:function:demopython), and a 'Function URL' (tbd). Below this are tabs for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'.

The second screenshot shows the 'Code source' tab of the Lambda function editor. It has a toolbar with 'File', 'Edit', 'Find', 'View', 'Go', 'Tools', 'Window', 'Test', 'Deploy', and a status message 'Changes not deployed'. The code editor window displays a file named 'lambda_function.py' with the following content:

```
def lambda_handler(event, context):
    print("Hello")
    print("Hello World")
```





Conclusion: Successfully created Lambda functions using Python.

Experiment no. 12

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

To start using AWS Lambda with Amazon S3, we need the following –

- Create S3 Bucket
- Create role which has permission to work with s3 and lambda
- Create lambda function and add s3 as the trigger.

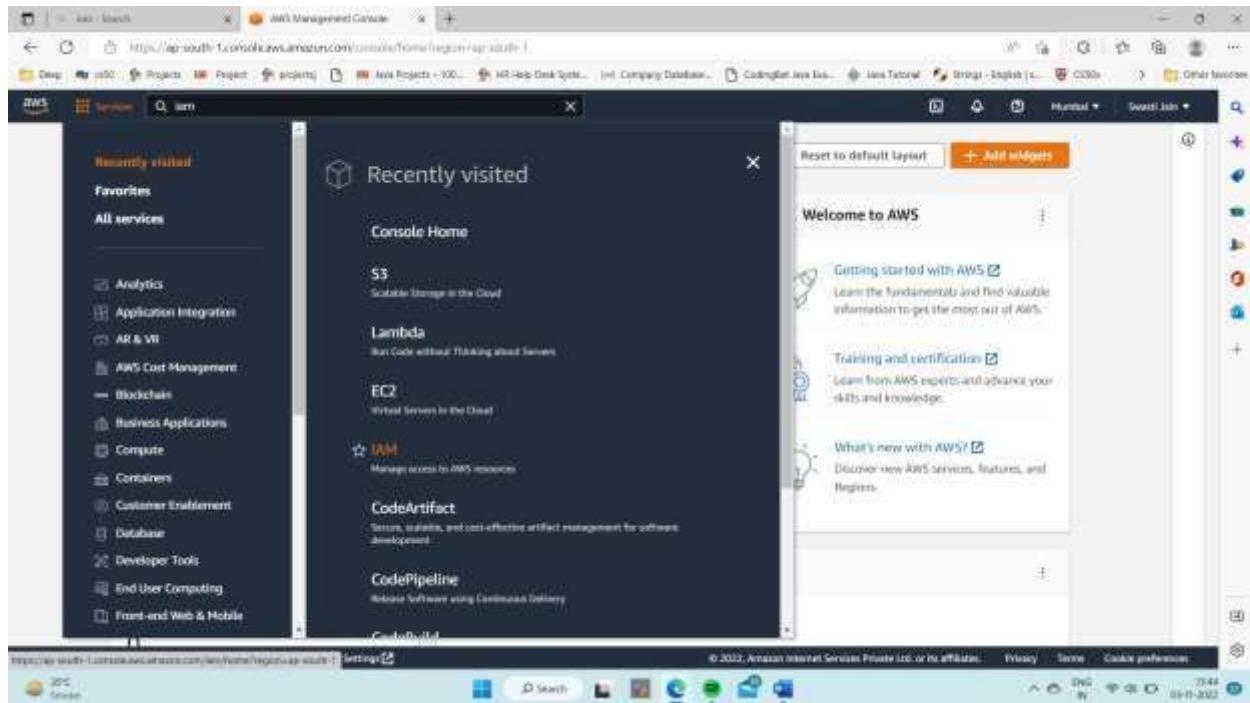
Creating S3 Bucket

The screenshot shows the AWS S3 Management Console. A green success message at the top states: "Successfully created bucket 'workingwithlambdaandS'. To upload files and folders, or to configure additional bucket settings choose View details." On the left, a sidebar menu includes options like Buckets, Access Points, Lambda Access Points, Multi-Region Access Points, Batch Operations, Access analysis for S3, Block Public Access settings for this account, Storage Lens, and AWS Marketplace for S3. The main area displays an "Account snapshot" and a "Buckets (4)" table. The table lists four buckets: "workingwithlambdaandS" (Bucket and objects are public, created Nov 5, 2022), "www-to-destination" (Public, created Sep 10, 2022), "www-test-attitard" (Objects can be public, created Sep 10, 2022), and "codepipeline-ap-south-1-868151265226" (Objects can be public, created Sep 10, 2022). A prominent orange "Create bucket" button is located at the top right of the Buckets table. The bottom of the screen shows the Windows taskbar with various pinned icons.

Create Role that Works with S3 and Lambda

To create role that works with S3 and Lambda, please follow the Steps given below :

Step 1: Go to AWS services and select IAM as shown below –



IAM dashboard

A screenshot of the IAM dashboard. On the left, under 'Security recommendations', there are two items: 'Add MFA for root user' (with a red warning icon) and 'Deactivate or delete access keys for root user' (with a red warning icon). Below these, the 'IAM resources' section shows counts for User groups (0), Users (1), Roles (3), Policies (0), and Identity providers (0). On the right, the 'AWS Account' section displays the Account ID (625453636107), Account Alias (625453636107), and a Sign-in URL (https://625453636107.signin.aws.amazon.com/console). A 'Quick Links' section at the bottom right includes links for 'My security credentials', 'Manage your access keys', and 'Tools'.

Step 2

Now, click IAM -> Roles as shown below –

The screenshot shows the AWS IAM Management Console. On the left, there's a navigation sidebar with sections like 'Access management' (highlighted), 'Identity providers', 'Account settings', 'Access reports', 'Analytics', and 'Organization activity'. The main area is titled 'Roles (5) info' and contains a table of roles. The table has columns for 'Role name', 'Trusted entities', and 'Last activity'. The roles listed are: 'AWSCodePipelineServiceRole-*ap-south-1-Pipeline*' (AWS Service: codepipeline), 'AWSServiceRoleForSupport' (AWS Service: support (Service-Linked Role)), 'AWSServiceRoleForTrustedAdvisor' (AWS Service: trustedadvisor (Service-Linked Role)), 'codebuild-*CodeBuild-user-role*' (AWS Service: codebuild), and 'demopriyanshirole-*BuBb7P9t*' (AWS Service: lambda). A 'Create role' button is visible at the top right of the table area.

This screenshot shows the 'Create New Role' wizard, Step 1: 'Select trusted entity'. It includes a progress bar with three steps: Step 1 (Select trusted entity), Step 2 (Add permissions), and Step 3 (Review, Name, and Options). Below the progress bar is a 'Trusted entity type' section with four options: 'AWS service' (selected), 'AWS account', 'Web identity', and 'SAML 2.0 Assertion'. Under 'AWS service', it says 'Allow AWS services like EC2, Lambda, or others to perform actions in your account'. Under 'Use case', it lists 'EC2' (Amazon EC2 instances to call AWS services in your account) and 'Lambda' (Select private functions to call AWS services on your behalf). At the bottom, there's a link to 'Use cases for other AWS services'.

Step 3

Add the permission from below :

Step 2: Add permissions

Permissions policy summary

Policy name	Type	Attached as
AmazonS3FullAccess	AWS managed	Permissions policy
AWSLambdaFullAccess	AWS managed	Permissions policy
CloudWatchFullAccess	AWS managed	Permissions policy

Tags

Add tags (optional) [Info](#)

Tag key-value pairs that you can add to this resource to help identify, organize, or search for resources.

0 tags associated with the resource.

Add tag

Learn more

Observe that we have chosen the following permissions – we have selected are

AmazonS3FullAccess, AWSLambdaFullAccess and CloudWatchFullAccess.

Step 4

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
- Analytic
- Settings
- Credential report
- Organization activity

IAM Roles

Roles (5) [Info](#)

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

Create role

Search

Role name	Trusted entities	Last used
AWSCodePipelineServiceRole-ug-ssn-1-Pipeline	AWS Service: codepipeline	63 days ago
AWSServiceRoleForSupport	AWS Service: support (Service-Linked Role)	-
AWSServiceRoleForTrustAdvisor	AWS Service: trustedadvisor (Service-Linked Role)	-
codebuild-codebuild-service-role	AWS Service: codebuild	53 days ago
demopython-role-fu8t7In	AWS Service: lambda	6 hours ago
lambdaawstwilio	AWS Service: lambda	-

Roles Anywhere [Info](#)

Authenticate your own AWS accounts and securely provide access to AWS services

Manage

Learn more

Create Lambda function and Add S3 Trigger

In this section, let us see how to create a Lambda function and add a S3 trigger to it. For this purpose, you will have to follow the Steps given below –

Step 1

Go to AWS Services and select Lambda as shown below –

The image consists of three vertically stacked screenshots of the AWS Lambda 'Create function' wizard.

Screenshot 1: The first step of the wizard, 'Create function'. It shows three options: 'Author from scratch' (selected), 'Use a blueprint', and 'Container image'. Below these are sections for 'Basic information' (Function name: 'lambdaWithS3bucket', Runtime: 'Node.js 16.x'), 'Architecture' (x86_64 selected), and 'Execution role' (Create a new role with basic Lambda permissions selected). A note at the bottom says '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 creating triggers.'

Screenshot 2: The second step of the wizard, 'Change default execution role'. It shows two sections: 'Execution role' (Create a new role with basic Lambda permissions selected) and 'Existing role' (LambdaDefaultRole selected). A note at the bottom says 'Select the LambdaDefaultRole role or the ARN of another role that has the LambdaBasicExecutionPolicy attached to it.'

Screenshot 3: The third step of the wizard, 'Advanced settings'. It contains four checkboxes: 'Enable Code signing' (unchecked), 'Enable function URL' (unchecked), 'Enable tags' (unchecked), and 'Enable VPC' (unchecked).

Step 2

Click Lambda and follow the process for adding Name. Choose the Runtime, Role etc. and create the function. The Lambda function that we have created is shown in the screenshot below –

The image shows two screenshots of the AWS Lambda service.

Screenshot 1: Lambda Function Creation Wizard

This screenshot shows the initial configuration steps for a new Lambda function:

- Architecture:** x86_64 (selected)
- Permissions:** 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.
- Execution role:** Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.
 - Create a new role with basic Lambda permissions
 - Use an existing role
 - Create a new role from AWS policy templates
- Existing role:** Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.
demolambdarole

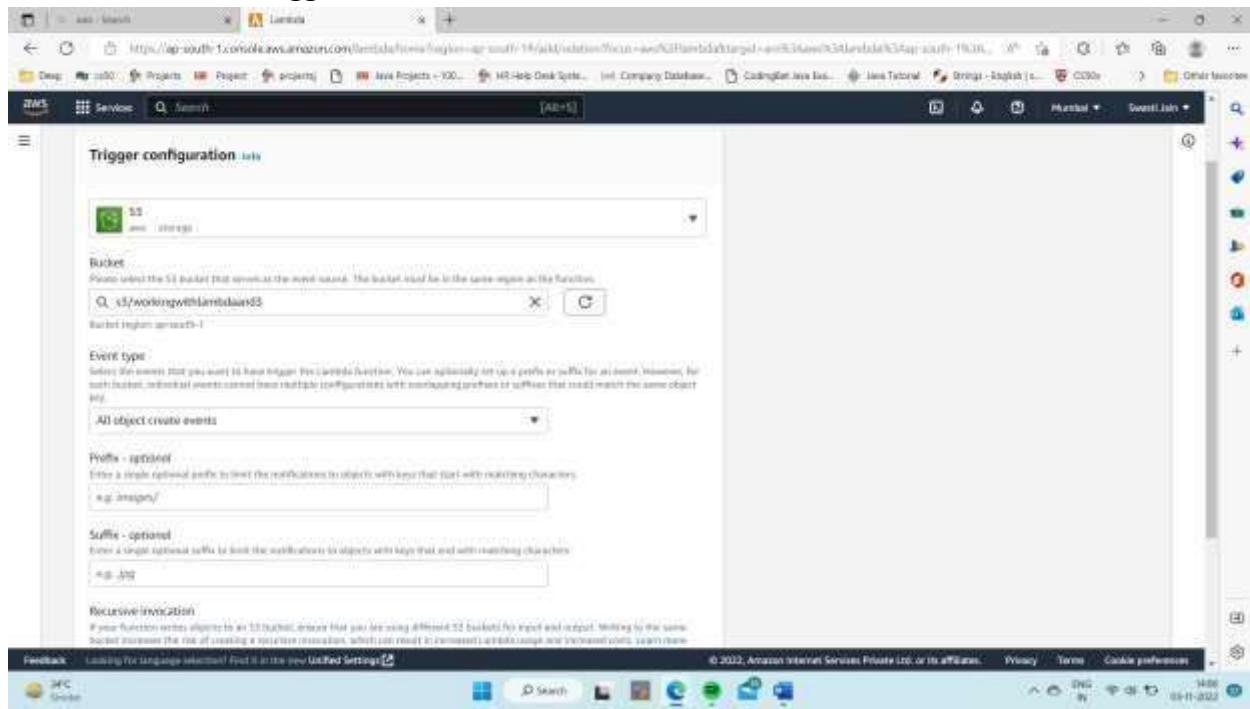
Screenshot 2: Lambda Function Overview

This screenshot shows the details of the newly created Lambda function, lambdawiths3bucket:

- Function overview:** lambdawiths3bucket
- Description:** (empty)
- Last modified:** 1 minute ago
- Function ARN:** arn:aws:lambda:ap-south-1:576802191139:function:lambdawiths3bucket
- Function URL:** (empty)
- Actions:** Throttle, Copy ARN, Actions ▾
 - Publish new version
 - Create alias
 - Export function
 - Delete function
- Code:** Test, Monitor, Configuration, Aliases, Versions
- Code source:** Upload from (dropdown menu)

Hence a lambda function is created

NOW Lets Create a trigger



Below is the code written in node.js to pinned a log about an object that has been added to s3.

The screenshot shows the AWS Lambda function editor with the file 'index.js' open. The code is as follows:

```
1 exports.handler = function(event, context, callback) {
2     console.log("Incoming Event: ", event);
3     const bucket = event.Records[0].s3.bucket.name;
4     const filename = decodeURIComponent(event.Records[0].s3.object.key.replace(/\+/g, ' '));
5     const message = `An image has been added to - ${bucket} -> ${filename}`;
6     console.log(message);
7     callback(null, message);
8 }
```

Now to test our lambda function with s3 trigger.

Step 1.add a image to s3 bucket

The screenshot shows the AWS S3 Management Console interface. The user is in the 'Upload' section of a bucket named 'workingwithlambda3'. The 'Destination' field is set to 's3://workingwithlambda3'. A file named 'Michael-Scott.jpg' (Type: Image/jpeg, Size: 142.6 KB) has been selected for upload. The status bar at the bottom indicates 'Upload successful'.

Upload

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, 142.6 KB)

All files and folders in this folder will be uploaded.

Name	Type	Size
Michael-Scott.jpg	Image/jpeg	142.6 KB

Destination

Destination:
s3://workingwithlambda3

[Destination details](#)

Feedback Looking for language selector? Find it in the new [Updated Settings](#)

Upload successful

[View details below](#)

Upload: status

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

Summary

Destination	Succeeded	Failed
s3://workingwithlambda3	1 File, 142.6 KB (100.00%)	0 Files, 0.0 (0%)

[Close](#)

Files and folders (1 Total, 142.6 KB)

Name	Folder	Type	Status	Error
Michael-Scott.jpg		Image/jpeg	Success	

Feedback Looking for language selector? Find it in the new [Updated Settings](#)

Step 2.

Now go to cloudwatch you will see the logs of newly happened activity in s3.

The screenshots show the AWS CloudWatch Log Stream interface. The first screenshot displays the 'Log group details' for the log group '/aws/lambda/lambdawiths3bucket'. It shows information like creation time (3 minutes ago), KMS key ID, metric filters (0), stored bytes (~49B), and subscription filters (AWS CloudWatch Metrics). The second screenshot shows the 'Log streams' tab with one stream listed: '2022/11/05/[SLATEST]J05e1bb63b44488bc77fe55cd6cb51'. The third screenshot shows the 'Log events' tab for the same stream, displaying three log entries:

Timestamp	Message
2022-11-05T14:18:34,034+05:30	START RequestId: aad5794-c2f7-4fb2-9ed7-6cd54410bba Version: \$LATEST
2022-11-05T14:18:34,035+05:30	END RequestId: aad5794-c2f7-4fb2-9ed7-6cd54410bba
2022-11-05T14:18:34,035+05:30	REPORT RequestId: aad5794-c2f7-4fb2-9ed7-6cd54410bba Duration: 0.16 ms Billed Duration: 0.0 ms Memory Size: 108 MB

Conclusion: Successfully created a Lambda function that will log “An Image has been added” once add you an object to a specific bucket in S3.