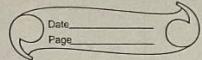


PRACTICAL 01

Name : Sonam chhabaidiya

class : D15A

Roll No:- 09



Adv. devOps
Experiment 01

Aim :- To understand the benefits of cloud infrastructure and setup AWS cloud 9 IDE, launch AWS cloud 9 IDE and perform collaboration.

Theory :- Amazon web services (aws) is a comprehensive cloud computing platform provided by Amazon. It offers wide range of cloud-based services, including computer power, storage and databases, machine learning and more.

Amazon EC2 is a core service with AWS that provides scalable virtual servers known as EC2 instances. These instances are designed to handle various workloads from basic web application to high performance computing tasks. EC2 allows users to launch and manage virtual servers in the cloud providing flexibility in terms of computing power and operating system.

An EC2 instance is a virtual server within the Amazon EC2 service. It represents a single unit of computing capacity in the AWS cloud.

Conclusion :- Thus, we have understood the benefits of cloud infra and setup AWS cloud 9 IDE. Due to private access, the S3 website was not accessible.

Teacher's Sign.: _____

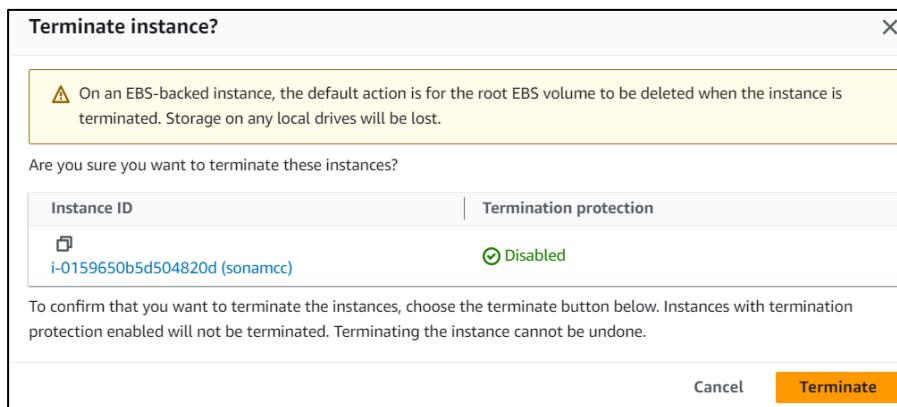
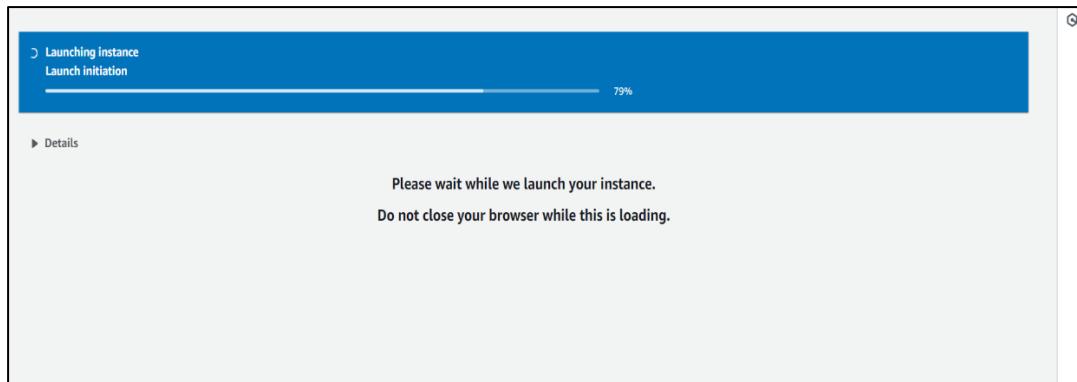
EC2 INSTANCES

The screenshot shows the AWS EC2 Dashboard in the US East (N. Virginia) Region. The left sidebar includes links for EC2 Global View, Events, Console-to-Code, Instances, Images, and Elastic Block Store. The main area displays 'Resources' with counts for Instances (running), Auto Scaling Groups, Dedicated Hosts, Elastic IPs, Instances, Key pairs, Load balancers, Placement groups, Security groups, Snapshots, and Volumes. Below this is a 'Launch instance' section with 'Launch instance' and 'Migrate a server' buttons, and a note about launching in the US East (N. Virginia) Region. To the right are sections for 'Account attributes' (Default VPC, Settings), 'Explore AWS' (Save up to 90% on EC2 with Spot Instances, Enable Best Price-Performance with AWS Graviton2), and 'Service health' (AWS Health Dashboard).

This screenshot shows the 'Launch an instance' wizard, Step 1: Set instance details. It includes fields for 'Name and tags' (Name: sonamcd), 'Application and OS Images (Amazon Machine Image)' (Software Image (AMI): Amazon Linux 2023.5.2...), 'Virtual server type (instance type)' (t3.micro), and 'Storage (volumes)' (1 volume(s) - 8 GiB). A note states: "Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below."

This screenshot shows the 'Launch an instance' wizard, Step 2: Set security and connectivity. It asks for a key pair: 'Existing key pair' (radio button), 'Create new key pair' (radio button, selected), and 'Proceed without key pair'. It also asks for a key pair name (SONA) and key pair type (RSA, selected). It specifies a private key file format (.pem, selected) for use with OpenSSH.

SONAM CHHABAIDIAY D15A 09



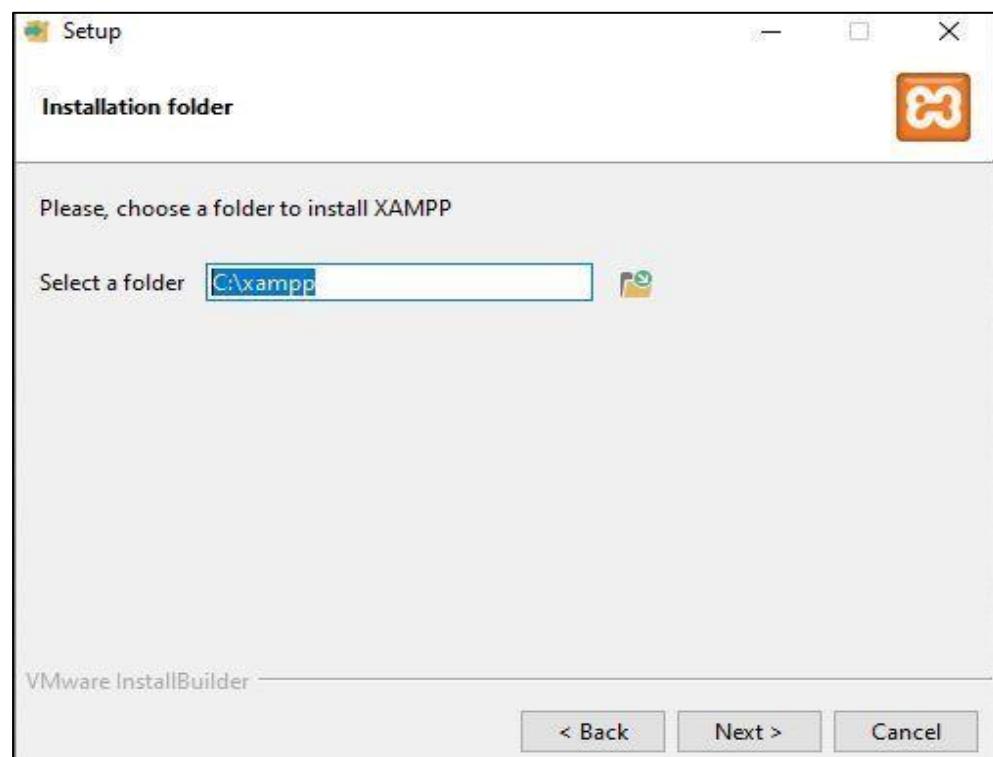
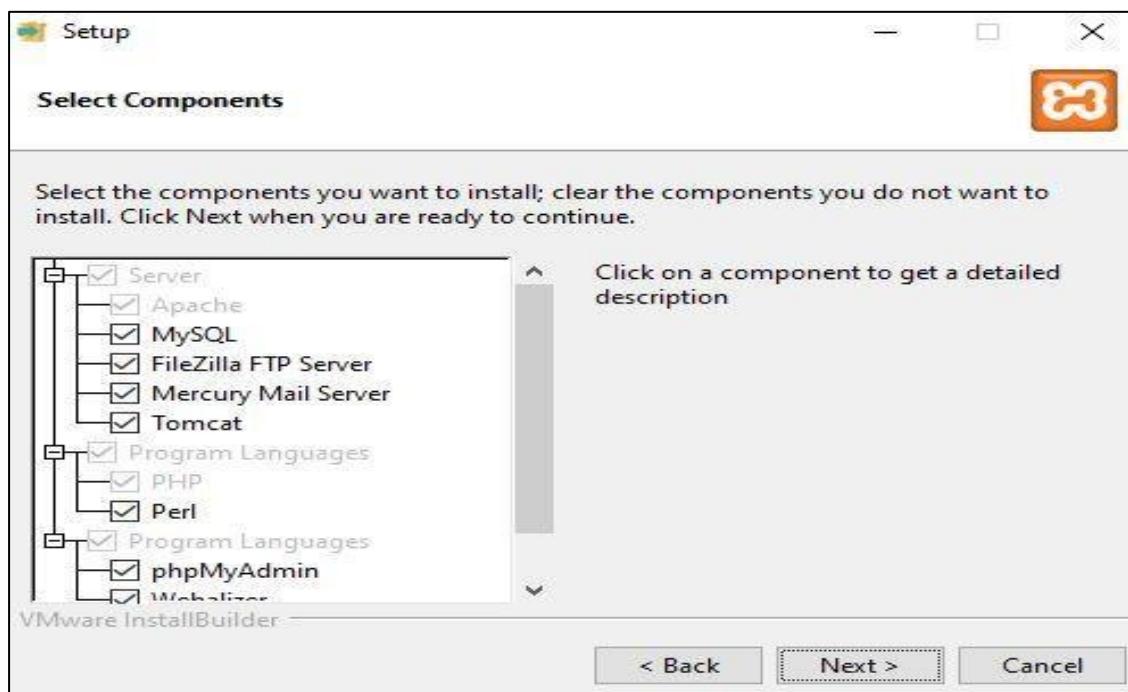
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
AAR-CICD	i-01470860556d94500	Running	t3.micro	2/2 checks passed	View alarms	eu-north-1b
<input checked="" type="checkbox"/> sonamcc	i-0159650b5d504820d	Terminated	t3.micro	Initializing	View alarms	eu-north-1b
	i-022af7c957d95ee84	Running	t3.micro	2/2 checks passed	View alarms	eu-north-1b

```
ubuntu@ip-172-31-5-93:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 39 not upgraded.
Need to get 1680 kB/2083 kB of archives.
After this operation, 8094 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-bin amd64 2.4.58-1ubuntu8.4 [1329 kB]
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-data all 2.4.58-1ubuntu8.4 [163 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-utils amd64 2.4.58-1ubuntu8.4 [97.1 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 amd64 2.4.58-1ubuntu8.4 [90.2 kB]
Fetched 1680 kB in 0s (17.0 MB/s)
Preconfiguring packages ...
Selecting previously unselected package libapr1t64:amd64.
(Reading database ... 67739 files and directories currently installed.)
Preparing to unpack .../0-libapr1t64_1.7.2-3.1build2_amd64.deb ...
Unpacking libapr1t64:amd64 (1.7.2-3.1build2) ...
Selecting previously unselected package libaprutil1t64:amd64.
Preparing to unpack .../1-libaprutil1t64_1.6.3-1.lubuntu7_amd64.deb ...
```

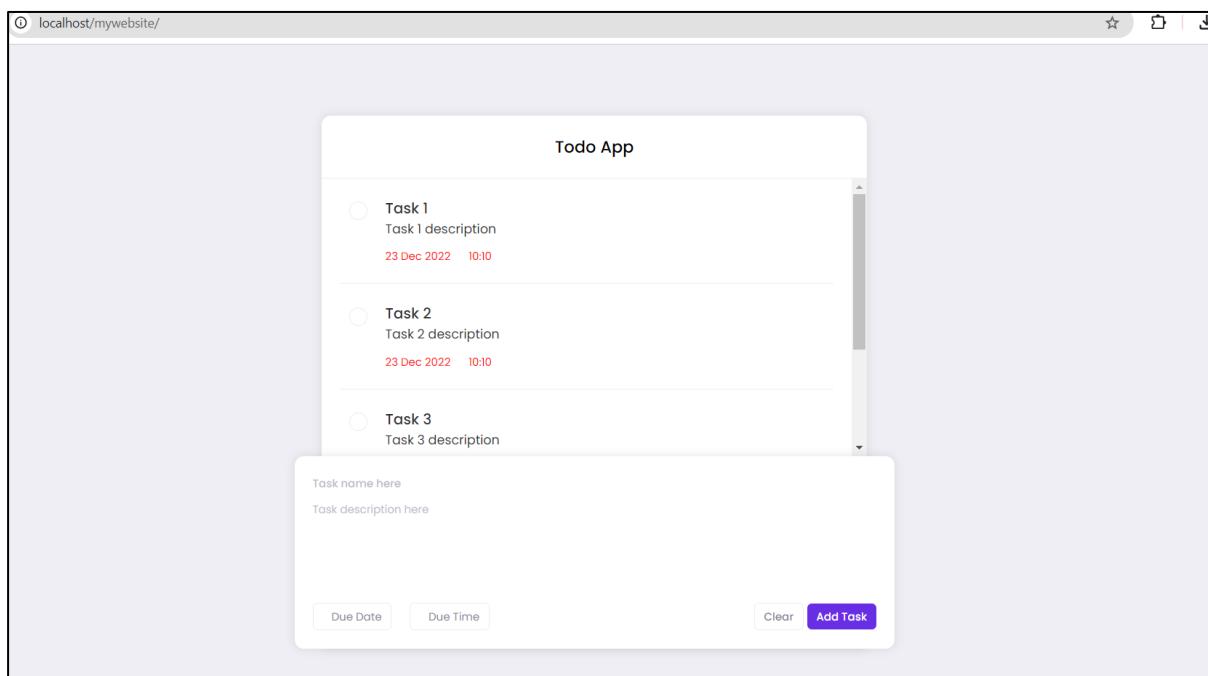
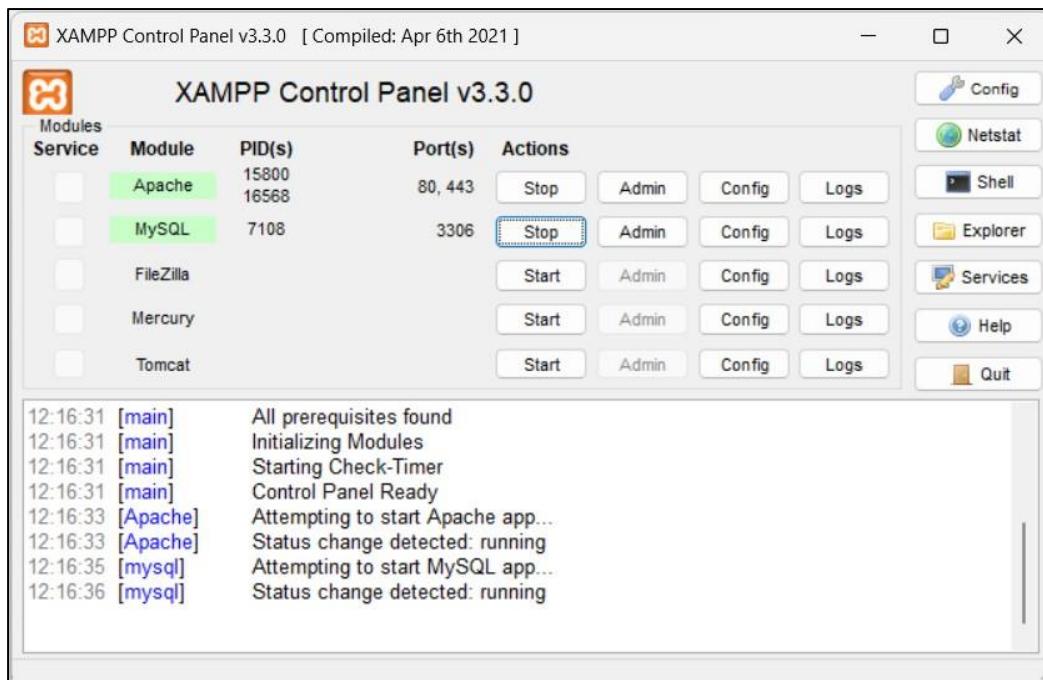
Part 1: a) To develop a website and host it on your local machine on a VM

- Steps to host Website using Xampp Server





SONAM CHHABAIDAY D15A 09



Part 1: b) Hosting a static website on Amazon S3

- Steps to host Website using Amazon S3

The screenshot shows the AWS S3 console interface. The top navigation bar has tabs for "General purpose buckets" (selected) and "Directory buckets". Below the tabs, there's a search bar with placeholder text "Find buckets by name" and a pagination indicator showing "1" of 1 page. A "Create bucket" button is prominently displayed in the top right.

Name	AWS Region	IAM Access Analyzer	Creation date
codepipeline-eu-north-1-636097867266	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 5, 2024, 20:55:50 (UTC+05:30)
sonamcc-todo-website	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 7, 2024, 14:34:59 (UTC+05:30)

The screenshot shows the AWS S3 bucket details page for "sonamcc-todo-website". At the top, it displays the URL "s3://sonamcc-todo-website" and file statistics: "6 files, 14.7 KB (100.00%)". Below this, there are tabs for "Files and folders" (selected) and "Configuration".

Files and folders (6 Total, 14.7 KB)

Name	Folder	Type	Size	Status	Error
add_tasks.p...	-	-	608.0 B	Succeeded	-
config.php	-	-	351.0 B	Succeeded	-
index.html	-	text/html	1.8 KB	Succeeded	-
index.php	-	-	2.7 KB	Succeeded	-
script.js	-	text/javascript	4.7 KB	Succeeded	-
style.css	-	text/css	4.5 KB	Succeeded	-

Object overview	
Owner	S3 URI
fd6b26c18c3c863e1c49c7ad460eaf4c9b5e8cda899e8a277fd8231e6410154e	s3://sonamcc-todo-website/index.html
AWS Region	Amazon Resource Name (ARN)
Europe (Stockholm) eu-north-1	arnaws:s3:::sonamcc-todo-website/index.html
Last modified	Entity tag (Etag)
August 7, 2024, 14:38:26 (UTC+05:30)	0777517de81bbdc7e8b7be2419fd5b0d
Size	Object URL
1.8 KB	https://sonamcc-todo-website.s3.eu-north-1.amazonaws.com/index.html
Type	
html	
Key	
index.html	

Not secure sonamcc-todo-website.s3-website.eu-north-1.amazonaws.com

Todo App

- Task 1
Task 1 description
23 Dec 2022 • 10:10
- Task 2
Task 2 description
23 Dec 2022 • 10:10
- Task 3
Task 3 description

ROLL NO: 09

Date _____
Page _____

Adv. DevOps.

Experiment : 02

Aim:- To Build a Application using AWS code Build and Deploy an S3/SEBS using AWS code Pipeline, Deploy Sample Application on EC2 instance using AWS codeDeploy.

Theory :-

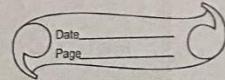
- AWS CodePipeline

- AWS CodePipeline is a fully managed continuous integration & continuous delivery service that automates the build, test & deploy phases of your release process.
- It allows you to define a sequence of actions, such as source retrieval, building the code, running tests & deploying the application.
- It integrates with other AWS services, making it a versatile tool for automating application deployments.

- AWS CodeBuild

- AWS CodeBuild is a fully managed build service in the cloud.
- It compiles your source code, runs unit tests & produces artifacts that are ready to deploy.
- CodeBuild scales continuously and processes multiple builds concurrently, so your builds are never left waiting in a queue.

Teacher's Sign.: _____



- AWS CodeDeploy

- AWS CodeDeploy automates application deployments, to various compute services such as EC2, Lambda & Serverless.
- It handles the complexity of updating your applications across multiple instances, with zero downtime.
- Deployment types.

(i) In-place Deployment :- update the application on each instance in place.

(ii) Blue/Green Deployment :- Deploys the application on new instances, allowing to test new version before routing traffic to it.

Conclusion :-

This experiment demonstrates the integration of various AWS service to build a robust CI/CD pipeline.

By automating the build, test, & deployment process using AWS CodeBuild, CodePipeline & CodeDeploy, we can achieve faster release, reduce errors & improve the overall quality of applications.

Teacher's Sign.: _____

- Creating two roles in AWS IAM for CI/CD pipeline setup

The screenshot shows the AWS search interface with the query 'iam'. The top navigation bar includes the AWS logo, a 'Services' button, a search bar containing 'iam', and account information for 'N. Virginia' and 'voclabs/user3396433=GUNJAL_TEJA'. The left sidebar has sections for EC2 Dashboard, EC2 Global View, Events, Console-to-Code (with a 'Preview' link), Instances (with 'Instances' and 'Instance Types' links), Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (with 'AMIs' and 'AMI Catalog' links), and more. The main search results page displays 'Services (11)' and 'Features (24)'. The 'IAM' service card is highlighted, showing its icon, name, description ('Manage access to AWS resources'), and 'Top features' (Groups, Users, Roles, Policies, Access Analyzer). Other cards shown include 'IAM Identity Center' and 'Resource Access Manager'. A 'See all 11 results' link is at the top right of the services section, and a 'See all 24 results' link is at the bottom right of the features section.

The screenshot shows the IAM Dashboard. The left sidebar includes 'Identity and Access Management (IAM)', 'Dashboard', 'Access management' (User groups, Users, Roles, Policies, Identity providers, Account settings), 'Access reports' (Access Analyzer, External access, Unused access, Analyzer settings, Credential report, Organization activity, Service control policies), and a search bar. The main dashboard area has a 'Security recommendations' section with a red exclamation mark icon, listing 'Add MFA for root user' (with a 'Add MFA' button) and 'Root user has no active access keys' (with a note about improving security). Below this is an 'IAM resources' section showing counts for User groups (0), Users (0), Roles (2), Policies (0), and Identity providers (0). A 'What's new' section notes the addition of policy checks for public and critical resource access. On the right, there are 'AWS Account' details (Account ID: 010928184174, Account Alias: Create, Sign-in URL: https://010928184174.signin.aws.amazon.com/console), 'Quick Links' (My security credentials, Manage access keys, MFA, and other credentials), and 'Tools' (with a 'Tools' link). The bottom of the screen shows the browser address bar with the URL https://voc-act-1.console.aws.amazon.com/iam/home?region=us-east-1#roles and the footer with copyright information for 2024 Amazon Web Services, Inc. or its affiliates, and links for Privacy, Terms, and Cookie preferences.

Identity and Access Management (IAM)

Roles (2) Info

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

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

Roles Anywhere Info

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

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

Step 2 Add permissions

Step 3 Name, review, and create

Trusted entity type

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

Use case

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

Service or use case

EC2

Choose a use case for the specified service.

Use case

- EC2: Allows EC2 instances to call AWS services on your behalf.
- EC2 Role for AWS Systems Manager: Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
- EC2 Spot Fleet Role: Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.
- EC2 - Spot Fleet Auto Scaling

IAM > Roles > Create role

Step 1 Select trusted entity

Step 2 Add permissions

Step 3 Name, review, and create

Add permissions Info

Permissions policies (1/945) Info

Choose one or more policies to attach to your new role.

Filter by Type: All types

Policy name	Type	Description
<input checked="" type="checkbox"/> AmazonEC2RoleforAWSCodeDeploy	AWS managed	Provides EC2 access to S3 bucket to ...
<input type="checkbox"/> AmazonEC2RoleforAWSCodeDeployLimited	AWS managed	Provides EC2 limited access to S3 buck...
<input type="checkbox"/> AWSCodeDeployDeployerAccess	AWS managed	Provides access to register and deploy ...
<input type="checkbox"/> AWSCodeDeployFullAccess	AWS managed	Provides full access to CodeDeploy res...
<input type="checkbox"/> AWSCodeDeployReadOnlyAccess	AWS managed	Provides read only access to CodeDepl...
<input type="checkbox"/> AWSCodeDeployRole	AWS managed	Provides CodeDeploy service access to ...
<input type="checkbox"/> AWSCodeDeployRoleForCloudFormation	AWS managed	Provides CodeDeploy service access to ...
<input type="checkbox"/> AWSCodeDeployRoleForECS	AWS managed	Provides CodeDeploy service wide acc...
<input type="checkbox"/> AWSCodeDeployRoleForECSLimited	AWS managed	Provides CodeDeploy service limited a...
<input type="checkbox"/> AWSCodeDeployRoleForLambda	AWS managed	Provides CodeDeploy service access to ...

Role details

Role name
Enter a meaningful name to identify this role.
EC2CodeDeploy

Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.

Step 1: Select trusted entities

Trust policy

```

1 < [           "version": "2012-10-17",
2   "Statement": [
3     {
4       "Effect": "Allow",
5       "Action": [
6         "sts:AssumeRole"
7       ],
8       "Principal": [
9         "Service",
10        "ec2.amazonaws.com"
11      ]
12    }
13  ]
14 ]
15 ]

```

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

Identity and Access Management (IAM)

IAM > Roles

Roles (3) Info

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

Role name	Trusted entities	Last activity
AWSServiceRoleForSupport	AWS Service: support (Service-Link)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service)	-
EC2CodeDeploy	AWS Service: ec2	-

Roles Anywhere Info

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

Access AWS from your non AWS workloads

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

X.509 Standard

Use your own existing PKI infrastructure or use AWS Certificate Manager Private Certificate Authority to authenticate identities.

Temporary credentials

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

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

Step 3 Name, review, and create

AWS service Allow AWS services like EC2, Lambda, or others to perform actions in this account.

AWS account Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

Web identity Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

SAML 2.0 federation Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

Custom trust policy Create a custom trust policy to enable others to perform actions in this account.

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

Service or use case
CodeDeploy

Choose a use case for the specified service.
Use case

CodeDeploy Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.

CodeDeploy for Lambda Allows CodeDeploy to route traffic to a new version of an AWS Lambda function version on your behalf.

CodeDeploy - ECS Allows CodeDeploy to read S3 objects, invoke Lambda functions, publish to SNS topics, and update ECS services on your behalf.

Next

IAM > Roles > Create role

Step 1 Select trusted entity

Step 2 Add permissions

Step 3 Name, review, and create

Add permissions

Permissions policies (1) Info
The type of role that you selected requires the following policy.

Policy name	Type
<input checked="" type="checkbox"/> AWSCodeDeployRole	AWS managed

Set permissions boundary - optional

Next

CloudShell Feedback

Step 1: Select trusted entities

Role details

Role name
Enter a meaningful name to identify this role.
CodeDeployRole

Description
Add a short explanation for this role.
Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.

Trust policy

```

1+ [{"Version": "2012-10-17", "Statement": [{"Sid": "", "Effect": "Allow", "Principal": {"Service": ["codedeploy.amazonaws.com"]}, "Action": ["sts:AssumeRole"]}], "2": "", "3": "", "4": "", "5": "", "6": "", "7": "", "8": "", "9": "", "10": "", "11": "", "12": "", "13": ""}]

```

Name, review, and create

Next

➤ Launching & Setting up EC2 Instance for Deployment

Search results for 'EC'

Services (112)

- EC2
- Security Hub
- Security Lake
- Direct Connect

Features (293)

- Direct Connect gateways
- AWS Private CA Connector for Active Directory

Resources (New)

Knowledge Articles (1,942)

Marketplace (440)

Blogs (23,090)

Events (696)

Tutorials (90)

See all 112 results ►

See all 293 results ►

Launch an instance [Info](#)

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

Name and tags [Info](#)

Name: AAR-CICD [Add additional tags](#)

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

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

Recent **Quick Start**

- Amazon Linux
- macOS
- Ubuntu
- Windows
- Red Hat
- SUSE Linux Enterprise Server

Summary

Number of instances: [Info](#) 1

Software Image (AMI): [Amazon Linux 2023 AMI 2023.5.2... read more](#) ami-05c3d60cb90f70

Virtual server type (instance type): t2.micro

Firewall (security group): default

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os.

Launch instance

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

aws Services Search [Alt+S]

Network [Info](#)
vpc-008c238273e437b32
Subnet [Info](#)
No preference (Default subnet in any availability zone)
Auto-assign public IP [Info](#)
Enable
Additional charges apply when outside of **free tier allowance**
Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
 Create security group Select existing security group
Common security groups [Info](#)
 Select security groups
 default sg-005e9d856101cf758 X
 VPC: vpc-008c238273e437b32
 Security groups that you add or remove here will be added to or removed from all your network interfaces.

Configure storage [Info](#) Advanced
1x 8 GiB gp3 Root volume (Not encrypted)

Summary
Number of instances [Info](#)
1
Software Image (AMI) Amazon Linux 2023 AMI 2023.5.2...read more ami-05c3dc660cb6907f0
Virtual server type (instance type) t2.micro
Firewall (security group) default
Storage (volumes) 1 volume(s) - 8 GiB
 ⓘ **Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance

Cancel **Launch instance** Review commands

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

aws Services Search [Alt+S]

Advanced details [Info](#)
Domain join directory [Info](#)
 Select

IAM instance profile [Info](#)
 EC2CodeDeploy [Info](#)
 arn:aws:iam::010928184174:instance-profile/EC2CodeDeploy

Hostname type [Info](#)
 IP name

DNS Hostname [Info](#)
 Enable IP name IPv4 (A record) DNS requests
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery [Info](#)
 Select

Shutdown behavior [Info](#)
 Stop

Stop - Hibernate behavior [Info](#)
 Select

Summary
Number of instances [Info](#)
1
Software Image (AMI) Amazon Linux 2023 AMI 2023.5.2...read more ami-05c3dc660cb6907f0
Virtual server type (instance type) t2.micro
Firewall (security group) default
Storage (volumes) 1 volume(s) - 8 GiB
 ⓘ **Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance

Cancel **Launch instance** Review commands

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

aws Services Search [Alt+S]

2
Allow tags in metadata [Info](#)
Select

User data - optional [Info](#)
Upload a file with your user data or enter it in the field.


```
#!/bin/bash
sudo yum -y update
sudo yum -y install ruby
sudo yum -y install wget
cd /home/ec2-user
wget https://aws-codedeploy-ap-south-1.s3.ap-south-1.amazonaws.com/latest/install
sudo chmod +x ./install
sudo ./install auto
sudo yum install -y python-pip
sudo pip install awscli
```

User data has already been base64 encoded

Summary
Number of instances [Info](#)
1
Software Image (AMI) Amazon Linux 2023 AMI 2023.5.2...read more ami-05c3dc660cb6907f0
Virtual server type (instance type) t2.micro
Firewall (security group) default
Storage (volumes) 1 volume(s) - 8 GiB
 ⓘ **Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance

Cancel **Launch instance** Review commands

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

Success
Successfully initiated launch of instance (i-06ee810b4baa23bd8)

Launch log

Next Steps

What would you like to do next with this instance, for example "create alarm" or "create backup"?

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

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

Instances (2) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
AAR-CICD	i-01470860556d94500	Running	t3.micro	2/2 checks passed	View alarms +	eu-north-1b	ec2-13-
	i-022af7c957d95ee84	Running	t3.micro	2/2 checks passed	View alarms +	eu-north-1b	ec2-16-

Select an instance

i-01470860556d94500 (AAR-CICD)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary

Instance ID i-01470860556d94500 (AAR-CICD)	Public IPv4 address 13.53.70.170 open address	Private IPv4 addresses 172.31.37.42
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-13-53-70-170.eu-north-1.compute.amazonaws.com open address
Hostname type IP name: ip-172-31-37-42.eu-north-1.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-37-42.eu-north-1.compute.internal	Elastic IP addresses -
Answer private resource DNS name IPv4 (A)	Instance type t3.micro	

- Creating an application and deployment group in AWS CodeDeploy

The screenshot shows the AWS Services search interface. The search bar at the top contains the text "CodeArtifact". Below the search bar, the sidebar lists various AWS services under categories like Instances, Images, and Network & Security. The main content area displays search results for "Code" across different service categories.

Services (32)	
Features (45)	See all 32 results ▶
Resources New	
Documentation (103,964)	
Knowledge Articles (137)	
Marketplace (5,843)	
Blogs (2,966)	
Events (52)	
Tutorials (15)	

Services	
Amazon Q Developer (Including Amazon CodeWhisperer) ☆	Build applications faster, and spend less time solving software development problems.
CodeCommit ☆	Store Code in Private Git Repositories
CodePipeline ☆	Release Software using Continuous Delivery
AWS Signer ☆	Ensuring trust and integrity of your code

Features	
Full repository analysis	See all 45 results ▶
Amazon CodeGuru feature	
Pull request code review	
Amazon CodeGuru feature	

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

The screenshot shows the AWS CodeDeploy Applications page. The left sidebar is titled "Developer Tools" and includes sections for CodeDeploy, Artifacts, Build, Deploy, Pipeline, and Settings. The main content area shows a table for managing applications, with a single row visible:

Applications		
Application name	Compute platform	Created
No results There are no results to display.		

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

The screenshot shows the 'Create application' configuration page in the AWS CodeDeploy console. The 'Application name' field contains 'AAR-CICD'. The 'Compute platform' dropdown is set to 'EC2/On-premises'. A 'Tags' section is present with a 'Add tag' button. At the bottom right are 'Cancel' and 'Create application' buttons.

The screenshot shows the 'AAR-CICD' application details page. It displays the application name 'AAR-CICD' and compute platform 'EC2/On-premises'. The 'Deployment groups' tab is selected, showing a table with columns: Name, Status, Last attempted deploy..., Last successful deploy..., and Trigger count. A message at the top states: 'Application created. In order to create a new deployment, you must first create a deployment group.' A 'Create deployment group' button is located at the bottom right of the deployment groups table.

Create deployment group

Application

Application
AAR-CI_CD
Compute type
EC2/On-premises

Deployment group name

Enter a deployment group name
AAR-CI_CD-DP
100 character limit

Service role

Enter a service role
Enter a service role with CodeDeploy permissions that grants AWS CodeDeploy access to your target instances.
arn:aws:iam::010928184174:role/CodeDeployRole

Deployment type

In-place
Updates the instances in the deployment group with the latest application revisions. During a deployment, each instance will be briefly taken offline for its update

Blue/green
Replaces the instances in the deployment group with new instances and deploys the latest application revision to them. After instances in the replacement environment are registered with a load balancer, instances from the original environment are deregistered and can be terminated.

Environment configuration

Select any combination of Amazon EC2 Auto Scaling groups, Amazon EC2 instances, and on-premises instances to add to this deployment

Amazon EC2 Auto Scaling groups

Amazon EC2 instances
1 unique matched instance. [Click here for details](#)

You can add up to three groups of tags for EC2 instances to this deployment group.
One tag group: Any instance identified by the tag group will be deployed to.
Multiple tag groups: Only instances identified by all the tag groups will be deployed to.

Tag group 1
Key: Name Value - optional: AAR-CI_CD
Add tag

Agent configuration with AWS Systems Manager

Complete the required prerequisites before AWS Systems Manager can install the CodeDeploy Agent.
Make sure the AWS Systems Manager Agent is installed on all instances and attach the required IAM policies to them. [Learn more](#)

Install AWS CodeDeploy Agent
 Never
 Only once
 Now and schedule updates

Basic scheduler Cron expression
14 Days

Deployment settings

Deployment configuration
Choose from a list of default and custom deployment configurations. A deployment configuration is a set of rules that determines how fast an application is deployed and the success or failure conditions for a deployment.
CodeDeployDefault.AllAtOnce or [Create deployment configuration](#)

Load balancer

Select a load balancer to manage incoming traffic during the deployment process. The load balancer blocks traffic from

- Setting up a CI/CD pipeline for deploying applications on EC2 using GitHub.

Pipeline settings

Pipeline name: AAR-CI_CD_PIPELINE

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

Execution mode: Choose the execution mode for your pipeline. This determines how the pipeline is run.

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

Service role:

- New service role: Create a service role in your account
- Existing service role: Choose an existing service role from your account

Variables

No variables defined at the pipeline level in this pipeline.

Add variable

You can add up to 50 variables.

Advanced settings

Artifact store:

- Default location: Create a default S3 bucket in your account.
- Custom location: Choose an existing S3 location from your account in the same region and account as your pipeline

Encryption key:

- Default AWS Managed Key: Use the AWS managed customer master key for CodePipeline in your account to encrypt the data in the artifact store.
- Customer Managed Key: To encrypt the data in the artifact store under an AWS KMS customer managed key, specify the key ID, key ARN, or alias ARN.

Next

Add source stage

Source

Source provider: GitHub (Version 2)

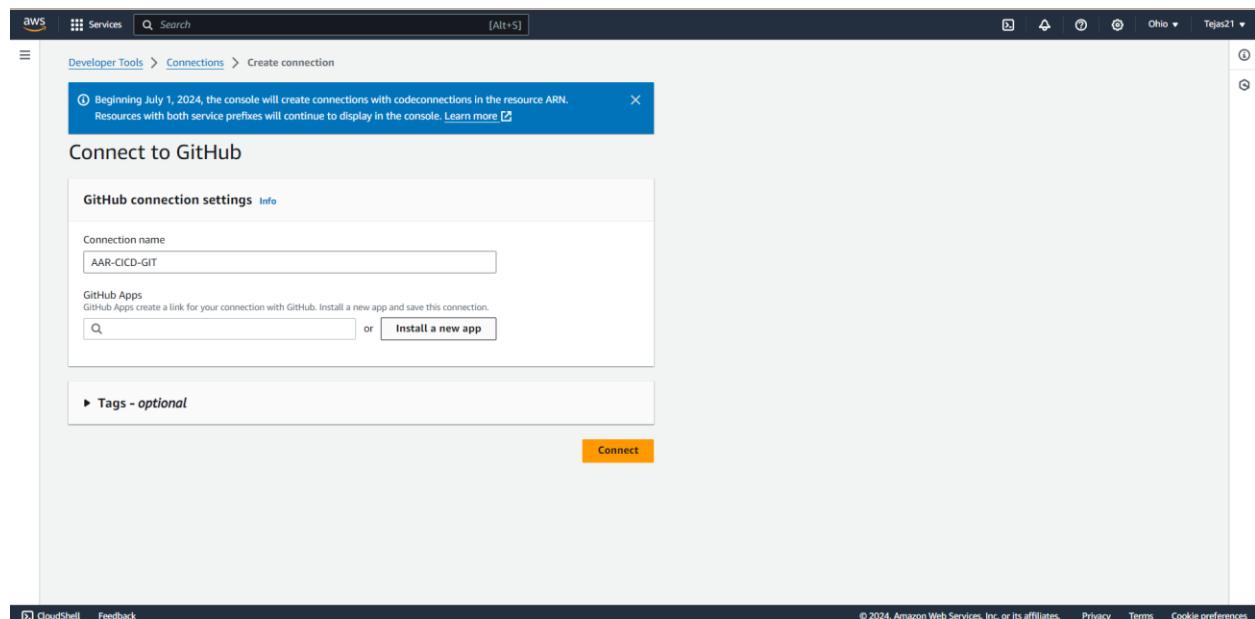
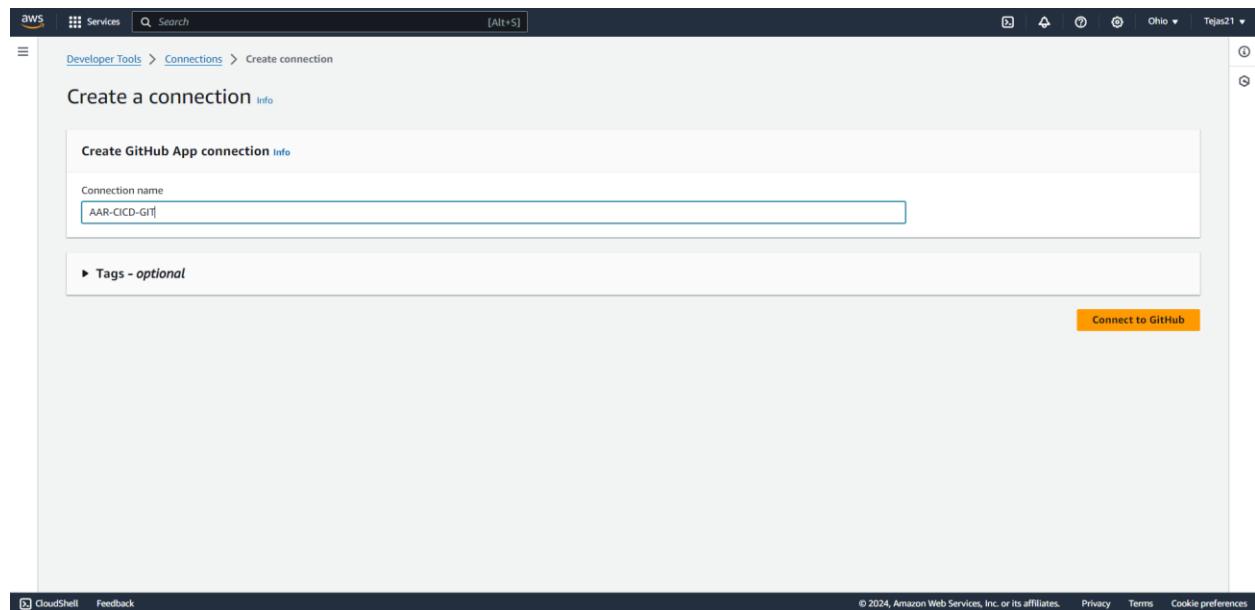
New GitHub version 2 (app-based) action: To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. [Learn more](#)

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

Repository name: Choose a repository in your GitHub account.

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

- Connecting and setting up GitHub for CI/CD deployment on AWS CodeDeploy.



Output artifact format
Choose the output artifact format.

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

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

Trigger

Trigger type
Choose the trigger type that starts your pipeline.

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

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

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

Event type
Choose the event type for the trigger that starts your pipeline.

Push

Pull request

Filter type
Choose the filter type for the event that starts your pipeline.

Branch

Tags

Branches
You can specify the target branch or branches you are pushing to. Use a comma to specify multiple entries.
Include

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose pipeline settings

Step 2 Add source stage

Step 3 **Add build stage**

Step 4 Add deploy stage

Step 5 Review

Add build stage Info Step 3 of 5

Build - optional

Build provider
This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.

CloudShell Feedback

Step 2 Add build stage

Step 3 **Add deploy stage**

Step 4 Review

Deploy

Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

AWS CodeDeploy

Region
US East (Ohio)

Input artifacts
Choose an input artifact for this action. [Learn more](#)

No more than 100 characters

Application name
Choose an application that you have already created in the AWS CodeDeploy console. Or create an application in the AWS CodeDeploy console and then return to this task.

AAR-CICD

Deployment group
Choose a deployment group that you have already created in the AWS CodeDeploy console. Or create a deployment group in the AWS CodeDeploy console and then return to this task.

AAR-CICD-DP

Configure automatic rollback on stage failure

Step 3: Add build stage

Build action provider

Build stage
No build

Step 4: Add deploy stage

Deploy action provider

Deploy action provider
AWS CodeDeploy

ApplicationName
AAR-CI_CD

DeploymentGroupName
AAR-CI_CD-DP

Configure automatic rollback on stage failure
Disabled

[Cancel](#) [Previous](#) [Create pipeline](#)

Success
Congratulations! The pipeline AAR-CI_CD-PIPELINE has been created.

[Create a notification rule for this pipeline](#)

AAR-CI_CD-PIPELINE

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded
Pipeline execution ID: 79f449a3-8c99-4865-acd6-4b44c7b71275

Source
GitHub (Version 2) [Edit](#)

Succeeded - Just now
[abfbba77d](#) [View details](#)

[Disable transition](#)

Deploy Failed
Pipeline execution ID: 79f449a3-8c99-4865-acd6-4b44c7b71275

[Start rollback](#) [Retry stage](#) [Retry failed actions](#)

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, and Elastic IPs.

The main content area displays an instance named "IMDSv2 Required". It shows the Instance ARN: arn:aws:ec2:us-east-2:010928184174:instance/i-06ee810b4baa25bd8. Below this, the "Security" tab is selected, showing the IAM Role (EC2CodeDeploy), Owner ID (010928184174), and Launch time (Sat Aug 10 2024 23:02:07 GMT+0530 (India Standard Time)).

The "Security details" section includes a table for Inbound rules and Outbound rules. The Inbound rules table has columns: Name, Security group rule ID, Port range, Protocol, Source, and Security groups. One rule is listed: sgr-0d969b954754bdd69, All, All, 0.0.0.0/0, sg-005e9d856101cf758, default. The Outbound rules table has similar columns and one rule listed: sgr-00b02179951be2472, All, All, 0.0.0.0/0, default.

The screenshot shows the "Edit inbound rules" interface for the security group "sg-005e9d856101cf758 - default". The title is "Edit inbound rules" with a "Info" link. A note says "Inbound rules control the incoming traffic that's allowed to reach the instance."

The main area is titled "Inbound rules" with an "Info" link. It lists three rules:

- sgr-0d969b954754bdd69: All traffic, All, All, Custom, Source: sg-005e9d856101cf758, Description: optional.
- : HTTP, TCP, 80, Anywhere..., Destination: 0.0.0.0/0, Description: optional.
- : SSH, TCP, 22, Anywhere..., Destination: 0.0.0.0/0, Description: optional.

Buttons at the bottom include "Add rule", "Cancel", "Preview changes", and "Save rules". A warning message at the bottom states: "⚠️ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only." The URL is https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#SecurityGroups:securityGroupDetails=sg-005e9d856101cf758.

The screenshot shows the "Most recent executions" page. It displays a single execution entry:

- Trigger:** CreatePipeline - root
- Pipeline execution ID:** 8a17e4dd
- Status:** Succeeded
- Last updated:** 6 days ago

A "Done" button is located at the bottom right of the card.

The screenshot shows the AWS CodePipeline console. On the left, a sidebar lists navigation options: Developer Tools (CodeCommit, CodeArtifact, CodeBuild, CodeDeploy), Pipeline (Pipeline, History, Settings, Go to resource, Feedback), and CloudShell/Feedback. The main area displays a pipeline execution with two stages: Source and Deploy. The Source stage is labeled "GitHub (Version 2)" and has a status of "Succeeded" with a timestamp of "2 minutes ago". The Deploy stage is labeled "AWS CodeDeploy" and also has a status of "Succeeded" with a timestamp of "1 minute 29s". Both stages have a "View details" button. A "Disable transition" button is located between the stages. A "Start rollback" button is on the right side of the Deploy stage. The pipeline ID is 80c71870-2c48-45fe-9614-614d5272d6a. The top right corner shows the user's name Tejas21 and location Ohio.

The screenshot shows a web browser window with a blue header bar. The address bar shows "Not secure 13.53.70.170". The main content area displays a large white text message: "Waaw, Congratulations Prabhakar you did it ...". Below this, smaller text reads: "This application was deployed using AWS CodeDeploy." and "For next steps, read the [AWS CodeDeploy Documentation](#)". The browser interface includes standard navigation buttons (back, forward, search) and a toolbar with icons for refresh, stop, and other functions.

- After committing changes in GitHub, the pipeline automatically starts deploying the updates, and the changes are applied without any manual intervention.

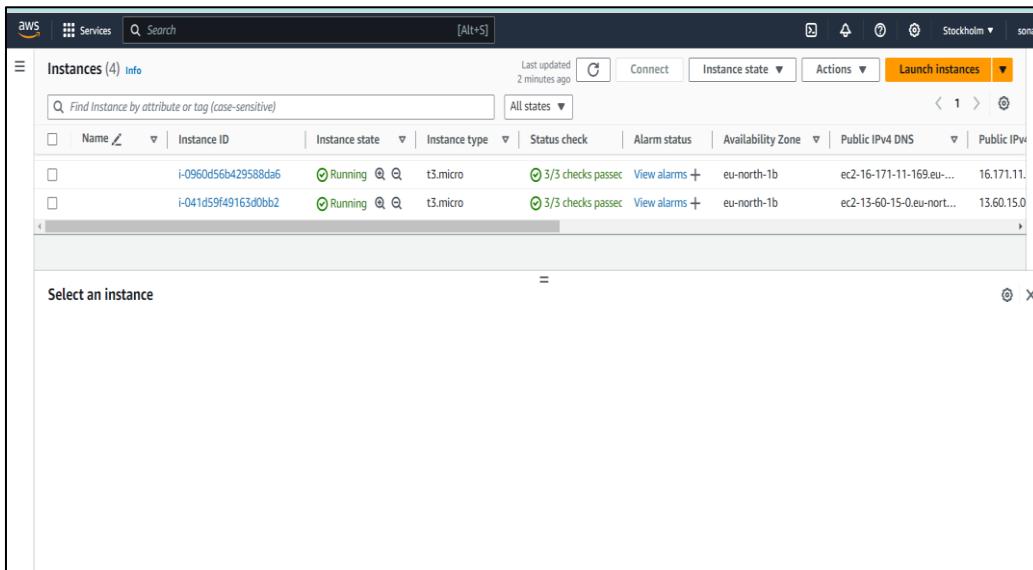
EXPERIMENT 3

Aim:

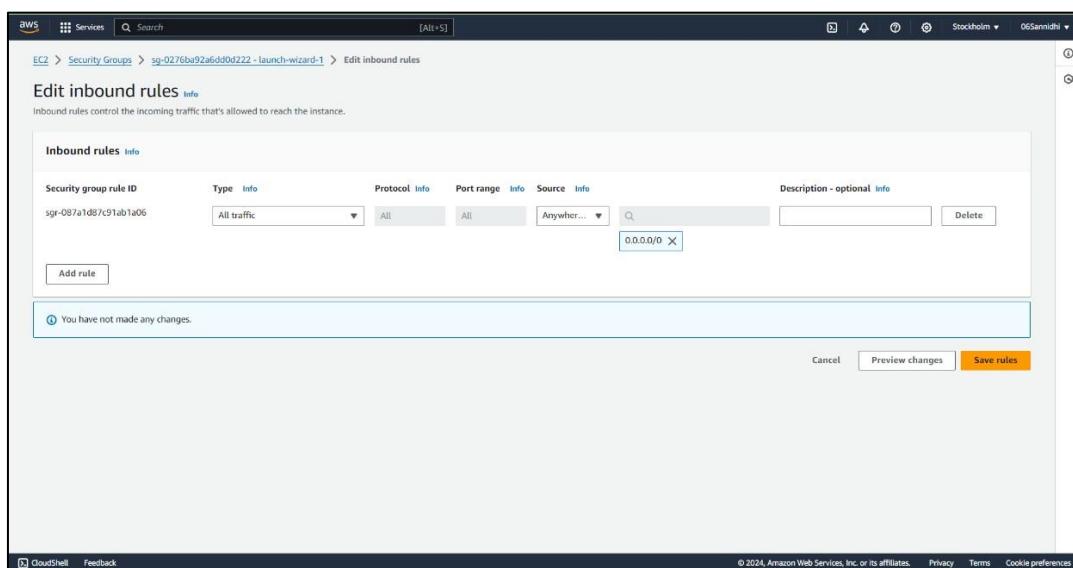
To understand the Kubernetes Cluster Architecture, install and Spin Up a KubernetesCluster on Linux Machines/Cloud Platforms.

Implementation

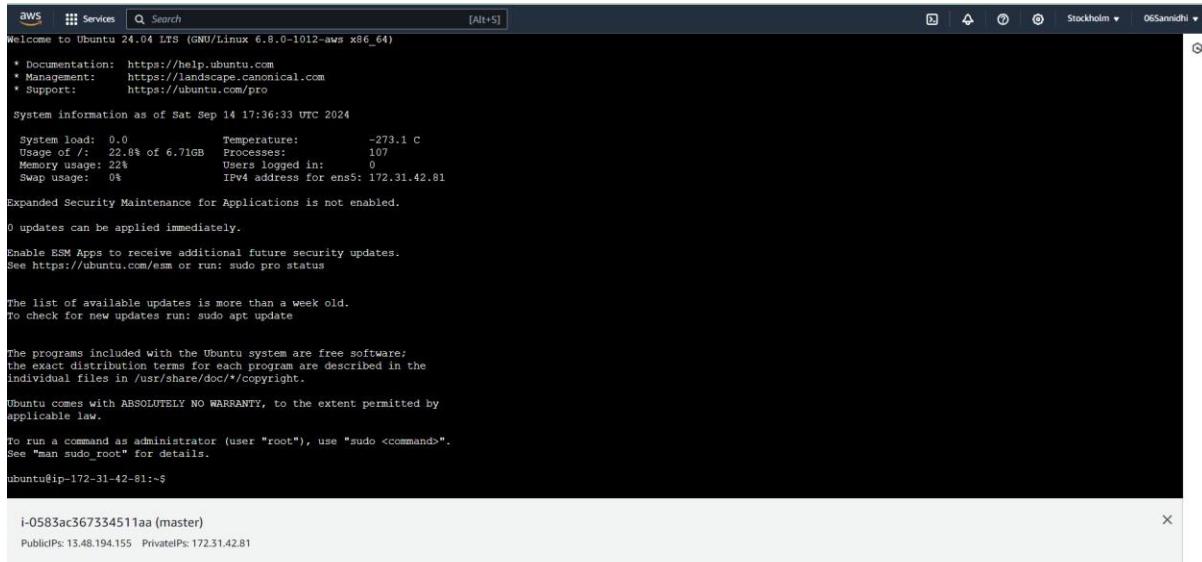
1) Create EC2 Ubuntu Instances on AWS. (Master and Worker)



2) Edit the Security Group Inbound Rules to allow SSH



3) AWS CLI for master and worker instances



```
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

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

System information as of Sat Sep 14 17:36:33 UTC 2024

System load: 0.0 Temperature: -273.1 C
Usage of /: 22.8% of 6.71GB Processes: 107
Memory usage: 22% Users logged in: 0
Swap usage: 0% IPv4 address for ens5: 172.31.42.81

Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

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

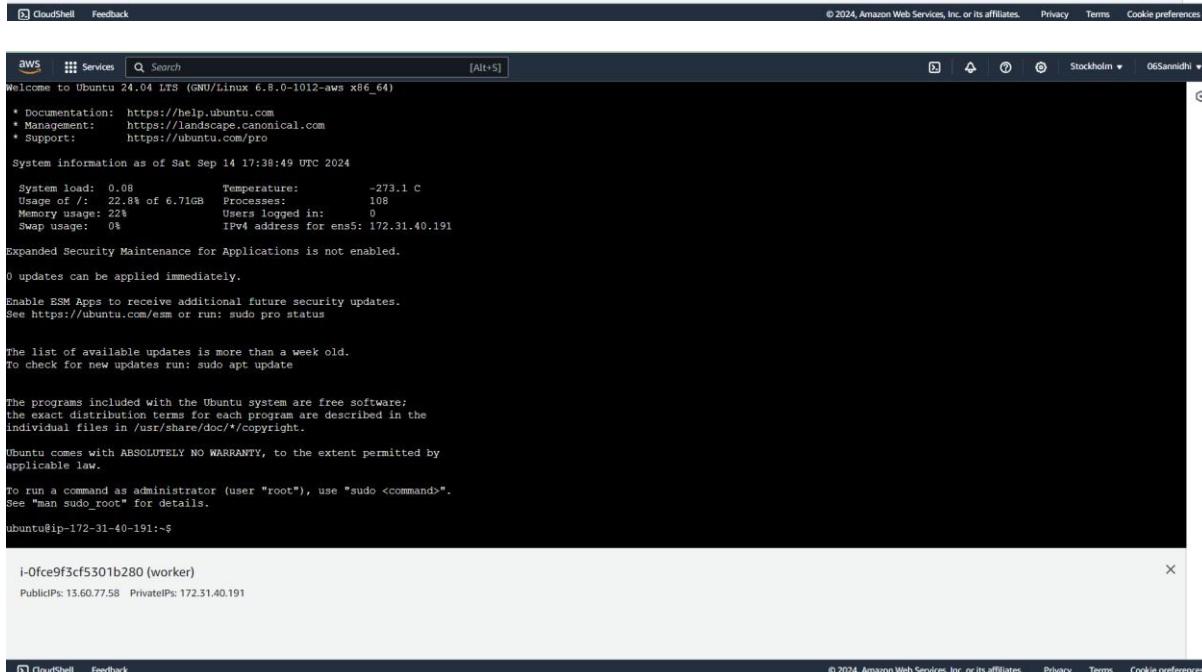
The list of available updates is more than a week old.
To check for new updates run: sudo apt update

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

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

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

ubuntu@ip-172-31-42-81:~$
```



```
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

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

System information as of Sat Sep 14 17:38:49 UTC 2024

System load: 0.08 Temperature: -273.1 C
Usage of /: 22.8% of 6.71GB Processes: 108
Memory usage: 22% Users logged in: 0
Swap usage: 0% IPv4 address for ens5: 172.31.40.191

Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

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

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

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

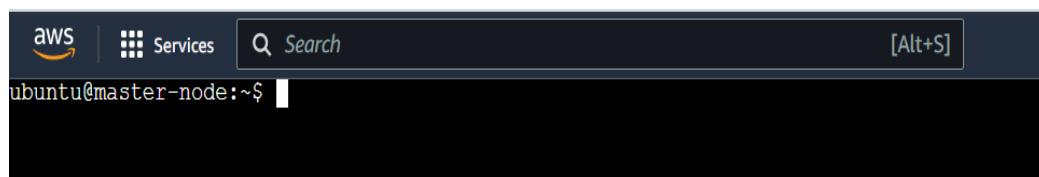
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

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

ubuntu@ip-172-31-40-191:~$
```

4) Assign Unique Hostname for Each Server Node

```
$ sudo hostnamectl set-hostname master-node
$ sudo hostnamectl set-hostname worker-1
```



```
ubuntu@master-node:~$
```

```
ubuntu@worker1:~$
```

Set up Docker (both master and worker)

5) Install Docker

\$ sudo apt-get update

```
ubuntu@master-node:~$ sudo apt-get update
Hit:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [377 kB]
Get:11 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:12 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:13 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:14 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [530 kB]
Get:15 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [128 kB]
Get:16 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [8352 B]
Get:17 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [372 kB]
Get:18 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [153 kB]
Get:19 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [45.0 kB]
Get:20 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [14.3 kB]
Get:21 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [353 kB]
Get:22 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [68.1 kB]
Get:23 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 c-n-f Metadata [424 B]
Get:24 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [14.4 kB]
```

\$ sudo apt-get install docker.io

```
ubuntu@master-node:~$ sudo apt-get install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
ifdown-ufw liblzo2-2 liblzo2-2:i386 liblzo2-2:amd64 liblzo2-2:armhf liblzo2-2:armv7hl
The following NEW packages will be installed:
bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 0 newly installed, 0 to remove and 139 not upgraded.
Need to get 76.8 MB of archives.
After this operation, 289 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 pigz amd64 2.8-1 [65.6 kB]
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 bridge-utils amd64 1.7.1-1ubuntu2 [33.9 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 runc amd64 1.1.12-0ubuntu3.1 [8599 kB]
Get:4 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 containerd amd64 1.7.12-0ubuntu4.1 [38.6 MB]
Get:5 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dns-root-data all 2023112702-willsync1 [4450 B]
Get:6 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dnsmasq-base amd64 2.90-2build2 [375 kB]
Get:7 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 docker.io amd64 24.0.7-0ubuntu4.1 [29.1 MB]
Get:8 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]
Fetched 76.8 MB in 1s (58.0 MB/s)
Preconfiguring packages...
Selecting previously unselected package pigz.
```

i-0960d56b429588da6 (Master)
PublicIPs: 16.171.11.169 PrivateIPs: 172.31.40.240

```
ubuntu@worker1:~$ sudo apt-get install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
ifdown-ufw liblzo2-2 liblzo2-2:i386 liblzo2-2:amd64 liblzo2-2:armhf liblzo2-2:armv7hl
The following NEW packages will be installed:
bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 0 newly installed, 0 to remove and 139 not upgraded.
Need to get 76.8 MB of archives.
After this operation, 289 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 pigz amd64 2.8-1 [65.6 kB]
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 bridge-utils amd64 1.7.1-1ubuntu2 [33.9 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 runc amd64 1.1.12-0ubuntu3.1 [8599 kB]
Get:4 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 containerd amd64 1.7.12-0ubuntu4.1 [38.6 MB]
Get:5 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dns-root-data all 2023112702-willsync1 [4450 B]
Get:6 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dnsmasq-base amd64 2.90-2build2 [375 kB]
Get:7 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/updeates/universe amd64 docker.io amd64 24.0.7-0ubuntu4.1 [29.1 MB]
Get:8 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]
Fetched 76.8 MB in 1s (67.1 MB/s)
Preconfiguring packages...
Selecting previously unselected package pigz.
```

i-041d59f49163d0bb2 (Worker-node)
PublicIPs: 13.60.15.0 PrivateIPs: 172.31.41.123

\$ docker —version

```

aws Services Search [Alt+S] Stockholm 06Sandhi
ubuntu@master-node:~$ docker --version
Command 'docker' not found, but can be installed with:
apt install docker
sudo apt install podman-docker # version 4.9-3ds1-1ubuntu0.1
sudo apt install docker.io # version 24.0.5-0ubuntu0.1
sudo apt install docker-compose additional versions.
ubuntu@master-node:~$ sudo apt-get update
[sudo] password for ubuntu:
Get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [126 kB]
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe arm64 Packages [10.0 MB]
Get:7 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe armhf Packages [3071 kB]
Get:8 http://security.ubuntu.com/ubuntu/public-security/main amd64 Packages [351 kB]
Get:9 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/public-security/main arm64 Packages [101 kB]
Get:10 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/public-security main amd64 Packages [249 kB]
Get:11 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble/multiverse translation-en [118 kB]
Get:12 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble/multiverse amd64 Components [60 kB]
Get:13 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble/multiverse amd64 c-n-f Metadata [8326 kB]
Get:14 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 Packages [502 kB]
Get:15 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble/main arm64 Packages [126 kB]
Get:16 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble/main armhf Packages [126 kB]
Get:17 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe amd64 Packages [366 kB]
Get:18 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe translation-en [90 kB]
Get:19 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe arm64 Packages [454 kB]
Get:20 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe armhf Packages [14.3 kB]
Get:21 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/restricted amd64 Packages [61.5 kB]
Get:22 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/restricted arm64 c-n-f Metadata [424 kB]
Get:23 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/restricted armhf Packages [10 kB]
Get:24 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe translation-en [3408 kB]
Get:25 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe amd64 Components [212 kB]
Get:26 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/universe amd64 c-n-f Metadata [532 kB]
Get:27 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/main amd64 Components [208 kB]
Get:28 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu/noble-backports/main amd64 components [208 kB]

i-0583ac567334511aa (master)
PublicIP: 13.48.194.155 PrivateIP: 172.31.42.81

```

6) Start and Enable Docker

```

$ sudo systemctl enable docker
$ sudo systemctl status docker
$ sudo systemctl start docker

```

```

aws Services Search [Alt+S] Stockholm
ubuntu@master-node:~$ sudo systemctl enable docker
ubuntu@master-node:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
     Active: active (running) since Thu 2024-09-19 15:32:18 UTC; 14min ago
TriggeredBy: • docker.socket
   Docs: https://docs.docker.com
 Main PID: 3938 (dockerd)
    Tasks: 9
   Memory: 63.1M (peak: 81.2M)
     CPU: 482ms
    CGroup: /system.slice/docker.service
            └─3938 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Sep 19 15:32:17 master-node systemd[1]: Starting docker.service - Docker Application Container Engine...
Sep 19 15:32:17 master-node dockerd[3938]: time="2024-09-19T15:32:17.923860899Z" level=info msg="Starting up"
Sep 19 15:32:17 master-node dockerd[3938]: time="2024-09-19T15:32:17.929761783Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-resolved, so using it"
Sep 19 15:32:18 master-node dockerd[3938]: time="2024-09-19T15:32:18.027582213Z" level=info msg="Loading containers: start."
Sep 19 15:32:18 master-node dockerd[3938]: time="2024-09-19T15:32:18.305044675Z" level=info msg="Loading containers: done."
Sep 19 15:32:18 master-node dockerd[3938]: time="2024-09-19T15:32:18.401360757Z" level=info msg="Docker daemon" commit="24.0.7-0ubuntu4.1 graphdriver=overlay2 version="24.0.7-0ubuntu4.1"
Sep 19 15:32:18 master-node dockerd[3938]: time="2024-09-19T15:32:18.402908962Z" level=info msg="Daemon has completed initialization"
Sep 19 15:32:18 master-node systemd[1]: Started docker.service - Docker Application Container Engine.
Sep 19 15:32:18 master-node dockerd[3938]: time="2024-09-19T15:32:18.453417910Z" level=info msg="API listen on /run/docker.sock"
lines 1-21/21 (END)

```

```

aws Services Search [Alt+S] Stockholm son
ubuntu@worker1:~$ sudo systemctl enable docker
ubuntu@worker1:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
     Active: active (running) since Thu 2024-09-19 15:32:43 UTC; 15min ago
TriggeredBy: • docker.socket
   Docs: https://docs.docker.com
 Main PID: 3488 (dockerd)
    Tasks: 9
   Memory: 32.1M (peak: 33.7M)
     CPU: 412ms
    CGroup: /system.slice/docker.service
            └─3488 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Sep 19 15:32:43 worker1 systemd[1]: Starting docker.service - Docker Application Container Engine...
Sep 19 15:32:43 worker1 dockerd[3488]: time="2024-09-19T15:32:43.160617326Z" level=info msg="Starting up"
Sep 19 15:32:43 worker1 dockerd[3488]: time="2024-09-19T15:32:43.161560743Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-resolved, so using it"
Sep 19 15:32:43 worker1 dockerd[3488]: time="2024-09-19T15:32:43.271723138Z" level=info msg="Loading containers: start."
Sep 19 15:32:43 worker1 dockerd[3488]: time="2024-09-19T15:32:43.519697388Z" level=info msg="Loading containers: done."
Sep 19 15:32:43 worker1 dockerd[3488]: time="2024-09-19T15:32:43.5857165402Z" level=info msg="Docker daemon" commit="24.0.7-0ubuntu4.1 graphdriver=overlay2 version="24.0.7-0ubuntu4.1"
Sep 19 15:32:43 worker1 dockerd[3488]: time="2024-09-19T15:32:43.588923202Z" level=info msg="Daemon has completed initialization"
Sep 19 15:32:43 worker1 dockerd[3488]: time="2024-09-19T15:32:43.633926974Z" level=info msg="API listen on /run/docker.sock"
Sep 19 15:32:43 worker1 systemd[1]: Started docker.service - Docker Application Container Engine.
lines 1-21/21 (END)

```

7) Install Kubernetes(both master and worker node)

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

```
ubuntu@master-node:~$ sudo apt-get update
Hit:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
ubuntu@master-node:~$ sudo apt-get install -y apt-transport-https ca-certificates curl

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
The following additional packages will be installed:
  libcurl3t64-gnutls libcurl4t64
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  curl libcurl3t64-gnutls libcurl4t64
3 upgraded, 0 newly installed, 0 to remove and 136 not upgraded.
Need to get 904 kB of archives.
After this operation, 38.9 kB of additional disk space will be used.
Get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 apt-transport-https all 2.7.14build2 [3974 B]
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 curl amd64 8.5.0-2ubuntu10.4 [227 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl4t64 amd64 8.5.0-2ubuntu10.4 [341 kB]
```

```
ubuntu@worker1:~$ sudo apt-get update
Hit:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
ubuntu@worker1:~$ sudo apt-get install -y apt-transport-https ca-certificates curl

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
The following additional packages will be installed:
  libcurl3t64-gnutls libcurl4t64
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  curl libcurl3t64-gnutls libcurl4t64
3 upgraded, 0 newly installed, 0 to remove and 136 not upgraded.
Need to get 904 kB of archives.
After this operation, 38.9 kB of additional disk space will be used.
Get:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 apt-transport-https all 2.7.14build2 [3974 B]
Get:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 curl amd64 8.5.0-2ubuntu10.4 [227 kB]
Get:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcurl4t64 amd64 8.5.0-2ubuntu10.4 [341 kB]
```

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

(download the google cloud public signing key)

(curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.31/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg)

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

(add the Kubernetes apt repository:)

(echo "deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.31/deb/" | sudo tee /etc/apt/sources.list.d/kubernetes.list)

```
ubuntu@worker1:~$ curl -fsSL https://pkgs.k8s.io/core/stable/v1.31/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg
ubuntu@worker1:~$ echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core/stable/v1.31/deb/' | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core/stable/v1.31/deb/
ubuntu@worker1:~$
```

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

\$ sudo apt-get update

\$ sudo apt-get install -y kubelet kubeadm kubectl

\$ sudo apt-mark hold kubelet kubeadm kubectl

```
ubuntu@master-node:~$ sudo apt-get update
Get:1 http://us-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Get:3 http://us-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:4 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb InRelease [1116 B]
Get:5 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb Packages [4065 B]
Fetched 0B in 0s (0B/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  cni-cni cri-tools kubernetes-cni
The following NEW packages will be installed:
  cni-cni cri-tools kubelet kubeadm kubectl kubernetes-cni
0 upgraded, 6 newly installed, 0 to remove and 130 not upgraded.
Need to get 87.4 MB of archives.
After this operation, 314 MB of additional disk space will be used.
Get:1 http://us-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 contrack amd64 1:1.4.8-1ubuntu1 [37.9 KB]
Get:2 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb cri-tools 1.31.1-1.1 [15.7 MB]
Get:3 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb kubelet 1.31.1-1.1 [114.0B]
Get:4 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb kubeadm 1.31.1-1.1 [11.2 MB]
Get:5 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb kubernetes-cni 1.5.1-1.1 [31.9 MB]
Get:6 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb kubectl 1.31.1-1.1 [15.2 MB]
Get:7 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb cni-cni 1.31.1-1.1 [1.0 MB]
Selecting previously unselected package contrack.
Unpacking database... 48112 files and directories currently installed.
Preparing to unpack .../0-contrack_1%{version}_1%{release}_amd64.deb ...
Unpacking contrack (1:1.4.8-1ubuntu1) ...
Selecting previously unselected package cri-tools.
Preparing to unpack .../1-cri-tools_1.31.1-1.1_amd64.deb ...
Unpacking cri-tools (1.31.1-1.1) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../2-kubelet_1.31.1-1.1_amd64.deb ...
Unpacking kubelet (1.31.1-1.1) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../3-kubeadm_1.31.1-1.1_amd64.deb ...
Unpacking kubeadm (1.31.1-1.1) ...
Selecting previously unselected package kubernetes-cni.
Preparing to unpack .../4-kubernetes-cni_1.5.1-1.1_amd64.deb ...
Unpacking kubernetes-cni (1.5.1-1.1) ...
Selecting previously unselected package kubectl.
Preparing to unpack .../5-kubectl_1.31.1-1.1_amd64.deb ...
Unpacking kubectl (1.31.1-1.1) ...
Setting up contrack (1:1.4.8-1ubuntu1) ...
Setting up cri-tools (1.31.1-1.1) ...
Setting up kubernetes-cni (1.5.1-1.1) ...
Setting up kubelet (1.31.1-1.1) ...
Setting up kubeadm (1.31.1-1.1) ...
Setting up kubectl (1.31.1-1.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...
Running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@master-node:~$ sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
ubuntu@master-node:~$
```

```
ubuntu@worker1:~$ sudo apt-get update
Get:1 http://us-north-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-north-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Get:3 http://us-north-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:4 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb InRelease [1116 B]
Get:5 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb Packages [4065 B]
Fetched 0B in 0s (0B/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  cni-cni cri-tools kubernetes-cni
The following NEW packages will be installed:
  cni-cni cri-tools kubelet kubeadm kubectl kubernetes-cni
0 upgraded, 6 newly installed, 0 to remove and 130 not upgraded.
Need to get 87.4 MB of archives.
After this operation, 314 MB of additional disk space will be used.
Get:1 http://us-north-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 contrack amd64 1:1.4.8-1ubuntu1 [37.9 KB]
Get:2 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb cri-tools 1.31.1-1.1 [15.7 MB]
Get:3 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb kubelet 1.31.1-1.1 [114.0B]
Get:4 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb kubeadm 1.31.1-1.1 [11.2 MB]
Get:5 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb kubernetes-cni 1.5.1-1.1 [31.9 MB]
Get:6 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb kubectl 1.31.1-1.1 [15.2 MB]
Get:7 https://prod-cdn.packages.k8s.io/repositories/issv/kubernetes/core/stable/v1.31/deb cni-cni 1.31.1-1.1 [1.0 MB]
Selecting previously unselected package contrack.
Unpacking database... 48112 files and directories currently installed.
Preparing to unpack .../0-contrack_1%{version}_1%{release}_amd64.deb ...
Unpacking contrack (1:1.4.8-1ubuntu1) ...
Selecting previously unselected package cri-tools.
Preparing to unpack .../1-cri-tools_1.31.1-1.1_amd64.deb ...
Unpacking cri-tools (1.31.1-1.1) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../2-kubelet_1.31.1-1.1_amd64.deb ...
Unpacking kubelet (1.31.1-1.1) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../3-kubeadm_1.31.1-1.1_amd64.deb ...
Unpacking kubeadm (1.31.1-1.1) ...
Selecting previously unselected package kubernetes-cni.
Preparing to unpack .../4-kubernetes-cni_1.5.1-1.1_amd64.deb ...
Unpacking kubernetes-cni (1.5.1-1.1) ...
Selecting previously unselected package kubectl.
Preparing to unpack .../5-kubectl_1.31.1-1.1_amd64.deb ...
Unpacking kubectl (1.31.1-1.1) ...
Setting up contrack (1:1.4.8-1ubuntu1) ...
Setting up cri-tools (1.31.1-1.1) ...
Setting up kubernetes-cni (1.5.1-1.1) ...
Setting up kubelet (1.31.1-1.1) ...
Setting up kubeadm (1.31.1-1.1) ...
Setting up kubectl (1.31.1-1.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...
Running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@worker1:~$ sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
ubuntu@worker1:~$
```

Kubernetes Deployment (master only)

- 8) Begin Kubernetes Deployment

```
$ sudo swapoff -a
```

- 9) Initialize Kubernetes on Master Node

```
$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16--ignore-preflight-errors=all
```

```
aws | Services | Search [Alt+S]
ubuntu@master-node:~$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16 --ignore-preflight-errors=all
[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
    [WARNING Mem]: the system RAM (914 MB) is less than the minimum 1700 MB
    [WARNING FileExisting-socat]: socat not found in system path
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your internet connection
[preflight] You can also perform this action beforehand using 'kubeadm config images pull'
W0919 18:23:42.736017 7569 checks.go:846] detected that the sandbox image "registry.k8s.io/pause:3.8" of the container runtime is inconsistent with that used by kubeadm. It is recommended to use "registry.k8s.io/pause:3.10" as the CRI sandbox image.
[certs] Using certificatebir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [kubernetes kubernetes.default kubernetes.default.svc kubernetes.default.svc.cluster.local master-node] and IPs [10.96.0.1 172.31.40.240]
[certs] Generating "apiserver-kubelet-client" certificate and key
[certs] Generating "front-proxy-ca" certificate and key
[certs] Generating "front-proxy-client" certificate and key
[certs] Generating "etcd/ca" certificate and key
[certs] Generating "etcd/server" certificate and key
[certs] etcd/server serving cert is signed for DNS names [localhost master-node] and IPs [172.31.40.240 127.0.0.1 ::1]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [localhost master-node] and IPs [172.31.40.240 127.0.0.1 ::1]
[certs] Generating "etcd/healthcheck-client" certificate and key
[certs] Generating "apiserver-etcd-client" certificate and key
```

- 10) Deploy Pod Network to Cluster

```
$ mkdir -p $HOME/.kube
```

```
$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
```

```
$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
$ kubectl apply -f
```

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

```
kubeadm join 172.31.40.240:6443 --token i0zoaj.tblkx57b8mg41aq3 \
    --discovery-token-ca-cert-hash sha256:b66cf6a507714d87b3012ab879b7af89f0d484df29bd6bcc7808e713a1c52fa
ubuntu@master-node:~$ mkdir -p $HOME/.kube
ubuntu@master-node:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
ubuntu@master-node:~$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
ubuntu@master-node:~$ ^C
ubuntu@master-node:~$ kubectl apply -f https://github.com/flannel-io/flannel/releases/latest/download/kube-flannel.yml
namespace/kube-flannel created
serviceaccount/flannel created
clusterrole.rbac.authorization.k8s.io/flannel created
clusterrolebinding.rbac.authorization.k8s.io/flannel created
configmap/kube-flannel-cfg created
daemonset.apps/kube-flannel-ds created
ubuntu@master-node:~$
```

```
$ kubectl get pods --all-namespaces
```

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
kube-flannel	kube-flannel-ds-gmnqm	1/1	Running	0	4m57s
kube-system	coredns-7c65d6fc9-bb6x4	1/1	Running	0	15m
kube-system	coredns-7c65d6fc9-zfsvw	1/1	Running	0	15m
kube-system	etcd-master-node	1/1	Running	0	15m
kube-system	kube-apiserver-master-node	1/1	Running	0	15m
kube-system	kube-controller-manager-master-node	1/1	Running	0	15m
kube-system	kube-proxy-k2ksj	0/1	CrashLoopBackOff	6 (2m40s ago)	15m
kube-system	kube-scheduler-master-node	1/1	Running	0	15m

Join Worker Node to Cluster (on worker node)

11) sudo kubeadm join 172.31.40.240:6443 --token i0zoaj.tblkx57b8mg41aq3 --discovery-token-ca-cert-hash sha256:b66cf6a507714d87b3012ab879b7af89f0d484df29bd6bccc7808e713a1c52fa – ignore-preflight-errors=all

```
ubuntu@worker1:~$ sudo kubeadm join 172.31.40.240:6443 --token i0zoaj.tblkx57b8mg41aq3 --discovery-token-ca-cert-hash sha256:b66cf6a507714d87b3012ab879b7af89f0d484df29bd6bccc7808e713a1c52fa --ignore-preflight-errors=all
[preflight] Running pre-flight checks
[WARNING FileExisting-socat]: socat not found in system path
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-check] Waiting for a healthy kubelet at http://127.0.0.1:10240/healthz. This can take up to 4m0s
[kubelet-check] The kubelet is healthy after 502.220002ms
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap

This node has joined the cluster:
* Certificate signing request was sent to apiserver and a response was received.
* The Kubelet was informed of the new secure connection details.

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

12) \$ kubectl get nodes (on master node)



NAME	STATUS	ROLES	AGE	VERSION
master-node	Ready	control-plane	39m	v1.31.1
worker1	Ready	<none>	9m44s	v1.31.1

we now have a Kubernetes cluster running across AWS EC2 Instances. This cluster can be used to further deploy applications and their loads being distributed across these machines.

Conclusion

Successfully understood the Kubernetes cluster architecture and deployed a Kubernetes cluster on Linux machines/cloud platforms, demonstrating seamless setup and orchestration.

EXPERIMENT 4

Aim

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

Implementation

- 1) In your Kubernetes cluster check if all nodes are connected.

```
$ kubectl get nodes
```

```
ubuntu@master-node:~$ kubectl get nodes
NAME     STATUS   ROLES      AGE    VERSION
master-node   Ready    control-plane   54m   v1.31.1
worker1       Ready    <none>    36m   v1.31.1
ubuntu@master-node:~$
```

- 2) Create deploy.yaml file

```
$ sudo nano deploy.yaml
```

- 3) Copy nginx-deployment.yaml file from Kubernetes site and paste in above file.

- 4) Create deployment

```
$ kubectl apply -f
```

- 5) Check if deployment is created and all pods are running

```
$ kubectl get deploy.po
```

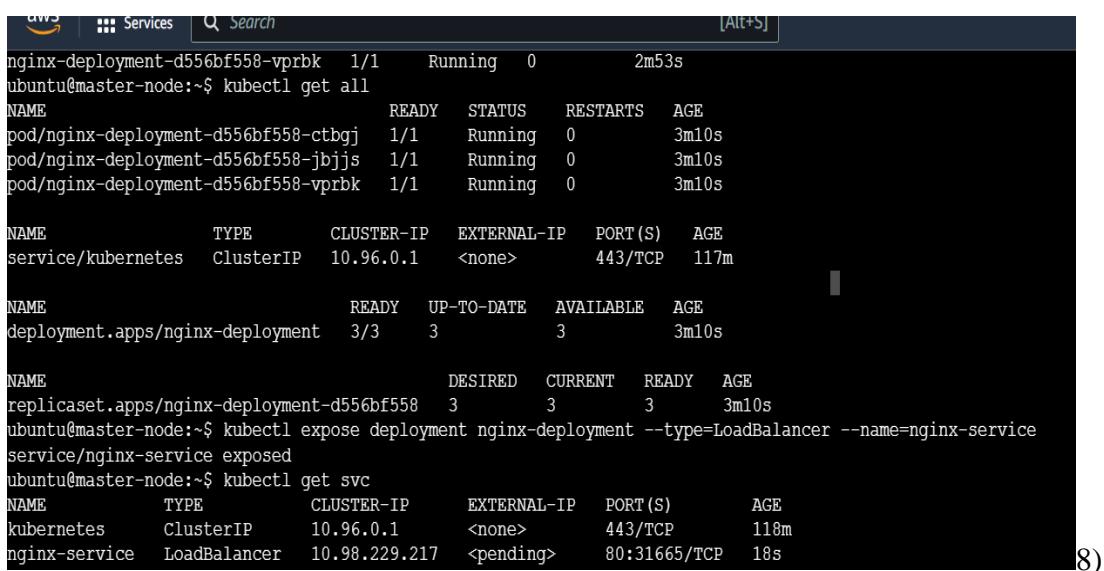
- 6) Expose the app to internet using load balancer service.

```
$ kubectl expose deployment.apps/nginx-deployment --type= "load balancer"
```

- 7) Kubectl get svc

```
i-0960d56b429588da6 (Master)
PublicIPs: 16.171.11.169 PrivateIPs: 172.31.40.240
ubuntu@master-node:~$ sudo nano deploy.yaml
ubuntu@master-node:~$ kubectl create -f deploy.yaml
error: the path "deploy.yaml" does not exist
ubuntu@master-node:~$ sudo nano deploy.yaml
ubuntu@master-node:~$ kubectl create -f deploy.yaml
deployment.apps/nginx-deployment created
ubuntu@master-node:~$ kubectl get deploy
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   3/3     3           3          68s
ubuntu@master-node:~$ kubectl expose deployment.apps/nginx-deployment --type= "load balancer"
error: there is no need to specify a resource type as a separate argument when passing arguments in resource/name form (e.g. 'kubectl get resource/<resource_name>' instead of 'kubectl get resource resource/<resource_name>')
ubuntu@master-node:~$ kubectl get deployments
kubectl get deployments
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   3/3     3           3          2m53s
NAME        READY   STATUS    RESTARTS   AGE
nginx-deployment-d556bf558-ctbgj   1/1     Running   0          2m53s
nginx-deployment-d556bf558-jbjjs   1/1     Running   0          2m53s
nginx-deployment-d556bf558-vprbk   1/1     Running   0          2m53s
ubuntu@master-node:~$ kubectl get all
NAME        READY   STATUS    RESTARTS   AGE
pod/nginx-deployment-d556bf558-ctbgj   1/1     Running   0          3m10s
pod/nginx-deployment-d556bf558-jbjjs   1/1     Running   0          3m10s
pod/nginx-deployment-d556bf558-vprbk   1/1     Running   0          3m10s
ubuntu@master-node:~$
```

```
GNU nano 7.2                                     deploy.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
          ports:
            - containerPort: 80
[ Read 21 lines ]
^G Help     ^O Write Out   ^W Where Is   ^X Cut       ^I Execute   ^C Location   M-U Undo   M-A Set Mark   M-[ To Bracket   M-Q Previous
X Exit     ^R Read File   ^N Replace   ^U Paste     ^J Justify   M-/ Go To Line M-Z Redo   M-B Copy     M-Q Where Was   M-T Next
```



```
aws | Services | Search [Alt+S]
nginx-deployment-d556bf558-vprbk 1/1 Running 0 2m53s
ubuntu@master-node:~$ kubectl get all
NAME                      READY   STATUS    RESTARTS   AGE
pod/nginx-deployment-d556bf558-ctbgj 1/1     Running   0          3m10s
pod/nginx-deployment-d556bf558-jbjjs 1/1     Running   0          3m10s
pod/nginx-deployment-d556bf558-vprbk 1/1     Running   0          3m10s

NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP  10.96.0.1   <none>        443/TCP  117m

NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx-deployment  3/3     3           3          3m10s

NAME           DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-deployment-d556bf558  3        3        3        3m10s
ubuntu@master-node:~$ kubectl expose deployment nginx-deployment --type=LoadBalancer --name=nginx-service
service/nginx-service exposed
ubuntu@master-node:~$ kubectl get svc
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP  10.96.0.1   <none>        443/TCP  118m
nginx-service LoadBalancer  10.98.229.217 <pending>     80:31665/TCP 18s
```

8)

8) deployed our Nginx server on our EC2 instance.



Conclusion

Successfully installed Kubectl and executed commands to manage the Kubernetes cluster, deploying the first application seamlessly.

EXPERIMENT 05

ROLL NO.	09	
Class:	DISHA	Date _____ Page _____

Experiment : 05

Aim:- To understand Terraform lifecycle, core concepts / terminologies and install it on a Linux machine and Windows.

Theory :-

Terraform is an Infrastructure as code (IaC) tool developed by HashiCorp that allows users to define and manage cloud and on-premises resources using declarative configuration files. Understanding its lifecycle and core concepts is essential for effectively utilizing Terraform in managing infrastructure.

Terraform Lifecycle

Terraform manages resources through a defined lifecycle that includes three primary stages:

- 1. **Apply (Create)**: This stage involves creating the resources as defined in the configuration files. Terraform communicates with the cloud provider's API to provision the resource.
- 2. **Update** : In this stage, Terraform makes modifications to existing resources.. This can involve incremental changes or complete replacements, depending on the nature of the update.

Teacher's Sign.: _____

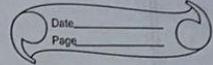
3. **destroy** :- This final stage removes the resources from the environment, ensuring that it is no longer managed by Terraform.

Terraform also provides a lifecycle meta-argument that allows users to control these stages more granularly. Options within this meta-argument include:-

- **create - before destroy** :- Creates a new instance is created before the old one is destroyed, which is useful for maintaining service availability during updates.
 - **prevent - destroy** :- Prevents accidental deletion of critical resources.
 - **ignore - changes** :- Allows Terraform to ignore changes made outside of its management for specified fields or entire objects.
- * **Core concepts**.

Several key concepts underpin Terraform's functionality

- **Configuration files** :- These files, written in HashiCorp configuration language (HCL), define the desired state of the infrastructure.



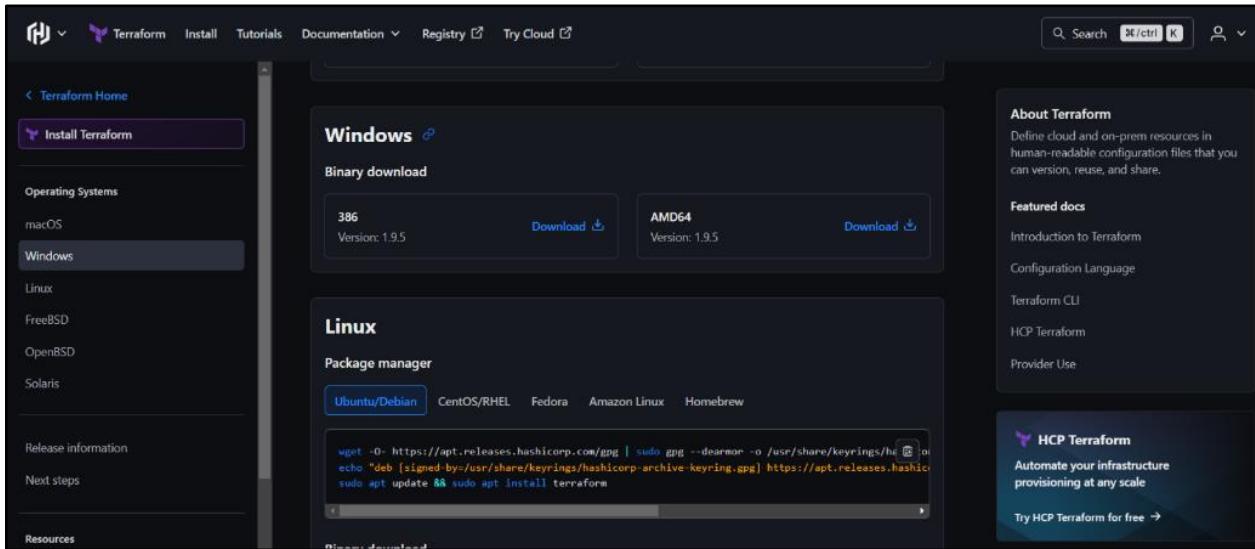
- Resources:- The fundamental building blocks in Terraform, representing various infrastructure components like virtual machines, storage accounts, and networking configurations.
- Data sources:- These provide external data to Terraform configurations, allowing users to reference existing resources or configurations from other objects.
- Modules:- Modules are reusable packages of Terraform configurations that enable users to group related resources for better organization and reusability.

Conclusion:-

By mastering these concepts and understanding the lifecycle of resources, users can effectively leverage Terraform to automate and manage their infrastructure in a safe and efficient manner.

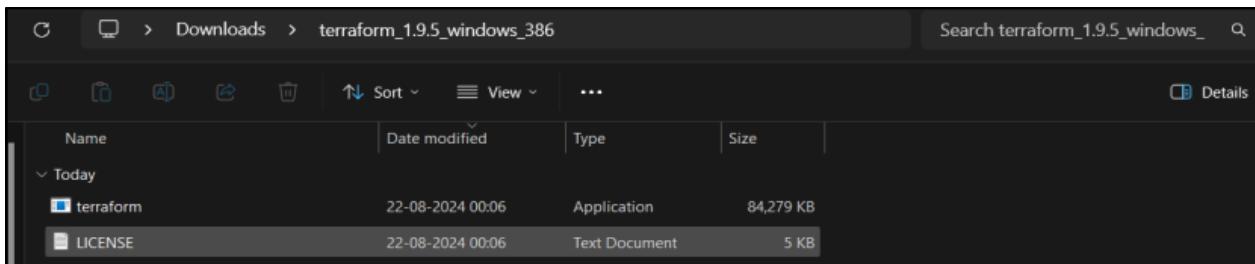
Implementation:

Step 1: Go to Terraform website and download 386 .



The screenshot shows the Terraform website's download section. On the left, a sidebar lists operating systems: macOS, Windows (selected), Linux, FreeBSD, OpenBSD, Solaris, Release information, Next steps, and Resources. The main content area is titled "Windows" and shows two download options: "386 Version: 1.9.5" and "AMD64 Version: 1.9.5". Below this is a section for "Linux" with a "Package manager" tab selected, showing links for Ubuntu/Debian, CentOS/RHEL, Fedora, Amazon Linux, and Homebrew. A terminal window displays the command to install Terraform via apt. To the right, there's an "About Terraform" section with a brief description and links to "Featured docs" like Introduction to Terraform, Configuration Language, Terraform CLI, HCP Terraform, and Provider Use. There's also a "HCP Terraform" section with a "Try HCP Terraform for free" button.

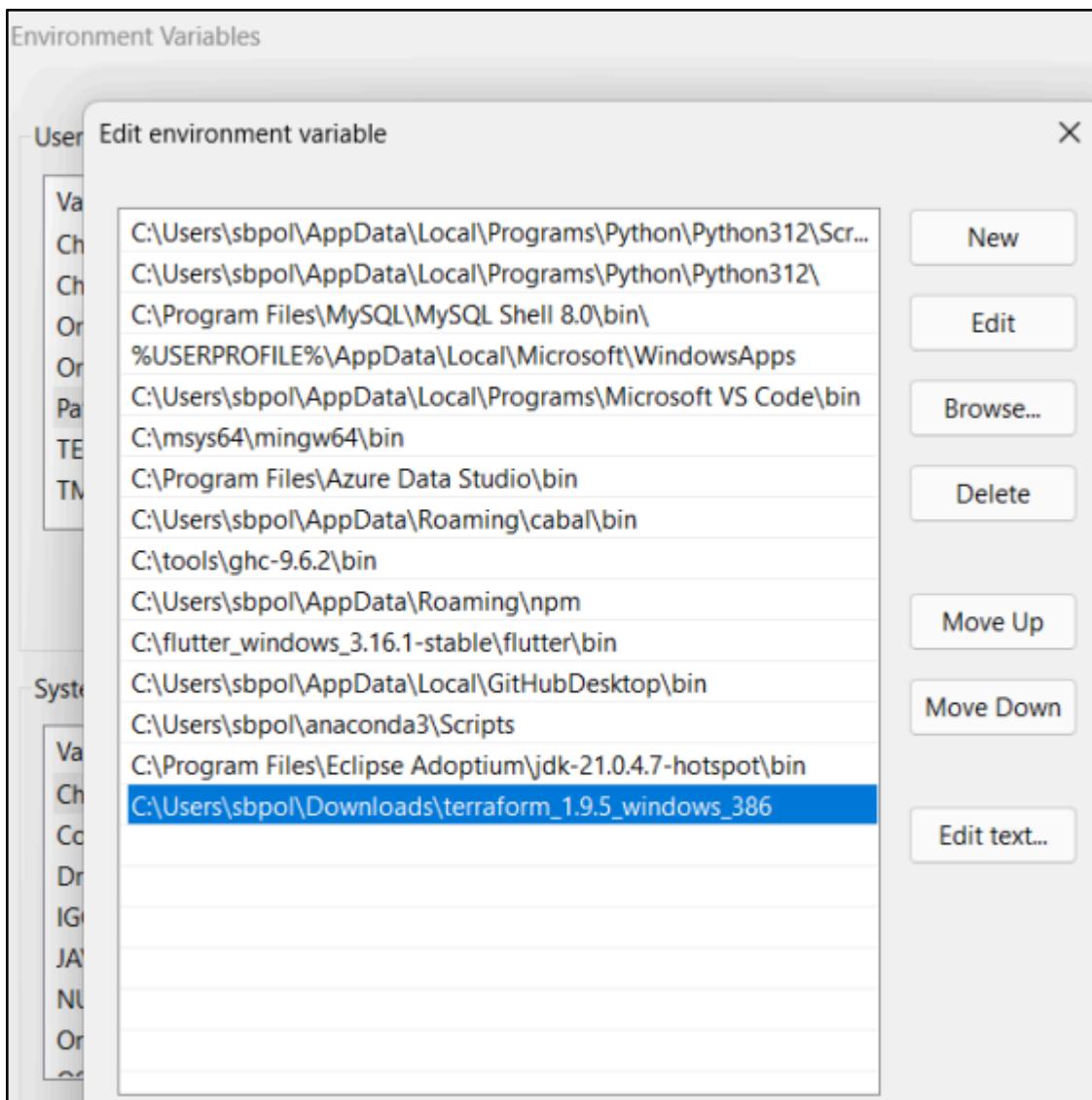
Step 2 : Extract the zip file and copy the path of the file.



The screenshot shows a file explorer window with the path "Downloads > terraform_1.9.5_windows_386". The contents of the folder are listed in a table:

Name	Date modified	Type	Size
terraform	22-08-2024 00:06	Application	84,279 KB
LICENSE	22-08-2024 00:06	Text Document	5 KB

Step 3: Edit environment variables and paste copied path.



Step 4 : Make sure Terraform is installed successfully.

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

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

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

All other commands:
  console    Try Terraform expressions at an interactive command prompt
  fmt        Reformat your configuration in the standard style
  force-unlock Release a stuck lock on the current workspace
  get        Install or upgrade remote Terraform modules
  graph      Generate a Graphviz graph of the steps in an operation
  import     Associate existing infrastructure with a Terraform resource
  login      Obtain and save credentials for a remote host
  logout     Remove locally-stored credentials for a remote host
  metadata   Metadata related commands
  output     Show output values from your root module
  providers  Show the providers required for this configuration
  refresh    Update the state to match remote systems
  show       Show the current state or a saved plan
  state     Advanced state management
```

```
Install the latest PowerShell for new features and improved security.

Loading personal and system profiles took 1476ms.
(base) PS C:\Users\sbpol> terraform --version
Terraform v1.9.5
on windows_386
```

EXPERIMENTS 06

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

Part A: Creating docker image using terraform

Step 1:Check Docker functionality

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

C:\Users\student>docker

Usage: docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

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

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

Check for the docker version with the following command.

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

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

Step 2:

Creating a new folder named ‘Docker’ in the ‘Terraform Scripts’ folder.

Creating a new docker.tf file using Atom editor and write the following contents into.

This will create a Ubuntu Linux container

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

Step 3: Execute Terraform Init command to initialize the resources

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

Terraform has been successfully initialized!

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

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

Step 4: Execute Terraform plan to see the available resources

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

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

Terraform will perform the following actions:

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

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

+ healthcheck (known after apply)

+ labels (known after apply)
}

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

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

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

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

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

Terraform will perform the following actions:

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

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

+ healthcheck (known after apply)

+ labels (known after apply)
}

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

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

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

Enter a value: yes

SONAM CHHABAIDIYA D15A 09

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

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

Before Executing Apply step:

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

After Executing Apply step:

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

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

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

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

Terraform will perform the following actions:

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

SONAM CHHABAIDIYA D15A 09

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

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

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

Do you really want to destroy all resources?

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

Enter a value: yes

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

Destroy complete! Resources: 2 destroyed.

Docker images After Executing Destroy step

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

ADVANCE DEVOPS EXP-7

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

Step-1: Open up Jenkins Dashboard on localhost, port 8080 or whichever port it is atfor you.

The screenshot shows the Jenkins dashboard with the following details:

- Left Sidebar:** Includes links for New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, My Views, and Restart Safely.
- Build Queue:** Shows 1 idle node and 2 idle nodes.
- Build Executor Status:** Shows 1 Built-In Node with 1 idle node.
- Central View:** A table of build jobs:

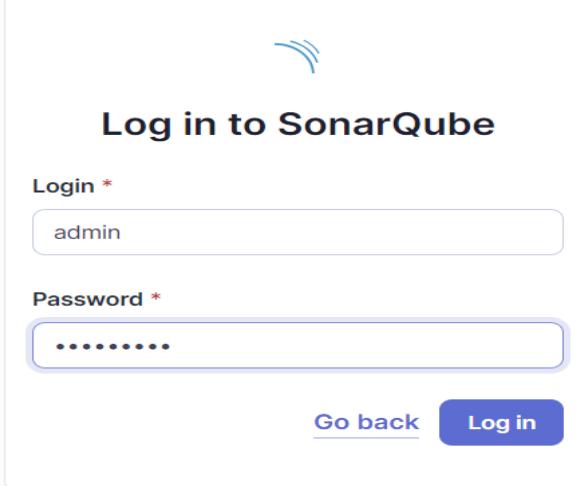
S	W	Name	Last Success	Last Failure	Last Duration
✓	☀️	first-job	14 sec #7407	10 days #5719	0.3 sec
✓	☁️	Git Test	5 days 9 hr #5	5 days 9 hr #4	1.9 sec
✓	☁️	MavenBuild	5 days 6 hr #10	5 days 6 hr #9	54 sec
✓	☀️	Scripted_Pipeline	5 days 9 hr #1	N/A	4.9 sec
✓	☀️	second-job	14 sec #8420	N/A	0.32 sec
✓	☀️	sonarqube	7 min 0 sec #1	N/A	1 min 12 sec
✓	☀️	third-job	14 sec #8889	10 days #6530	0.27 sec

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

```
C:\Windows\System32>docker -v
Docker version 27.2.0, build 3ab4256

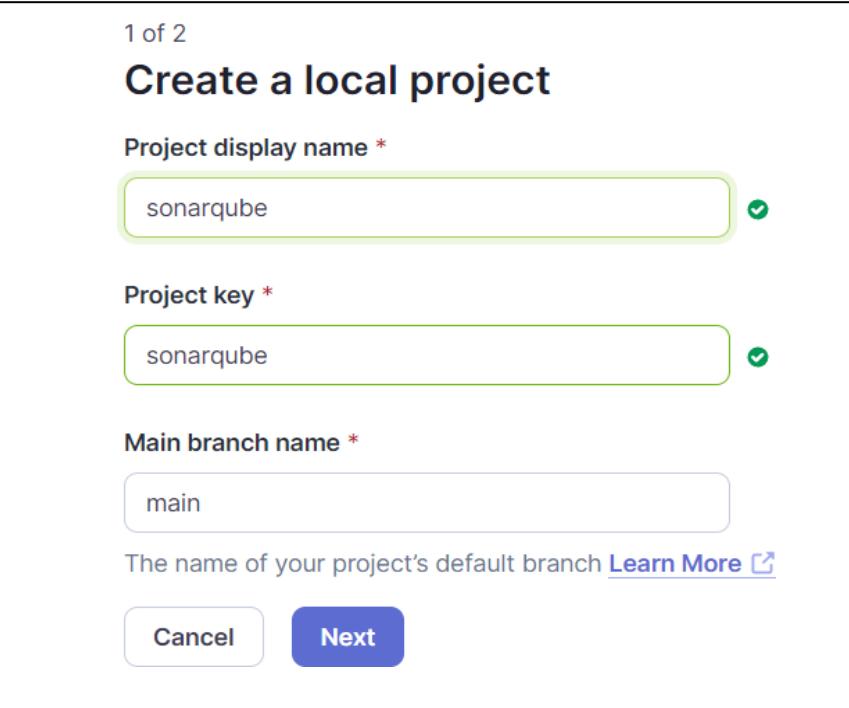
C:\Windows\System32>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
Unable to find image 'sonarqube:latest' locally
latest: Pulling from library/sonarqube
7478e0ac0f23: Pull complete
90a925ab929a: Pull complete
7d9a34308537: Pull complete
80338217a4ab: Pull complete
1a5fd5c7e184: Pull complete
7b87d6fa783d: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Downloaded newer image for sonarqube:latest
1b0713c1db8319472cb04c7bb7753025f7e75b97c484e85bab3c5161d8416f0
```

Step-3: Once the container is up and running, you can check the status of SonarQube at localhost port 9000. The login id is “admin” and the password is also “admin”.



The screenshot shows the SonarQube login interface. At the top is the Sonar logo, followed by a light blue decorative graphic resembling a stylized wave or sound waves. Below this is the heading "Log in to SonarQube". There are two input fields: "Login *" containing "admin" and "Password *" containing a series of dots. At the bottom are two buttons: "Go back" and a larger "Log in" button.

Step-4: Create a local project in SonarQube with the name sonarqube



1 of 2

Create a local project

Project display name *

Project key *

Main branch name *

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

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

2 of 2

Set up project for Clean as You Code

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

Choose the baseline for new code for this project

Use the global setting

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

Define a specific setting for this project

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

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

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

[Back](#) [Create project](#)

Step-5: Setup the project and come back to Jenkins Dashboard. Go to Manage Jenkins → Plugins and search for SonarQube Scanner in Available Plugins and install it.

The screenshot shows the Jenkins Manage Jenkins > Plugins interface. A search bar at the top contains the text "sonarqube". On the left, there's a sidebar with options: Updates (23), Available plugins (selected), Installed plugins, and Advanced settings. The main area displays a table for the SonarQube Scanner plugin. The table has columns for Install, Name (SonarQube Scanner 2.17.2), and Released (7 mo 13 days ago). Below the table, a note says: "This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality." There are "Install" and "Uninstall" buttons at the top right of the table row.

Step-6: Under 'Manage Jenkins → System', look for SonarQube Servers and enter these details. Name : sonarqube, Server URL : http://localhost:9000

Sonam chhabaidiya D15A 09

SonarQube servers

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

Environment variables

SonarQube installations

List of SonarQube installations

Name	sonarqube
Server URL	Default is http://localhost:9000
<input type="text" value="http://localhost:9000"/>	
Server authentication token	SonarQube authentication token. Mandatory when anonymous access is disabled.
<input type="text" value="- none -"/>	
+ Add ▾	
Advanced ▾	

Step-7: Search for SonarQube Scanner under Global Tool Configuration. Choose the latest configuration and choose Install automatically. Manage Jenkins → Tools → SonarQube Scanner Installation.

SonarQube Scanner installations

Add SonarQube Scanner

SonarQube Scanner

Name

sonarqube

Install automatically ?

Install from Maven Central

Version

SonarQube Scanner 6.2.0.4584

Add Installer ▾

Add SonarQube Scanner

Step-8: After the configuration, create a New Item in Jenkins, choose a freestyle project named sonarqube.

New Item

Enter an item name
sonarqube

Select an item type

-  **Freestyle project**
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
-  **Maven project**
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
-  **Pipeline**
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
-  **Multi-configuration project**
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
-  **Folder**
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
-  **Multibranch Pipeline**
Creates a set of Pipeline projects according to detected branches in one SCM repository.
-  **Organization Folder**
Creates a set of multibranch project subfolders by scanning for repositories

OK

Step-9: Choose this GitHub repository in Source Code Management.

https://github.com/shazforiot/MSBuild_firstproject.git. It is a sample hello-world project with no vulnerabilities and issues, just to test the integration.

Source Code Management

Configure

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

None

Git

Repositories

Repository URL ?

Credentials ?

- none -

+ Add *

Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

Add Branch

Repository browser ?

(Auto)

Save Apply

The screenshot shows the Jenkins configuration interface for a build job. Under the 'Source Code Management' section, the 'Git' option is selected. The 'Repository URL' is set to 'https://github.com/shazforiot/MSBuild_firstproject.git'. The 'Branches to build' field contains '*/master'. The 'Repository browser' is set to '(Auto)'. There are 'Save' and 'Apply' buttons at the bottom.

Step-10: Under Build-> Execute SonarQube Scanner, enter these Analysis Properties. Mention the SonarQube Project Key, Login, Password, Source path and Host URL.

sonar.projectKey=sonarqube

sonar.login=admin

sonar.password=Sona@2004

sonar.sources=.

sonar.host.url=http://localhost:9000

Sonam chhabaidiya D15A 09

Build Steps

Execute SonarQube Scanner

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

Path to project properties ?
sonar.projectKey=sonarqube
sonar.login=admin
sonar.password=Sona@2004
sonar.sources=.
sonar.host.url=http://localhost:9000

Analysis properties ?
Additional arguments ?
JVM Options ?

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

Global Permissions

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

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

4 of 4 shown

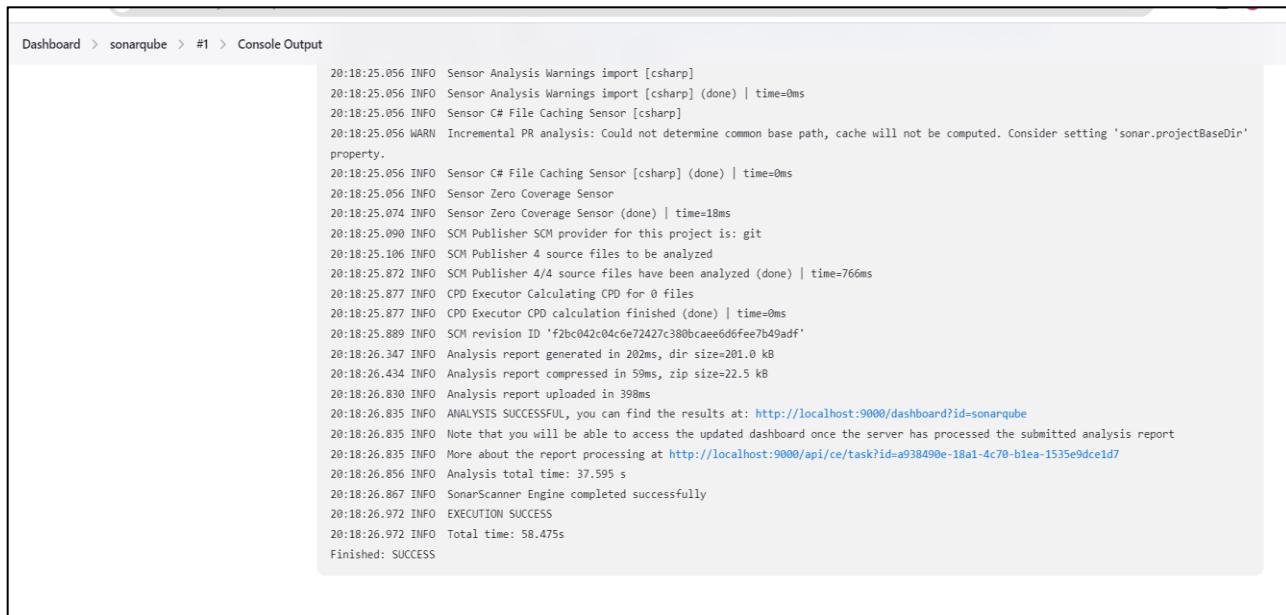
Sonam chhabaidiya D15A 09

Step-12: Run The Build and check the console output.

The screenshot shows the Jenkins dashboard for the 'sonarqube' project. On the left, there's a sidebar with options like Status, Changes, Workspace, Build Now, Configure, Delete Project, SonarQube, and Rename. The main area displays the project name 'sonarqube' with a green checkmark icon. It includes a 'SonarQube' logo and a 'Permalinks' section with a bulleted list of recent builds. Below that is a 'Build History' section with a table showing one build (#1) from Sep 29, 2024, at 8:17 PM. At the bottom right of the main area, there are links for 'Atom feed for all' and 'Atom feed for failures'.

The screenshot shows the Jenkins 'Console Output' page for build #1 of the 'sonarqube' project. The sidebar on the left has options for Status, Changes, Console Output (which is selected and highlighted in blue), Edit Build Information, Delete build '#1', Timings, and Git Build Data. The main content area is titled 'Console Output' with a green checkmark icon. It shows the build log starting with 'Started by user sonam chhabaidiya'. The log details the build process, including cloning the repository from GitHub, fetching upstream changes, and unpacking the SonarScanner CLI. The log concludes with the command 'Dsonar.host.url=http://localhost:9000 -Dsonar.projectKey=sonarqube -Dsonar.login=admin -Dsonar.host.url=http://localhost:9000 -Dsonar.sources=. -Dsonar.password=Sona@2004 -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\jenkins\workspace\sonarqube'.

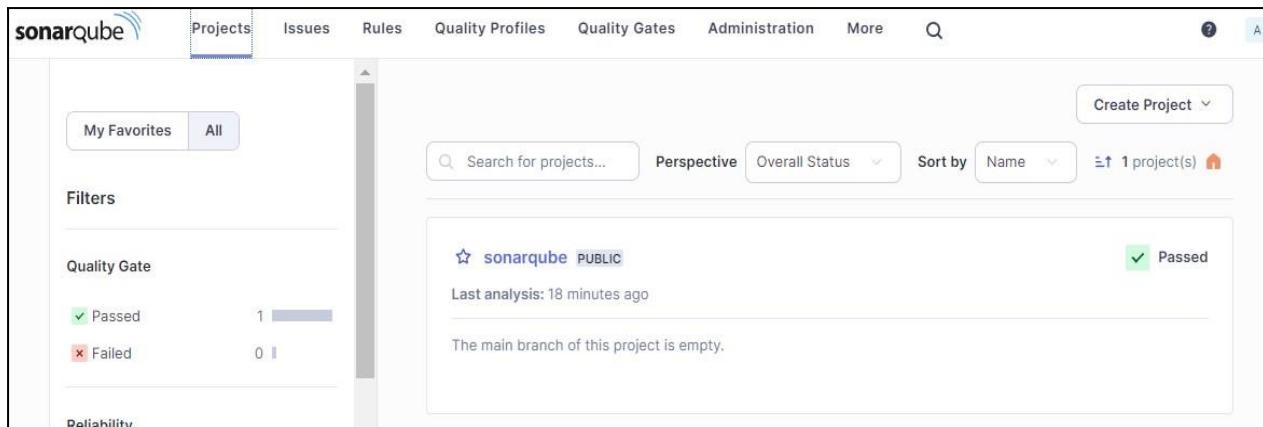
Sonam chhabaidiya D15A 09



The screenshot shows the SonarQube interface with the path "Dashboard > sonarqube > #1 > Console Output". The console output window displays a log of analysis results:

```
20:18:25.056 INFO Sensor Analysis Warnings import [csharp]
20:18:25.056 INFO Sensor Analysis Warnings import [csharp] (done) | time=0ms
20:18:25.056 INFO Sensor C# File Caching Sensor [csharp]
20:18:25.056 WARN Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir' property.
20:18:25.056 INFO Sensor C# File Caching Sensor [csharp] (done) | time=0ms
20:18:25.056 INFO Sensor Zero Coverage Sensor
20:18:25.074 INFO Sensor Zero Coverage Sensor (done) | time=18ms
20:18:25.090 INFO SCM Publisher SCM provider for this project is: git
20:18:25.106 INFO SCM Publisher 4 source files to be analyzed
20:18:25.872 INFO SCM Publisher 4/4 source files have been analyzed (done) | time=766ms
20:18:25.877 INFO CPD Executor Calculating CPD for 0 files
20:18:25.877 INFO CPD Executor CPD calculation finished (done) | time=0ms
20:18:25.889 INFO SCM revision ID 'f2bc042c04c6e72427c380bcae66dfee7b49adf'
20:18:26.347 INFO Analysis report generated in 202ms, dir size=201.0 kB
20:18:26.434 INFO Analysis report compressed in 59ms, zip size=22.5 kB
20:18:26.830 INFO Analysis report uploaded in 398ms
20:18:26.835 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarqube
20:18:26.835 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
20:18:26.835 INFO More about the report processing at http://localhost:9000/api/ce/task?id=a938490e-18a1-4c70-b1ea-1535e9dce1d7
20:18:26.856 INFO Analysis total time: 37.595 s
20:18:26.867 INFO SonarScanner Engine completed successfully
20:18:26.972 INFO EXECUTION SUCCESS
20:18:26.972 INFO Total time: 58.475s
Finished: SUCCESS
```

Step-13: Once the build is complete, check the project in SonarQube.



The screenshot shows the SonarQube interface with the "Projects" tab selected. The main area displays the "sonarqube" project, which is marked as "PUBLIC" and "Passed". The project status is shown as "Last analysis: 18 minutes ago" and "The main branch of this project is empty." The sidebar includes sections for "My Favorites" and "All" projects, and various filters like "Quality Gate" and "Reliability".

Sonam chhabaidiya D15A 09

The screenshot shows the SonarQube dashboard for the 'main' project. At the top, there's a navigation bar with links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. Below the navigation is a breadcrumb trail: sonarqube / main. The main content area has tabs for Overview, Issues, Security Hotspots, Measures, Code, and Activity, with 'Overview' being the active tab. On the right side of the header, there are Project Settings and Project Information dropdowns. The main content area displays the following information:

- Quality Gate:** Passed (green checkmark icon)
- Last analysis:** 3 minutes ago
- New Code:** Overall Code (selected tab)
- Security:** 0 Open issues (severity breakdown: 0 H, 0 M, 0 L)
- Reliability:** 0 Open issues (severity breakdown: 0 H, 0 M, 0 L)
- Maintainability:** 0 Open issues (severity breakdown: 0 H, 0 M, 0 L)
- Accepted issues:** 0
- Coverage:** (indicated by a question mark icon)
- Duplications:** 0.0%

A yellow warning box at the top states: "⚠️ The last analysis has warnings. See details".

ADVANCE DEVOPS EXPERIMENT 08

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

Step 1: Log in to sonarqube portal and create a local project.

The screenshot shows the 'Create a local project' step in the SonarQube interface. It includes fields for Project display name (sonarqube-pipeline), Project key (sonarqube-pipeline), and Main branch name (main). A 'Next' button is visible at the bottom.

The screenshot shows the 'Set up project for Clean as You Code' step in the SonarQube interface. It provides options for choosing a baseline for new code: 'Use the global setting' (selected) or 'Define a specific setting for this project'. Under 'Use the global setting', it shows 'Previous version' as the option chosen. A note states: 'Any code that has changed since the previous version is considered new code. Recommended for projects following regular versions or releases.' A 'Next' button is visible at the bottom right.

Step 2: Go to [download sonarscanner](#) to download sonar scanner

Sonam chhabaidiya D15A 09

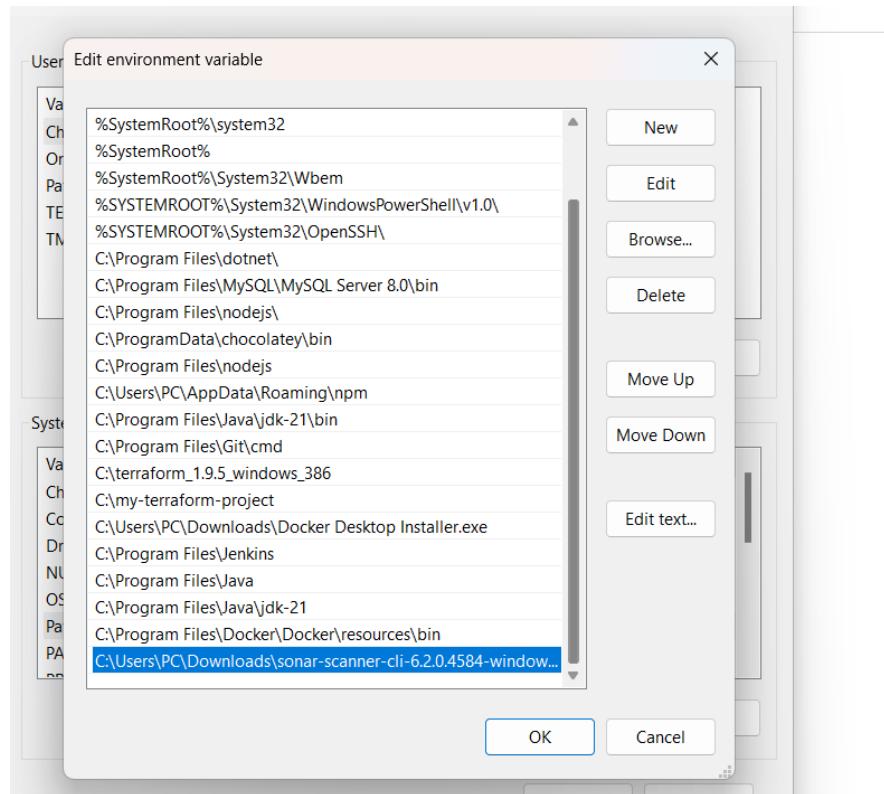
The screenshot shows the SonarScanner CLI documentation page on the SonarQube website. It features a sidebar with navigation links for SonarQube and SonarScanner, and a main content area with three release entries:

- 6.2** (2024-09-17): Support PKCS12 truststore generated with OpenSSL. Download scanner for: Linux x64, Linux AArch64, Windows x64, macOS x64, macOS AArch64, Docker Any (Requires a pre-installed JVM). **Windows x64** is highlighted with a red box.
- 6.1** (2024-06-27): macOS and Linux AArch64 distributions. Download scanner for: Linux x64, Linux AArch64, **Windows x64**, macOS x64, macOS AArch64, Docker Any (Requires a pre-installed JVM). **Windows x64** is highlighted with a red box.
- 6.0** (2024-06-04): New bootstrapping mechanism and JRE provisioning with SonarQube 10.6+ and SonarCloud. Download scanner for: Linux x64, Windows x64, macOS x64, Docker, Any (Requires a pre-installed JVM).

A sidebar on the right lists "On this page" links: Configuring your project, Running SonarScanner CLI from the zip file, Running SonarScanner CLI from the Docker image, Scanning C, C++, or Objective-C projects, Sample projects, Alternatives to sonar-project.properties, Alternate analysis directory, Advanced configuration, and Troubleshooting.

After the download is complete, extract the file and copy the path to bin folder Go to environment variables, system variables and click on path

Add a new path, paste the path copied earlier.



Step 3: Create a New Item in Jenkins, choose Pipeline.

Dashboard > All > New Item

New Item

Enter an item name
sonarqube-pipeline

Select an item type

- Freestyle project**
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
- Maven project**
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
- Pipeline**
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Multi-configuration project**
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
- Folder**
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

OK

localhost:8080/job/sonarqube-pipeline/configure

Dashboard > sonarqube-pipeline > Configuration

Configure

Pipeline

Definition
Pipeline script

Script ?
try sample Pipeline...

```
1 // Use SonarQube environment
2 withSonarQubeEnv('sonarqube') {
3     // Run the SonarScanner for code analysis
4     bat """
5         C:/Users/varya/Downloads/sonar-scanner-clt-6.1.0.4477-windows-x64/sonar-scanner-6.1.0.4477-windows-x64/bin/sonar-sca
6         -Dsonar.login=admin ^
7         -Dsonar.password=Sonar@2004 ^
8         -Dsonar.projectKey=sonarqube-test1 ^
9         -Dsonar.exclusions=vendor/**,resources/**,java ^
10        -Dsonar.host.url=http://127.0.0.1:9000
11    }
12 }
```

Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

Step 4: Save the pipeline and build it.

The screenshot shows the Jenkins Pipeline 'sonarqube-pipeline' dashboard. On the left, there's a sidebar with various options like Status, Changes, Build Now, Configure, Delete Pipeline, Full Stage View, SonarQube, Stages, Rename, and Pipeline Syntax. The main area is titled 'Stage View' and shows two stages: 'Cloning the GitHub Repo' (3s) and 'SonarQube Analysis' (30s). Below the stages, a timeline shows two builds: #2 (Sep 29, 21:05) and #1 (Sep 29, 21:01). Build #2 is successful ('1min 0s'), while build #1 failed ('1s'). The 'Build History' section at the bottom lists the last four builds. A 'Permalinks' section provides links to the latest builds.

Build	Date	Time	Status	Duration
#2	Sep 29	21:05	Success	1min 0s
#1	Sep 29	21:01	Failed	1s
Last build (#2)	Sep 29, 2024	9:05 PM	Success	1 min 18 sec ago
Last stable build (#2)	Sep 29, 2024	9:05 PM	Success	1 min 18 sec ago
Last successful build (#2)	Sep 29, 2024	9:05 PM	Success	1 min 18 sec ago
Last failed build (#1)	Sep 29, 2024	9:05 PM	Failed	5 min 20 sec ago

Console output:

The screenshot shows the Jenkins Pipeline 'sonarqube-pipeline' build #2 console output. The sidebar includes options like Status, Changes, Console Output (which is selected), Edit Build Information, Delete build '#2', Timings, Git Build Data, Pipeline Overview, Pipeline Console, Restart from Stage, Replay, Pipeline Steps, Workspaces, and Previous Build. The main area is titled 'Console Output' and displays the command-line logs for the build. The logs show the pipeline starting, cloning the GitHub repository, and fetching changes from the remote Git repository.

```

Started by user sonam chhabaidiya
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\ProgramData\Jenkins\.jenkins\workspace\sonarqube-pipeline
[Pipeline] {
[Pipeline] stage
[Pipeline] {
  (Cloning the GitHub Repo)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
> C:\Program Files\Git\bin\git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\sonarqube-pipeline\.git # timeout=10
Fetching changes from the remote Git repository
> C:\Program Files\Git\bin\git.exe config remote.origin.url https://github.com/shazforiot/MSBuild_firstproject.git # timeout=10
Fetching upstream changes from https://github.com/shazforiot/MSBuild_firstproject.git
> C:\Program Files\Git\bin\git.exe --version # timeout=10
> git --version # 'git' version 2.46.0.windows.1'
> C:\Program Files\Git\bin\git.exe fetch --tags --force --progress -- https://github.com/shazforiot/MSBuild_firstproject.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> C:\Program Files\Git\bin\git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision f2bc042c04c6e72427c380bcae6d6fee7b49adf (refs/remotes/origin/master)
> C:\Program Files\Git\bin\git.exe config core.sparsecheckout # timeout=10
> C:\Program Files\Git\bin\git.exe checkout -f f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
> C:\Program Files\Git\bin\git.exe branch -a -v --no-abbrev # timeout=10

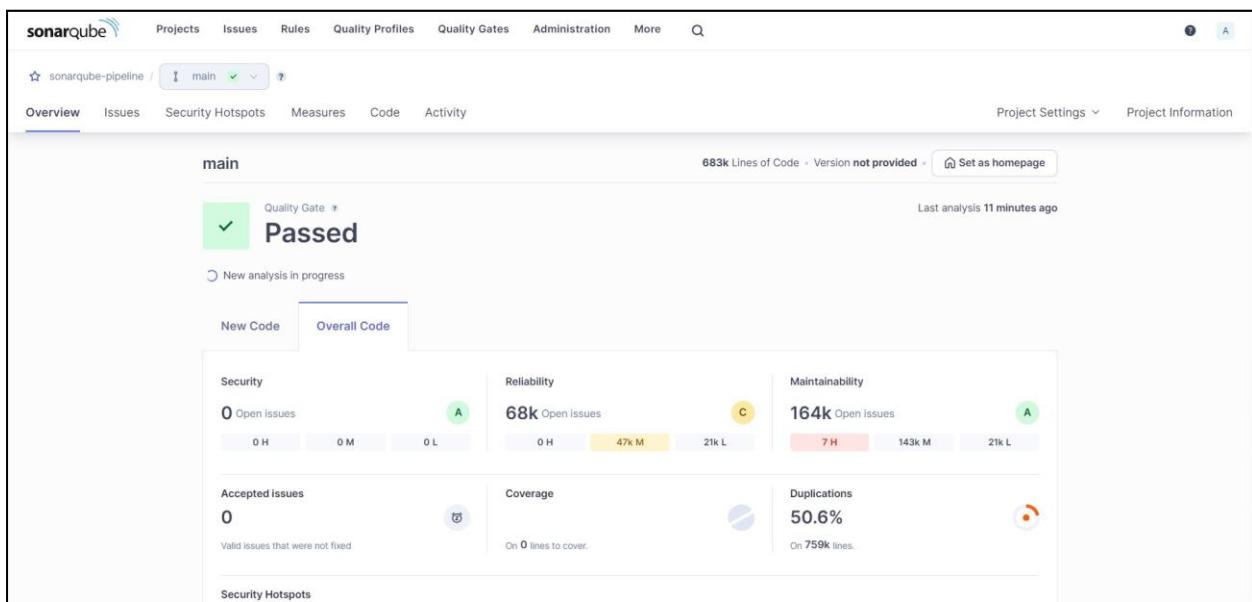
```

Sonam chhabaidiya D15A 09

```
Dashboard > sonarqube-pipeline > #2

21:06:33.871 INFO Sensor Zero Coverage Sensor (done) | time=19ms
21:06:33.884 INFO SCM Publisher SCM provider for this project is: git
21:06:33.884 INFO SCM Publisher 4 source files to be analyzed
21:06:34.583 INFO SCM Publisher 4/4 source files have been analyzed (done) | time=699ms
21:06:34.592 INFO CPD Executor Calculating CPD for 0 files
21:06:34.592 INFO CPD Executor CPD calculation finished (done) | time=0ms
21:06:34.609 INFO SCM revision ID 'f2bc042c04c6e72427c380bcaee6d6fee7b49ad'
21:06:35.023 INFO Analysis report generated in 194ms, dir size=199.9 kB
21:06:35.126 INFO Analysis report compressed in 74ms, zip size=22.5 kB
21:06:36.435 INFO Analysis report uploaded in 1307ms
21:06:36.437 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://127.0.0.1:9000/dashboard?id=sonarqube-test1
21:06:36.441 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
21:06:36.441 INFO More about the report processing at http://127.0.0.1:9000/api/ce/task?id=b707dbf1-7b05-4f30-9d7d-962c0f708a55
21:06:36.461 INFO Analysis total time: 46.199 s
21:06:36.466 INFO SonarScanner Engine completed successfully
21:06:36.543 INFO EXECUTION SUCCESS
21:06:36.543 INFO Total time: 58.331s
[Pipeline]
[Pipeline] // withSonarQubeEnv
[Pipeline]
[Pipeline] // stage
[Pipeline]
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Step 5: After that, check the project in SonarQube



Under different tabs, check all different issues with the code.

Sonam chhabaidiya D15A 09

The screenshot displays two views of the SonarQube interface for the project 'sonarqube-pipeline'.

Measures Tab: This view provides an overview of various code quality metrics. The sidebar shows Reliability, Maintainability, Security Review, Duplications, Size, and Complexity. The main panel shows the overall code status: Open Issues (210,549), Confirmed Issues (0), Accepted Issues (0), and False Positive Issues (0). A detailed list of open issues is shown for the 'gameoflife' module, including gameoflife-acceptance-tests (4 issues), gameoflife-build (0 issues), gameoflife-core (603 issues), gameoflife-deploy (0 issues), and gameoflife-web (209,940 issues). A file named pom.xml has 2 issues.

Issues Tab: This view lists specific code issues. It includes a 'Filters' section with 'My Issues' and 'All' options, and a 'Bulk Change' button. The main area shows a list of issues categorized by severity and type. One issue is highlighted: "Insert a <!DOCTYPE> declaration to before this <html> tag." (Reliability, Consistency, user-experience) with 197k effort. Other listed issues include "Remove this deprecated "width" attribute." (Maintainability, Consistency, html5 obsolete), "Remove this deprecated "align" attribute." (Maintainability, Consistency, html5 obsolete), and another "Remove this deprecated "align" attribute." (Maintainability, Consistency, html5 obsolete).

Sonam chhabaidiya D15A 09

The screenshot shows the SonarQube interface for the project 'sonarqube-pipeline'. The 'Issues' tab is selected. On the left, a sidebar displays 'My Issues' and 'All' issues. A 'Filters' section is open, showing 'Issues in new code' and a expanded 'Clean Code Attribute' section. The 'Intentionality' filter is selected, showing 14k issues. Other filters include 'Consistency' (197k), 'Adaptability' (0), and 'Responsibility' (0). Below this is a 'Software Quality' section with 'Security' (0), 'Reliability' (14k), and 'Maintainability' (15). On the right, a list of issues is shown under the file 'gameoflife-acceptance-tests/Dockerfile'. One issue is highlighted with a blue border, labeled 'Use a specific version tag for the image.' It has a 'Maintainability' status and is marked as 'Not assigned'. A modal window titled 'Introducing Clean Code Attributes' is displayed, explaining that Clean Code attributes are characteristics your code must have to be considered Clean Code. It shows 1 of 5 items, with a 'Next' button. The top right of the screen shows '13,887 issues' and '59d effort'.

This screenshot shows the same SonarQube interface for the 'sonarqube-pipeline' project. The 'Issues' tab is selected. The sidebar and filters are identical to the first screenshot. On the right, a list of issues is shown under 'gameoflife-core/build/reports/tests/all-tests.html'. Two issues are highlighted with blue borders. The first is 'Add "lang" and/or "xml:lang" attributes to this "<html>" element' under 'Reliability' with 'Intentionality' status and 'wcag2-a' accessibility. The second is 'Add "<th>" headers to this "<table>".' under 'Reliability' with 'Intentionality' status and 'wcag2-a' accessibility. Both issues are marked as 'Not assigned'. The top right shows '13,872 issues' and '59d effort'.

Sonam chhabaidiya D15A 09

The screenshot shows the SonarQube interface for the project 'sonarqube-pipeline' under the 'main' branch. The 'Issues' tab is selected. On the left, there are several filter panels:

- Filters**: Includes 'Issues in new code' and a 'Clean Code Attribute' section for Intentionality (15), Consistency (164k), Adaptability (0), and Responsibility (0). It also has sections for Software Quality (Security 0, Reliability 14k, Maintainability 15).
- Code Smells**: A list of 15 issues found in 'gameoflife-acceptance-tests/Dockerfile'. Each item includes a checkbox, the issue title, its priority (Intentionality), and a detailed description.

At the top right, it says '15 issues' and '44min effort'.

This screenshot is similar to the one above, but the 'Severity' filter is expanded. The 'Type' section shows 15 Code Smells, 0 Bugs, and 0 Vulnerabilities. Other collapsed filters include Scope, Status, Security Category, and Creation Date.

The right side displays the same list of 15 code smell issues from 'gameoflife-acceptance-tests/Dockerfile' with their respective details.

Sonam chhabaidiya D15A 09

The screenshot shows the SonarQube interface for the project 'sonarqube-pipeline'. The 'Security Hotspots' tab is selected. A single security hotspot is listed:

- Status:** To review
- Description:** This security hotspot needs to be reviewed to assess whether the code poses a risk.
- Review button:** Review
- Category:** Permission
- Assignee:** Not assigned

The hotspot details are as follows:

```
FROM tomcat:8-jre8
...
RUN rm -rf /usr/local/tomcat/webapps/*
COPY target/gameoflife.war /usr/local/tomcat/webapps/ROOT.war
EXPOSE 8080
CMD ["catalina.sh", "run"]
```

Review priority: Medium

- Permission: The tomcat image runs with root as the default user. Make sure it is safe here.
- Encryption of Sensitive Data
- Others

3 of 3 shown

The screenshot shows the SonarQube interface for the project 'sonarqube-pipeline'. The 'Measures' tab is selected. On the left, a sidebar displays various metrics and reports:

- Reliability
- Maintainability
- Security Review
- Duplications
- Overview
- Overall Code
 - Density: 50.6%
 - Duplicated Lines: 384,007
 - Duplicated Blocks: 42,808
 - Duplicated Files: 979
- Size

The main panel displays a report titled 'Duplicated Lines (%)':

	Duplicated Lines (%)	Duplicated Lines
gameoflife-acceptance-tests	0.0%	0
gameoflife-build	0.0%	0
gameoflife-core	9.6%	374
gameoflife-deploy	0.0%	0
gameoflife-web	50.9%	383,633
pom.xml	0.0%	0

View as: Tree | Select files | Navigate | 6 files

Sonam chhabaidiya D15A 09

The screenshot shows the SonarQube interface for the project `sonarqube-pipeline`. The left sidebar displays various metrics under the `Measures` tab, including:

- Security Review: 1,112 issues
- Duplications: Overview
- Overall Code:
 - Density: 50.6%
 - Duplicated Lines: 384,007
 - Duplicated Blocks: 42,808
 - Duplicated Files: 979
- Size
- Complexity:
 - Cyclomatic Complexity: 1,112

The right panel shows the file structure for the project, with a total of 6 files:

- `gameoflife-acceptance-tests`
- `gameoflife-build`
- `gameoflife-core`: 18 files
- `gameoflife-deploy`
- `gameoflife-web`
- `pom.xml`: 1,094 files

At the bottom of the right panel, it says "6 of 6 shown".

ADVANCE DEVOPS EXPERIMENT 9

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

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

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

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

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

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

Sonam chhabaidiya D15A 09

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

Package	Architecture	Version	Repository	Size
Installing:				
httpd	x86_64	2.4.62-1.amzn2023	amazonlinux	48 k
php8_3	x86_64	8.3.10-1.amzn2023.0.1	amazonlinux	10 k
Installing dependencies:				
apr	x86_64	1.7.2-2.amzn2023.0.2	amazonlinux	129 k
apr-util	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	98 k
generic-logos-htpd	noarch	18.0.0-12.amzn2023.0.3	amazonlinux	19 k
httpd-core	x86_64	2.4.62-1.amzn2023	amazonlinux	1.4 M
httpd-filesystem	noarch	2.4.62-1.amzn2023	amazonlinux	14 k
httpd-tools	x86_64	2.4.62-1.amzn2023	amazonlinux	81 k
libbrotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	315 k
libsodium	x86_64	1.0.19-4.amzn2023	amazonlinux	176 k
libxslt	x86_64	1.1.34-5.amzn2023.0.2	amazonlinux	241 k
mailman	noarch	2.1.49-3.amzn2023.0.3	amazonlinux	32 k

Step 4: Update and install the required packages Use the following commands: **sudo yum update** **sudo yum install httpd php** **sudo yum install gcc glibc glibc-common** **sudo yum install gd gd-devel**

```
[ec2-user@ip-172-31-33-14 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:20:32 ago on Thu Sep 26 08:42:17 2024.
Package glibc-2.34-52.amzn2023.0.11.x86_64 is already installed.
Package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
Installing:				
gcc	x86_64	11.4.1-2.amzn2023.0.2	amazonlinux	32 M
Installing dependencies:				
annobin-docs	noarch	10.93-1.amzn2023.0.1	amazonlinux	92 k
annobin-plugin-gcc	x86_64	10.93-1.amzn2023.0.1	amazonlinux	887 k
cpp	x86_64	11.4.1-2.amzn2023.0.2	amazonlinux	10 M
go	x86_64	8.0.4-5.amzn2023.0.2	amazonlinux	105 k
glibc-devel	x86_64	2.34-52.amzn2023.0.11	amazonlinux	27 k
glibc-headers-x86	noarch	2.34-52.amzn2023.0.11	amazonlinux	427 k
guile22	x86_64	2.2.7-2.amzn2023.0.3	amazonlinux	6.4 M
kernel-headers	x86_64	6.1.109-118.189.amzn2023	amazonlinux	1.4 M
liblempc	x86_64	1.2.1-2.amzn2023.0.2	amazonlinux	62 k
libtool-ltdl	x86_64	2.4.7-1.amzn2023.0.3	amazonlinux	38 k
libxmlcrmt-devel	x86_64	4.4.33-7.amzn2023	amazonlinux	32 k

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

Package	Architecture	Version	Repository	Size
Installing:				
gd	x86_64	2.3.3-5.amzn2023.0.3	amazonlinux	139 k
gd-devel	x86_64	2.3.3-5.amzn2023.0.3	amazonlinux	38 k
Installing dependencies:				
brotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	314 k
brotli-devel	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	31 k
bzip2-devel	x86_64	1.0.8-6.amzn2023.0.2	amazonlinux	214 k
cairo	x86_64	1.17.6-2.amzn2023.0.1	amazonlinux	684 k
cmake-filesystem	x86_64	3.22.2-1.amzn2023.0.4	amazonlinux	16 k
fontconfig	x86_64	2.13.94-2.amzn2023.0.2	amazonlinux	273 k
fontconfig-devel	x86_64	2.13.94-2.amzn2023.0.2	amazonlinux	128 k
fonts-filesystem	noarch	1:2.0.5-12.amzn2023.0.2	amazonlinux	9.5 k
freetype	x86_64	2.13.94-2.amzn2023.0.1	amazonlinux	423 k

Step 5: Create a new nagios user by writing the following commands
sudo adduser -m nagios **sudo passwd nagios**

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

Step 6: Create a new user group using **sudo groupadd nagcmd** and Add users to the group using the following commands:

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

```
Complete!
[ec2-user@ip-172-31-33-14 ~]$ sudo adduser -m nagios
[ec2-user@ip-172-31-33-14 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-33-14 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-33-14 ~]$ sudo usermod -a -G nagcmd nagios
[ec2-user@ip-172-31-33-14 ~]$ sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-33-14 ~]$ mkdir downloads
[ec2-user@ip-172-31-33-14 ~]$ cd downloads
[ec2-user@ip-172-31-33-14 downloads]$ wget https://sourceforge.net/projects/nagios/files/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz/download?use_mirror=excellmedia
--2024-09-26 09:15:54-- https://sourceforge.net/projects/nagios/files/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz/download?use_mirror=excellmedia
Resolving sourceforge.net (sourceforge.net)... 172.64.150.145, 104.18.37.111, 2606:4700:4400::6812:256f, ...
Connecting to sourceforge.net (sourceforge.net)|172.64.150.145|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz?ts=gAAAAABm9SZKFW7LwD1QAJ2jNzqmSJwAPM1mQ-eAJYK8zNmrv ifVkhbsV-qOfPsLUyICC6yvdHu6UeIyvN2sVGUtr9BeQ%3D%3D&use_mirror=excellmedia&r= [following]
```

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

Commands - **mkdir ~/downloads**

cd ~/downloads

Also download Nagios and plugin source files

Commands - **wget**

https://assets.nagios.com/downloads/nagioscore/releases/nagios-

**4.4.6.tar.gz wget https://nagios-plugins.org/download/nagios-plugins-
2.3.3.tar.gz**

Sonam chhabaidiya D15A 09

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

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

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

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

Step 8-Extract the nagios source file with the following commands
tar zxvf nagios-4.4.6.tar.gz cd nagios-4.4.6

Then run the configuration script with the following command

/configure --with-command-group=nagcmd

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

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

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

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

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

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

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

Commands -

```
./sudo make install sudo make
install-init sudo make install-
config sudo make install-
commandmode
```

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

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

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

*** External command directory configured ***

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

Step 11-Edit the Config File to Change the Email Address Commands

-

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

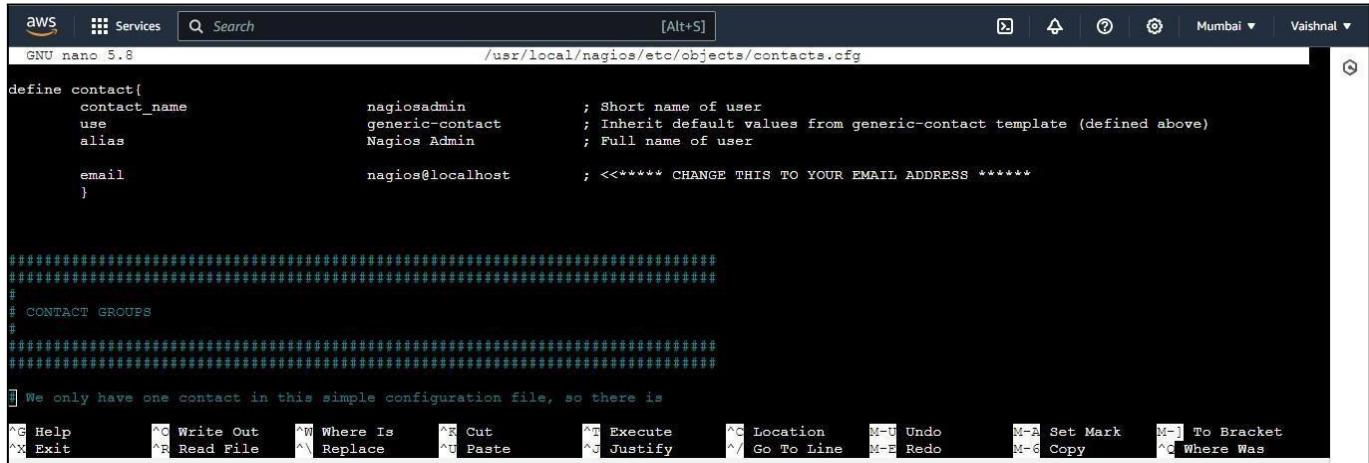
- Change the email address in the contacts.cfg file to your preferred email

Step 12-Configure the Web Interface

Commands - **sudo make**

install-webconf

Sonam chhabadiya D15A 09



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

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

    email                nagios@localhost ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****

}

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

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

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

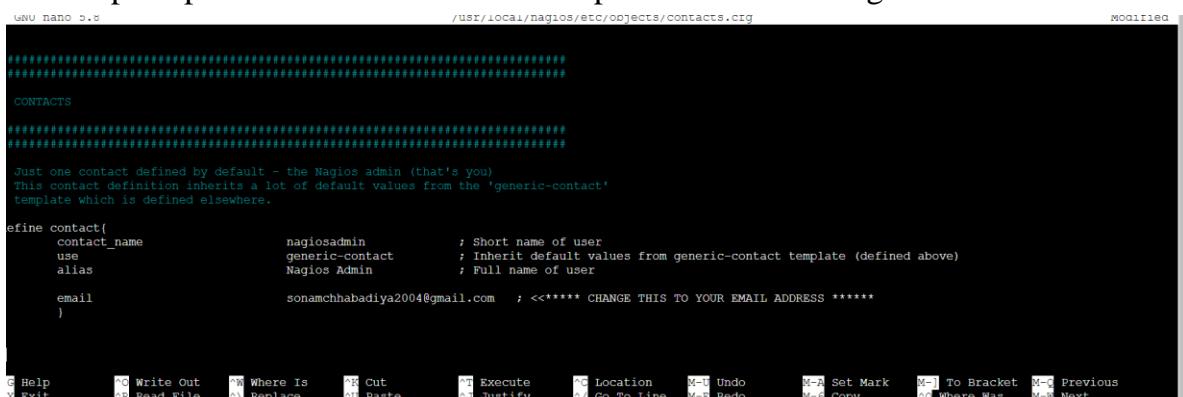
Help Write Out Where Is Cut Execute Location Undo Set Mark To Bracket

Exit Read File Replace Paste Justify Go To Line Redo Copy Where Was

Step 13-Create a Nagios Admin Account

Commands -

sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin • You will be prompted to enter and confirm the password for the nagiosadmin user



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

#####
# CONTACTS
#
#####

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

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

    email                sonamchhabadiya2004@gmail.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****

}

Modified
```

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

Help Write Out Where Is Cut Execute Location Undo Set Mark To Bracket

Exit Read File Replace Paste Justify Go To Line Redo Copy Where Was

M-Q Previous M-N Next

Step 14-. Extract the Plugins Source File

Commands - **cd**

~/downloads

tar zxvf nagios-plugins-2.3.3.tar.gz

cd nagios-plugins-2.3.3

```
*** External command directory configured ***

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

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

Step 15-19. Compile and Install Plugins

Commands -

**./configure --with-nagios-user=nagios --with-nagios-group=nagios make
sudo make install**

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

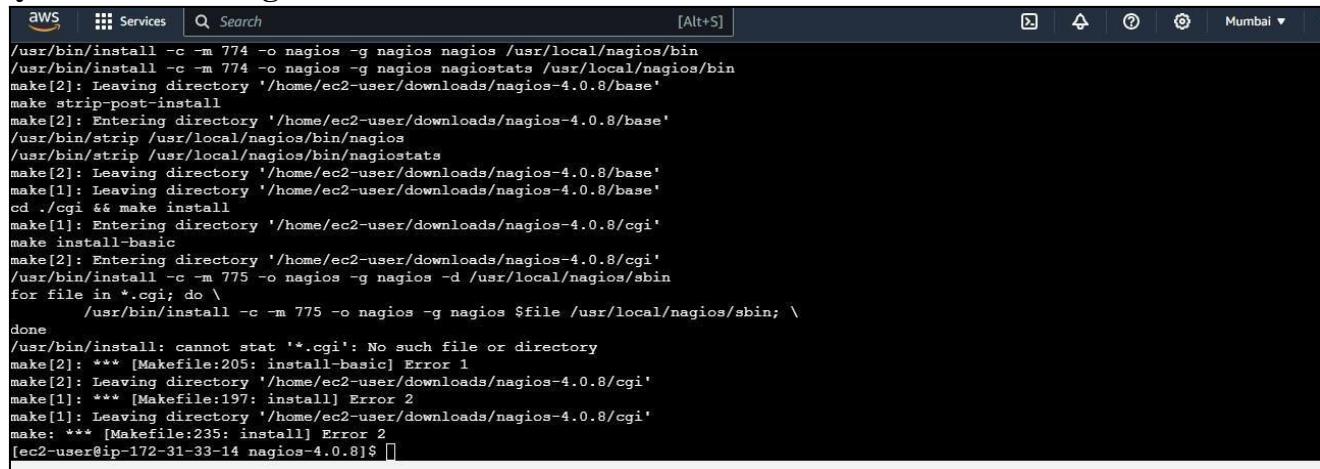
Step 16-Start Nagios

Commands - sudo chkconfig

--add nagios sudo chkconfig

nagios on

**sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg sudo
systemctl start nagios**



```
aws Services Search [Alt+S] Mumbai ▾
/usr/bin/install -c -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -m 774 -o nagios -g nagios nagiosstats /usr/local/nagios/bin
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
make strip-post-install
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/base'
/usr/bin/strip /usr/local/nagios/bin/nagios
/usr/bin/strip /usr/local/nagios/bin/nagiosstats
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/sbin
for file in *.cgi; do \
    /usr/bin/install -c -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin; \
done
/usr/bin/install: cannot stat '*.cgi': No such file or directory
make[2]: *** [Makefile:205: install-basic] Error 1
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
make[1]: *** [Makefile:197: install] Error 2
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
make: *** [Makefile:235: install] Error 2
[ec2-user@ip-172-31-33-14 nagios-4.0.8]$ []
```

Step 17-Access Nagios Web Interface

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

Nagios® Core™
Version 4.4.6
April 28, 2020
Check for updates

A new version of Nagios Core is available!
Visit nagios.org to download Nagios 4.5.5.

Get Started

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

Quick Links

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

Latest News

Don't Miss...

ADVANCE DEVOPS EXP-10

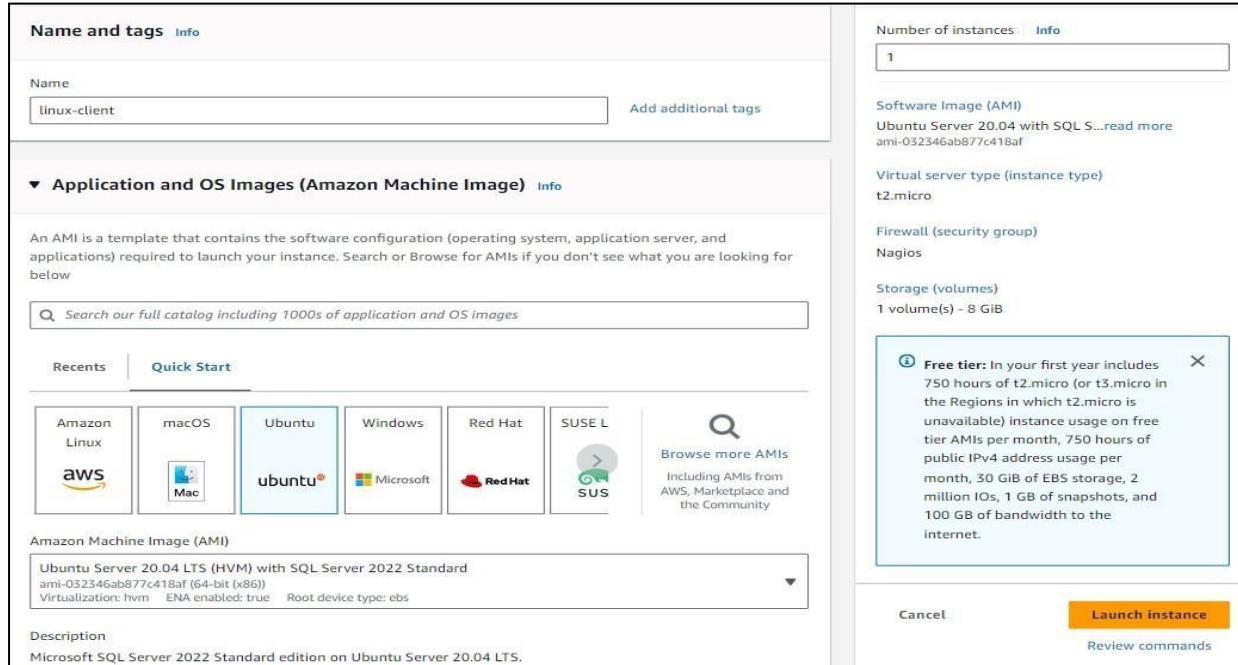
Aim: To perform Port, Service monitoring, Windows/Linux server monitoring using Nagios.

Step-1. Confirm Nagios is Running on the Server. sudo systemctl status nagios Proceed if you see that Nagios is active and running.

```
[ec2-user@ip-172-31-90-152 nagios-plugins-2.3.3]$ cd
[ec2-user@ip-172-31-90-152 ~]$ sudo systemctl restart nagios
[ec2-user@ip-172-31-90-152 ~]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Mon 2024-09-30 19:41:36 UTC; 7s ago
     Docs: https://www.nagios.org/documentation
 Process: 80238 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Process: 80239 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 80240 (nagios)
   Tasks: 6 (limit: 1112)
  Memory: 4.0M
    CPU: 15ms
   CGroup: /system.slice/nagios.service
           └─80240 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
               ├─80241 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               ├─80242 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               ├─80243 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               ├─80244 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               └─80245 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: qh: core query handler registered
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: qh: echo service query handler registered
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: qh: help for the query handler registered
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Successfully registered manager as @wproc with query handler
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Registry request: name=Core Worker 80244;pid=80244
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Registry request: name=Core Worker 80243;pid=80243
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Registry request: name=Core Worker 80242;pid=80242
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Registry request: name=Core Worker 80241;pid=80241
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: Successfully launched command file worker with pid 80245
```

Step-2. Create an Ubuntu 20.04 Server EC2 Instance



Step-3: Verify Nagios Process on the Server

```
[ec2-user@ip-172-31-80-215 nagios-plugins-2.3.3]$ ps -ef | grep nagios
nagios  68654      1  0 20:29 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios  68655  68654  0 20:29 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios  68656  68654  0 20:29 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios  68657  68654  0 20:29 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios  68658  68654  0 20:29 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios  68659  68654  0 20:29 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user  69588  26447  0 20:44 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-80-215 nagios-plugins-2.3.3]$
```

Step-4: Become Root User and Create Directories

`sudo su`, `mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts` and to copy the same config

`file- cp /usr/local/nagios/etc/objects/localhost.cfg,`

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

```
[ec2-user@ip-172-31-80-215 nagios-plugins-2.3.3]$ sudo su
[root@ip-172-31-80-215 nagios-plugins-2.3.3]# mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-80-215 nagios-plugins-2.3.3]# cp /usr/local/nagios/etc/objects/localhost.cfg
cp: missing destination file operand after '/usr/local/nagios/etc/objects/localhost.cfg'
Try 'cp --help' for more information.
[root@ip-172-31-80-215 nagios-plugins-2.3.3]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-80-215 nagios-plugins-2.3.3]#
```

i-0ae1aae975bae3b7a (nagios-host)

Step-5: Edit the Configuration File

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

- Change hostname to linuxserver everywhere in the file
- Change address to the public IP address of your linux-client.
- Change host_group name under hostgroup to linux_server

```
#####
# HOST DEFINITION
#
#####

# Define a host for the local machine

define host {

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

    host_name    linuxserver
    alias        linuxserver
    address      35.174.139.220
}

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

# Define an optional hostgroup for Linux machines

define hostgroup {

    hostgroup_name   linux-servers1 ; The name of the hostgroup
    alias            Linux Servers   ; Long name of the group
    members          localhost       ; Comma separated list of hosts that belong to this group
}

[ Read 187 lines ]
^G Help      ^O Write Out     ^W Where Is      ^K Cut           ^I Execute      ^C Location      M-U Undo      M-A Set Mark   M-J To
^X Exit      ^R Read File      ^V Replace       ^U Paste         ^J Justify      ^Y Go To Line    M-E Redo      M-G Copy      ^Q Where
```

Step-6: Update Nagios Configuration sudo

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

Add the command - cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/

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

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

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

Step-7: Verify Configuration Files sudo /usr/local/nagios/bin/nagios -v

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

```
[ec2-user@ip-172-31-80-215 ~]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org
Reading configuration data...
    Read main config file okay...
Warning: Duplicate definition found for service 'HTTP' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg')
Warning: Duplicate definition found for service 'SSH' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg'),
Warning: Duplicate definition found for service 'Swap Usage' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg')
Warning: Duplicate definition found for service 'Current Load' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg')
Warning: Duplicate definition found for service 'Total Processes' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg')
Warning: Duplicate definition found for service 'Current Users' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg')
Warning: Duplicate definition found for service 'Root Partition' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg')
Warning: Duplicate definition found for service 'PING' on host 'localhost' (config file '/usr/local/nagios/etc/objects/localhost.cfg')
    Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
    Checked 8 services.
    Checked 2 hosts.
    Checked 2 host groups.
    Checked 0 service groups.
    Checked 1 contacts.
    Checked 1 contact groups.
    Checked 24 commands.
    Checked 5 time periods.
    Checked 0 host escalations.
    Checked 0 service escalations.
Checking for circular paths...
    Checked 2 hosts
    Checked 0 service dependencies
    Checked 0 host dependencies
    Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors: 0
```

Step-8: Restart Nagios Service sudo
systemctl restart nagios

Step-9: SSH into the Client Machine

Use SSH or EC2 Instance Connect to access the linux-client.

Step-10: Update Package Index and Install Required Packages sudo

apt update -y sudo apt install gcc -y

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

```
ubuntu@ip-172-31-86-24:~$ sudo apt update -y
sudo apt install gcc -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [380 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:9 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [83.1 kB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [4560 B]
Get:11 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [274 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [535 kB]
Get:18 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [116 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [130 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [8652 B]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [379 kB]
```

Step-11: Edit NRPE Configuration File

Commands -

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

Add your Nagios host IP address under allowed_hosts:

```
allowed_hosts=<Nagios_Host_IP>
#
# 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=127.0.0.1,35.174.139.220

#
# 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
# option.
#
# *** ENABLING THIS OPTION IS A SECURITY RISK! ***
# Read the SECURITY file for information on some of the security implications
# of enabling this variable.
#
# Values: 0=do not allow arguments, 1=allow command arguments
dont_blame_nrpe=0
```

Step-12: Restart NRPE Server

Commands -

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

Step-13:Check Nagios Dashboard

Open your browser and navigate to http://<Nagios_Host_IP>/nagios.

Log in with nagiosadmin and the password you set earlier.

You should see the new host linuxserver added.

Click on Hosts to see the host details.

Click on Services to see all services and ports being monitored

Nagios® Core™
Daemon running with PID 71172

Nagios® Core™
Version 4.4.6
April 28, 2020
Check for updates

A new version of Nagios Core is available!
Visit nagios.org to download Nagios 4.5.5

Get Started

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

Quick Links

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

Latest News

Don't Miss...

Copyright © 2010-2020 Nagios Core Development Team and Community Contributors. Copyright © 1999-2009 Ethan Galstad. See the THANKS file for more information on contributors.

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:

Current Network Status

Last Updated: Mon Sep 30 21:16:41 UTC 2024
Updated every 30 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

[View Service Status Detail For All Host Groups](#)
[View Status Overview For All Host Groups](#)
[View Status Summary For All Host Groups](#)
[View Status Grid For All Host Groups](#)

Limit Results: 100 ▾

Host **	Status ***	Last Check ***	Duration ***	Status Information
linuxserver	UP	09-30-2024 21:14:52	0d 0h 1m 49s	PING OK - Packet loss = 0%, RTA = 0.98 ms
localhost	UP	09-30-2024 21:14:01	0d 0h 47m 2s	PING OK - Packet loss = 0%, RTA = 0.04 ms

Results 1 - 2 of 2 Matching Hosts

Host Status Totals

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

Service Status Totals

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

Host Status Details For All Host Groups

Current Network Status

Last Updated: Mon Sep 30 21:21:11 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

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

Limit Results: 100 ▾

Host **	Service ***	Status ***	Last Check ***	Duration ***	Attempt ***	Status Information
localhost	Current Load	OK	09-30-2024 21:20:16	0d 0h 50m 55s	1/4	OK - load average: 0.00, 0.00, 0.00
localhost	Current Users	OK	09-30-2024 21:20:54	0d 0h 50m 17s	1/4	USERS OK - 1 users currently logged in
HTTP		WARNING	09-30-2024 21:19:31	0d 0h 46m 40s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.001 second response time
PING		OK	09-30-2024 21:17:09	0d 0h 49m 2s	1/4	PING OK - Packet loss = 0%, RTA = 0.04 ms
Root Partition		OK	09-30-2024 21:17:46	0d 0h 48m 25s	1/4	DISK OK - free space: / 6080 MB (74.91% inode=98%)
SSH		OK	09-30-2024 21:18:24	0d 0h 47m 47s	1/4	SSH OK - OpenSSH_8.7 (protocol 2.0)
Swap Usage		CRITICAL	09-30-2024 21:17:01	0d 0h 44m 10s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
Total Processes		OK	09-30-2024 21:19:39	0d 0h 46m 32s	1/4	PROCS OK: 36 processes with STATE = RSZDT

Results 1 - 8 of 8 Matching Services

ADVANCED DEVOPS

Experiment 11

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

Theory:

AWS Lambda

AWS Lambda is a serverless computing service provided by Amazon Web Services (AWS).

Users of AWS Lambda create functions, self-contained applications written in one of the supported languages and runtimes, and upload them to AWS Lambda, which executes those

functions in an efficient and flexible manner. The Lambda functions can perform any kind of computing task, from serving web pages and processing streams of data to calling APIs and integrating with other AWS services.

The concept of “serverless” computing refers to not needing to maintain your own servers to run these functions. AWS Lambda is a fully managed service that takes care of all the infrastructure

for you. And so “serverless” doesn’t mean that there are no servers involved: it just means that the servers, the operating systems, the network layer and the rest of the infrastructure have already been taken care of so that you can focus on writing application code.

Features of AWS Lambda

- AWS Lambda easily scales the infrastructure without any additional configuration. It reduces the operational work involved.

- It offers multiple options like AWS S3, CloudWatch, DynamoDB, API Gateway, Kinesis,

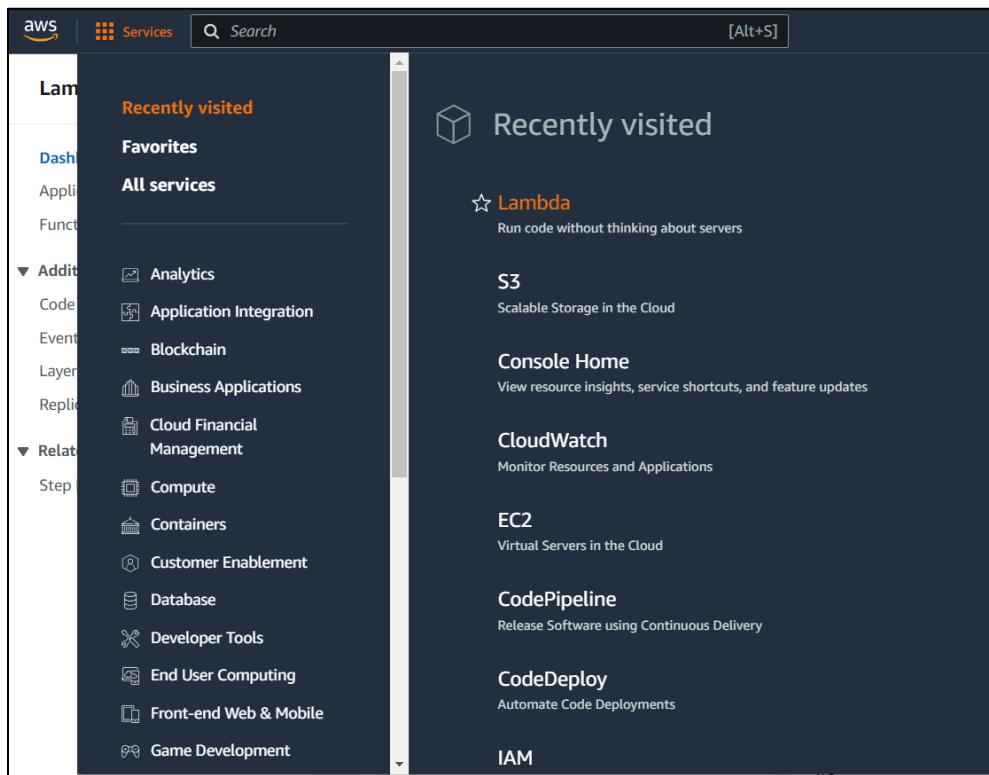
CodeCommit, and many more to trigger an event.

- You don’t need to invest upfront. You pay only for the memory used by the lambda function and minimal cost on the number of requests hence cost-efficient.

- AWS Lambda is secure. It uses AWS IAM to define all the roles and security policies.

- It offers fault tolerance for both services running the code and the function. You do not have to worry about the application down.

1. Go to your AWS account and search for Lamda



Functions (5)					
Last fetched 10 seconds ago					
Function name	Description	Package type	Runtime	Last modified	
MainMonitoringFunction	-	Zip	Python 3.8	3 months ago	Actions
RedshiftEventSubscription	Create Redshift event subscription to SNS Topic.	Zip	Python 3.8	3 months ago	Actions
RedshiftOverwatch	Deletes Redshift Cluster if the count is more than 2.	Zip	Python 3.8	3 months ago	Actions
RoleCreationFunction	Create SLR if absent	Zip	Python 3.8	3 months ago	Actions
ModLabRole	updates LabRole to allow it to assume itself	Zip	Python 3.8	3 months ago	Actions

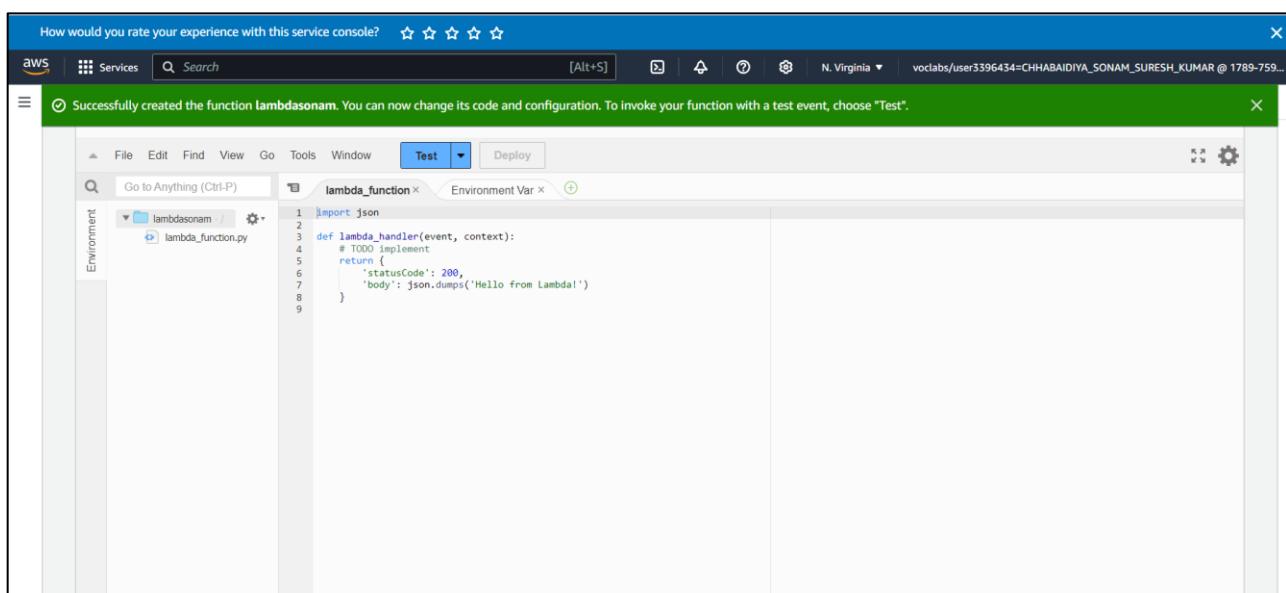
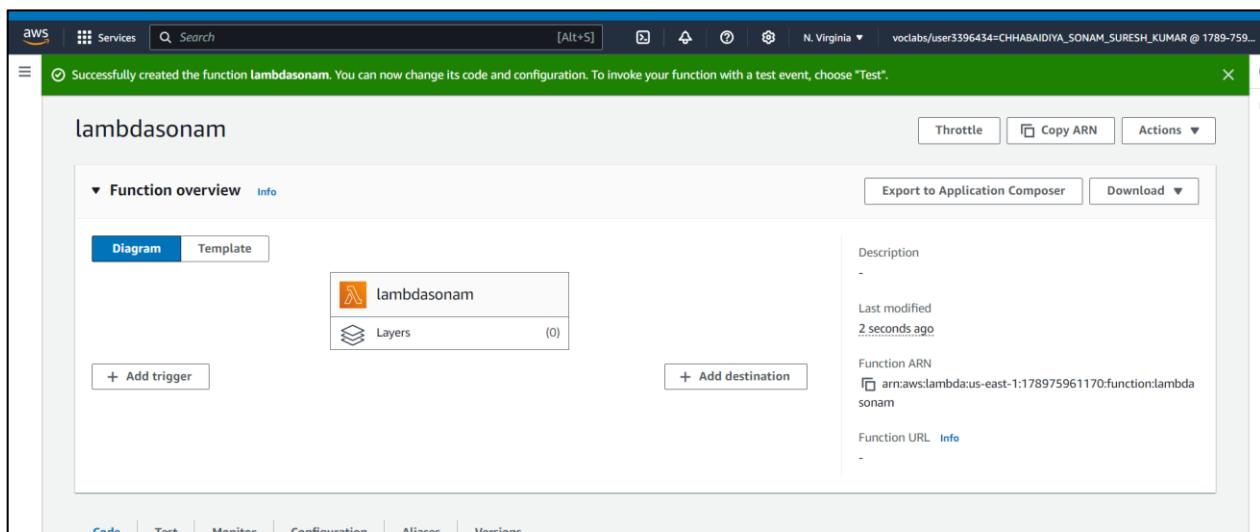
2. Create a new Lambda function

The screenshot shows the 'Create function' wizard in the AWS Lambda console. The top navigation bar includes 'Services', 'Search', and the region 'N. Virginia'. The main section is titled 'Create function' with a 'Basic information' sub-section. Under 'Function name', the value 'lambdasonam' is entered. Under 'Runtime', 'Python 3.12' is selected. Under 'Architecture', 'x86_64' is chosen. There are three tabs at the top: 'Author from scratch' (selected), 'Use a blueprint', and 'Container image'.

Use Labrole as user role.

The screenshot shows the 'Permissions' configuration page for a Lambda function. It includes sections for 'Change default execution role', 'Execution role' (where 'Use an existing role' is selected with 'LabRole' chosen), and 'Additional Configurations' (where 'Enable Code signing' is unchecked). A sidebar on the left lists existing roles: 'arn:aws:iam::123456789012:root' and 'arn:aws:lambda:us-east-1:123456789012:LabRole'.

Sonam Chhabaidiya D15A09



3. Create a new test event and save it.

Configure test event

A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.

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

Test event action

Create new event Edit saved event

Event name

sonam_event

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

Event sharing settings

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

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

Template - optional

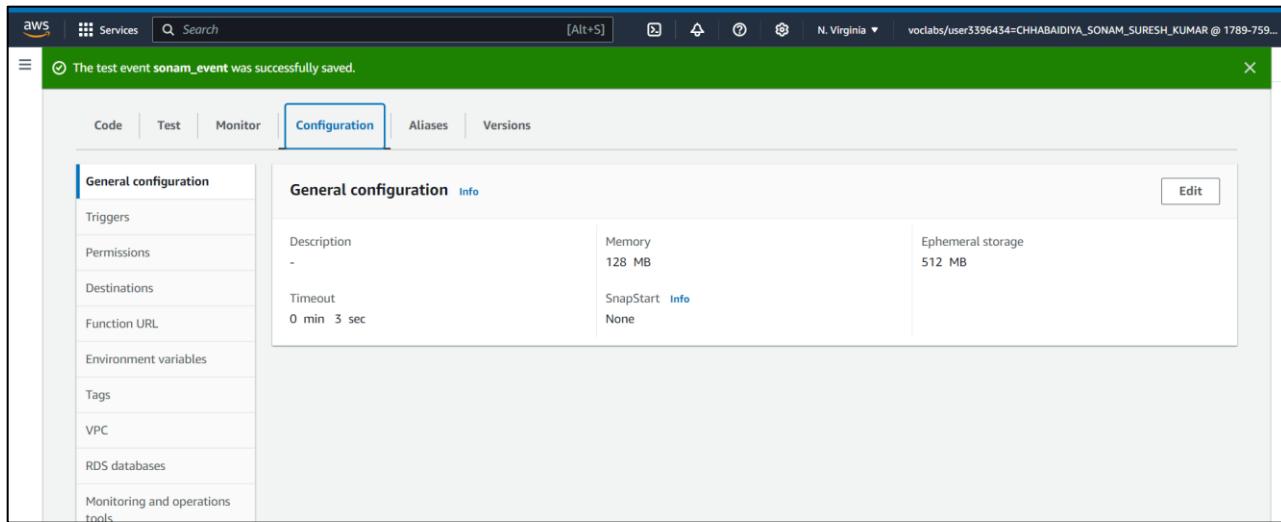
hello-world

Event JSON

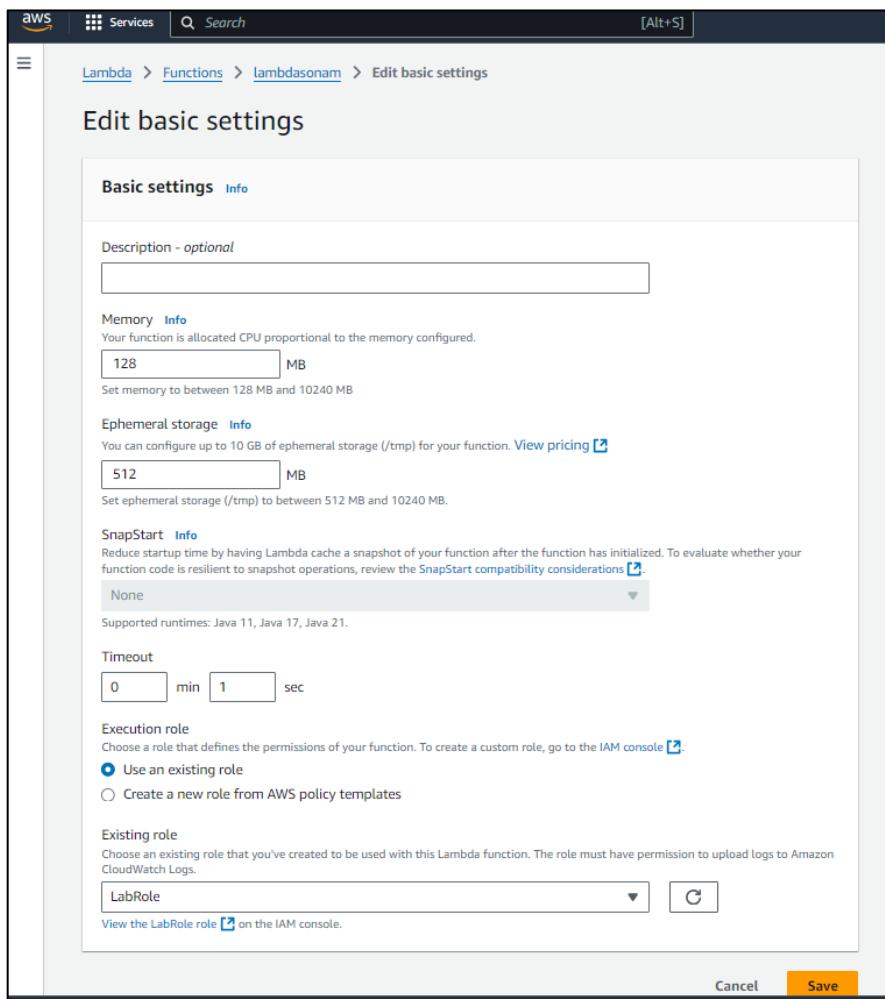
```
1 [{}]
2   "key1": "value1",
3   "key2": "value2",
4   "key3": "value3"
5 ]
```

Format JSON

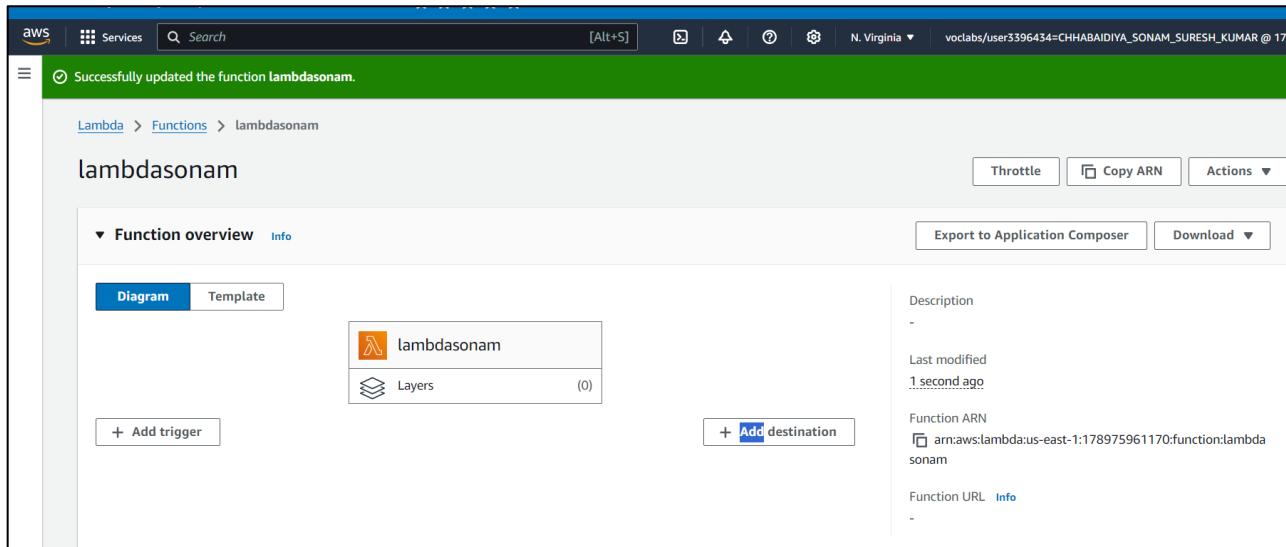
Cancel [Invoke](#) [Save](#)



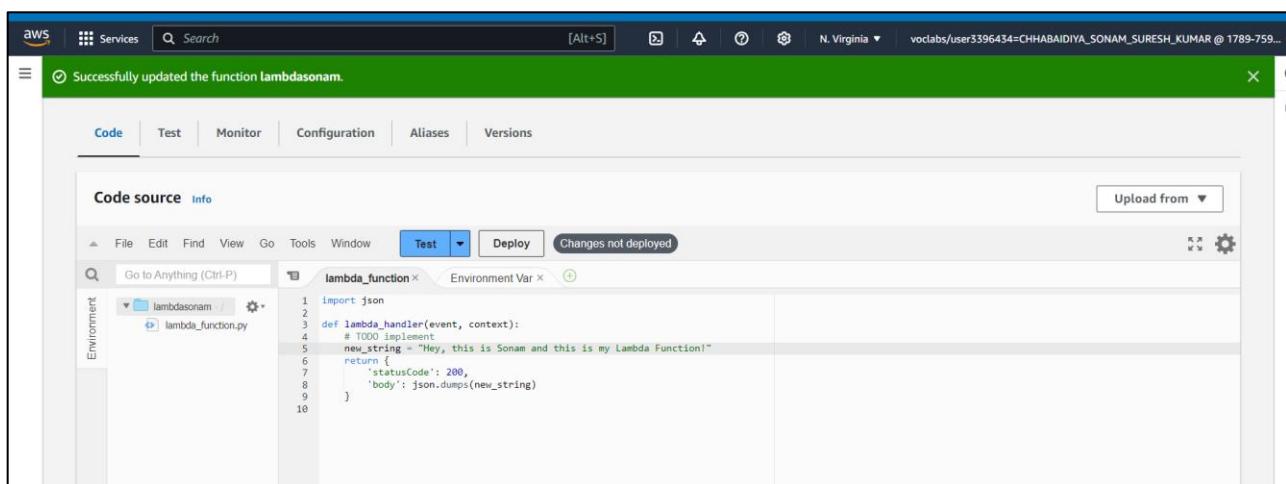
4. Change the basic settings



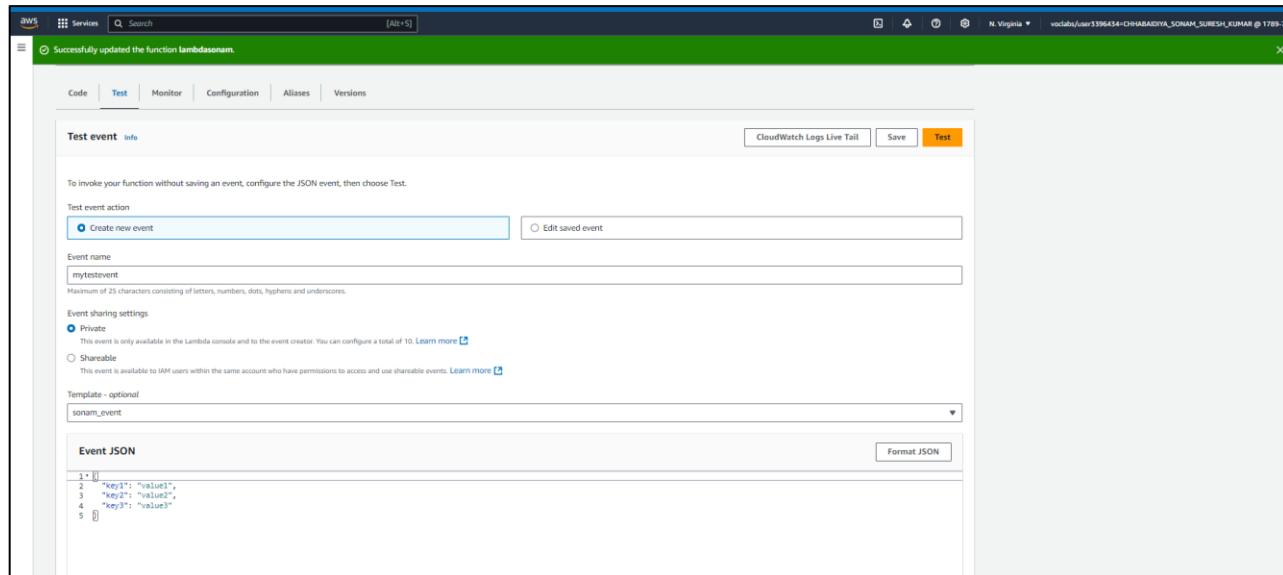
Sonam Chhabaidiya D15A09



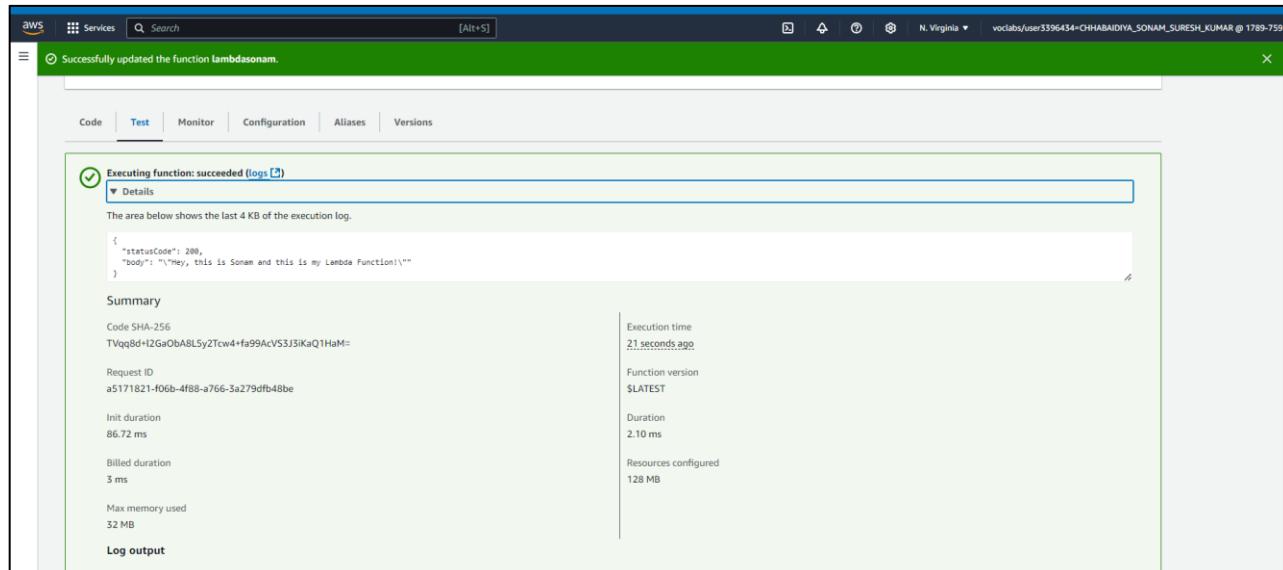
5. Edit the code as required and **ctrl+ s** to save it. Click on Deploy



6. Create a new test and click on Test



7. After this, the result can be seen



Sonam Chhabaidiya D15A09

Sonam Chhabaidiya D15A09

ADVANCE DEVOPS

Experiment No 12

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

1. Create an AWS S3 bucket and save it.

The screenshot shows the 'Create bucket' wizard. In the 'General configuration' step, the 'Bucket name' field is filled with 'sonabucket'. Below it, there's a note about naming rules and a 'Choose bucket' button for copying settings from another bucket. In the 'Object Ownership' step, the 'ACLs disabled (recommended)' option is selected, with a note explaining that all objects belong to the account owner. The 'Copy settings from existing bucket - optional' section is also visible.

The screenshot shows the 'Amazon S3 > Buckets' page. A green banner at the top indicates 'Successfully created bucket "sonabucket"'. Below, there's an 'Account snapshot' summary and a table of 'General purpose buckets'. The 'sonabucket' is listed with its details: Name 'sonabucket', AWS Region 'US East (Ohio) us-east-2', IAM Access Analyzer link, and Creation date 'October 19, 2024, 18:09:44 (UTC+05:30)'. There are buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'.

Name	AWS Region	IAM Access Analyzer	Creation date
codepipeline-eu-north-1-636097867266	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 5, 2024, 20:55:50 (UTC+05:30)
sonabucket	US East (Ohio) us-east-2	View analyzer for us-east-2	October 19, 2024, 18:09:44 (UTC+05:30)
sonamcc-todo-website	Europe (Stockholm) eu-north-1	View analyzer for eu-north-1	August 7, 2024, 14:34:59 (UTC+05:30)

2. Create an AWS Lambda Function, use python 3.11

The screenshot shows the 'Create function' wizard in the AWS Lambda console. The 'Info' tab is selected. Under 'Create function', there are three options: 'Author from scratch' (selected), 'Use a blueprint', and 'Container image'. The 'Basic information' section includes fields for 'Function name' (set to 'sonam_lambda'), 'Runtime' (set to 'Python 3.12'), and 'Architecture' (set to 'x86_64'). A sidebar titled 'Create a simple web app' provides a tutorial on how to build a simple web application using Lambda.

The screenshot shows the 'Function overview' page for the 'sonam_lambda' function. It displays the function's ARN (arn:aws:lambda:us-east-2:010928207831:function:sonam_lambda), the last modified time (5 minutes ago), and the function URL. The sidebar on the right continues the 'Create a simple web app' tutorial.

3. Change the code such that it prints “An image has been added to the bucket” when triggered.

4. Add S3 bucket in your triggers and select the name of your S3 bucket.

Add trigger

Trigger configuration

Bucket
Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.
s3/sonabucket

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

Prefix - optional
Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters. Any special characters must be URL encoded.
sonam's image

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

Create a simple web app
In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

[Learn more](#) [Start tutorial](#)

sonam_lambda

Function overview

Description: -

Last modified: 3 hours ago

Function ARN: arn:aws:lambda:us-east-2:010928207831:function:sonam_lambda

Function URL: Info

Code | Test | Monitor | Configuration | Aliases | Versions

Tutorials

Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

[Learn more](#)

[Start tutorial](#)

5. Go to your S3 bucket

Execution role

Role name: sonam_lambda-role-9qc3us19

Resource summary

To view the resources and actions that your function has permission to access, choose a service.

Amazon CloudWatch Logs

Resource	Actions
arn:aws:logs:us-east-2:010928207831:*	Allow: logs:CreateLogGroup
arn:aws:logs:us-east-2:010928207831:log-group:/aws/lambda/sonam_lambda:*	Allow: logs:CreateLogStream Allow: logs:PutLogEvents

Resource-based policy statements (1) Info

Resource-based policies grant other AWS accounts and services permissions to access your Lambda resources.

Statement ID	Principal	PrincipalOrgID	Conditions	Action
Lambda-13c40a2...	s3.amazonaws.com	-	StringEquals, Arn...	lambda:InvokeFunction

6. Upload an image inside the s3 bucket

The screenshot shows the AWS S3 console interface for uploading files. The URL in the address bar is `us-east-2.console.aws.amazon.com/s3/upload/sonabucket?region=us-east-2&bucketType=general`. The navigation path is `AWS > Services > Buckets > sonabucket > upload`. The main area is titled "Upload" with a "Info" link. A note says: "Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)". Below this is a large dashed box for dragging and dropping files, with "Drag and drop files and folders you want to upload here, or choose Add files or Add folder." instructions. A "Files and folders" table shows one item: "Screenshot 2023-10-24 210832.png" (1 Total, 841.5 KB). The table has columns for Name, Folder, Type, and Status. The status is "image/png". Below the table is a "Destination" section with "Destination" set to "s3://sonabucket". A "Destination details" link is present.

The screenshot shows the AWS S3 console after the upload was successful. A green banner at the top says "Upload succeeded" with a "View details below." link. Below the banner, the title is "Upload: status" with a "Close" button. A note says: "The information below will no longer be available after you navigate away from this page." The "Summary" section shows the destination "s3://sonabucket" with a status of "Succeeded" and "1 file, 841.5 KB (100.00%)". The "Failed" section shows "0 files, 0 B (0%)". Below this is a "Files and folders" table with one item: "Screenshot 2023-10-24 210832.png" (1 Total, 841.5 KB). The table has columns for Name, Folder, Type, Size, Status, and Error. The status is "Succeeded".

7. Go back to your Lambda function and under monitor select 'open cloudwatch logs'
Here you can see the message: An image has been added to the bucket.

The screenshot shows the AWS CloudWatch Log Events interface. The left sidebar is titled "CloudWatch" and includes sections for Dashboards, Alarms, Logs (with "Log groups" selected), Metrics, X-Ray traces, Events, Application Signals, Network monitoring, and Insights. The main content area shows a log group path: CloudWatch > Log groups > /aws/lambda/sonam_lambda > 2024/10/19/[\$.LATEST]. The title bar indicates the region is Ohio and the log group is sonam_lambda. The interface includes a search bar, filter buttons (Actions, Start tailing, Create metric filter), and a timestamp range selector (1m, 30m, 1h, 12h, Custom, UTC timezone). The log events table has columns for Timestamp and Message. The first event is timestamped 2024-10-19T12:55:59.137Z and contains the message "INIT_START Runtime Version: python3.12.v36 Runtime Version ARN: arn:aws:lambda:us-east-2::runtime:188d9ca2e2714ff5637bd2bbe...". Subsequent events show START, END, and REPORT requests for different request IDs, all with similar timestamps and ARNs. The last event is timestamped 2024-10-19T12:59:12.065Z and contains the message "REPORT RequestId: 3c1ba014-db19-4dc7-b132-a27d4a547228 Duration: 1.71 ms Billed Duration: 2 ms Memory Size: 128 MB Max Mem...".

Timestamp	Message
2024-10-19T12:55:59.137Z	INIT_START Runtime Version: python3.12.v36 Runtime Version ARN: arn:aws:lambda:us-east-2::runtime:188d9ca2e2714ff5637bd2bbe...
2024-10-19T12:55:59.228Z	START RequestId: cda3195f-bd32-4780-a664-aa6277b12944 Version: \$.LATEST
2024-10-19T12:55:59.243Z	END RequestId: cda3195f-bd32-4780-a664-aa6277b12944
2024-10-19T12:55:59.243Z	REPORT RequestId: cda3195f-bd32-4780-a664-aa6277b12944 Duration: 1.83 ms Billed Duration: 2 ms Memory Size: 128 MB Max Mem...
2024-10-19T12:59:12.056Z	START RequestId: 3c1ba014-db19-4dc7-b132-a27d4a547228 Version: \$.LATEST
2024-10-19T12:59:12.065Z	END RequestId: 3c1ba014-db19-4dc7-b132-a27d4a547228
2024-10-19T12:59:12.065Z	REPORT RequestId: 3c1ba014-db19-4dc7-b132-a27d4a547228 Duration: 1.71 ms Billed Duration: 2 ms Memory Size: 128 MB Max Mem...