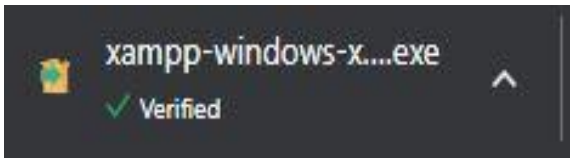
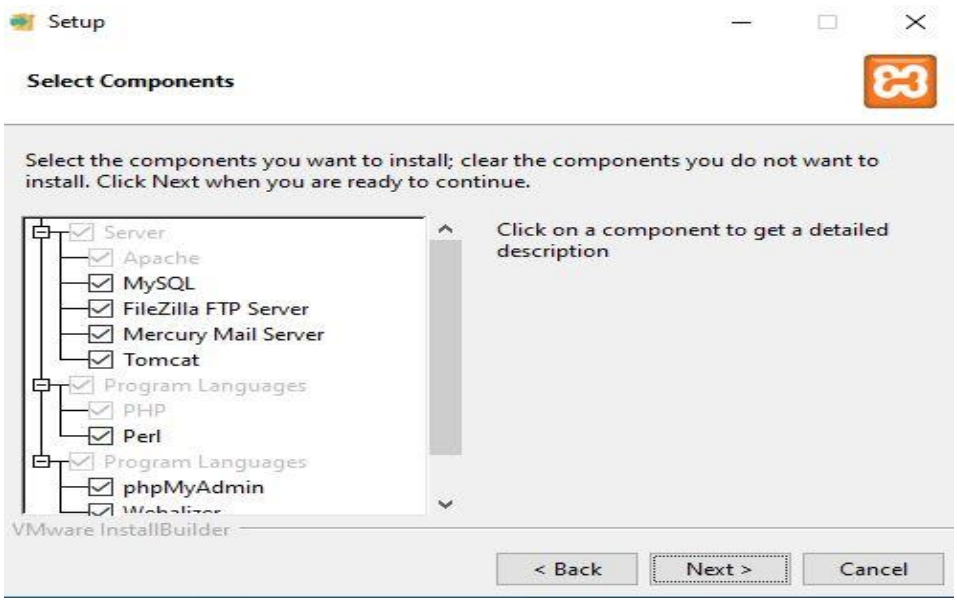
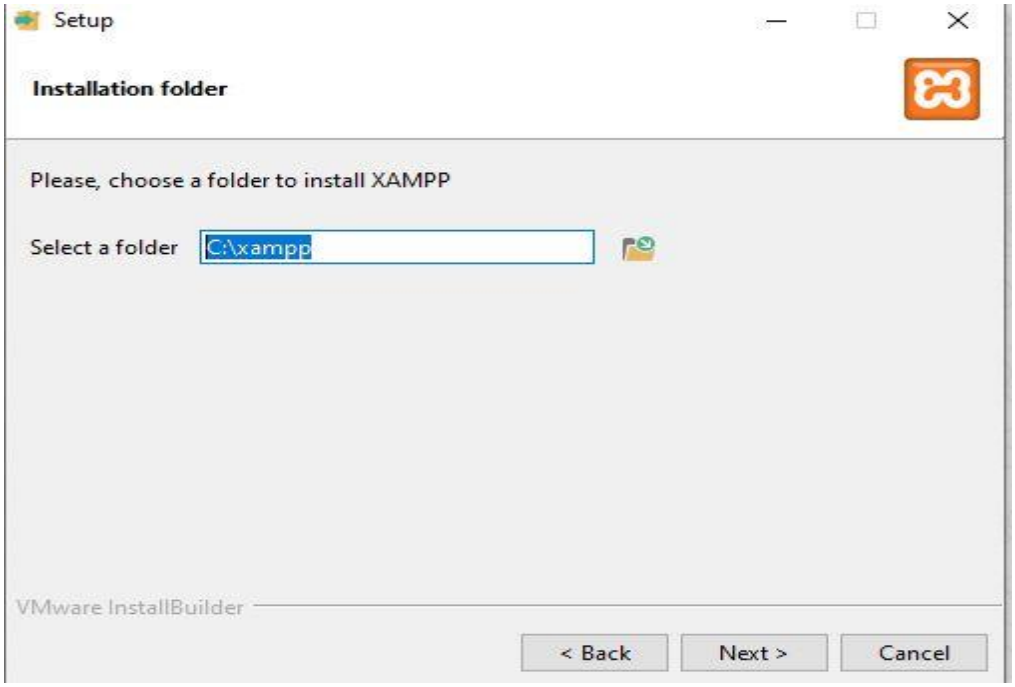
EXPERIMENT NO: - 01

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

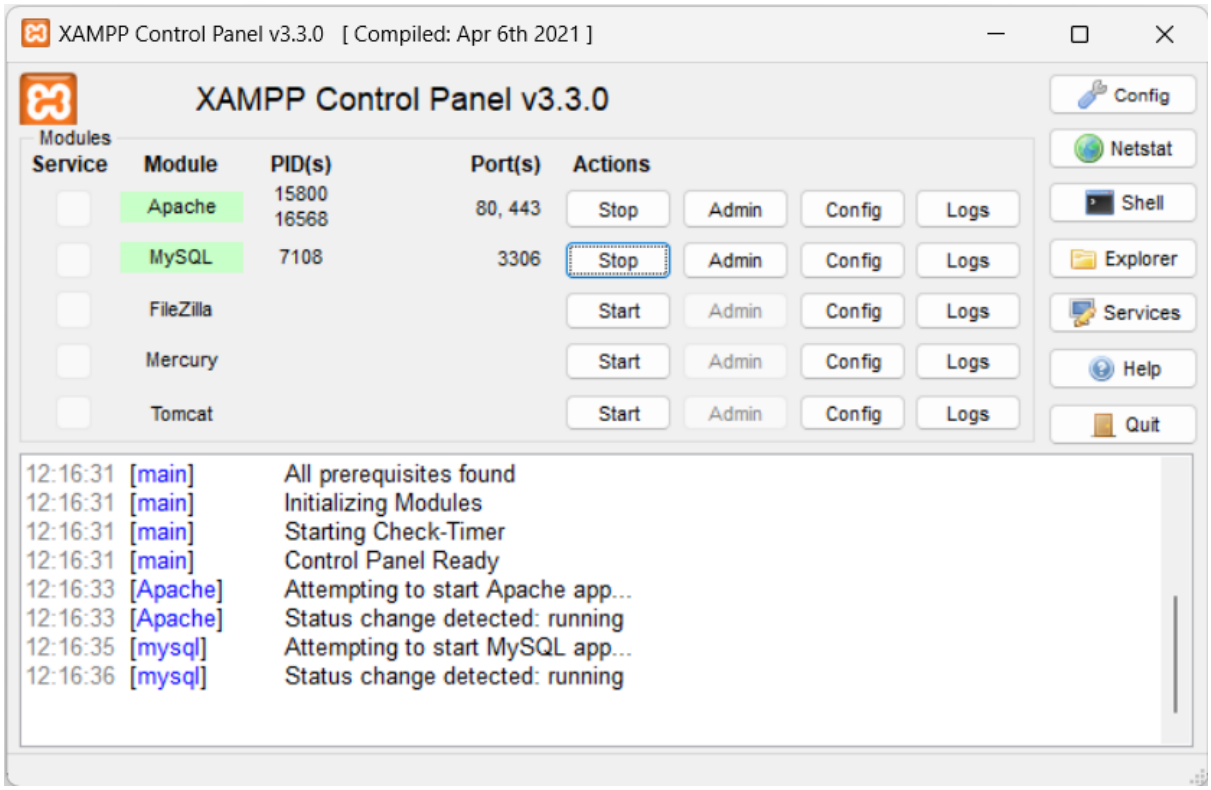
- Steps to host Website using Xampp Server

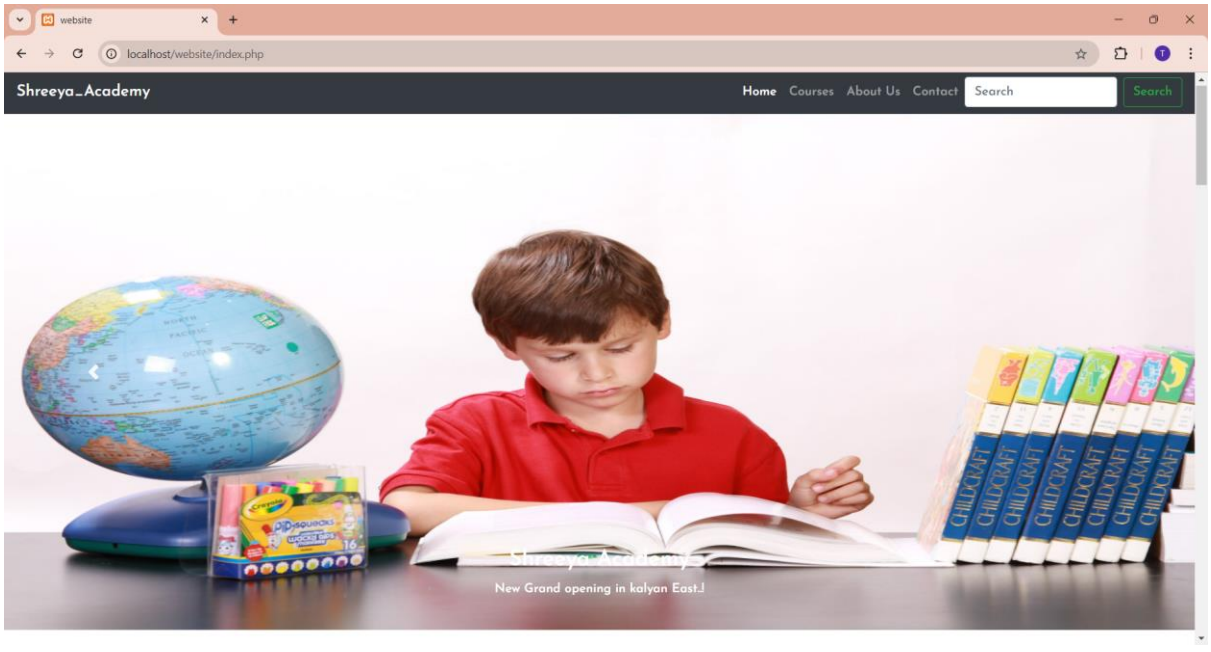




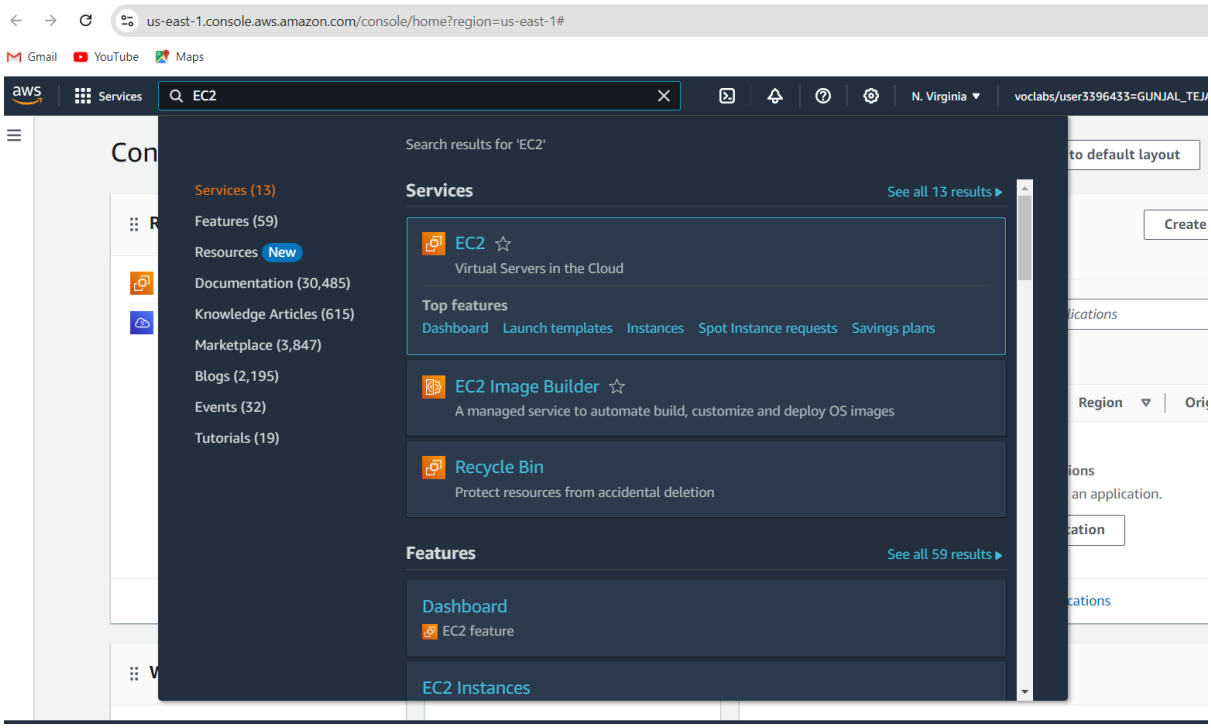

Setup
Installation folder
Please, choose a folder to install XAMPP
Select a folder
C:\xampp
VMware InstallBuilder
< Back
Next >
Cancel

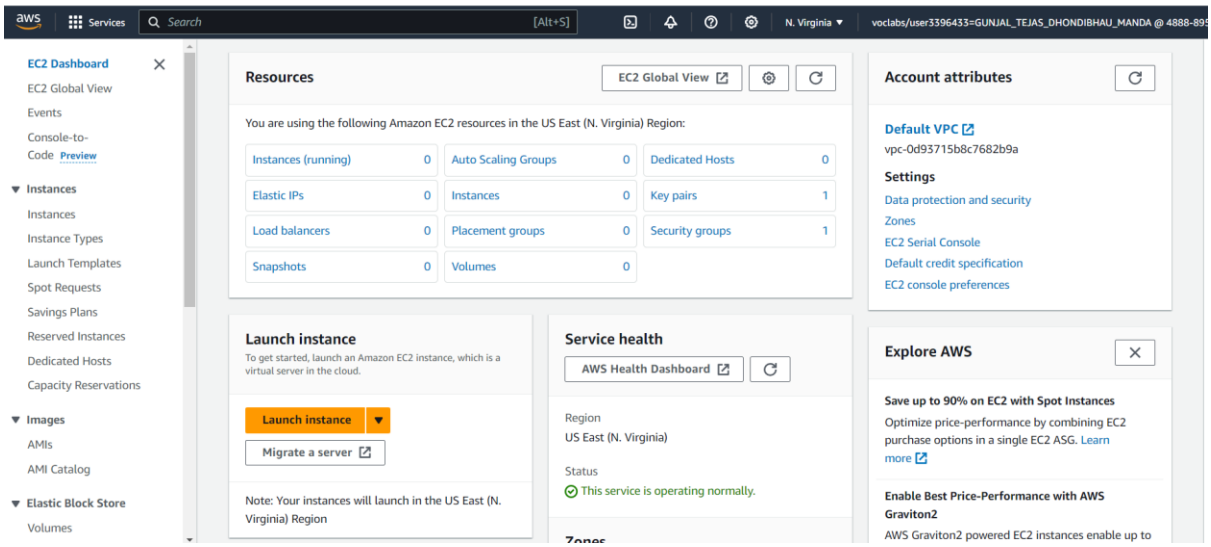

Setup
Welcome to XAMPP!
XAMPP is an easy to install Apache distribution containing MySQL, PHP and Perl
Installing
Unpacking files
VMware InstallBuilder
< Back
Next >
Cancel

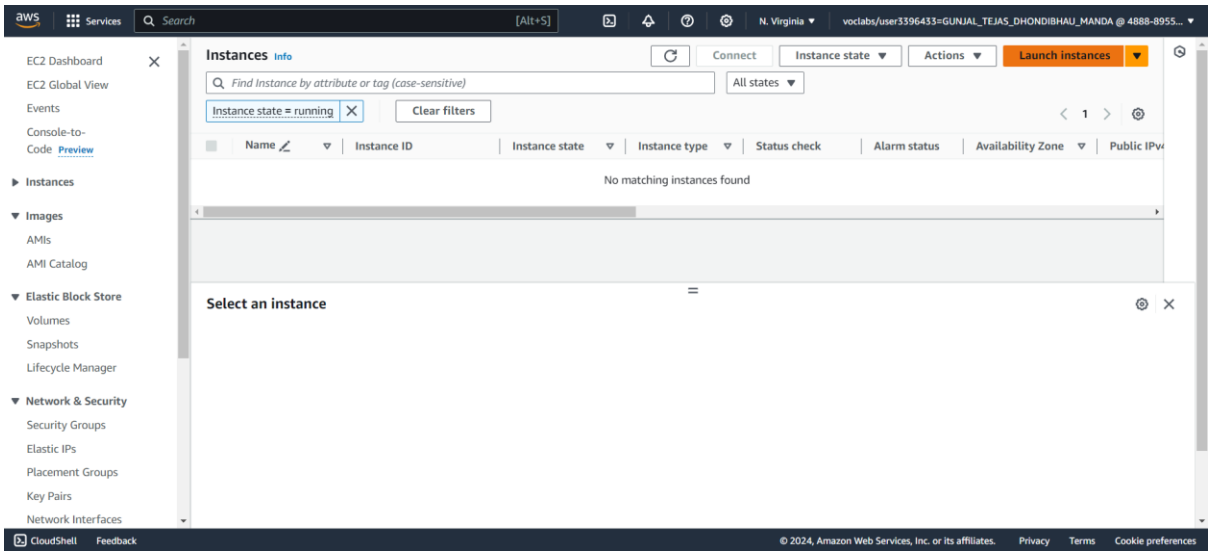

XAMPP Control Panel v3.3.0 [Compiled: Apr 6th 2021]
XAMPP Control Panel v3.3.0
Modules
Service
Module
PID(s)
Port(s)
Actions
Apache
15800
16568
80, 443
Stop
Admin
Config
Logs
MySQL
7108
3306
Stop
Admin
Config
Logs
FileZilla
Start
Admin
Config
Logs
Mercury
Start
Admin
Config
Logs
Tomcat
Start
Admin
Config
Logs
Config
Netstat
Shell
Explorer
Services
Help
Quit
12:16:31 [main] All prerequisites found
12:16:31 [main] Initializing Modules
12:16:31 [main] Starting Check-Timer
12:16:31 [main] Control Panel Ready
12:16:33 [Apache] Attempting to start Apache app...
12:16:33 [Apache] Status change detected: running
12:16:35 [mysql] Attempting to start MySQL app...
12:16:36 [mysql] Status change detected: running

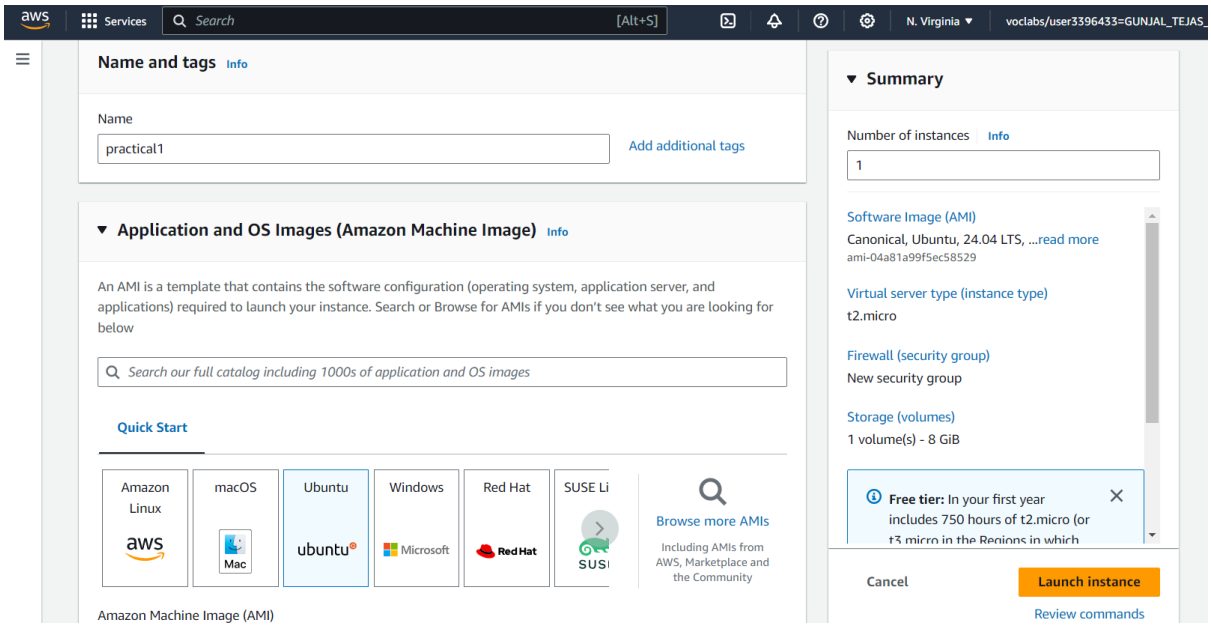

website
localhost/website/index.php
Shreeya_Academy
Home
Courses
About Us
Contact
Search
Search
New Grand opening in kalyan East.!

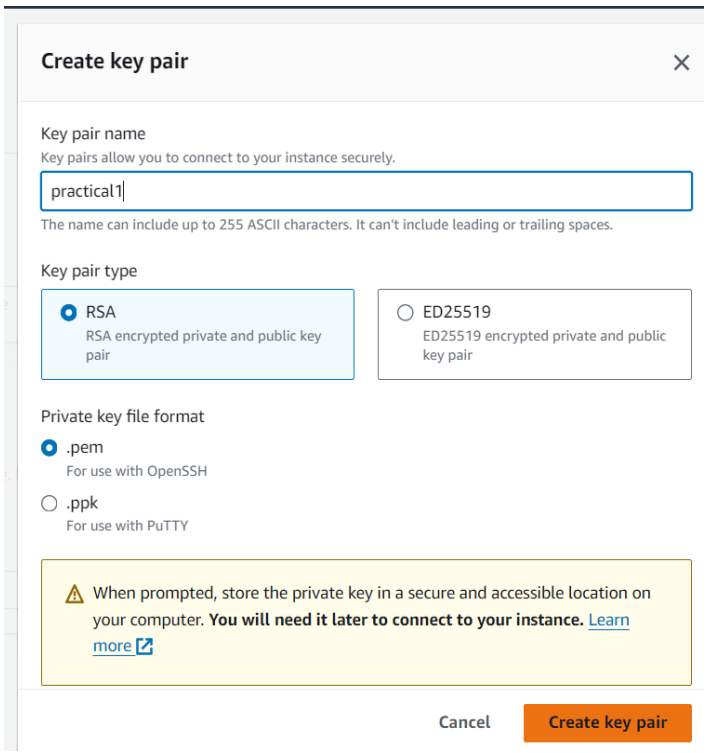
➢ Launching EC2 on AWS

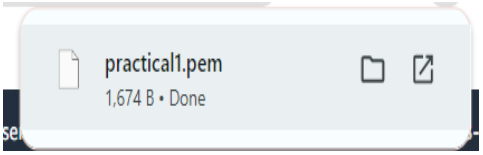


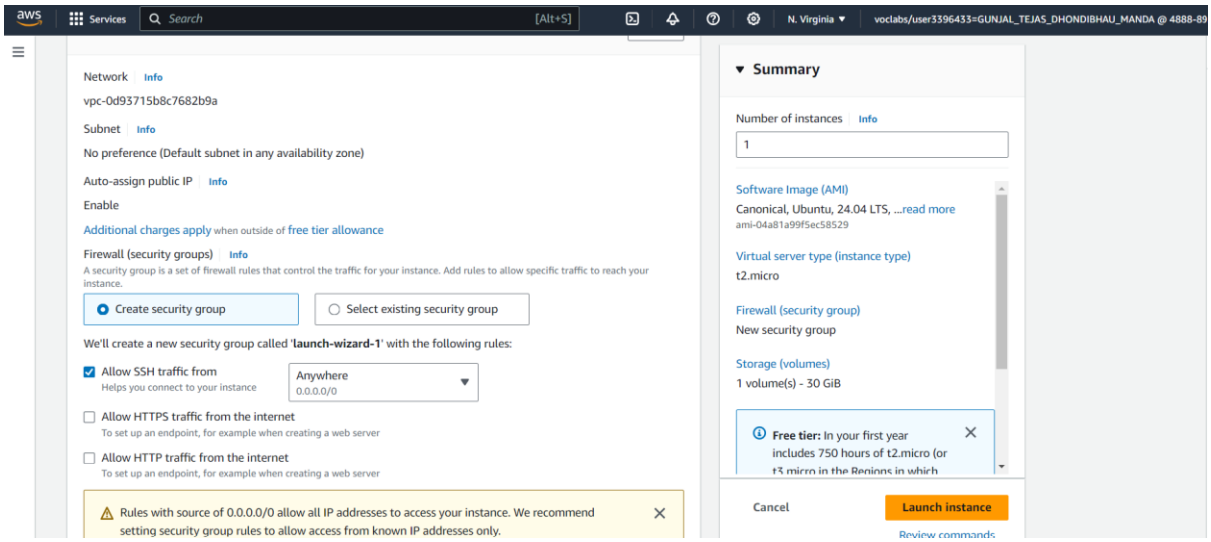


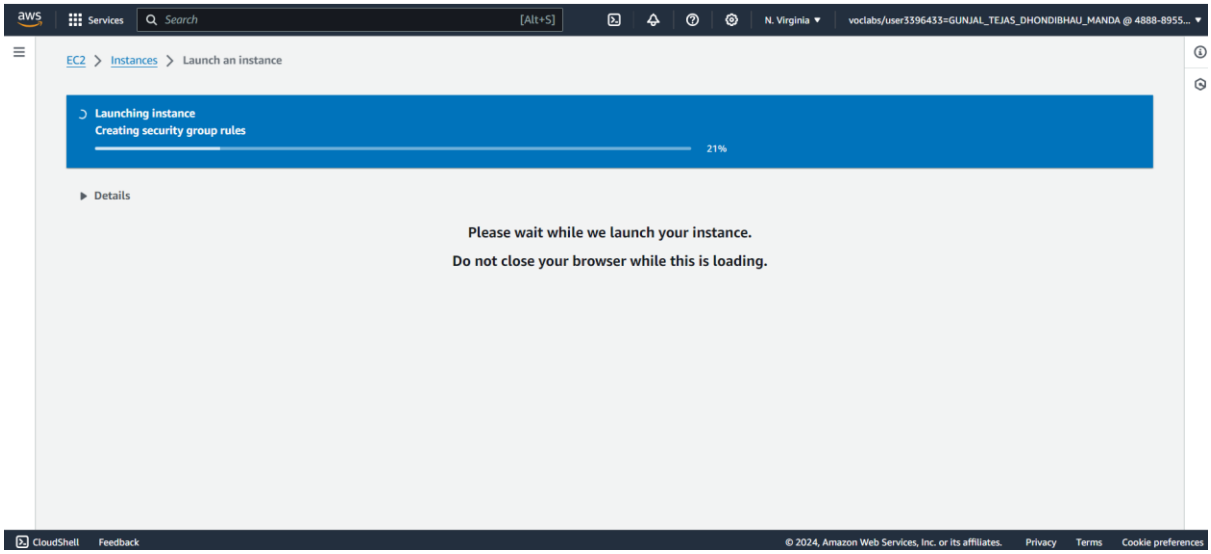

Services
Search
[Alt+S]
N. Virginia
EC2 Dashboard
EC2 Global View
Events
Console-to-Code Preview
Instances
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Instances Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance state = running
Clear filters
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv
No matching instances found
Select an instance
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

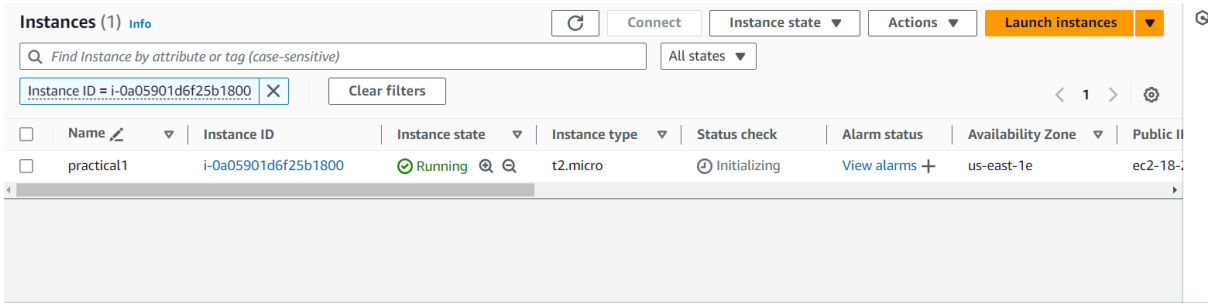

Services
Search
[Alt+S]
N. Virginia
Name and tags Info
Name
practical1
Add additional tags
Application and OS Images (Amazon Machine Image) Info
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below
Search our full catalog including 1000s of application and OS images
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Li
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Summary
Number of instances Info
1
Software Image (AMI)
Canonical, Ubuntu, 24.04 LTS, ...read more
ami-04a81a99f5ec58529
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or
Cancel
Launch instance
Review commands

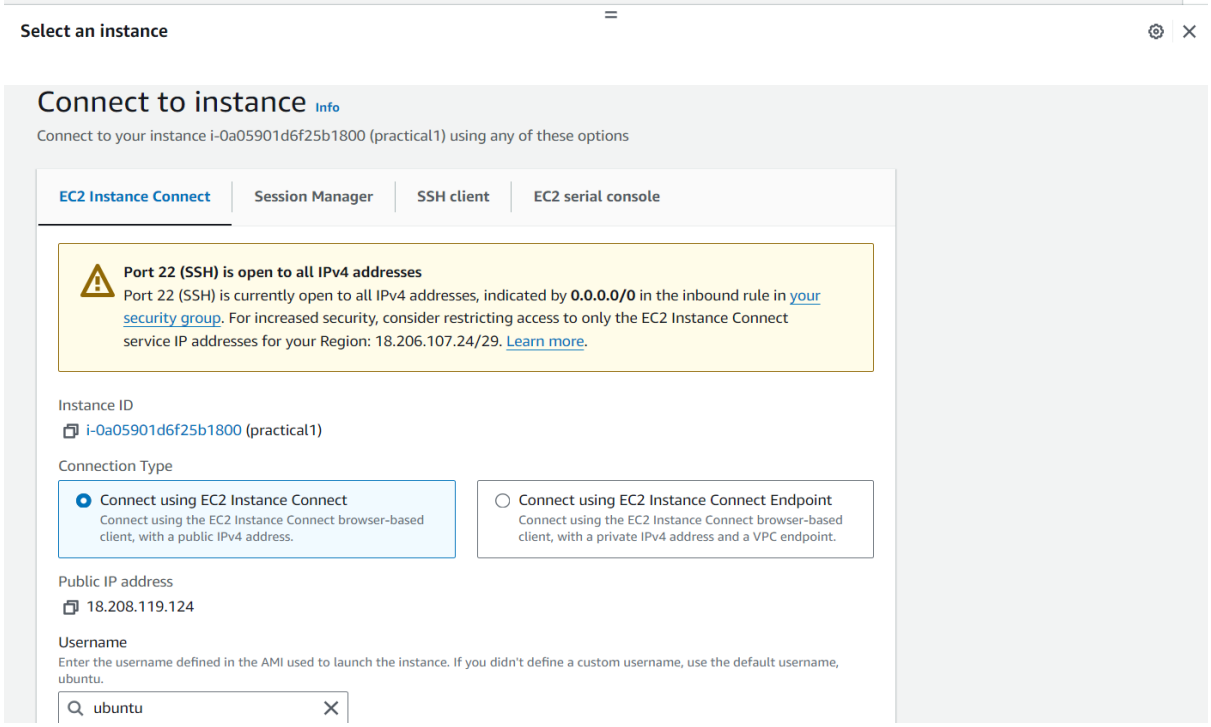

Create key pair
Key pair name
Key pairs allow you to connect to your instance securely.
practical1
The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.
Key pair type
RSA
RSA encrypted private and public key pair
ED25519
ED25519 encrypted private and public key pair
Private key file format
.pem
For use with OpenSSH
.ppk
For use with PuTTY
When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. Learn more
Cancel
Create key pair


practical1.pem
1,674 B • Done

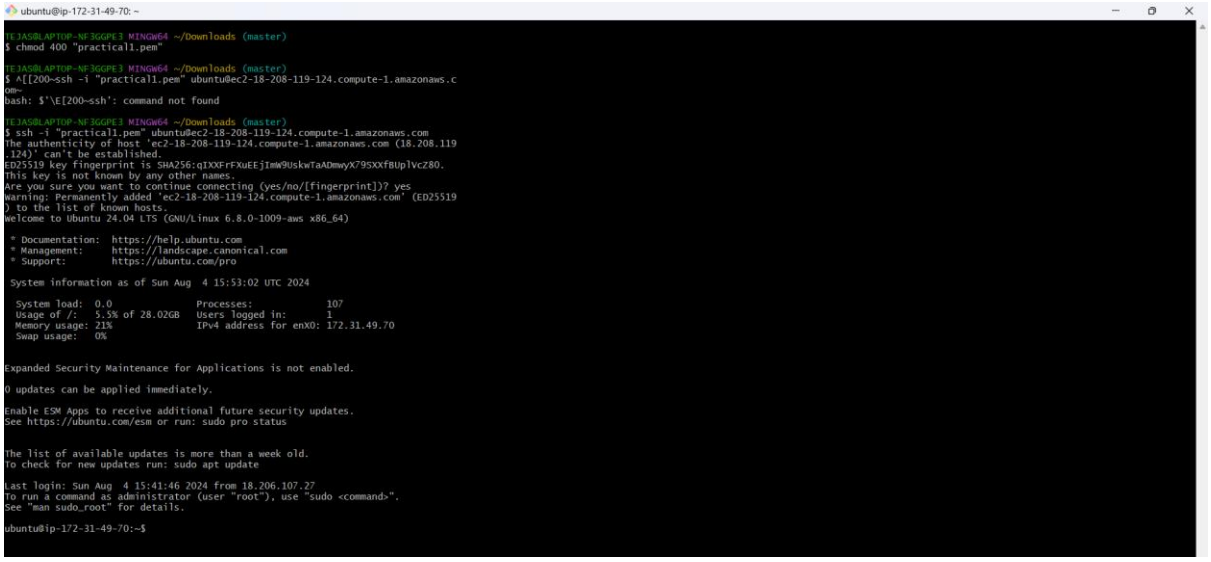

Services
Search
[Alt+S]
N. Virginia
Network Info
vpc-0d93715b8c7682b9a
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall 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
We'll create a new security group called 'launch-wizard-1' with the following rules:
Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0
Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server
Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.
Summary
Number of instances Info
1
Software Image (AMI)
Canonical, Ubuntu, 24.04 LTS, ...read more
ami-04a81a99f5ec58529
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 30 GiB
Free tier: In your first year includes 750 hours of t2.micro (or
Cancel
Launch instance


Services
Search
[Alt+S]
N. Virginia
EC2 > Instances > Launch an instance
Launching instance
Creating security group rules
21%
Details
Please wait while we launch your instance.
Do not close your browser while this is loading.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Instances (1) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance ID = i-0a05901d6f25b1800
Clear filters
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
practical1
i-0a05901d6f25b1800
Running
t2.micro
Initializing
View alarms
us-east-1e

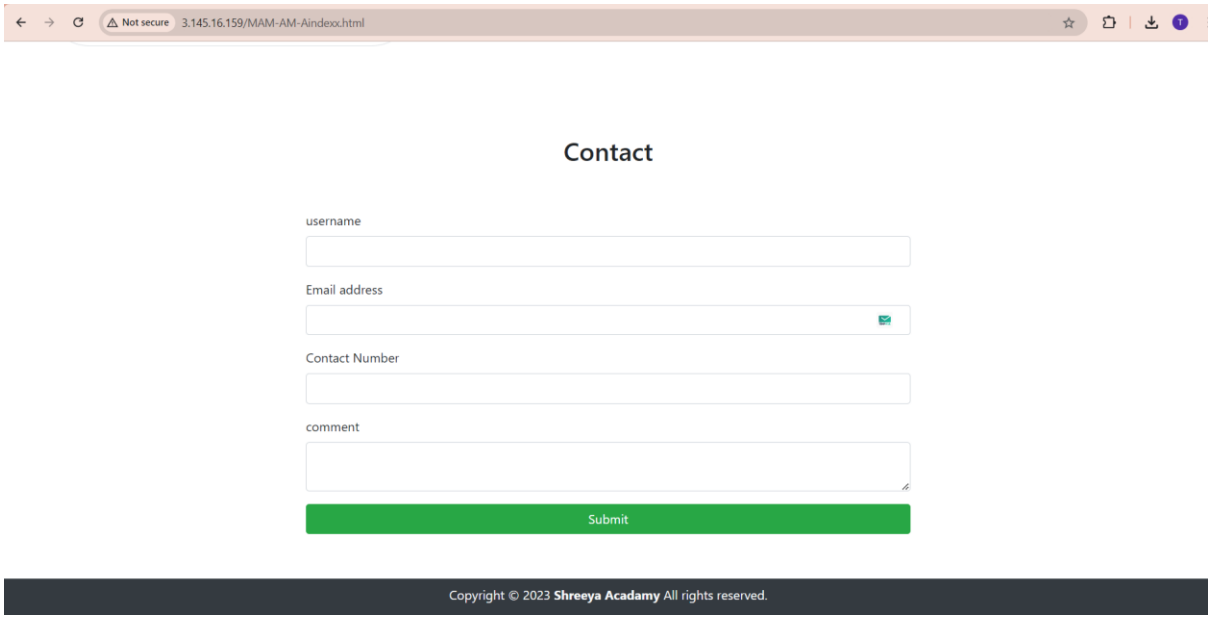

Select an instance
Connect to instance Info
Connect to your instance i-0a05901d6f25b1800 (practical1) using any of these options
EC2 Instance Connect
Session Manager
SSH client
EC2 serial console
Port 22 (SSH) is open to all IPv4 addresses
Port 22 (SSH) is currently open to all IPv4 addresses, indicated by 0.0.0.0/0 in the inbound rule in your security group. For increased security, consider restricting access to only the EC2 Instance Connect service IP addresses for your Region: 18.206.107.24/29. Learn more.
Instance ID
i-0a05901d6f25b1800 (practical1)
Connection Type
Connect using EC2 Instance Connect
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.
Connect using EC2 Instance Connect Endpoint
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.
Public IP address
18.208.119.124
Username
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.
ubuntu

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

ubuntu@ip-172-31-49-70:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        29G  1.6G   27G   6% /
tmpfs           479M     0  479M   0% /dev/shm
tmpfs           192M  868K  191M   1% /run
tmpfs           5.0M     0  5.0M   0% /run/lock
/dev/xvda16     881M   76M  744M  10% /boot
/dev/xvda15     105M  6.1M   99M   6% /boot/efi
tmpfs            96M   12K   96M   1% /run/user/1000
ubuntu@ip-172-31-49-70:~$ htop
ubuntu@ip-172-31-49-70:~$ ls
ubuntu@ip-172-31-49-70:~$
```




```
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_1.6.3-1.1ubuntu7_amd64.deb ...
```

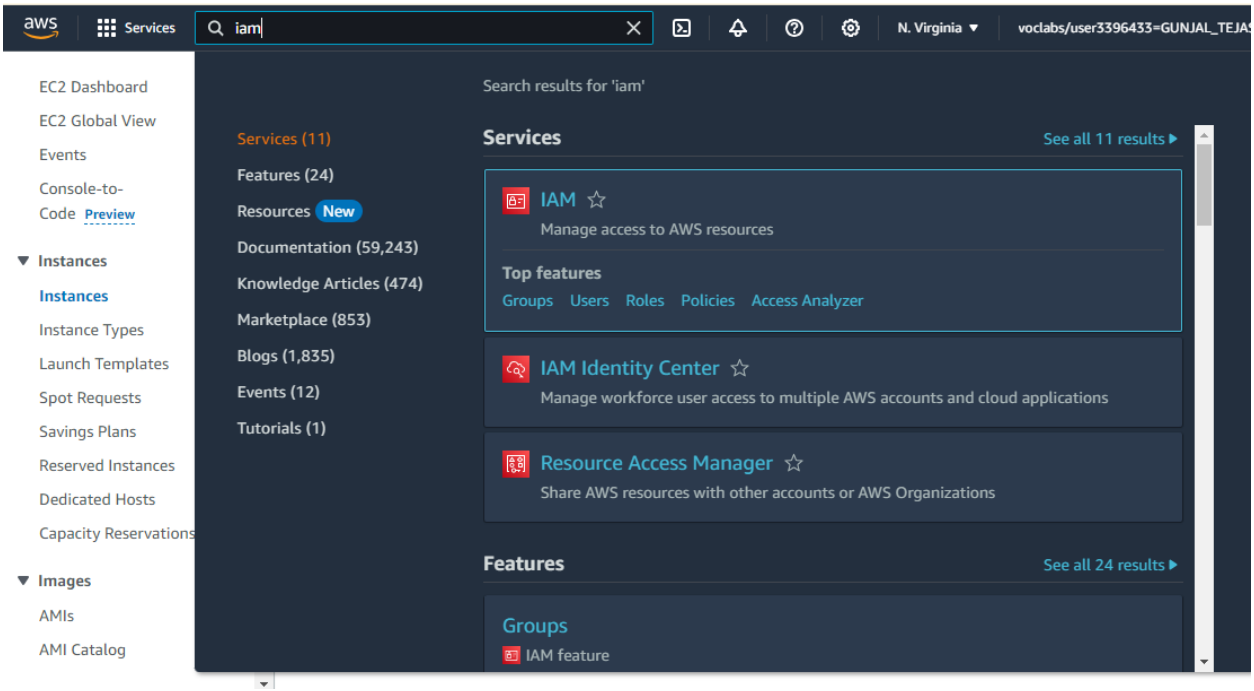
```
Running kernel seems to be up-to-date.

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@ip-172-31-5-93:~$ cd /var/www/html
ubuntu@ip-172-31-5-93:/var/www/html$ ls
index.html
ubuntu@ip-172-31-5-93:/var/www/html$ sudo rm index.html
ubuntu@ip-172-31-5-93:/var/www/html$ sudo nano indexx.html
ubuntu@ip-172-31-5-93:/var/www/html$
```

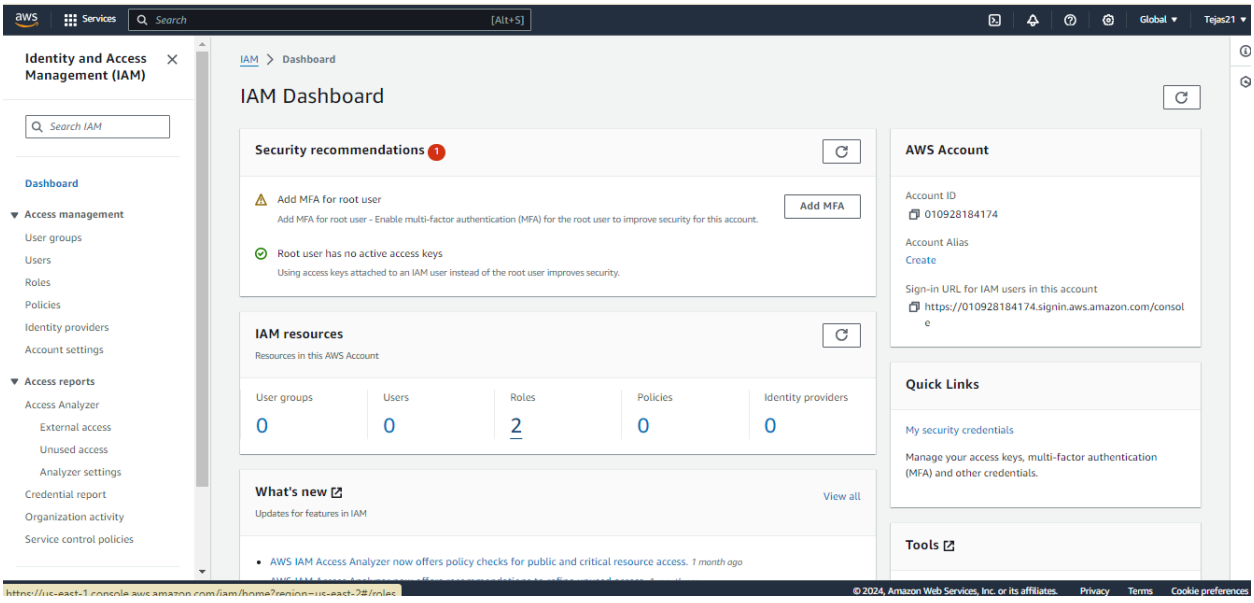


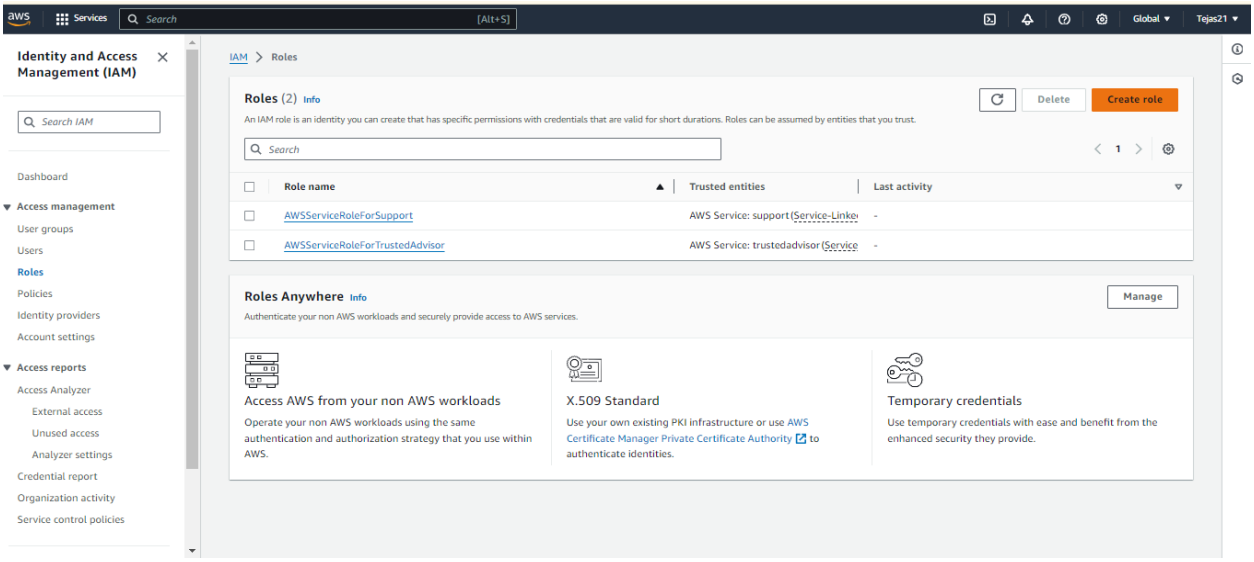
EXPERIMENT NO :- 02

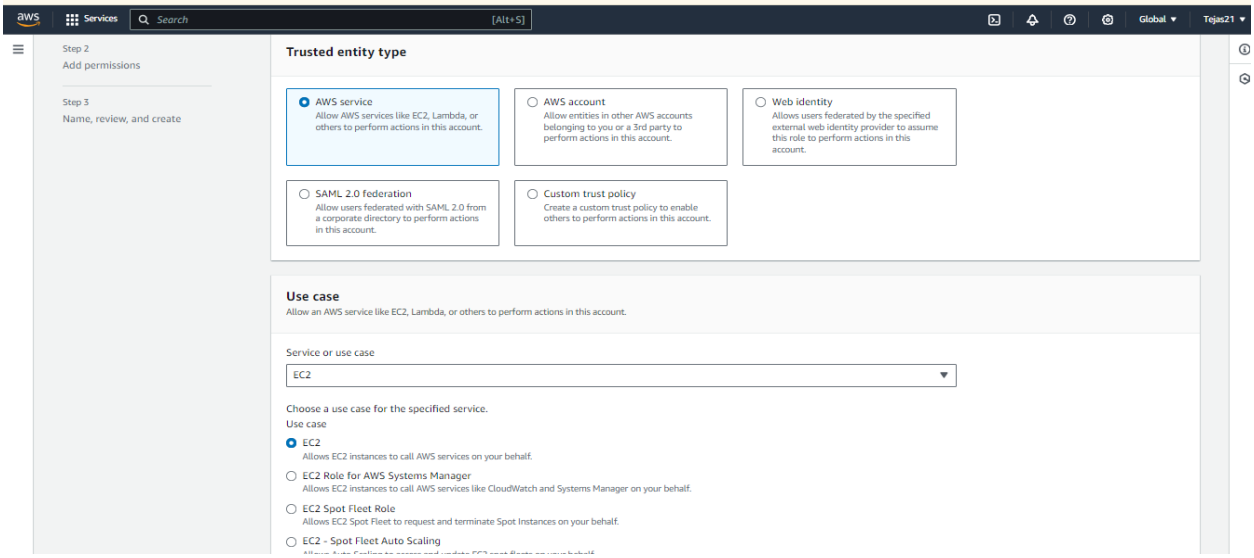
AIM:- To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS Code Pipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

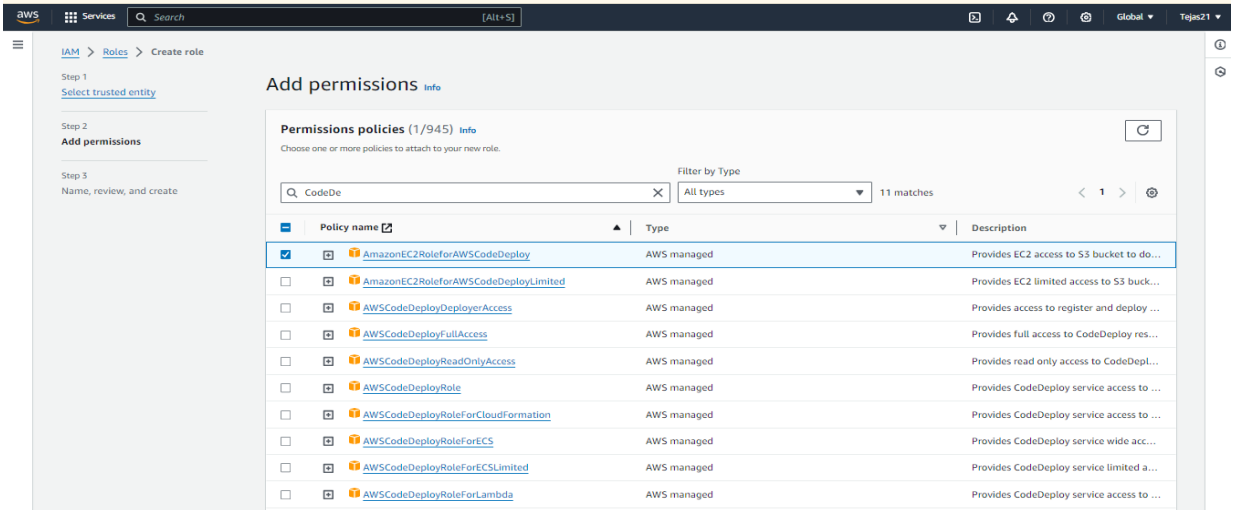
- Creating two roles in AWS IAM for CI/CD pipeline setup

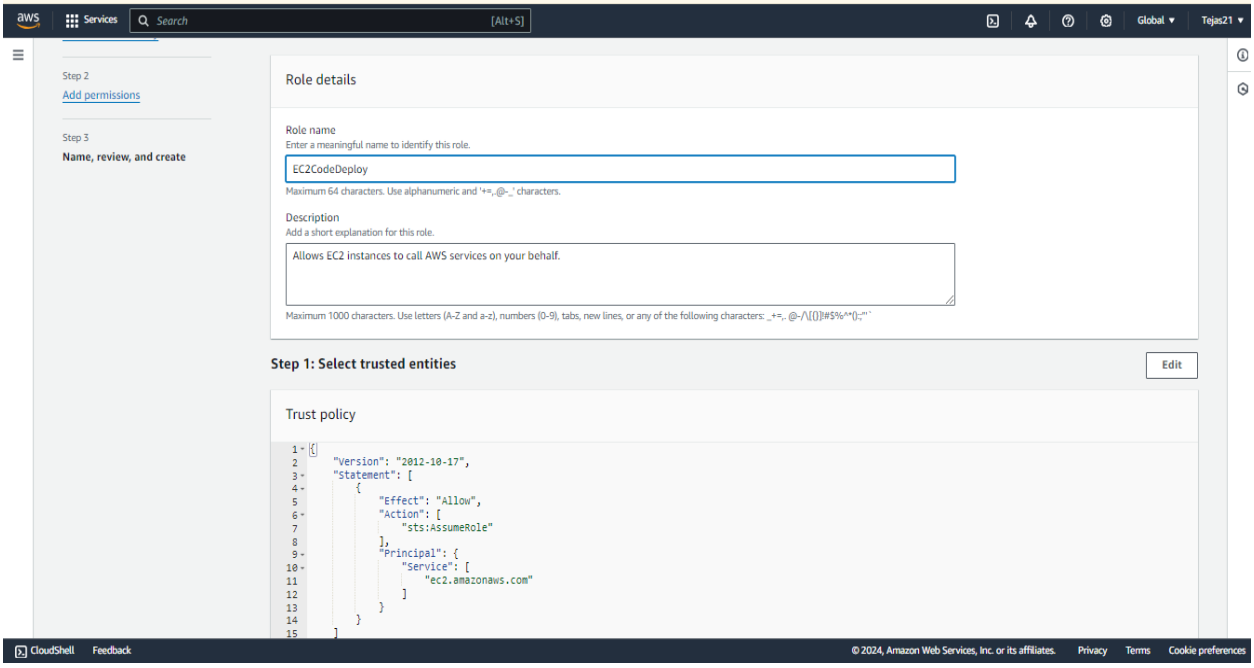


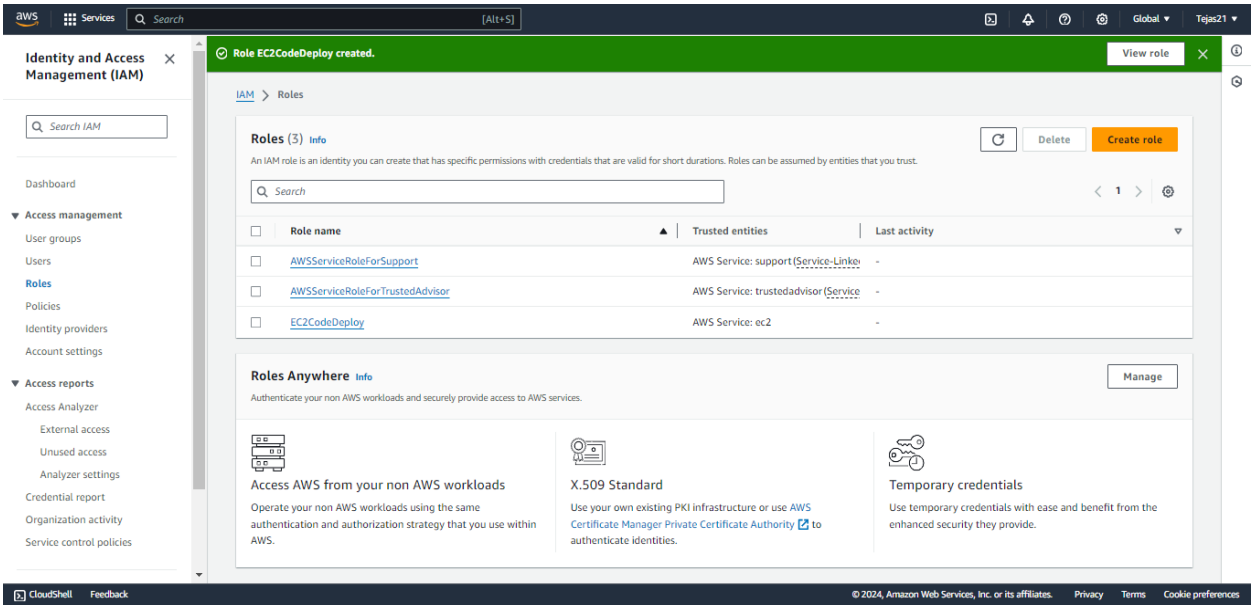


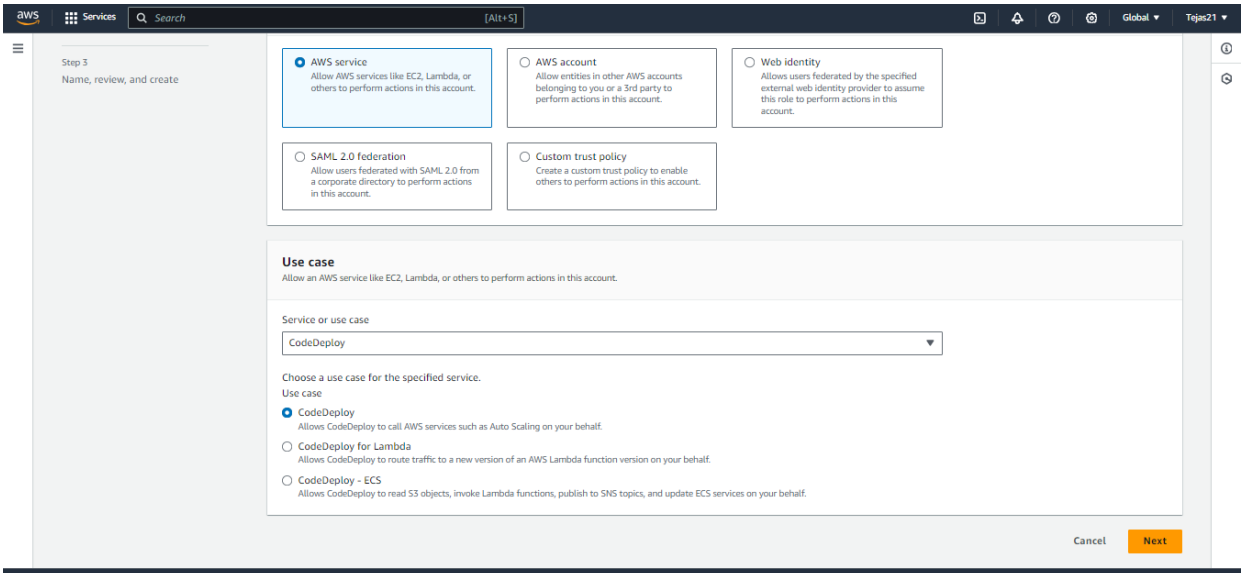

Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Credential report
Organization activity
Service control policies
IAM > Roles
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.
Delete
Create role
Search
Role name
Trusted entities
Last activity
AWSServiceRoleForSupport
AWS Service: support (Service-Linke
AWSServiceRoleForTrustedAdvisor
AWS Service: trustedadvisor (Service
Roles Anywhere Info
Authenticate your non AWS workloads and securely provide access to AWS services.
Manage
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.

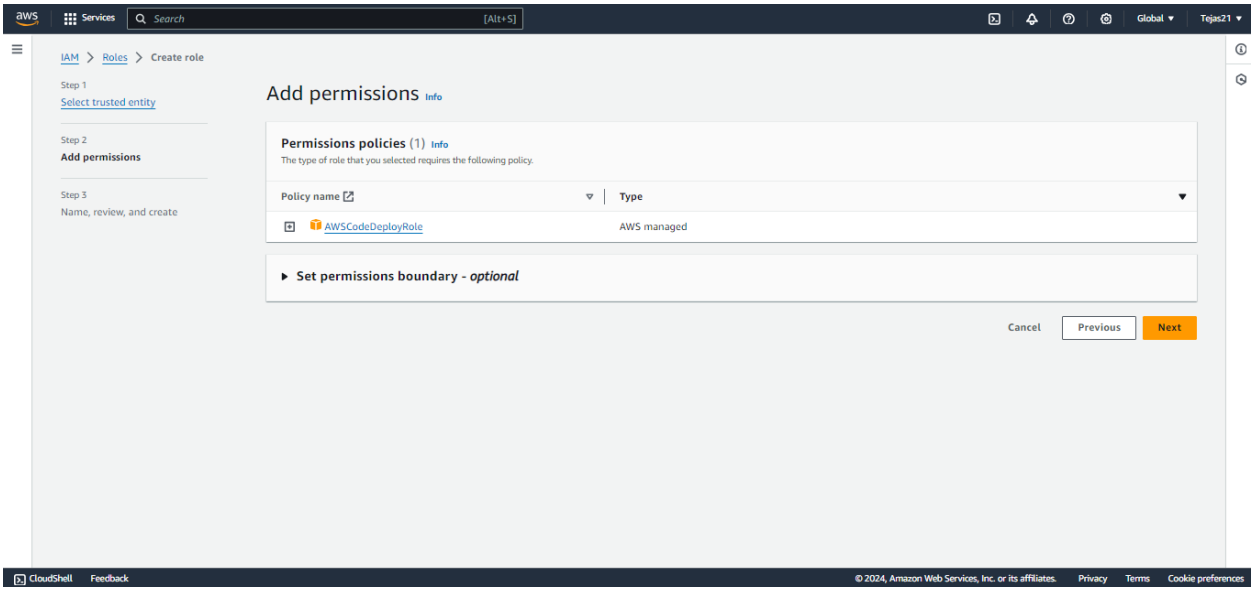

Step 2
Add permissions
Step 3
Name, review, and create
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
Web identity
Allows 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

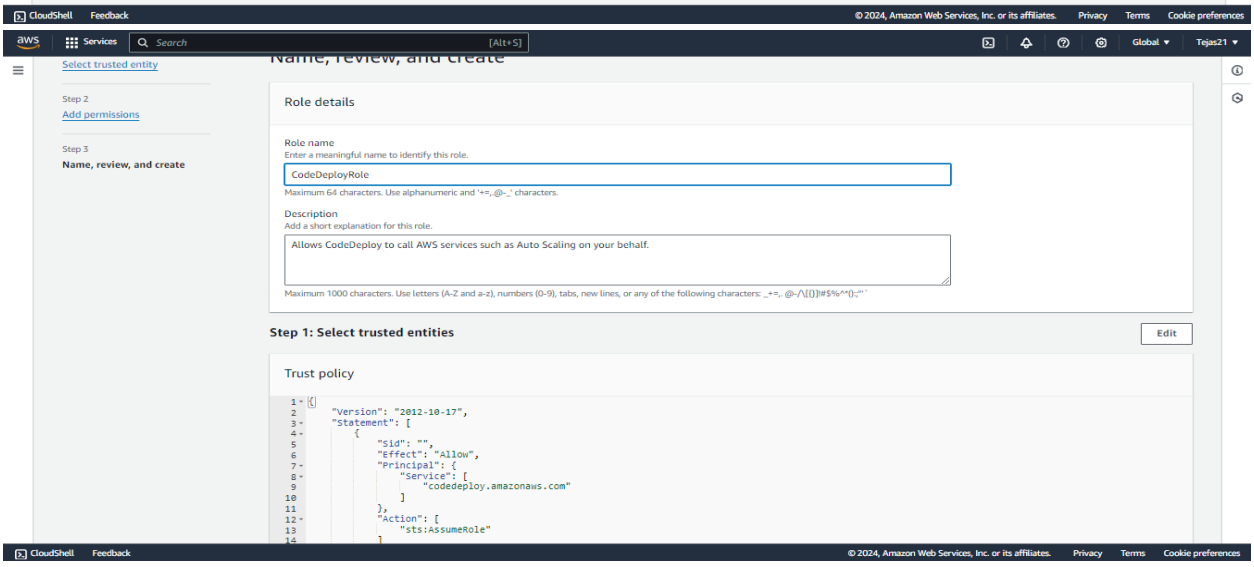

IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (1/945) Info
Choose one or more policies to attach to your new role.
Filter by Type
CodeDe
All types
11 matches
Policy name
Type
Description
AmazonEC2RoleforAWSCodeDeploy
AWS managed
Provides EC2 access to S3 bucket to do...
AmazonEC2RoleforAWSCodeDeployLimited
AWS managed
Provides EC2 limited access to S3 buck...
AWSCodeDeployDeployerAccess
AWS managed
Provides access to register and deploy ...
AWSCodeDeployFullAccess
AWS managed
Provides full access to CodeDeploy res...
AWSCodeDeployReadOnlyAccess
AWS managed
Provides read only access to CodeDepl...
AWSCodeDeployRole
AWS managed
Provides CodeDeploy service access to ...
AWSCodeDeployRoleForCloudFormation
AWS managed
Provides CodeDeploy service access to ...
AWSCodeDeployRoleForECS
AWS managed
Provides CodeDeploy service wide acc...
AWSCodeDeployRoleForECSLimited
AWS managed
Provides CodeDeploy service limited a...
AWSCodeDeployRoleForLambda
AWS managed
Provides CodeDeploy service access to ...


Step 2
Add permissions
Step 3
Name, review, and create
Role details
Role name
Enter a meaningful name to identify this role.
EC2CodeDeploy
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=, .@-/\[{}]!#$%^*();:"'`
Step 1: Select trusted entities
Edit
Trust policy
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"sts:AssumeRole"
],
"Principal": {
"Service": [
"ec2.amazonaws.com"
]
}
}
]
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

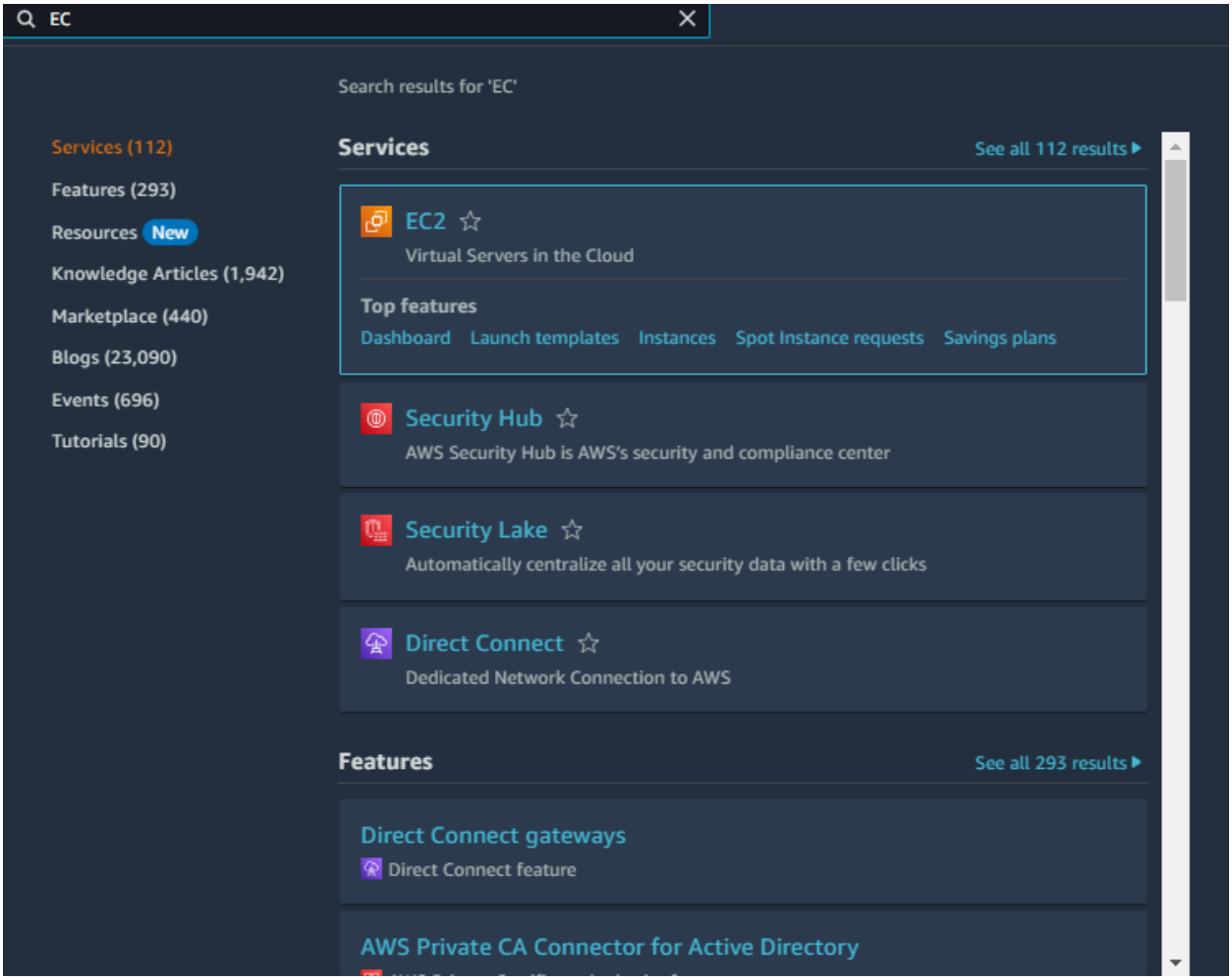

Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Credential report
Organization activity
Service control policies
Role EC2CodeDeploy created.
View role
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.
Delete
Create role
Search
Role name
Trusted entities
Last activity
AWSServiceRoleForSupport
AWS Service: support (Service-Linke
AWSServiceRoleForTrustedAdvisor
AWS Service: trustedadvisor (Service
EC2CodeDeploy
AWS Service: ec2
Roles Anywhere Info
Manage
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.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

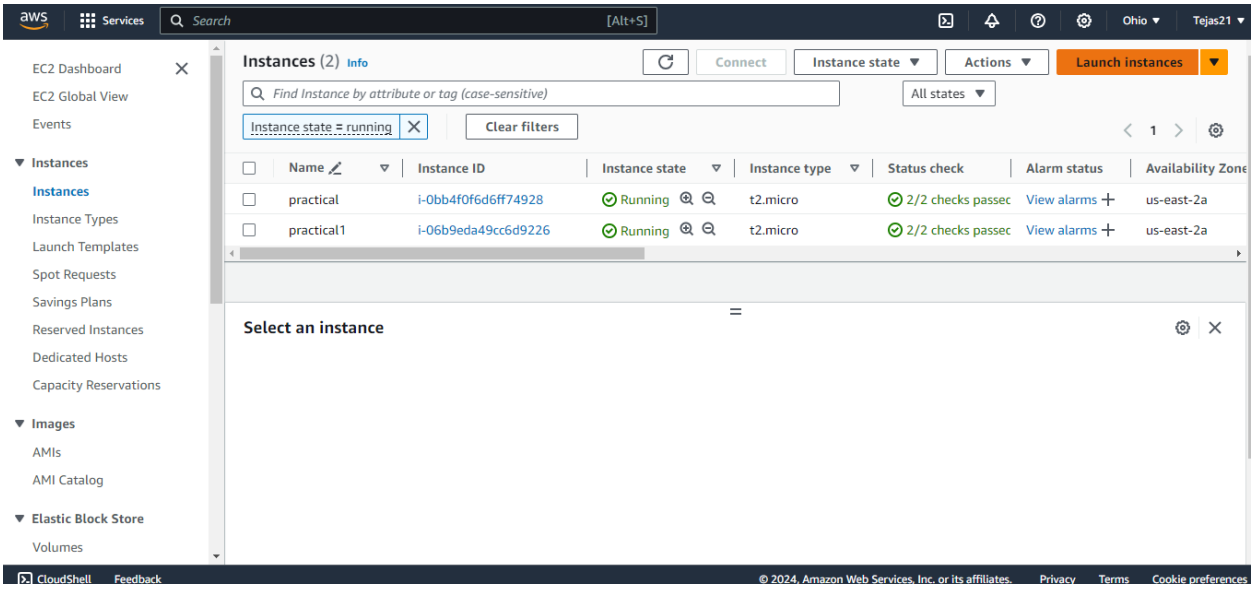

Step 3
Name, review, and create
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
Web identity
Allows users federated by the specified external 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.
Cancel
Next

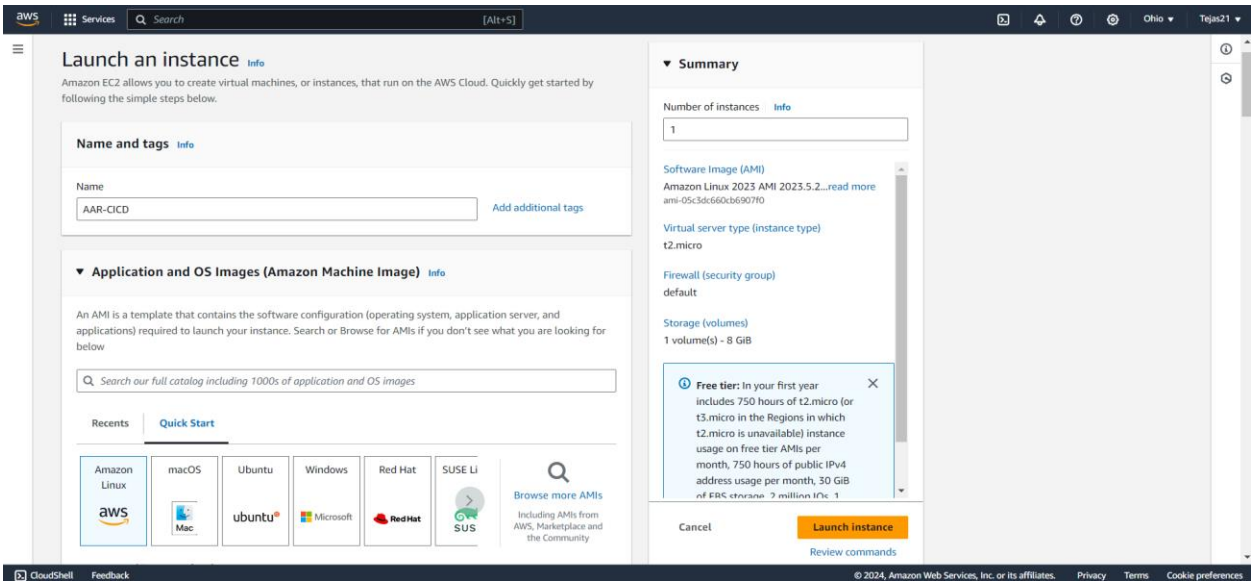

IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (1) Info
The type of role that you selected requires the following policy.
Policy name
Type
AWSCodeDeployRole
AWS managed
Set permissions boundary - optional
Cancel
Previous
Next

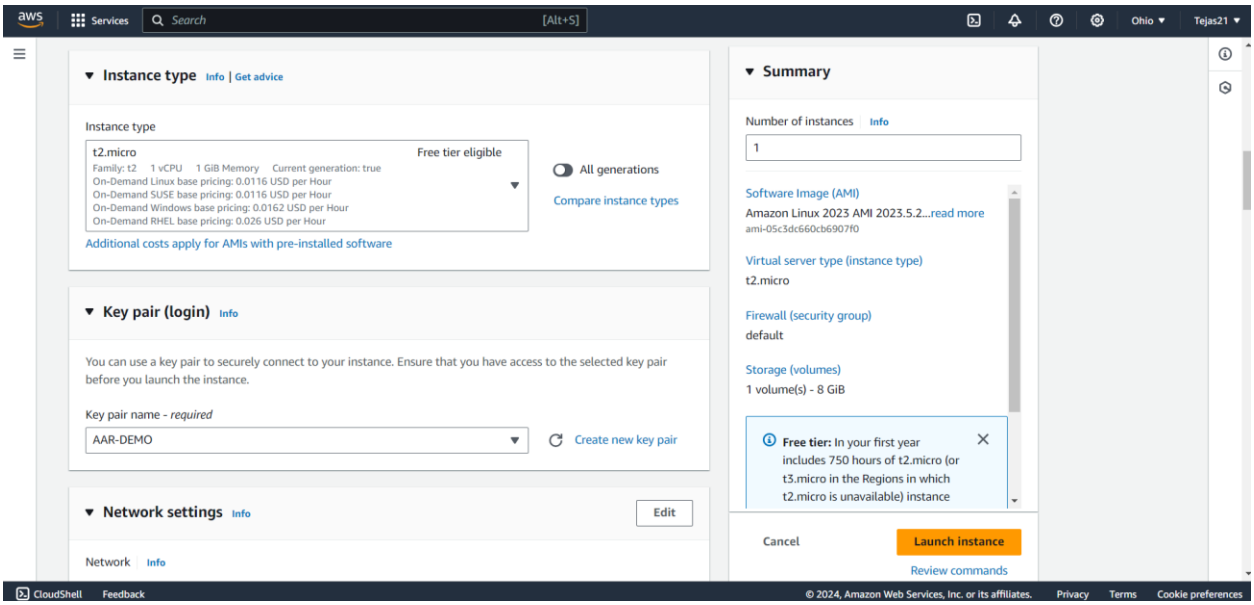

Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Role details
Role name
Enter a meaningful name to identify this role.
CodeDeployRole
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.
Step 1: Select trusted entities
Edit
Trust policy
{
"Version": "2012-10-17",
"Statement": [
{
"Sid": "",
"Effect": "Allow",
"Principal": {
"Service": [
"codedeploy.amazonaws.com"
]
},
"Action": [
"sts:AssumeRole"
]

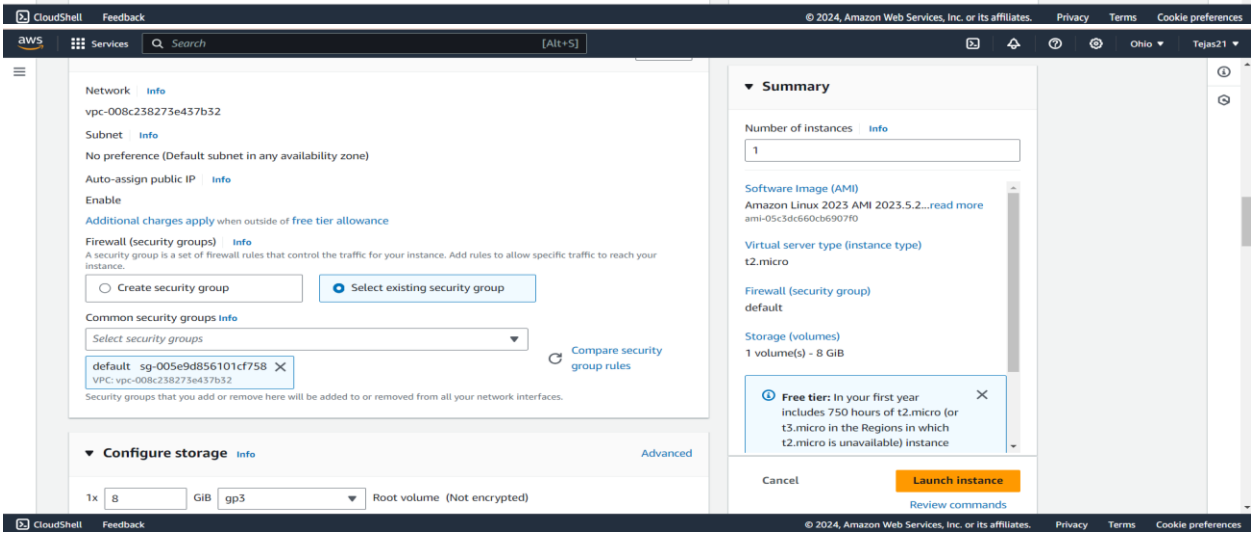
➢ Launching & Setting up EC2 Instance for Deployment

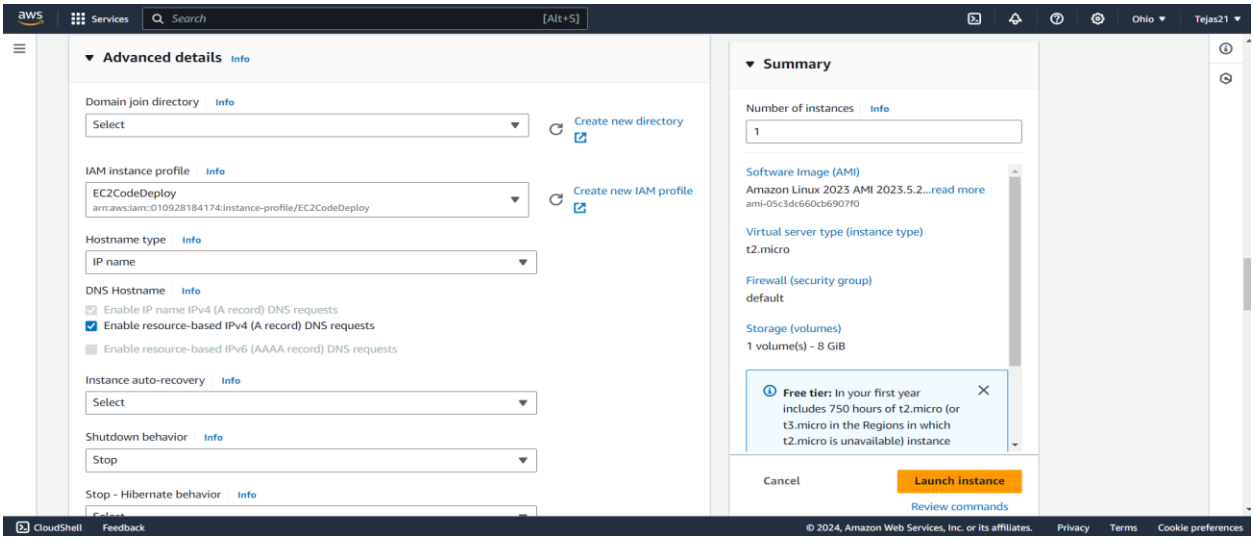


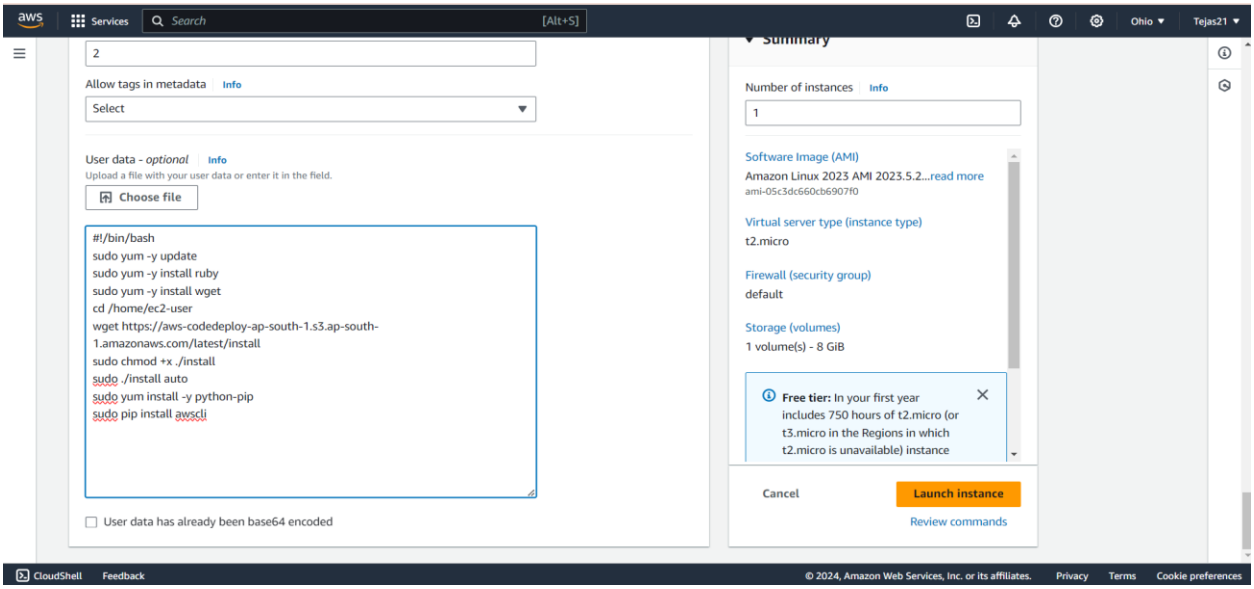


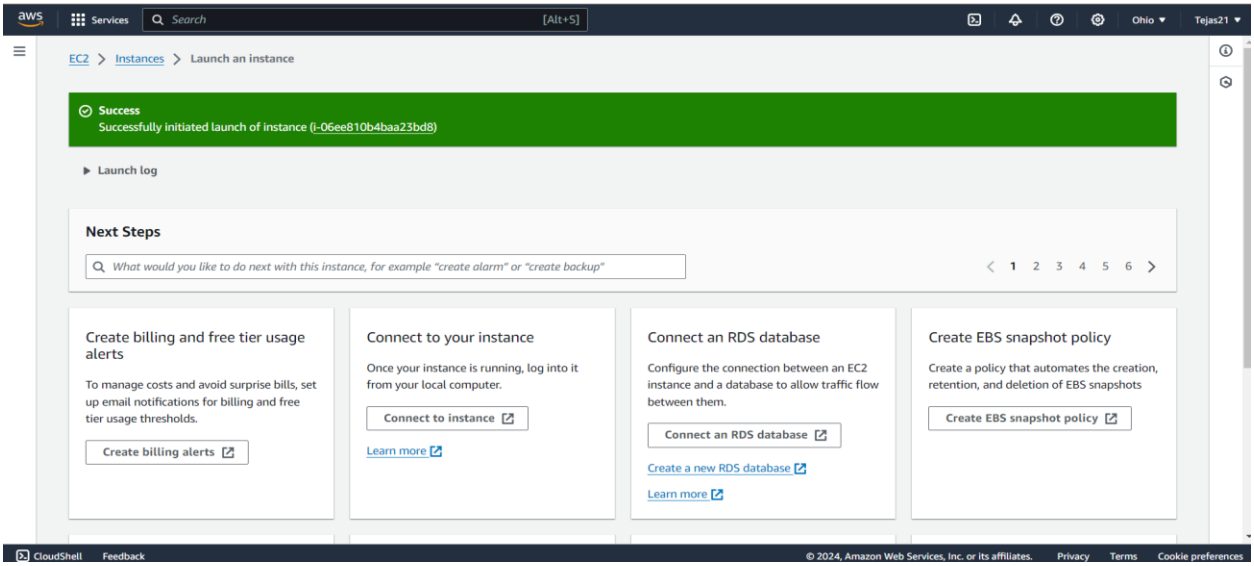

Launch an instance Info
Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.
Name and tags Info
Name
AAR-CICD
Add additional tags
Application and OS Images (Amazon Machine Image) Info
An AMI is a template that 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
Search our full catalog including 1000s of application and OS images
Recents
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Li
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
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 usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB
Cancel
Launch instance
Review commands
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Instance type Info | Get advice
Instance type
t2.micro
Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0116 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour
All generations
Compare instance types
Additional costs apply for AMIs with pre-installed software
Key pair (login) Info
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.
Key pair name - required
AAR-DEMO
Create new key pair
Network settings Info
Edit
Network Info
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
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

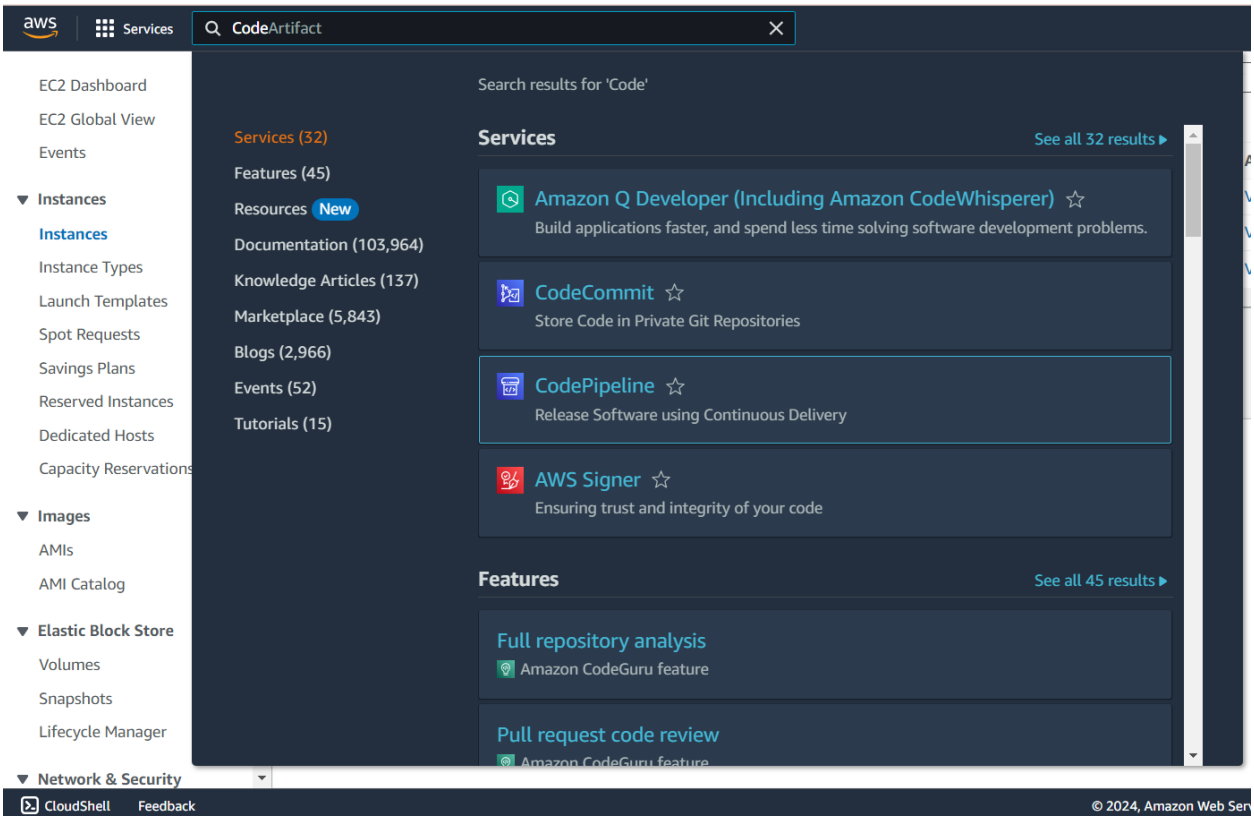

Network Info
vpc-008c238273e437b32
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for 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
VPC: vpc-008c238273e437b32
Compare security group rules
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
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

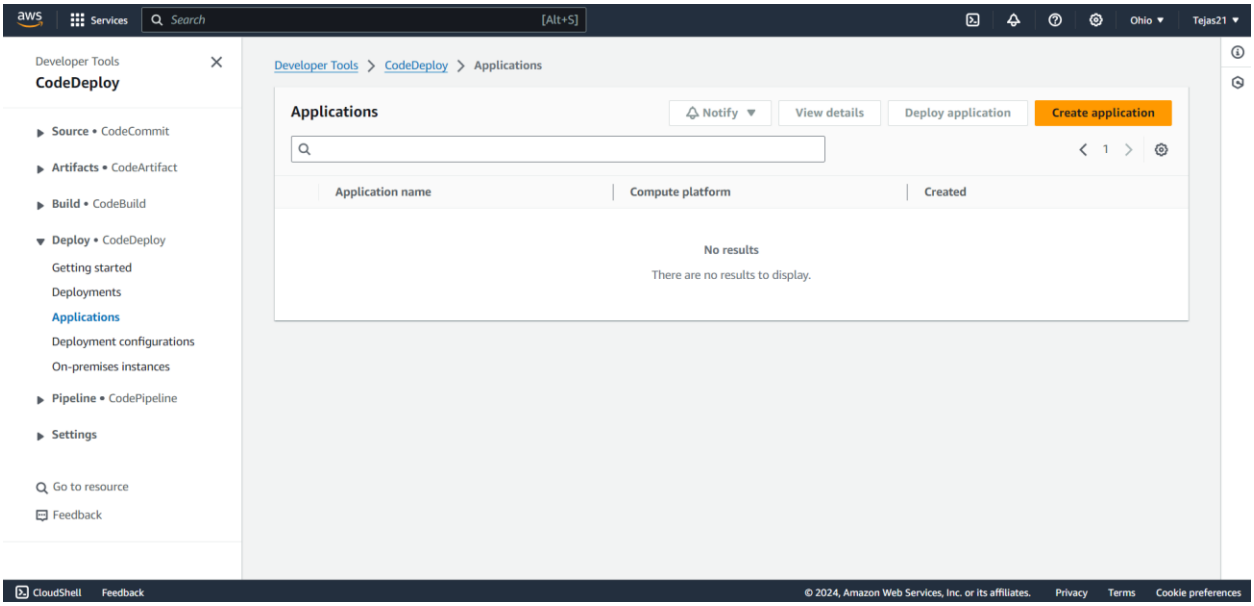

Advanced details Info
Domain join directory Info
Select
Create new directory
IAM instance profile Info
EC2CodeDeploy
arn:aws:iam::010928184174:instance-profile/EC2CodeDeploy
Create new IAM profile
Hostname type Info
IP name
DNS Hostname Info
Enable IP name IPv4 (A record) DNS requests
Enable resource-based 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
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

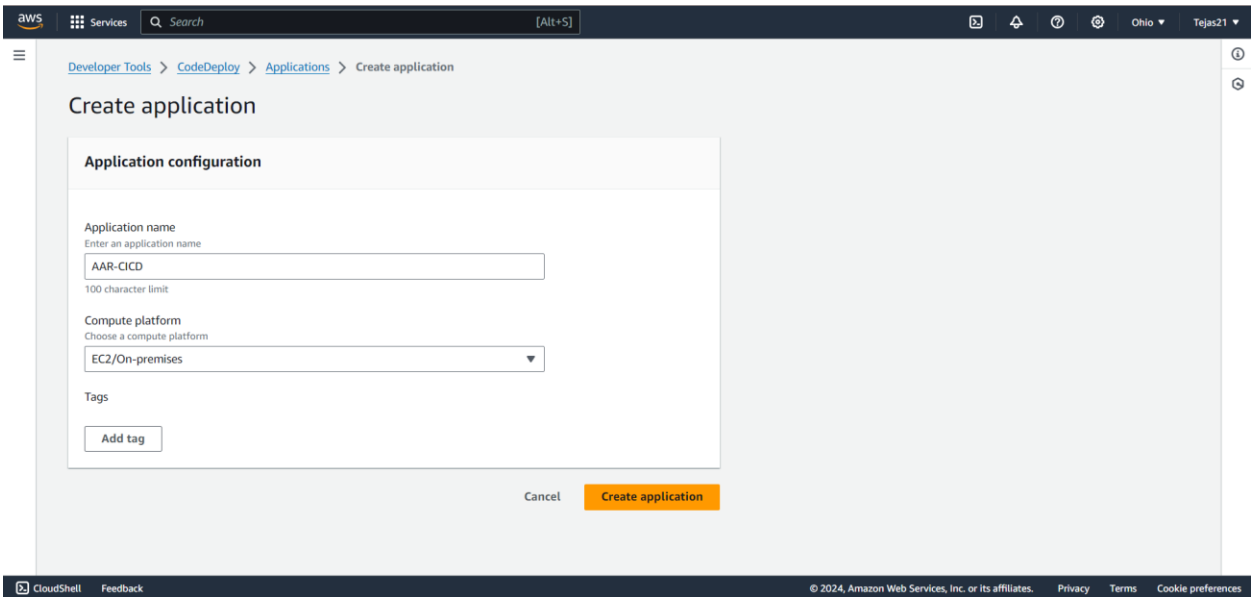

2
Allow tags in metadata Info
Select
User data - optional Info
Upload a file with your user data or enter it in the field.
Choose file
#!/bin/bash
sudo yum -y update
sudo yum -y install ruby
sudo yum -y install wget
cd /home/ec2-user
wget https://aws-codedeploy-ap-south-1.s3.ap-south-1.amazonaws.com/latest/install
sudo chmod +x ./install
sudo ./install auto
sudo yum install -y python-pip
sudo pip install awscli
User data has already been base64 encoded
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

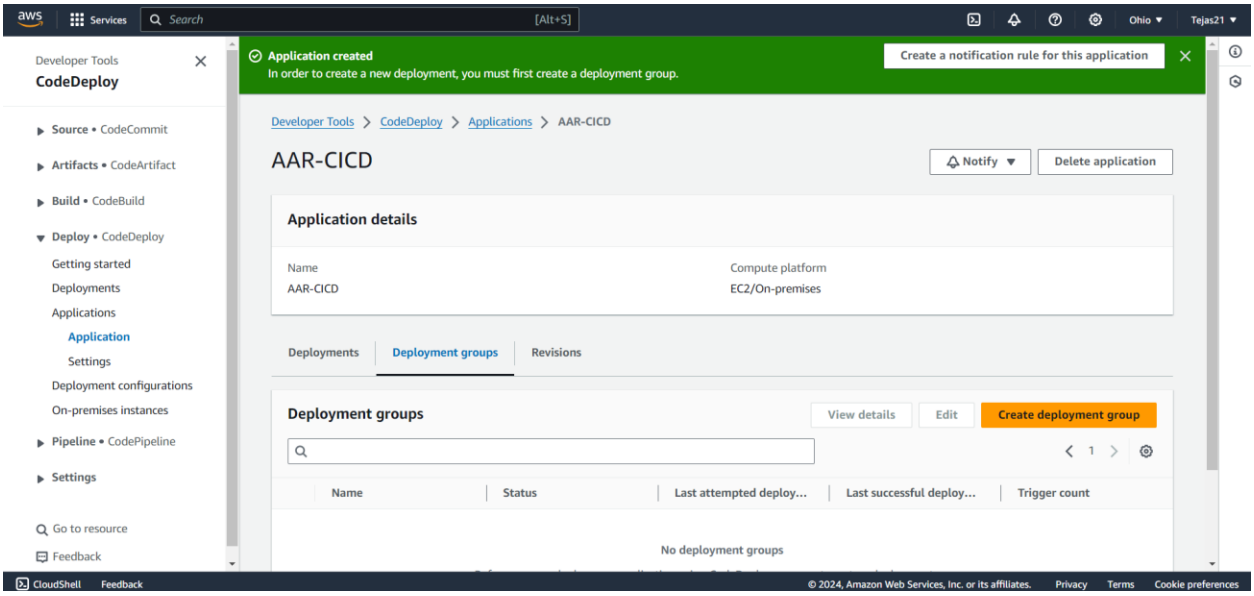

EC2 > Instances > Launch an instance
Success
Successfully initiated launch of instance (i-06ee810b4baa23bd8)
Launch log
Next Steps
What would you like to do next with this instance, for example "create alarm" or "create backup"
Create billing and free tier usage alerts
To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.
Create billing alerts
Connect to your instance
Once your instance is running, log into it from your local computer.
Connect to instance
Learn more
Connect an RDS database
Configure the connection between an EC2 instance and a database to allow traffic flow between them.
Connect an RDS database
Create a new RDS database
Learn more
Create EBS snapshot policy
Create a policy that automates the creation, retention, and deletion of EBS snapshots
Create EBS snapshot policy

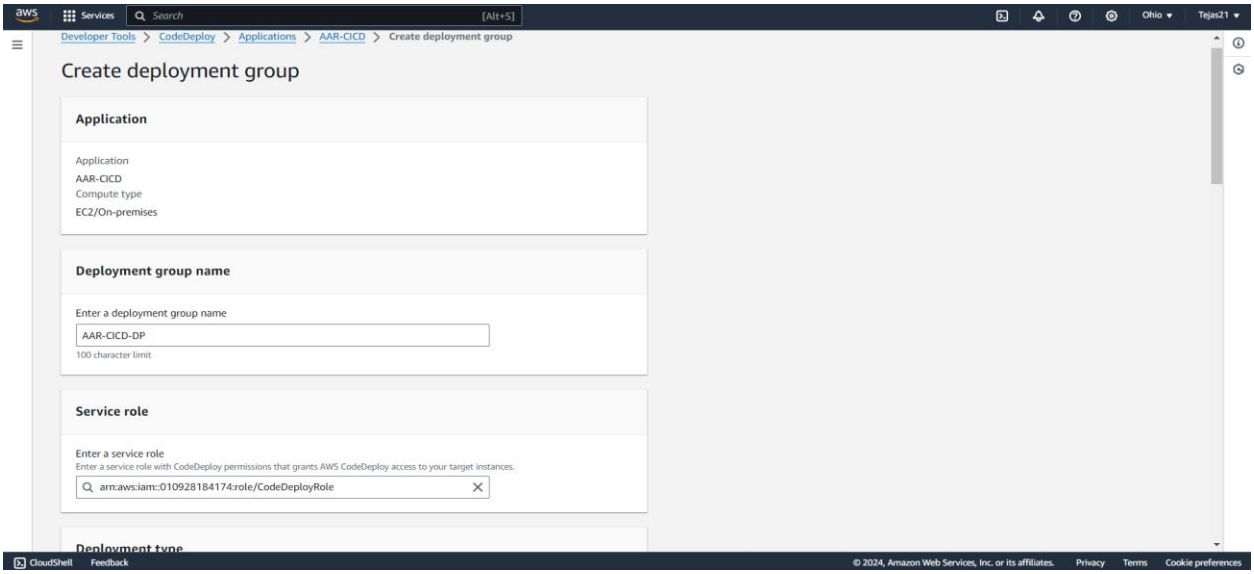
➢ Creating an application and deployment group in AWS CodeDeploy

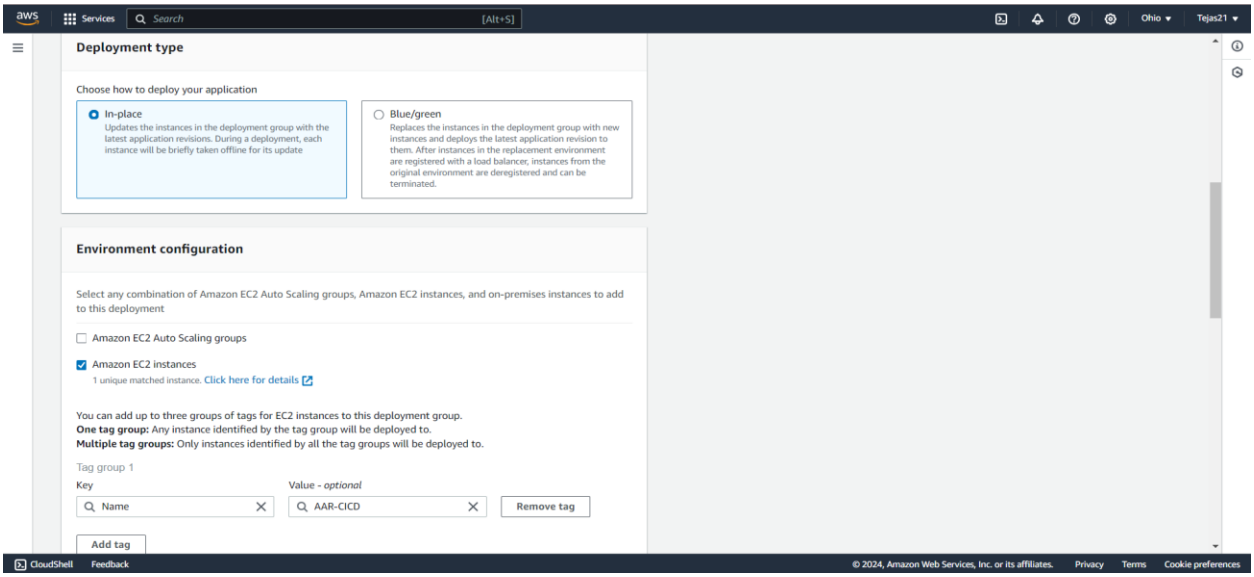


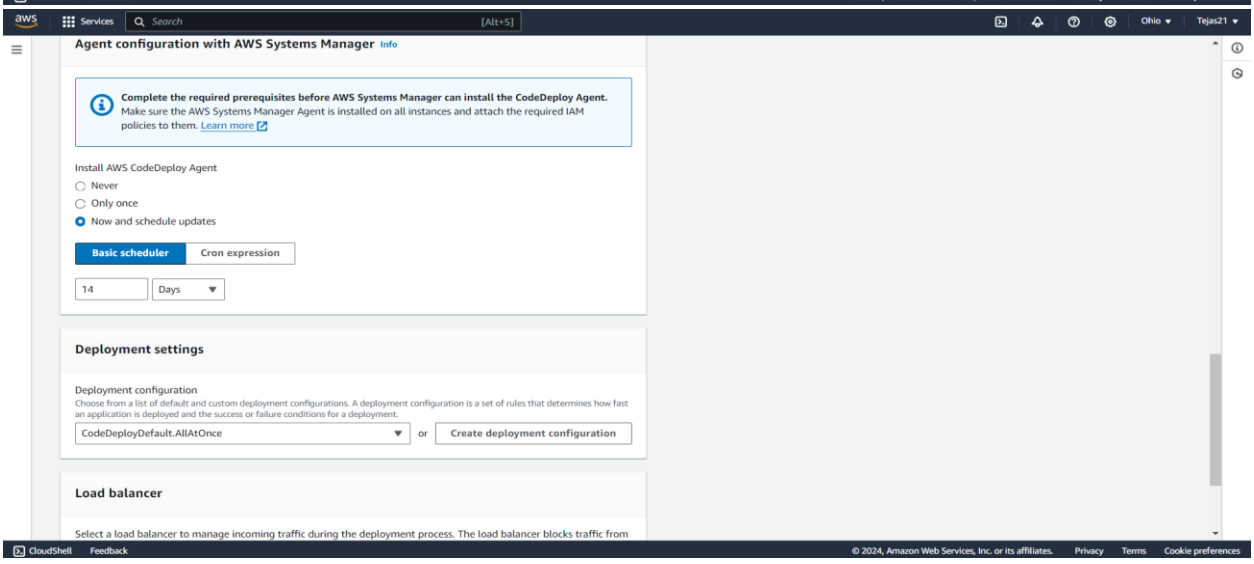



Services
Search
[Alt+S]
Ohio
Tejas21
Developer Tools > CodeDeploy > Applications > Create application
Create application
Application configuration
Application name
Enter an application name
AAR-CICD
100 character limit
Compute platform
Choose a compute platform
EC2/On-premises
Tags
Add tag
Cancel
Create application
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

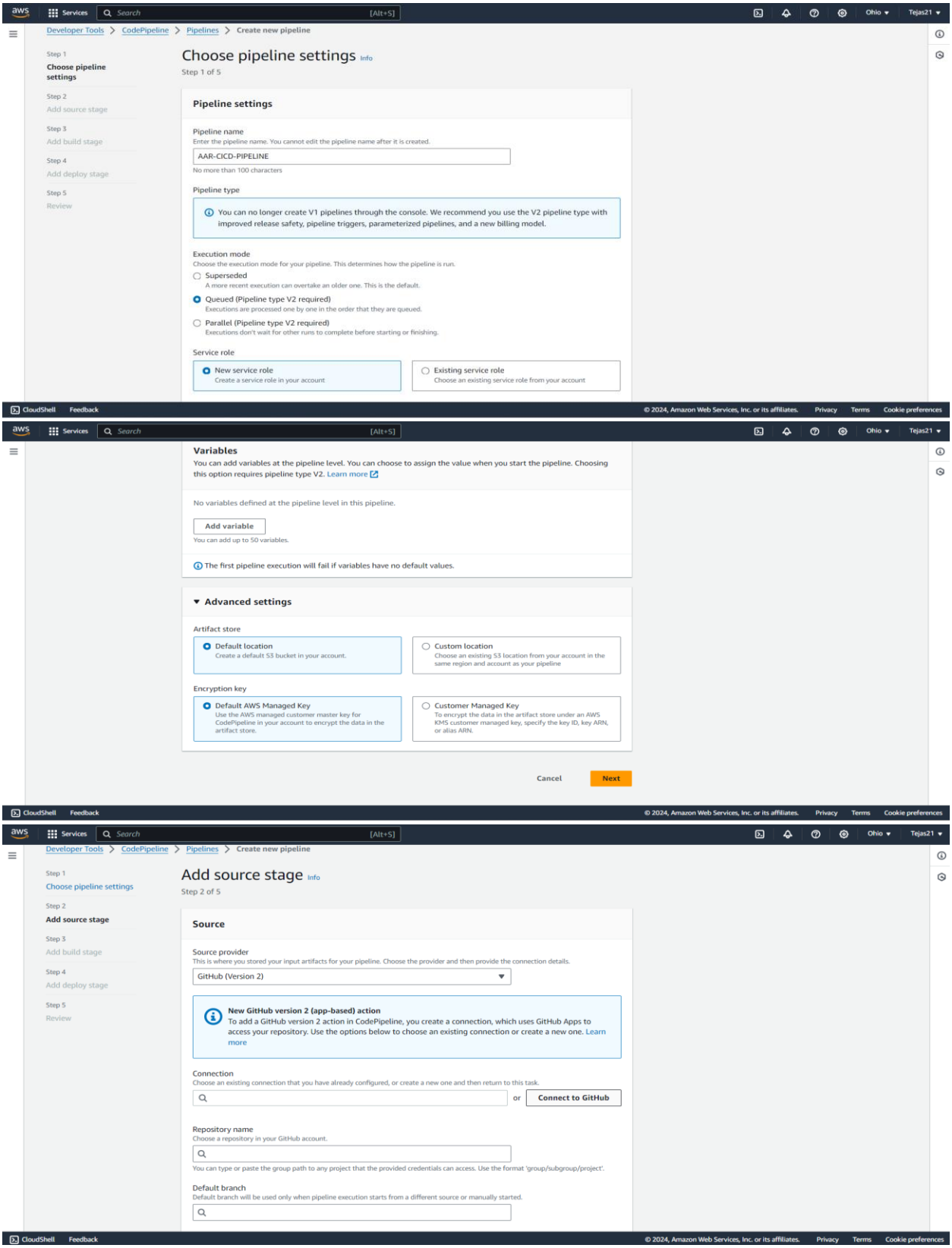

Services
Search
[Alt+S]
Ohio
Tejas21
Developer Tools
CodeDeploy
Source • CodeCommit
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Applications
Application
Settings
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Settings
Go to resource
Feedback
Application created
In order to create a new deployment, you must first create a deployment group.
Create a notification rule for this application
Developer Tools > CodeDeploy > Applications > AAR-CICD
AAR-CICD
Notify
Delete application
Application details
Name
AAR-CICD
Compute platform
EC2/On-premises
Deployments
Deployment groups
Revisions
Deployment groups
View details
Edit
Create deployment group
Name
Status
Last attempted deploy...
Last successful deploy...
Trigger count
No deployment groups
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Services
Search
[Alt+S]
Ohio
Tejas21
Developer Tools > CodeDeploy > Applications > AAR-CICD > Create deployment group
Create deployment group
Application
Application
AAR-CICD
Compute type
EC2/On-premises
Deployment group name
Enter a deployment group name
AAR-CICD-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
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

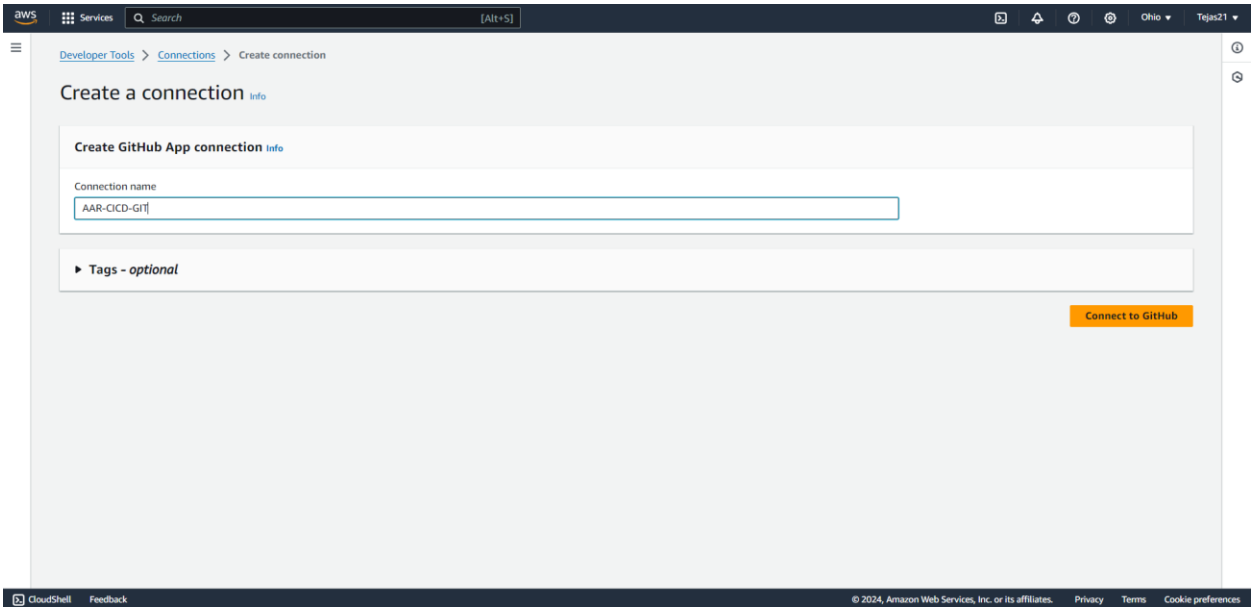

Services
Search
[Alt+S]
Ohio
Tejas21
Deployment type
Choose how to deploy your application
In-place
Updates the instances in the deployment group with the latest application revisions. During a deployment, each instance will be briefly taken offline for its update
Blue/green
Replaces the instances in 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
Value - optional
Name
AAR-CICD
Remove tag
Add tag
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

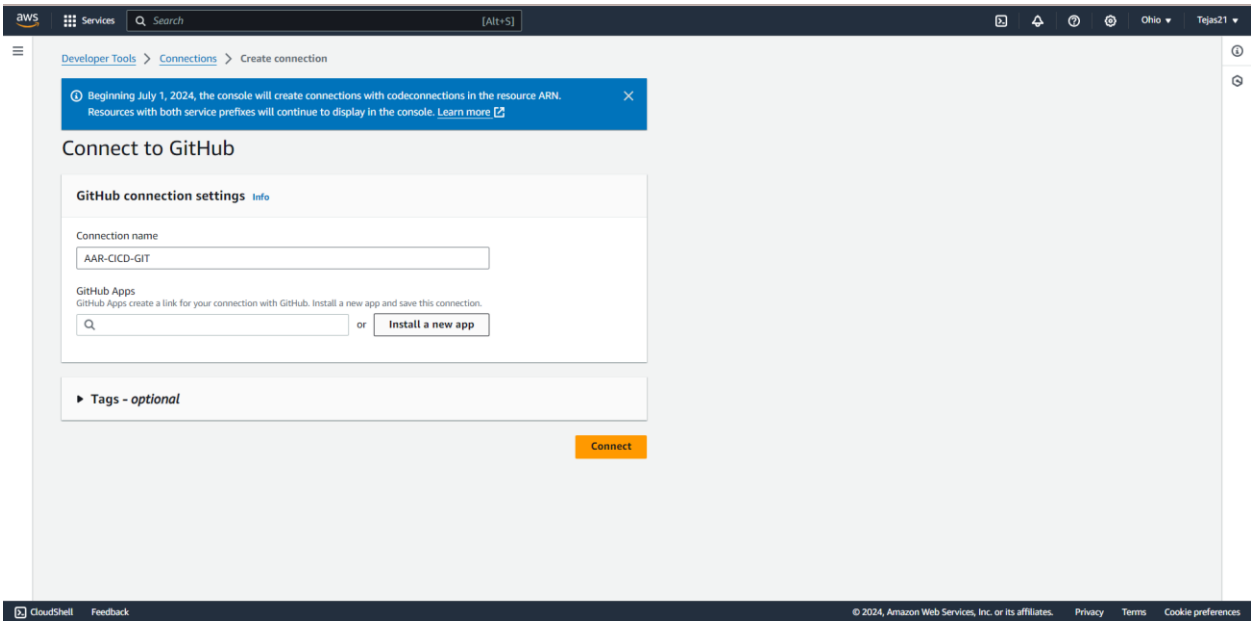

Services
Search
[Alt+S]
Ohio
Tejas21
Agent configuration with AWS Systems Manager Info
Complete the required prerequisites before AWS Systems Manager can install the CodeDeploy Agent.
Make sure the AWS Systems Manager Agent is installed on all instances and attach the required IAM policies to them. Learn 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
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

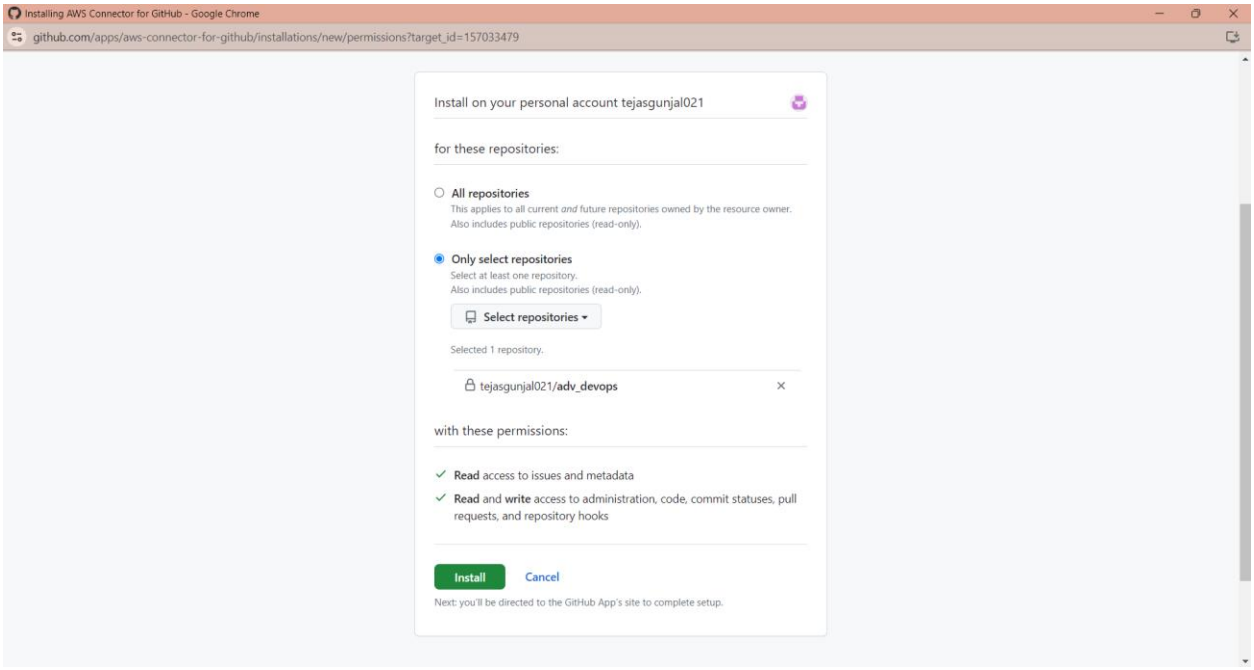
- Setting up a CICD pipeline for deploying applications on EC2 using GitHub.

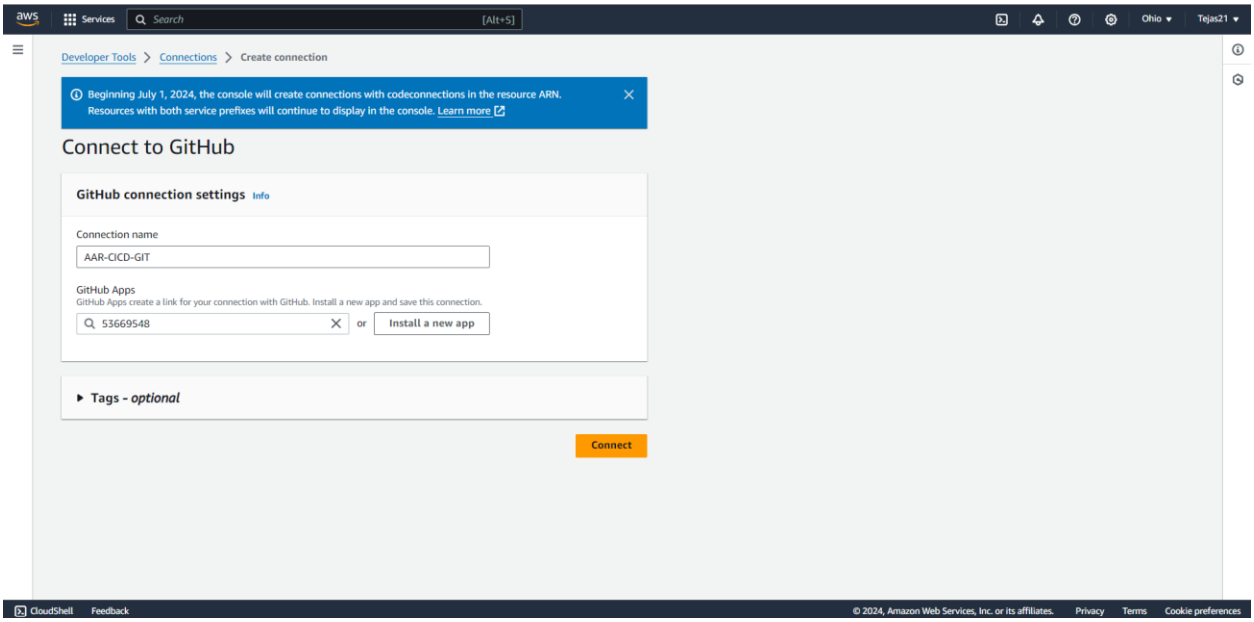


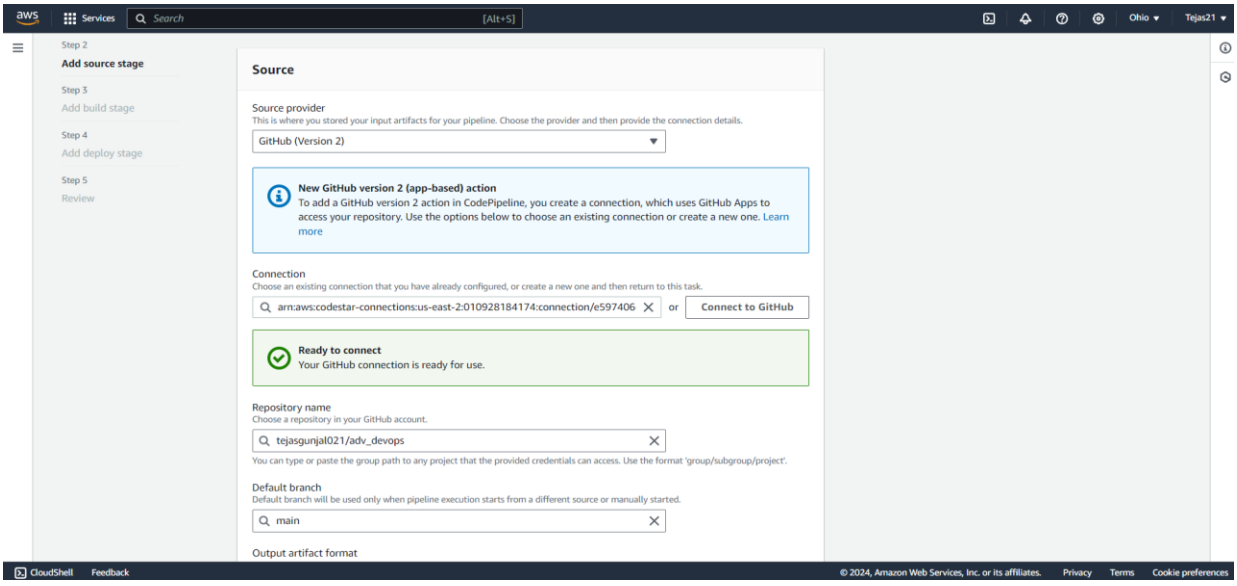
- Connecting and setting up GitHub for CI/CD deployment on AWS CodeDeploy.

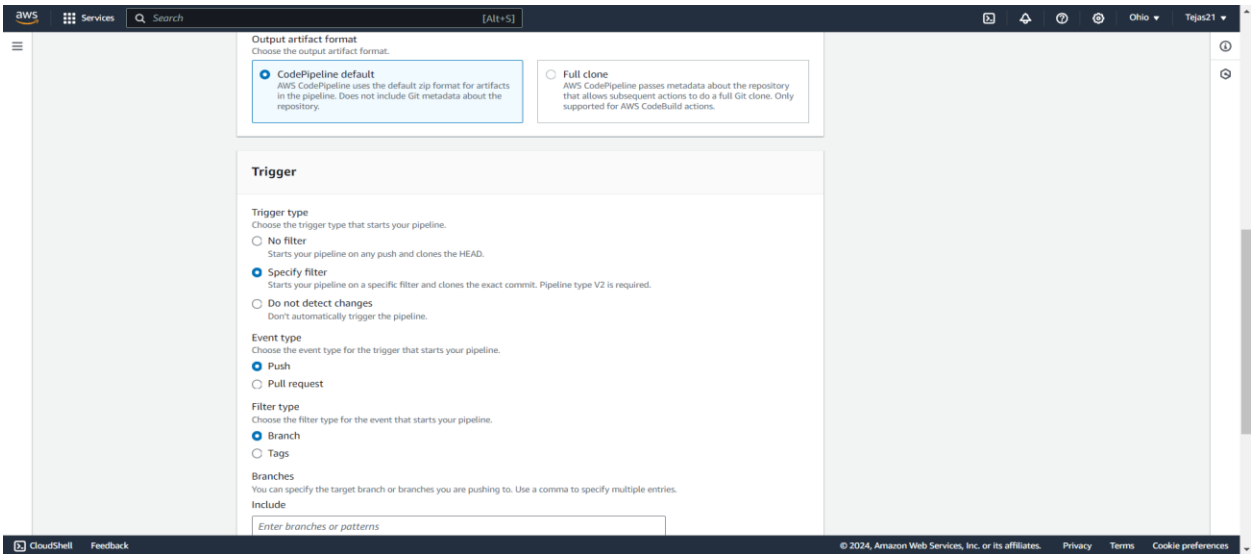


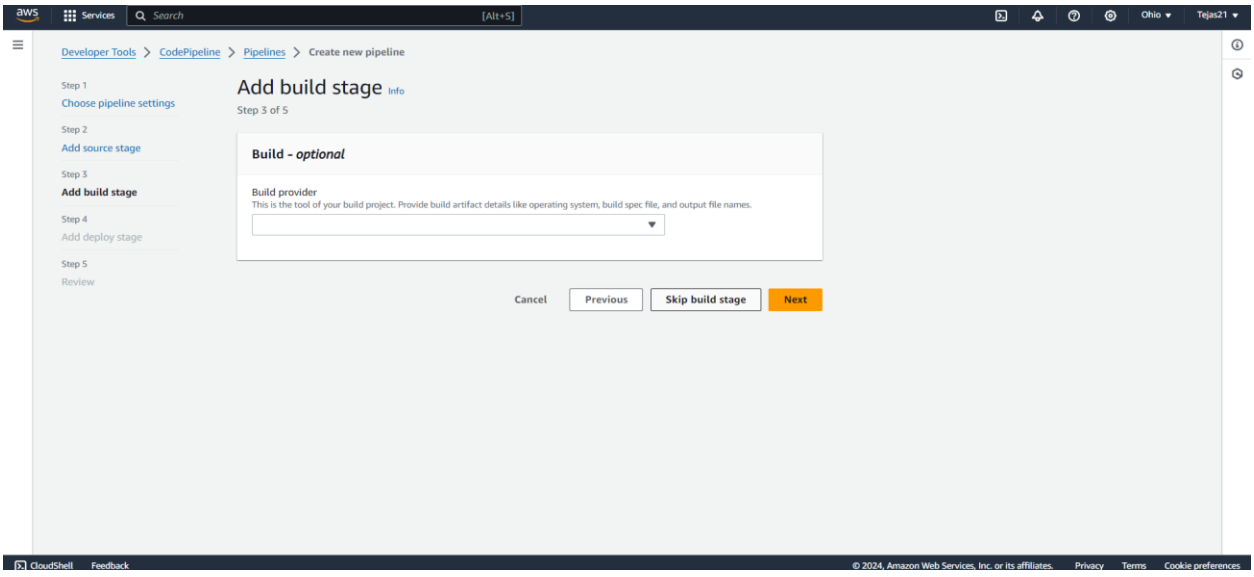


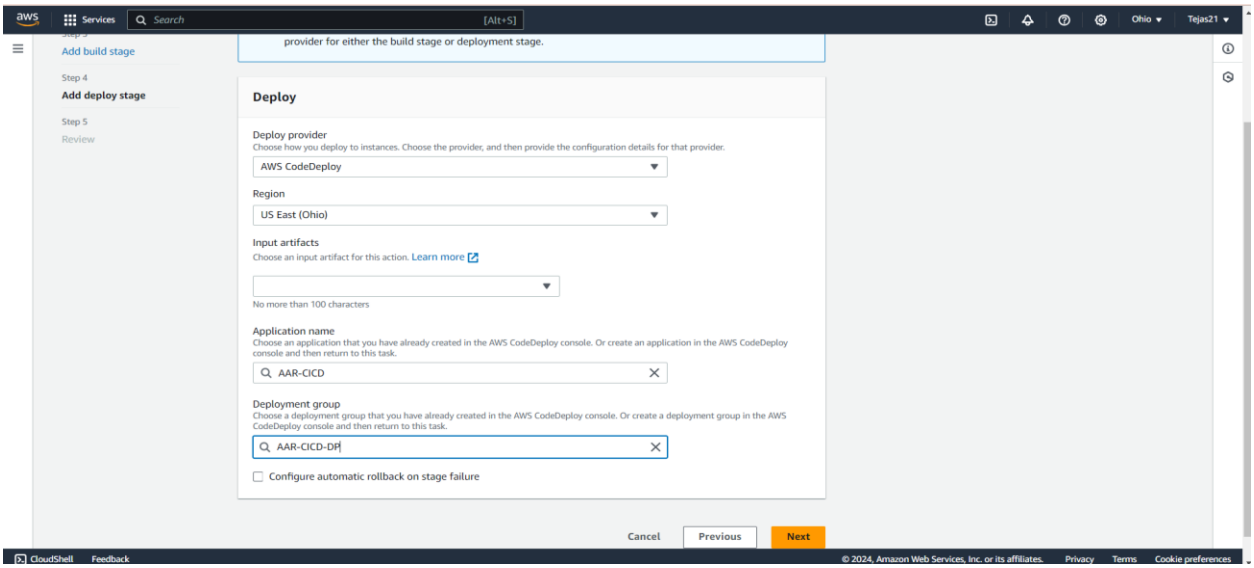

Installing AWS Connector for GitHub - Google Chrome
github.com/apps/aws-connector-for-github/installations/new/permissions?target_id=157033479
Install on your personal account tejasgunjal021
for these repositories:
All repositories
This applies to all current and future repositories owned by the resource owner.
Also includes public repositories (read-only).
Only select repositories
Select at least one repository.
Also includes public repositories (read-only).
Select repositories
Selected 1 repository.
tejasgunjal021/adv_devops
with these permissions:
Read access to issues and metadata
Read and write access to administration, code, commit statuses, pull requests, and repository hooks
Install
Cancel
Next: you'll be directed to the GitHub App's site to complete setup.

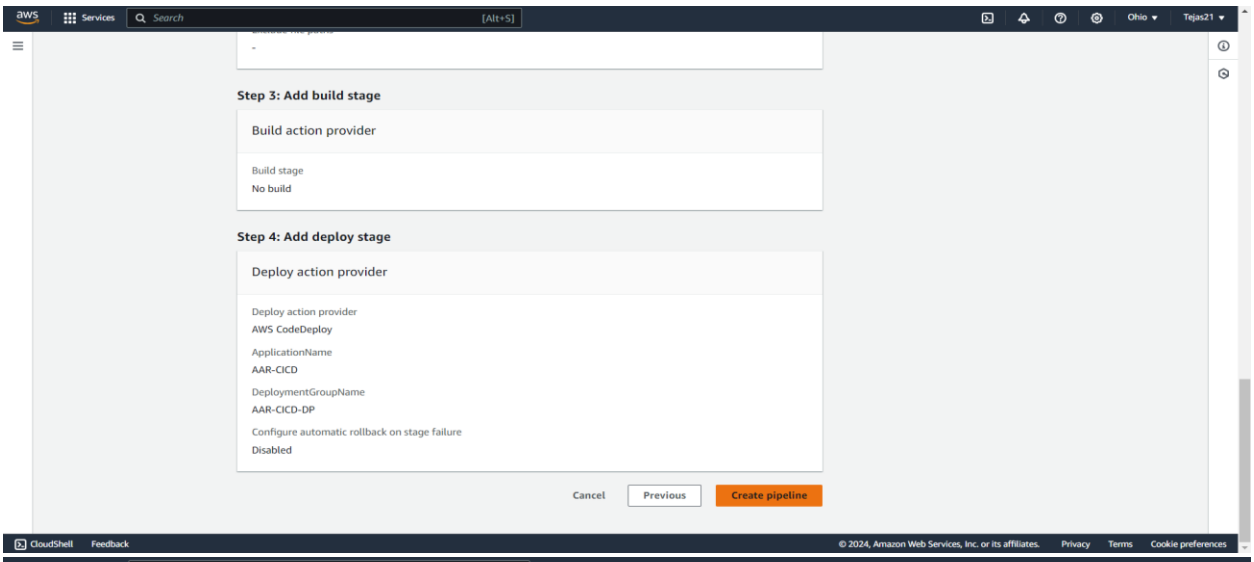

aws
Services
Search
[Alt+S]
Ohio
Tejas21
Developer Tools > Connections > Create connection
Beginning July 1, 2024, the console will create connections with codeconnections in the resource ARN.
Resources with both service prefixes will continue to display in the console. Learn more
Connect to GitHub
GitHub connection settings Info
Connection name
AAR-CICD-GIT
GitHub Apps
GitHub Apps create a link for your connection with GitHub. Install a new app and save this connection.
53669548
or
Install a new app
Tags - optional
Connect
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

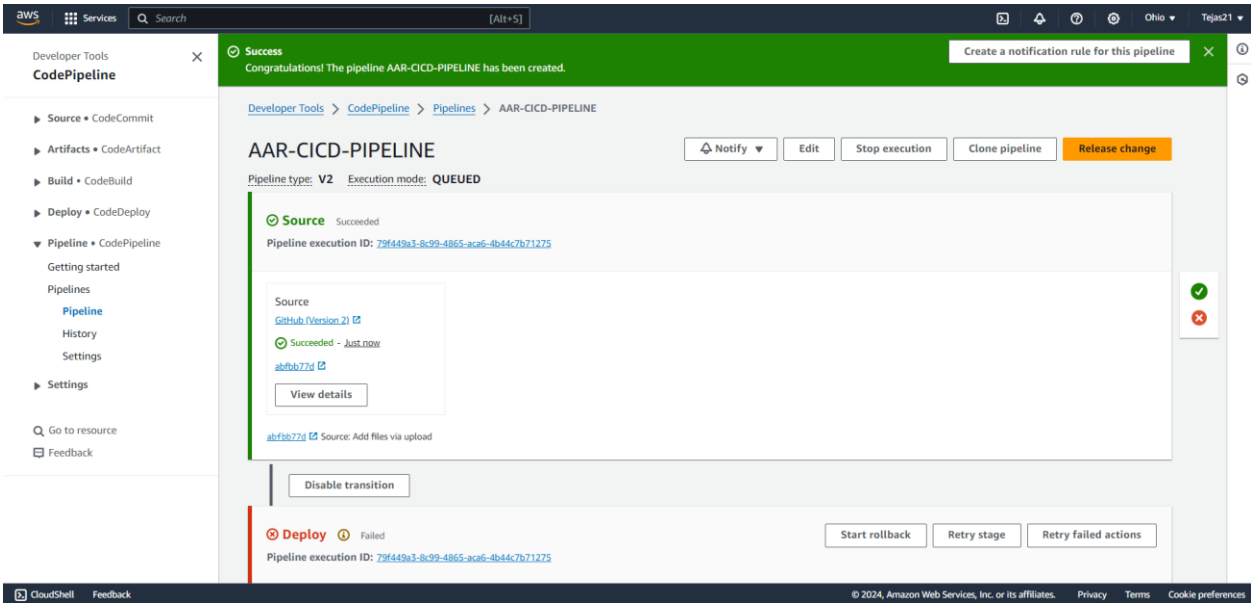

Services
Search
[Alt+S]
Ohio
Tejas21
Step 2
Add source stage
Step 3
Add build stage
Step 4
Add deploy stage
Step 5
Review
Source
Source provider
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.
GitHub (Version 2)
New GitHub version 2 (app-based) action
To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. Learn more
Connection
Choose an existing connection that you have already configured, or create a new one and then return to this task.
arn:aws:codestar-connections:us-east-2:010928184174:connection/e597406
or
Connect to GitHub
Ready to connect
Your GitHub connection is ready for use.
Repository name
Choose a repository in your GitHub account.
tejasgunjal021/adv_devops
You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.
Default branch
Default branch will be used only when pipeline execution starts from a different source or manually started.
main
Output artifact format
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

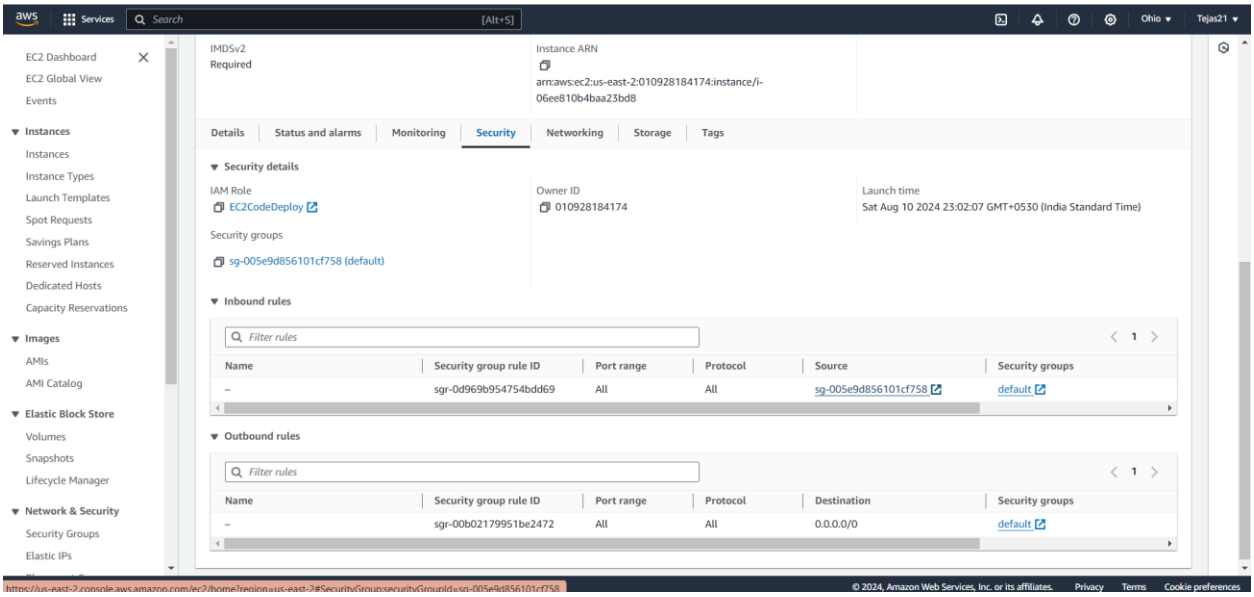

Services
Search
[Alt+S]
Ohio
Tejas21
Output artifact format
Choose the output artifact format.
CodePipeline default
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.
Full clone
AWS CodePipeline passes metadata about the repository 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
Enter branches or patterns
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

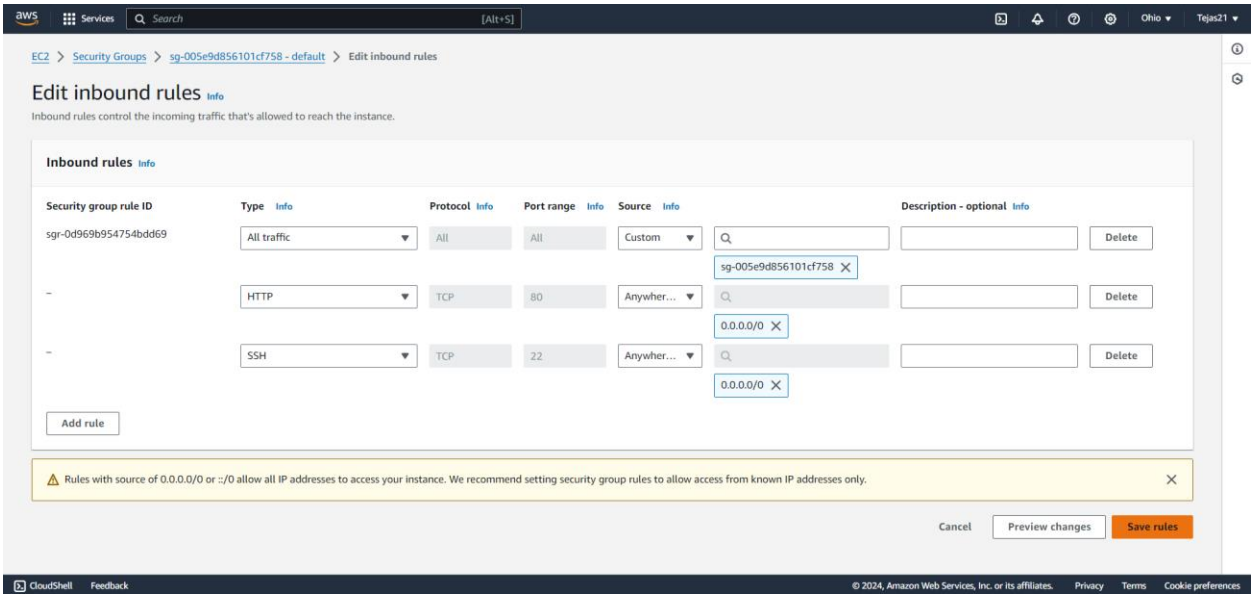

Services
Search
[Alt+S]
Ohio
Tejas21
Developer Tools
CodePipeline
Pipelines
Create new pipeline
Step 1
Choose pipeline settings
Step 2
Add source stage
Step 3
Add build stage
Step 4
Add deploy stage
Step 5
Review
Add build stage
Info
Step 3 of 5
Build - optional
Build provider
This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.
Cancel
Previous
Skip build stage
Next
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

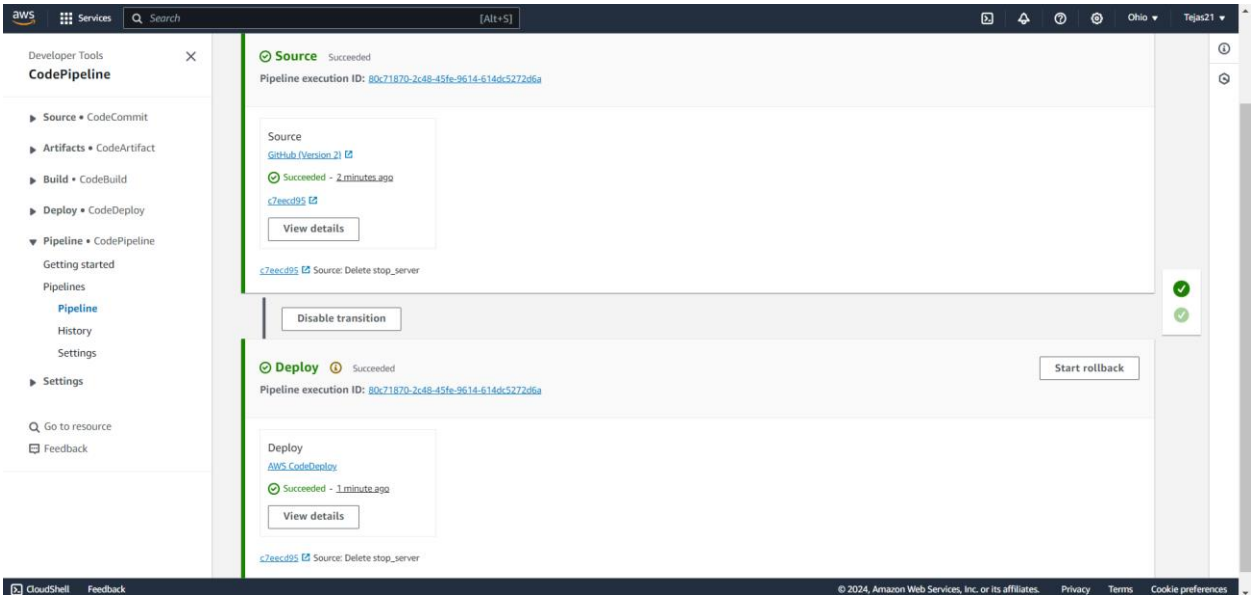

Add build stage
Step 4
Add deploy stage
Step 5
Review
provider for either the build stage or deployment stage.
Deploy
Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.
AWS CodeDeploy
Region
US East (Ohio)
Input artifacts
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
Cancel
Previous
Next

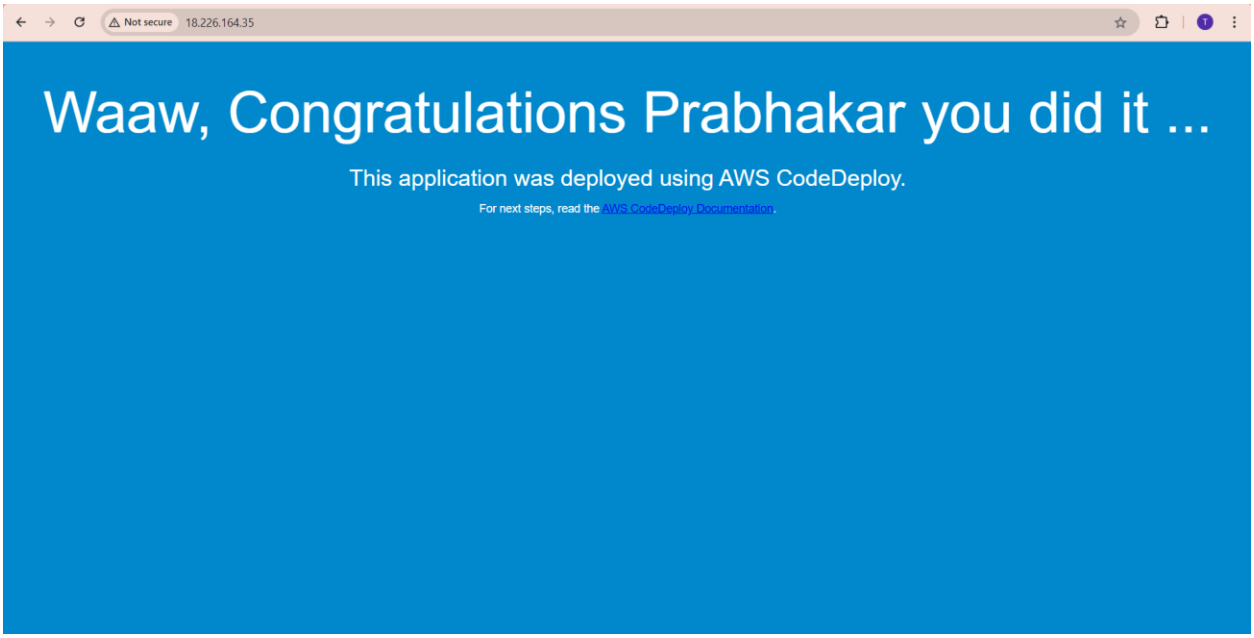

Step 3: Add build stage
Build action provider
Build stage
No build
Step 4: Add deploy stage
Deploy action provider
Deploy action provider
AWS CodeDeploy
ApplicationName
AAR-CICD
DeploymentGroupName
AAR-CICD-DP
Configure automatic rollback on stage failure
Disabled
Cancel
Previous
Create pipeline


Developer Tools
CodePipeline
Source • CodeCommit
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Settings
Settings
Go to resource
Feedback
Success
Congratulations! The pipeline AAR-CICD-PIPELINE has been created.
Create a notification rule for this pipeline
Developer Tools > CodePipeline > Pipelines > AAR-CICD-PIPELINE
AAR-CICD-PIPELINE
Notify
Edit
Stop execution
Clone pipeline
Release change
Pipeline type: V2
Execution mode: QUEUED
Source
Succeeded
Pipeline execution ID: 79f449a3-8c99-4865-aca6-4b44c7b71275
Source
GitHub (Version 2)
Succeeded - Just now
abfbb77d
View details
abfbb77d Source: Add files via upload
Disable transition
Deploy
Failed
Pipeline execution ID: 79f449a3-8c99-4865-aca6-4b44c7b71275
Start rollback
Retry stage
Retry failed actions

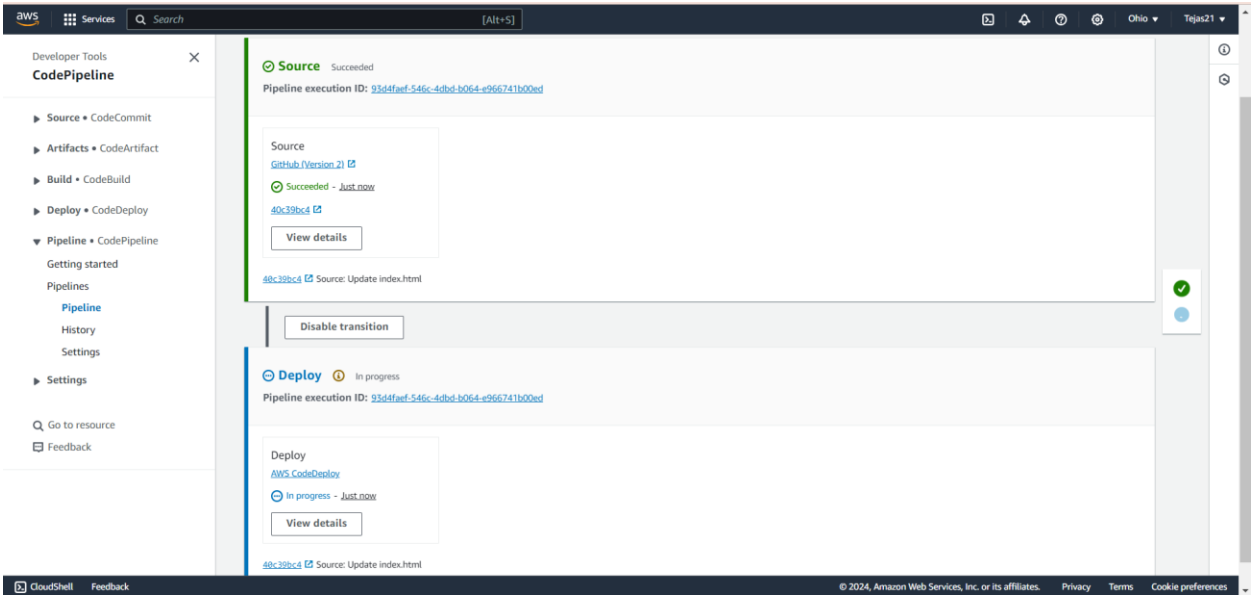

Services
Search
[Alt+S]
Ohio
Tejas21
EC2 Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
IMDSv2
Required
Instance ARN
arn:aws:ec2:us-east-2:010928184174:instance/i-06ee810b4baa23bd8
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Security details
IAM Role
EC2CodeDeploy
Owner ID
010928184174
Launch time
Sat Aug 10 2024 23:02:07 GMT+0530 (India Standard Time)
Security groups
sg-005e9d856101cf758 (default)
Inbound rules
Filter rules
Name
Security group rule ID
Port range
Protocol
Source
Security groups
–
sgr-0d969b954754bdd69
All
All
sg-005e9d856101cf758
default
Outbound rules
Filter rules
Name
Security group rule ID
Port range
Protocol
Destination
Security groups
–
sgr-00b02179951be2472
All
All
0.0.0.0/0
default
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

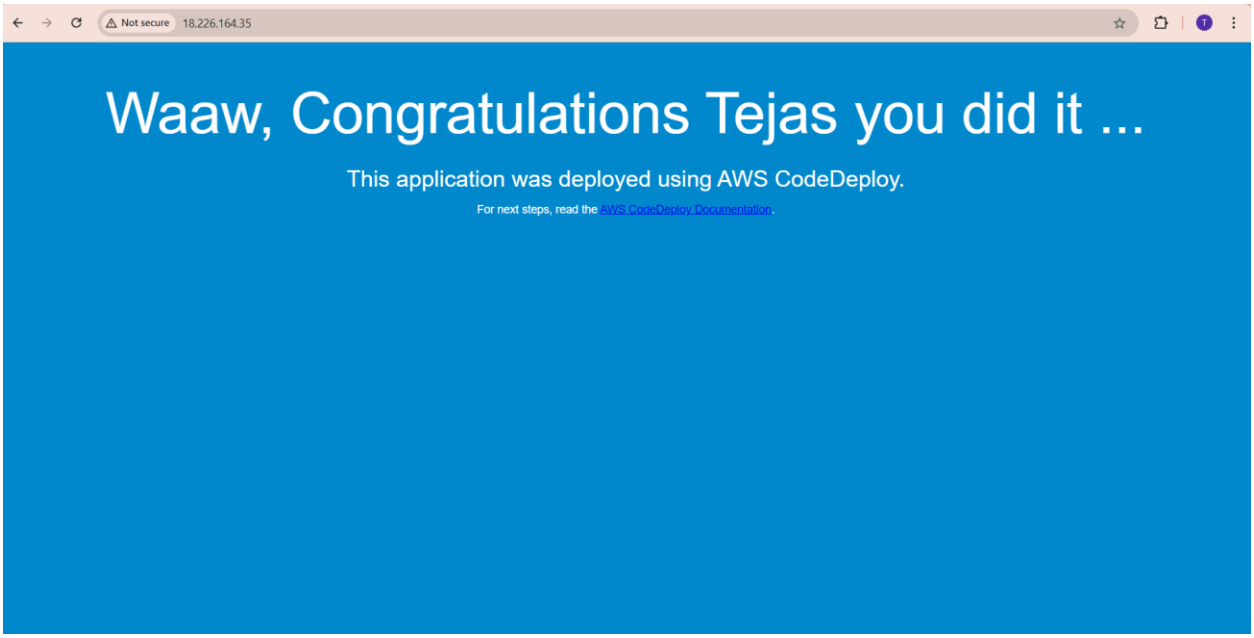

Services
Search
[Alt+S]
Ohio
Tejas21
EC2
Security Groups
sg-005e9d856101cf758 - default
Edit inbound rules
Edit inbound rules
Info
Inbound rules control the incoming traffic that's allowed to reach the instance.
Inbound rules
Info
Security group rule ID
Type
Info
Protocol
Info
Port range
Info
Source
Info
Description - optional
Info
sgr-0d969b954754bdd69
All traffic
All
All
Custom
sg-005e9d856101cf758
Delete
–
HTTP
TCP
80
Anywher...
0.0.0.0/0
Delete
–
SSH
TCP
22
Anywher...
0.0.0.0/0
Delete
Add rule
Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.
Cancel
Preview changes
Save rules
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
Ohio
Tejas21
Developer Tools
CodePipeline
Source • CodeCommit
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Settings
Settings
Go to resource
Feedback
Source Succeeded
Pipeline execution ID: 80c71870-2c48-45fe-9614-614dc5272d6a
Source
GitHub (Version 2)
Succeeded - 2 minutes ago
c7eecd95
View details
c7eecd95 Source: Delete stop_server
Disable transition
Deploy Succeeded
Start rollback
Pipeline execution ID: 80c71870-2c48-45fe-9614-614dc5272d6a
Deploy
AWS CodeDeploy
Succeeded - 1 minute ago
View details
c7eecd95 Source: Delete stop_server
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Not secure
18.226.164.35
Waaw, Congratulations Prabhakar you did it ...
This application was deployed using AWS CodeDeploy.
For next steps, read the AWS CodeDeploy Documentation

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





EXPERIMENT NO: - 03

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

Understanding Kubernetes Cluster Architecture

Introduction: -

Kubernetes is an open-source platform used to deploy and maintain a group of containers in a virtualized environment. In practice, Kubernetes is most commonly used alongside Docker for better control and implementation of containerized applications. Containerized applications "bundle" applications together with all its files, libraries, and packages required for it to run reliably and efficiently on different platforms. However, it operates at the container level rather than at the hardware level.

The name Kubernetes is derived from a Greek term meaning 'helmsman' or 'pilot.' True to this word, Kubernetes provides the guiding force for developer platforms to transition from virtual machines (VMs) to containers and the statically scheduled to the dynamically scheduled. This means no more manual integration and configuration when you move from a testing environment to an actual production environment or from on-premise to the cloud! The Kubernetes logical compute environment offers common services to all the applications in the cluster as part of the ecosystem for the software to run consistently.

Features of Kubernetes

- Automates various manual processes and controls server hosting and launching
- Manages containers offer security, and networking and storage services
- Monitors and continuously checks the health of nodes and containers
- Automates rollback for changes that go wrong
- Mounts and adds a storage system to run apps

Purpose of Kubernetes

The primary purpose of Kubernetes is to enable developers to write and deploy applications that can run seamlessly across multiple operating environments. Traditionally, application performance and deployment were tightly coupled to specific infrastructures, often requiring adherence to cloud provider-specific constructs and back-end storage systems. This dependency resulted in infrastructure lock-in, limiting flexibility and scalability

Kubernetes addresses this challenge by abstracting the underlying infrastructure, allowing developers to deploy cloud-native applications in containers without restrictions. This means applications can be managed and scaled consistently across different environments—be it in the cloud, on-premises, or in hybrid setups—providing true infrastructure independence and operational flexibility.

Working of Kubernetes

Before exploring how Kubernetes operates, it's essential to grasp the concept of containers and their significance in modern application development. A container is a small, lightweight virtual machine (VM) that does not have device drivers and shares its operating system among the applications. It is a good way to bundle and run applications in a production environment. However, you need to manage these containers in a proper way so that there is no downtime. This is where Kubernetes comes to the rescue.

Kubernetes works as a "container orchestration system" that manages the lifecycle of containerized applications and automates the deployment of several containers. Containers running the same applications are usually grouped together into Pods. There can be one or multiple containers in a single Pod and each of them shares the same IP address and resources such as memory and storage. By grouping the containers in this manner, Kubernetes eliminates the need to cram multiple functionalities in one single container. There is a dedicated container orchestrator which supervises these groups and ensures that they operate correctly.

Kubernetes in DevOps

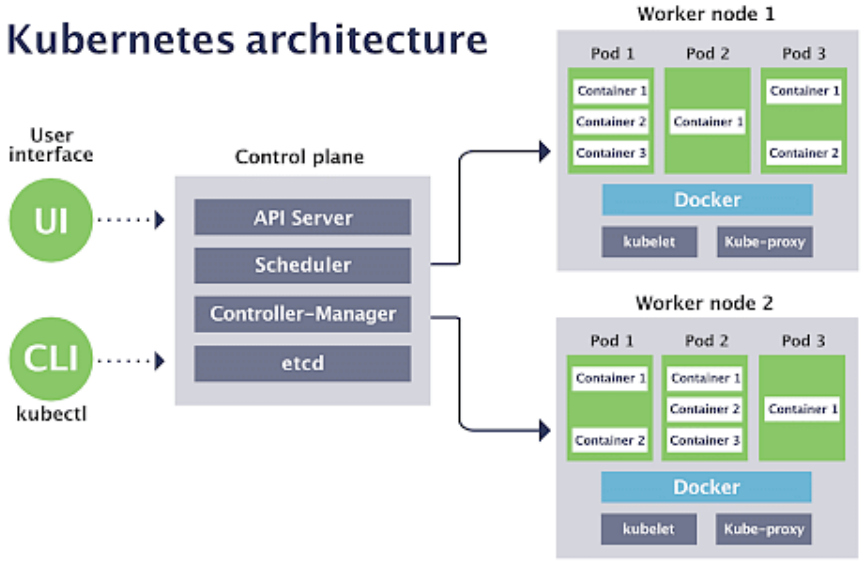
Kubernetes is more than just a container orchestration tool; it's a powerful enabler of DevOps practices. By bridging the gap between IT operations and development, Kubernetes fosters a collaborative DevOps environment, ensuring that software and its dependencies are shared seamlessly across different environments.

Kubernetes facilitates various stages of the software lifecycle, enhancing the build-test-deploy timeline:

- **Developer Environment:** Helps run software consistently in any setting, ensuring that applications behave the same across different environments.
- **QA/Testing Process:** Coordinates pipelines between test and production environments, streamlining the testing process.
- **Sys-Admin:** Once configured, Kubernetes runs anything, simplifying system administration tasks.
- **Operations:** Provides a comprehensive solution for building, shipping, and scaling software, making operations smoother and more efficient.

Kubernetes Cluster Architecture

A Kubernetes cluster consists of a set of worker machines, called nodes, that run containerized applications. The cluster is managed by a control plane, which is responsible for maintaining the desired state of the cluster, such as which applications are running and where they are running.



Control Plane: The control plane is the brain of the Kubernetes cluster, responsible for managing the desired state of the cluster, making decisions on scheduling, and responding to cluster events.

- **API Server:** The API server acts as the front-end for the Kubernetes control plane. It exposes the Kubernetes API, which is used by both internal components and external users (via CLI or UI) to communicate with the cluster.
- **Scheduler:** The scheduler is responsible for assigning newly created pods to nodes in the cluster. It evaluates the resource requirements of the pods against the available resources on the nodes, ensuring optimal placement.
- **Controller Manager:** This component runs various controllers that manage the state of the cluster. For example, it ensures that the number of pod replicas matches the desired configuration and handles node failures.
- **etcd:** A key-value store that holds all the configuration data for the Kubernetes cluster, including the current state and the desired state of the objects in the cluster. It is essential for maintaining cluster consistency and recovery.

Worker Nodes: Worker nodes are the machines where the application workloads run. Each worker node contains the services necessary to run pods and communicate with the control plane.

- **Kubelet:** The kubelet is an agent that runs on each worker node. It ensures that containers are running in a pod as expected by the control plane. It communicates with the API server to receive instructions and report back the status of the node and its workloads.
- **Kube-proxy:** Kube-proxy is a network proxy that runs on each worker node. It manages the networking for the pods, ensuring that each pod can communicate with others, both within and outside the cluster.
- **Docker (or other container runtimes):** Docker is the container runtime that runs and manages the containers on the worker nodes. It pulls container images from a registry, starts and stops containers, and manages container storage and networking.

User Interfaces: Users interact with the Kubernetes cluster through two primary interfaces:

- **UI (User Interface):** A graphical interface that provides an easy way to manage and monitor the cluster.
- **CLI (Command-Line Interface, e.g., kubectl):** A more powerful tool for managing the cluster, allowing users to interact with the API server directly via command-line commands.

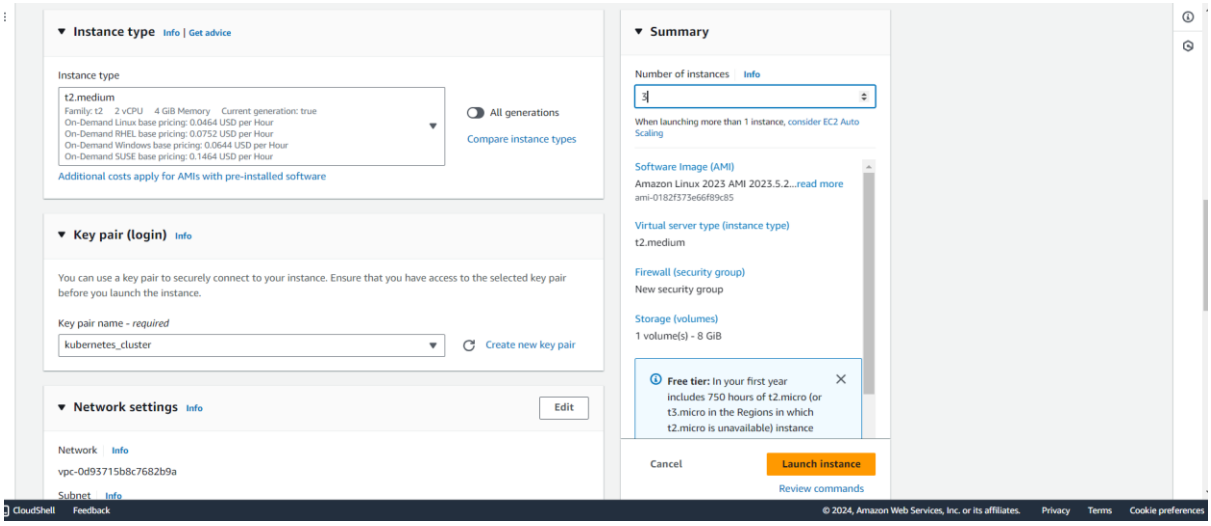
This architecture allows Kubernetes to abstract away the underlying infrastructure, providing a consistent and scalable environment for deploying and managing containerized applications across different environments.

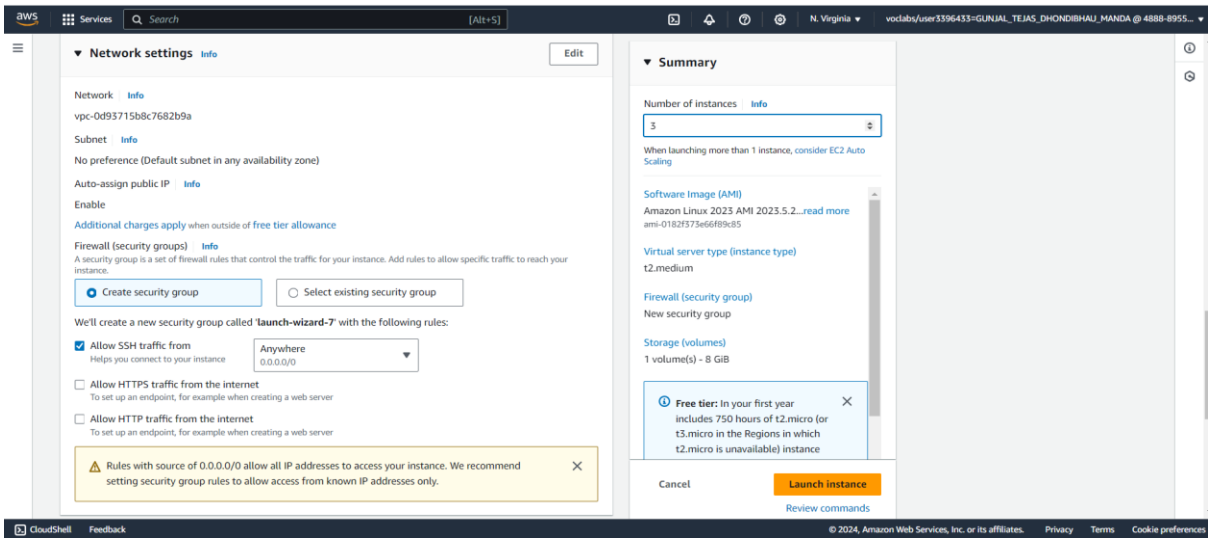
Conclusion: -

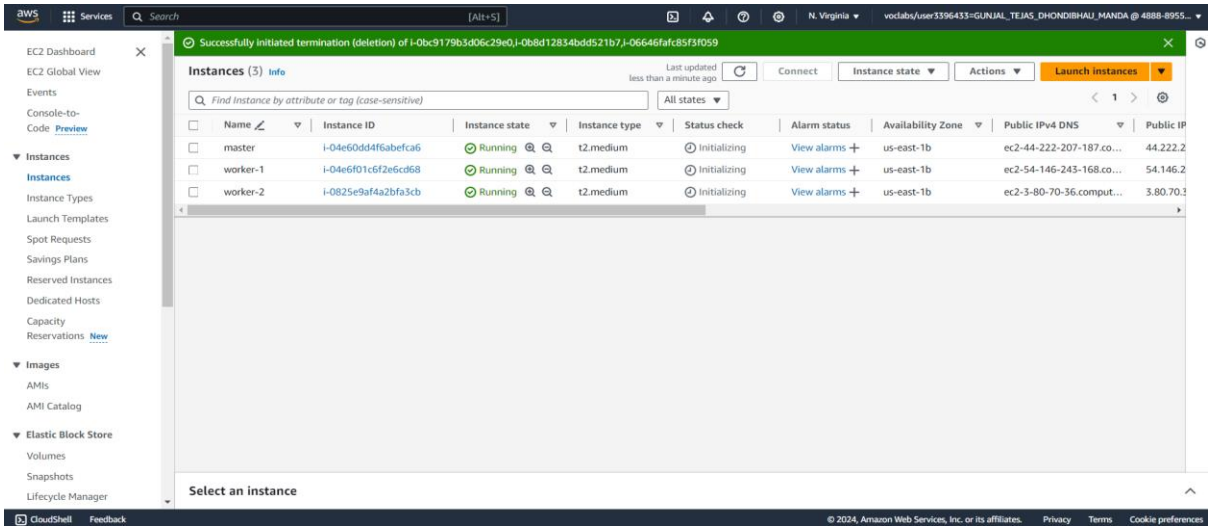
In conclusion, Kubernetes is a powerful, open-source platform that automates the deployment and management of containerized applications. Its architecture ensures seamless operation across diverse environments, promoting flexibility and scalability. By abstracting the underlying infrastructure and supporting DevOps practices, Kubernetes enhances the software development lifecycle. This results in streamlined operations, reduced downtime, and consistent application performance.

Install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms.

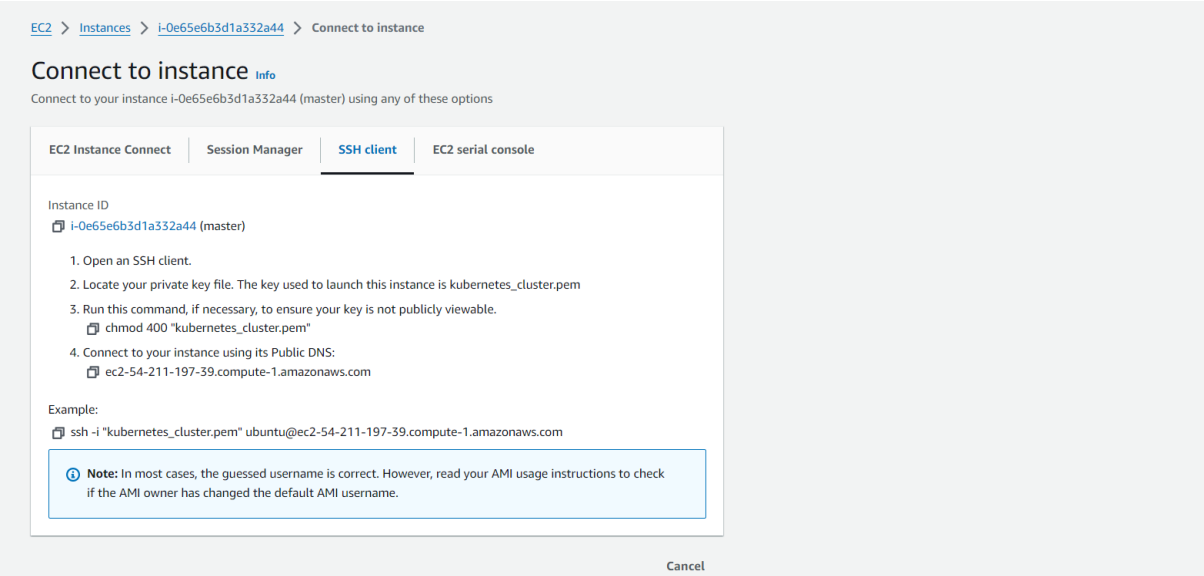
- Create 3 EC2 instances, one for the master node and two for the worker nodes







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

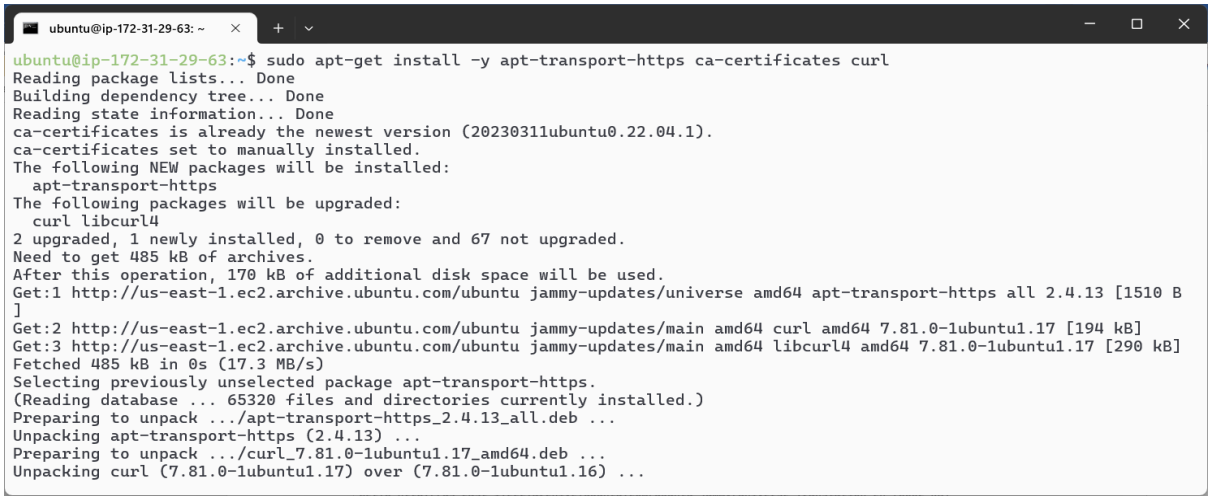



```
ubuntu@ip-172-31-29-63: ~
Microsoft Windows [Version 10.0.22631.4112]
(c) Microsoft Corporation. All rights reserved.

C:\Users\TEJAS\Downloads>ssh -i "kubernetes_cluster.pem" ubuntu@ec2-54-211-197-39.compute-1.amazonaws.com
The authenticity of host 'ec2-54-211-197-39.compute-1.amazonaws.com (54.211.197.39)' can't be established.
ED25519 key fingerprint is SHA256:E7RRK6LxQFHSnbWLb8zgsskyIdjOYXpInW0bO8/5vnA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-211-197-39.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1022-aws x86_64)

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

 System information as of Sun Sep 15 17:48:58 UTC 2024

  System load:  0.28              Processes:             115
  Usage of /:   20.7% of 7.57GB   Users logged in:       0
  Memory usage: 5%                IPv4 address for eth0: 172.31.29.63
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.
```

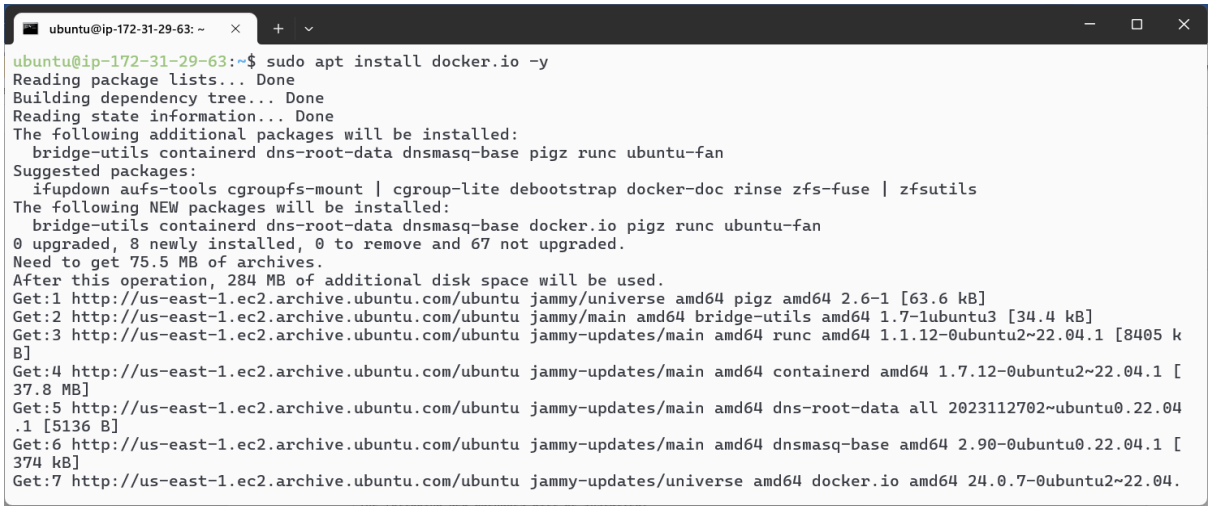
- Update the package manager on all nodes:

```
ubuntu@ip-172-31-29-63: ~
ubuntu@ip-172-31-29-63:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1806 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2023 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [295 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [13.3 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2377 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [352 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [17.8 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [2437 kB]
Get:19 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [409 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [584 B]
Get:21 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [902 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [419 kB]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [616 B]
```

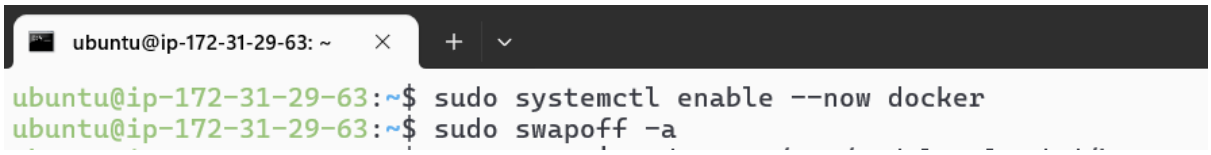
- Installing Required Packages for HTTPS and Certificate Transport on Ubuntu



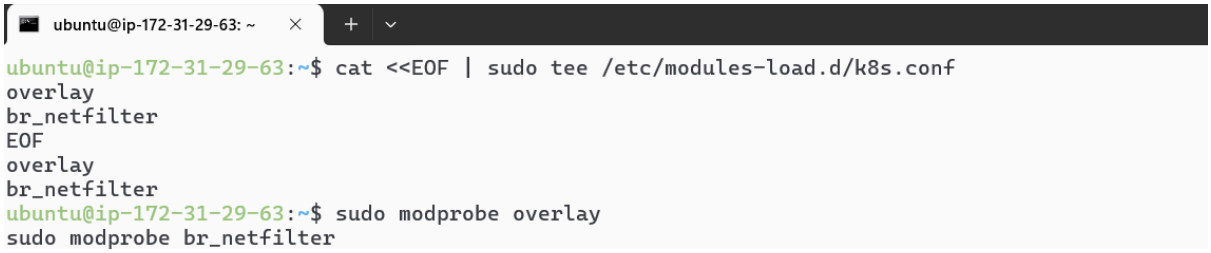
- Installing Docker on Ubuntu



- Enabling Docker and Disabling Swap on Ubuntu



- Load necessary kernel modules for networking and iptables:



- Configure sysctl settings for Kubernetes networking:

```
ubuntu@ip-172-31-29-63: ~
ubuntu@ip-172-31-29-63:~$ cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-iptables  = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.ipv4.ip_forward                 = 1
EOF
net.bridge.bridge-nf-call-iptables  = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.ipv4.ip_forward                 = 1
```

```
ubuntu@ip-172-31-29-63: ~
ubuntu@ip-172-31-29-63:~$ sudo sysctl --system
* Applying /etc/sysctl.d/10-console-messages.conf ...
kernel.printk = 4 4 1 7
* Applying /etc/sysctl.d/10-ipv6-privacy.conf ...
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.default.use_tempaddr = 2
* Applying /etc/sysctl.d/10-kernel-hardening.conf ...
kernel.kptr_restrict = 1
* Applying /etc/sysctl.d/10-magic-sysrq.conf ...
kernel.sysrq = 176
* Applying /etc/sysctl.d/10-network-security.conf ...
net.ipv4.conf.default.rp_filter = 2
net.ipv4.conf.all.rp_filter = 2
* Applying /etc/sysctl.d/10-ptrace.conf ...
kernel.yama.ptrace_scope = 1
* Applying /etc/sysctl.d/10-zeropage.conf ...
vm.mmap_min_addr = 65536
* Applying /etc/sysctl.d/50-cloudimg-settings.conf ...
net.ipv4.neigh.default.gc_thresh2 = 15360
net.ipv4.neigh.default.gc_thresh3 = 16384
net.netfilter.nf_conntrack_max = 1048576
* Applying /usr/lib/sysctl.d/50-default.conf ...
kernel.core_uses_pid = 1
net.ipv4.conf.default.rp_filter = 2
```

- Install Kubernetes tools on all nodes.

```
ubuntu@ip-172-31-29-63: ~
ubuntu@ip-172-31-29-63:~$ curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.29/deb/Release.key | sudo gpg --dearmor -o /et
c/apt/keyrings/kubernetes-apt-keyring.gpg
echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /' | sud
o tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /

ubuntu@ip-172-31-29-63:~$ sudo apt-get update -y
sudo apt-get install -y kubelet="1.29.0-*" kubectl="1.29.0-*" kubeadm="1.29.0-*"
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:5 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.29/deb  InRelease [1189 B]
Get:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.29/deb  Packages [14.0 kB]
Fetched 15.1 kB in 0s (31.9 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Selected version '1.29.0-1.1' (isv:kubernetes:core:stable:v1.29:pkgs.k8s.io [amd64]) for 'kubelet'
Selected version '1.29.0-1.1' (isv:kubernetes:core:stable:v1.29:pkgs.k8s.io [amd64]) for 'kubectl'
Selected version '1.29.0-1.1' (isv:kubernetes:core:stable:v1.29:pkgs.k8s.io [amd64]) for 'kubeadm'
The following additional packages will be installed:
  conntrack cri-tools ebtables kubernetes-cni socat
The following NEW packages will be installed:
```

```
ubuntu@ip-172-31-29-63:~$ sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
```

ONLY ON MASTER NODE

- Initialize the Kubernetes Cluster on Master Node

```
ubuntu@ip-172-31-29-63: ~
ubuntu@ip-172-31-29-63:~$ sudo kubeadm init --pod-network-cidr=10.244.0.0/16
I0915 17:58:19.113429    3356 version.go:256] remote version is much newer: v1.31.0; falling back to: stable-1.29
[init] Using Kubernetes version: v1.29.8
[preflight] Running pre-flight checks
[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 in beforehand using 'kubeadm config images pull'
W0915 17:58:28.696218    3356 checks.go:835] detected that the sandbox image "registry.k8s.io/pause:3.8" of the containe
r runtime is inconsistent with that used by kubeadm. It is recommended that using "registry.k8s.io/pause:3.9" as the CRI
 sandbox image.
[certs] Using certificateDir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [ip-172-31-29-63 kubernetes kubernetes.default kubernetes.default
.svc kubernetes.default.svc.cluster.local] and IPs [10.96.0.1 172.31.29.63]
[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 [ip-172-31-29-63 localhost] and IPs [172.31.29.63 127.0.0.1 ::1
]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [ip-172-31-29-63 localhost] and IPs [172.31.29.63 127.0.0.1 ::1]
```

```
ubuntu@ip-172-31-29-63: ~

Your Kubernetes control-plane has initialized successfully!

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

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

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

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

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

kubeadm join 172.31.29.63:6443 --token 05rvqu.66v5246nmoe81d5e \
        --discovery-token-ca-cert-hash sha256:ff6e3056ea0d41a598919b1be9dbe765739c40a5aa6d37b5e4cbc45b1256c1c7
```

- Set up kubectl on the master node

```
ubuntu@ip-172-31-29-63:~$ mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
ubuntu@ip-172-31-29-63:~$ kubectl get nodes
NAME              STATUS     ROLES           AGE   VERSION
ip-172-31-29-63   NotReady   control-plane   3m    v1.29.0
ubuntu@ip-172-31-29-63:~$
```

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

```
ubuntu@ip-172-31-29-63:~$ sudo systemctl restart kubelet
ubuntu@ip-172-31-29-63:~$ mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
cp: overwrite '/home/ubuntu/.kube/config'? y
ubuntu@ip-172-31-29-63:~$ kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-fl
annel.yml
namespace/kube-flannel created
clusterrole.rbac.authorization.k8s.io/flannel created
clusterrolebinding.rbac.authorization.k8s.io/flannel created
serviceaccount/flannel created
configmap/kube-flannel-cfg created
daemonset.apps/kube-flannel-ds created
ubuntu@ip-172-31-29-63:~$
```

```
ubuntu@ip-172-31-29-63:~$ kubeadm token create --print-join-command
kubeadm join 172.31.29.63:6443 --token avprdu.7xw3ox1x6l9w6zfw --discovery-token-ca-cert-hash sha256:ff6e3056ea0d41a5989
19b1be9dbe765739c40a5aa6d37b5e4cbc45b1256c1c7
ubuntu@ip-172-31-29-63:~$
```

ONLY ON WORKER NODES

- Join Worker Nodes to the Cluster

```
ubuntu@ip-172-31-20-115:~$ sudo kubeadm reset pre-flight checks
W0915 18:24:16.207647    3366 preflight.go:56] [reset] WARNING: Changes made to this host by 'kubeadm init' or 'kubeadm
join' will be reverted.
[reset] Are you sure you want to proceed? [y/N]: y
[preflight] Running pre-flight checks
W0915 18:24:24.608981    3366 removeetcdmember.go:106] [reset] No kubeadm config, using etcd pod spec to get data direct
ory
[reset] Deleted contents of the etcd data directory: /var/lib/etcd
[reset] Stopping the kubelet service
[reset] Unmounting mounted directories in "/var/lib/kubelet"
[reset] Deleting contents of directories: [/etc/kubernetes/manifests /var/lib/kubelet /etc/kubernetes/pki]
[reset] Deleting files: [/etc/kubernetes/admin.conf /etc/kubernetes/super-admin.conf /etc/kubernetes/kubelet.conf /etc/k
ubernetes/bootstrap-kubelet.conf /etc/kubernetes/controller-manager.conf /etc/kubernetes/scheduler.conf]

The reset process does not clean CNI configuration. To do so, you must remove /etc/cni/net.d

The reset process does not reset or clean up iptables rules or IPVS tables.
If you wish to reset iptables, you must do so manually by using the "iptables" command.

If your cluster was setup to utilize IPVS, run ipvsadm --clear (or similar)
to reset your system's IPVS tables.
```

- On the worker nodes, run the command provided by the master node during initialization . It looks something like this: sudo kubeadm join :6443 --token --discovery-token-ca-cert-hash sha256:

```
ubuntu@ip-172-31-20-115:~$ sudo  kubeadm join 172.31.29.63:6443 --token avprdu.7xw3ox1x6l9w6zfw --discovery-token-ca-cer
t-hash sha256:ff6e3056ea0d41a598919b1be9dbe765739c40a5aa6d37b5e4cbc45b1256c1c7 --v=5
I0915 18:26:15.816542    3382 join.go:413] [preflight] found NodeName empty; using OS hostname as NodeName
I0915 18:26:15.816650    3382 initconfiguration.go:122] detected and using CRI socket: unix:///var/run/containerd/contai
nerd.sock
[preflight] Running pre-flight checks
I0915 18:26:15.816686    3382 preflight.go:93] [preflight] Running general checks
I0915 18:26:15.816765    3382 checks.go:280] validating the existence of file /etc/kubernetes/kubelet.conf
I0915 18:26:15.816774    3382 checks.go:280] validating the existence of file /etc/kubernetes/bootstrap-kubelet.conf
I0915 18:26:15.816784    3382 checks.go:104] validating the container runtime
I0915 18:26:15.833043    3382 checks.go:639] validating whether swap is enabled or not
I0915 18:26:15.833129    3382 checks.go:370] validating the presence of executable crictl
I0915 18:26:15.833152    3382 checks.go:370] validating the presence of executable conntrack
I0915 18:26:15.833171    3382 checks.go:370] validating the presence of executable ip
I0915 18:26:15.833203    3382 checks.go:370] validating the presence of executable iptables
I0915 18:26:15.833227    3382 checks.go:370] validating the presence of executable mount
I0915 18:26:15.833242    3382 checks.go:370] validating the presence of executable nsenter
I0915 18:26:15.833276    3382 checks.go:370] validating the presence of executable ebtables
I0915 18:26:15.833292    3382 checks.go:370] validating the presence of executable ethtool
I0915 18:26:15.833320    3382 checks.go:370] validating the presence of executable socat
I0915 18:26:15.833340    3382 checks.go:370] validating the presence of executable tc
```

```
ootstrap-kubelet.conf
I0915 18:26:15.933343    3382 kubelet.go:136] [kubelet-start] writing CA certificate at /etc/kubernetes/pki/ca.crt
I0915 18:26:15.933723    3382 kubelet.go:157] [kubelet-start] Checking for an existing Node in the cluster with name "ip
-172-31-20-115" and status "Ready"
I0915 18:26:15.936219    3382 kubelet.go:172] [kubelet-start] Stopping the kubelet
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...
I0915 18:26:17.103583    3382 cert_rotation.go:137] Starting client certificate rotation controller
I0915 18:26:17.104245    3382 kubelet.go:220] [kubelet-start] preserving the crisocket information for the node
I0915 18:26:17.104261    3382 patchnode.go:31] [patchnode] Uploading the CRI Socket information "unix:///var/run/contain
erd/containerd.sock" to the Node API object "ip-172-31-20-115" as an annotation

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

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

ubuntu@ip-172-31-20-115:~$
```

- Verify the Cluster
  Once the worker node joins, check the status on the master node

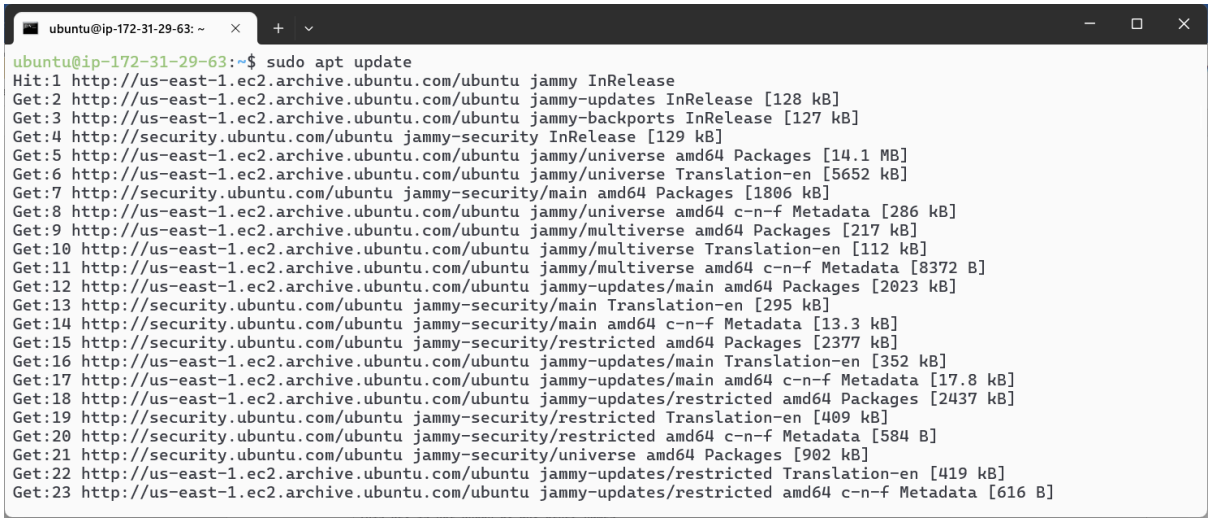
```
ubuntu@ip-172-31-29-63:~$ kubectl get nodes
NAME               STATUS   ROLES           AGE   VERSION
ip-172-31-20-115   Ready    <none>          27m   v1.29.0
ip-172-31-20-200   Ready    <none>          24m   v1.29.0
ip-172-31-29-63    Ready    control-plane   55m   v1.29.0
ubuntu@ip-172-31-29-63:~$
```

EXPERIMENT NO: - 04

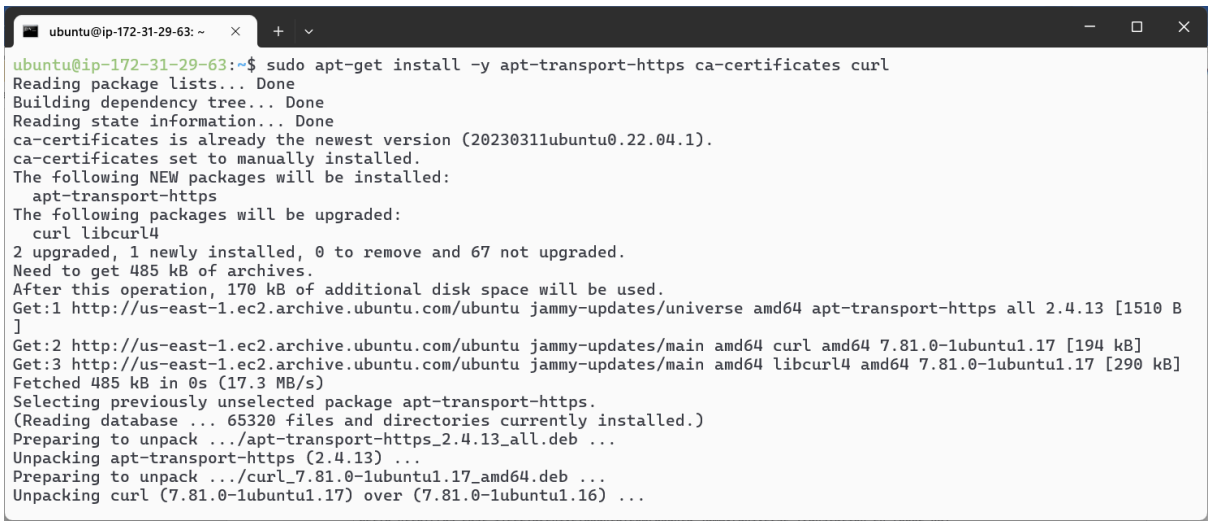
AIM:- To install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy Your First Kubernetes Application.

➢ Install Kubectl on Ubuntu

Installing Required Packages for HTTPS and Certificate Transport on Ubuntu



```
ubuntu@ip-172-31-29-63: ~
ubuntu@ip-172-31-29-63:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1806 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2023 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [295 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [13.3 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2377 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [352 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [17.8 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [2437 kB]
Get:19 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [409 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [584 B]
Get:21 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [902 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [419 kB]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [616 B]
```



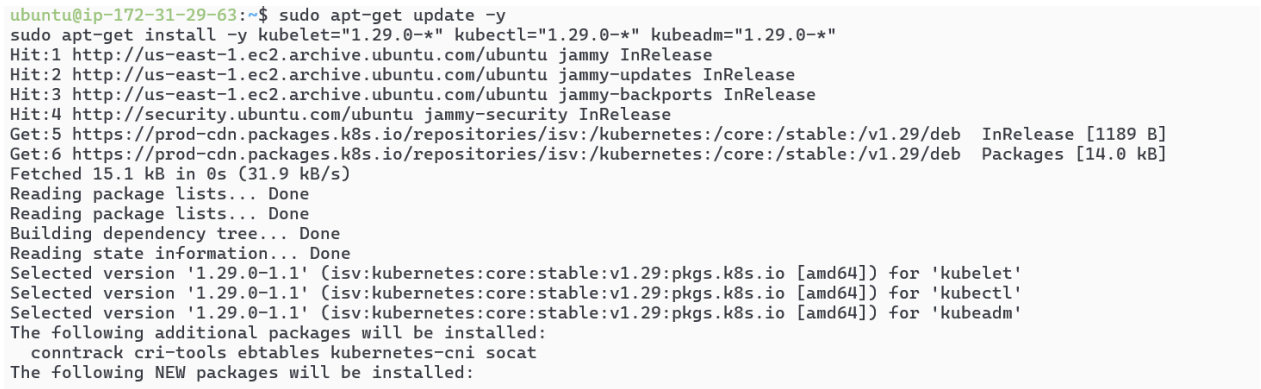
```
ubuntu@ip-172-31-29-63: ~
ubuntu@ip-172-31-29-63:~$ sudo apt-get install -y apt-transport-https ca-certificates curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  curl libcurl4
2 upgraded, 1 newly installed, 0 to remove and 67 not upgraded.
Need to get 485 kB of archives.
After this operation, 170 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.13 [1510 B
]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.17 [194 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libcurl4 amd64 7.81.0-1ubuntu1.17 [290 kB]
Fetched 485 kB in 0s (17.3 MB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 65320 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.13_all.deb ...
Unpacking apt-transport-https (2.4.13) ...
Preparing to unpack .../curl_7.81.0-1ubuntu1.17_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.17) over (7.81.0-1ubuntu1.16) ...
```

Add the GPG key & repository for Kubernetes

```
ubuntu@ip-172-31-29-63: ~
ubuntu@ip-172-31-29-63:~$ curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.29/deb/Release.key | sudo gpg --dearmor -o /et
c/apt/keyrings/kubernetes-apt-keyring.gpg
echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /' | sud
o tee /etc/apt/sources.list.d/kubernetes.list
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /
```

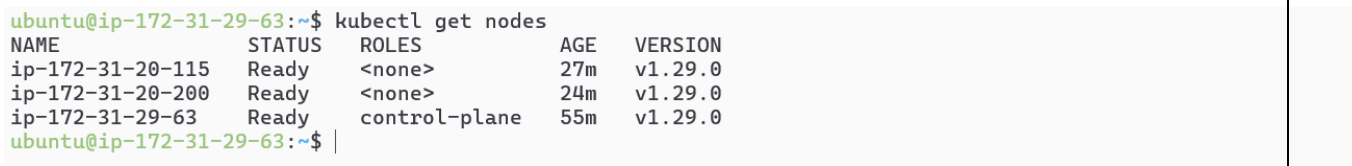
Configure sysctl settings for Kubernetes networking

Install Kubernetes tools on all nodes.



- **Set up Kubernetes Cluster**

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

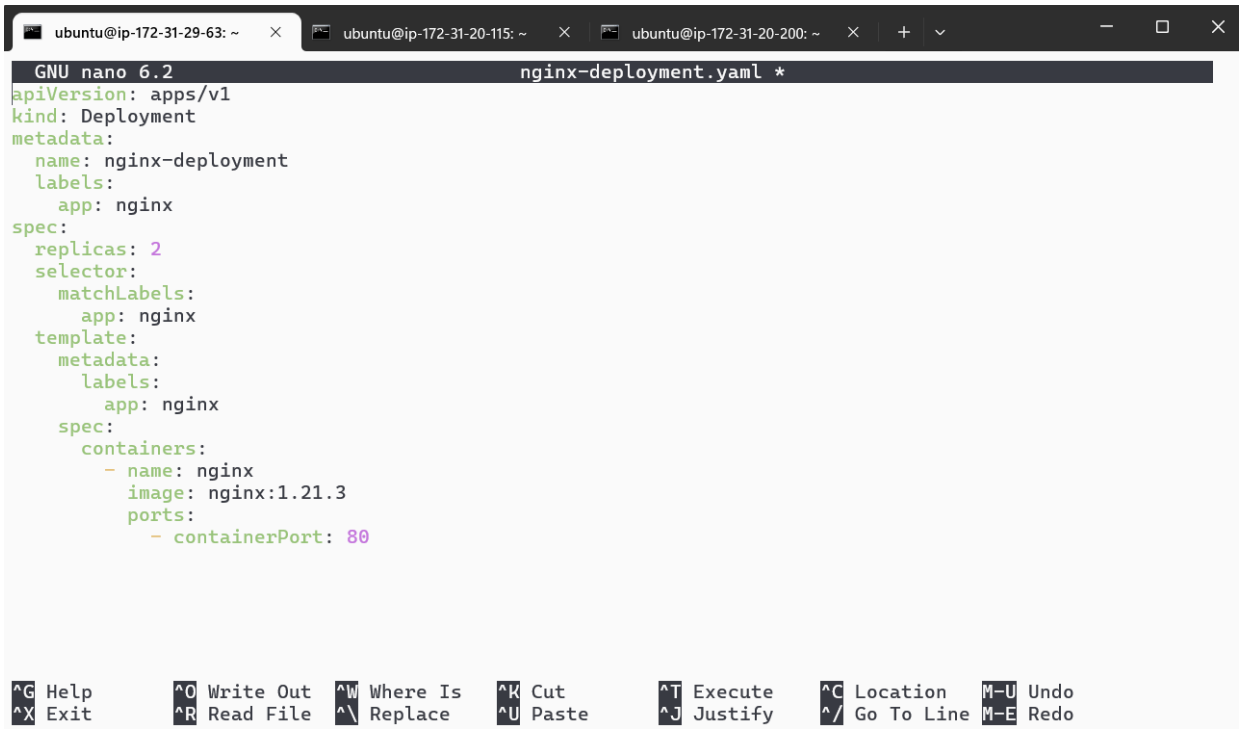


➢ Deploying Your Application on Kubernetes

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

```
ubuntu@ip-172-31-29-63:~$ nano nginx-deployment.yaml
ubuntu@ip-172-31-29-63:~$
```

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



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

```
ubuntu@ip-172-31-29-63:~$ nano nginx-service.yaml
ubuntu@ip-172-31-29-63:~$
```

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

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

```
ubuntu@ip-172-31-29-63:~$ kubectl apply -f nginx-deployment.yaml --validate=false
deployment.apps/nginx-deployment created
```

```
ubuntu@ip-172-31-29-63:~$ kubectl apply -f nginx-service.yaml --validate=false
service/nginx-service created
ubuntu@ip-172-31-29-63:~$
```

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

```
ubuntu@ip-172-31-29-63:~$ kubectl get deployments
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   1/2     2            1           9m25s
ubuntu@ip-172-31-29-63:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS      AGE
nginx-deployment-6b4d6fdbf-6w4bm    1/1     Running   4 (98s ago)   9m18s
nginx-deployment-6b4d6fdbf-bhcwm    1/1     Running   4 (70s ago)   9m18s
ubuntu@ip-172-31-29-63:~$ kubectl get services
NAME            TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes      ClusterIP      10.96.0.1       <none>        443/TCP        111m
nginx-service   LoadBalancer   10.110.88.111   <pending>     80:30132/TCP   110s
ubuntu@ip-172-31-29-63:~$
```

Describe the deployment

```
ubuntu@ip-172-31-29-63:~$ kubectl get deployments
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   2/2     2            2           11m
ubuntu@ip-172-31-29-63:~$ kubectl describe deployment
Name:                   nginx-deployment
Namespace:              default
CreationTimestamp:      Sun, 15 Sep 2024 19:39:41 +0000
Labels:                 app=nginx
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=nginx
Replicas:               2 desired | 2 updated | 2 total | 1 available | 1 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
   nginx:
    Image:        nginx:1.21.3
    Port:         80/TCP
    Host Port:    0/TCP
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Progressing    True    NewReplicaSetAvailable
  Available      False   MinimumReplicasUnavailable
OldReplicaSets:  <none>
NewReplicaSet:   nginx-deployment-6b4d6fdbf (2/2 replicas created)
Events:
  Type    Reason             Age   From                   Message
  ----    ------             ----  ----                   -------
  Normal  ScalingReplicaSet  11m   deployment-controller  Scaled up replica set nginx-deployment-6b4d6fdbf to 2
ubuntu@ip-172-31-29-63:~$
```

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

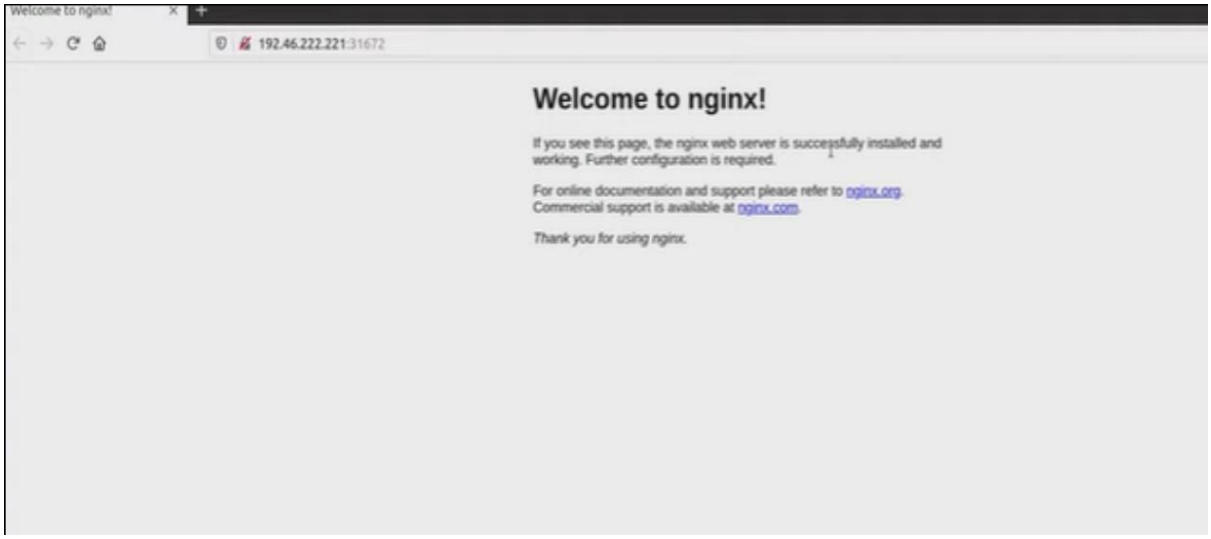
```
ubuntu@ip-172-31-29-63:~$ kubectl get service
NAME            TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes      ClusterIP      10.96.0.1       <none>        443/TCP        114m
nginx-service   LoadBalancer   10.110.88.111   <pending>     80:30132/TCP   4m59s
ubuntu@ip-172-31-29-63:~$
```

Forward the Service Port to Your Local Machine

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

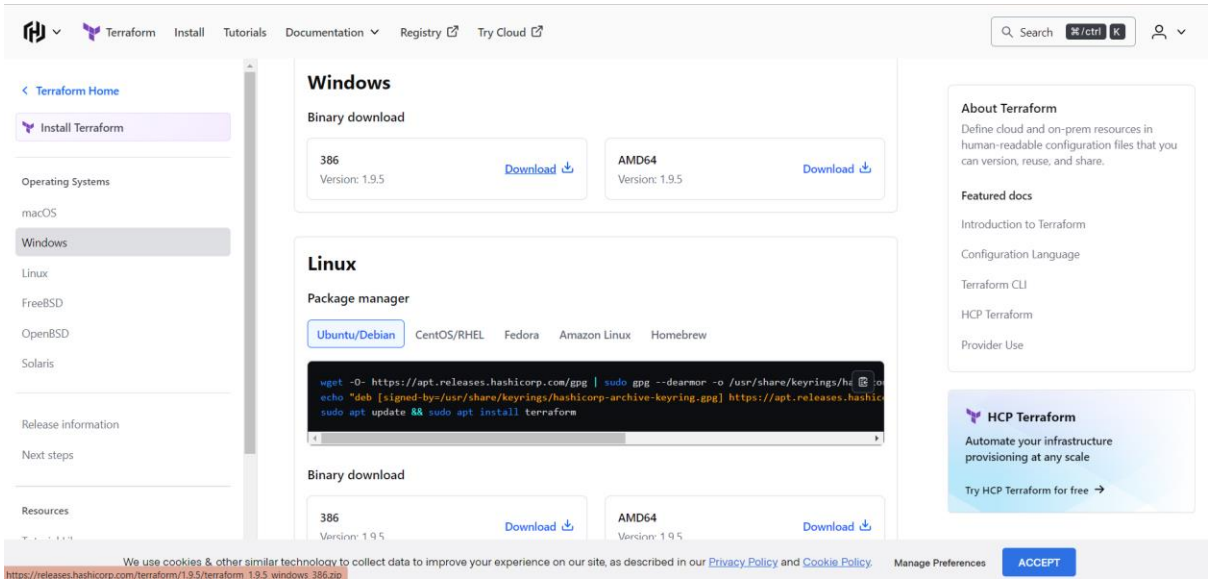
```
ubuntu@ip-172-31-45-227:~$ kubectl port-forward service/nginx-service 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
^Cubuntu@ip-172-31-45-227:~kubectl port-forward service/nginx-service 8081:8080
Forwarding from 127.0.0.1:8081 -> 80
Forwarding from [::1]:8081 -> 80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

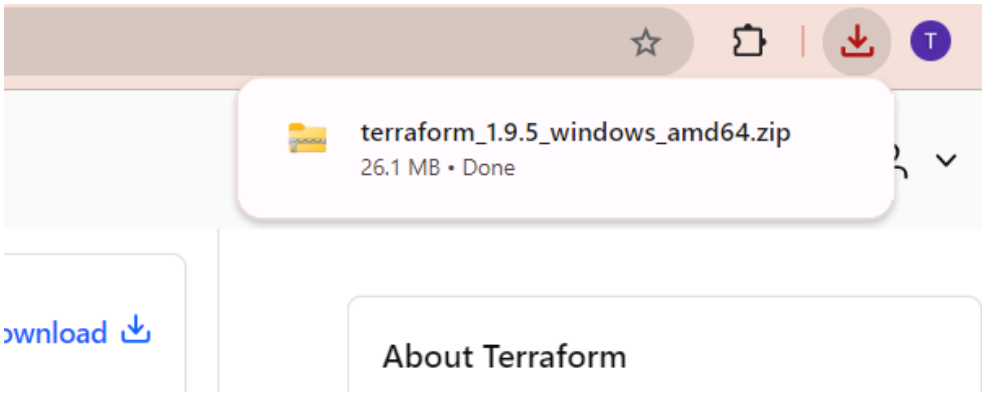
Access the Application Locally
Open a Web Browser: Now open your web browser and go to the following URL: http://localhost:8080 You should see the application (in this case, Nginx) that you have deployed running in the Kubernetes cluster, served locally via port 8080.



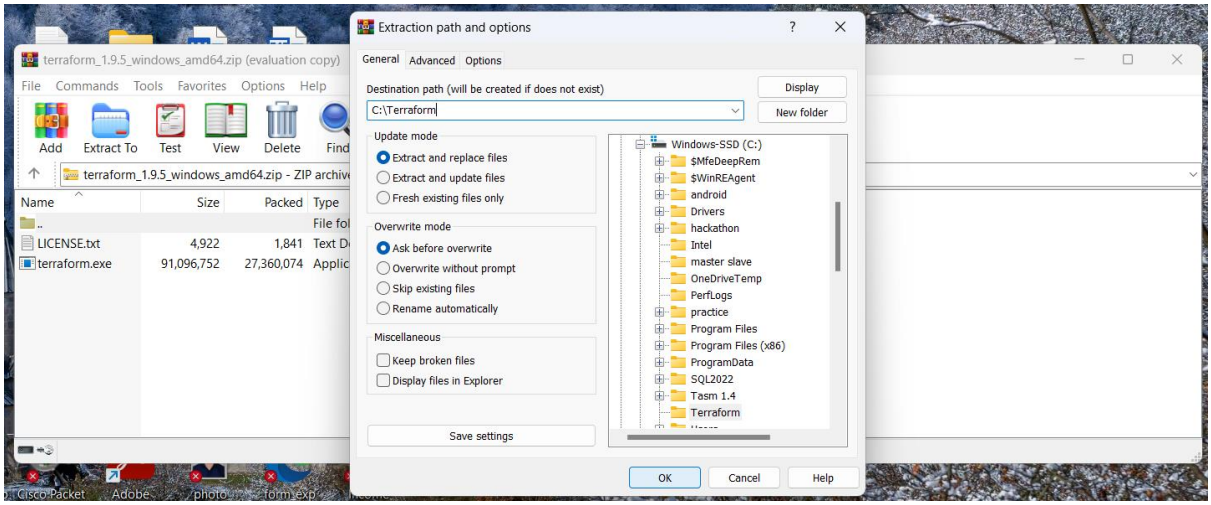
EXPERIMENT NO: - 05

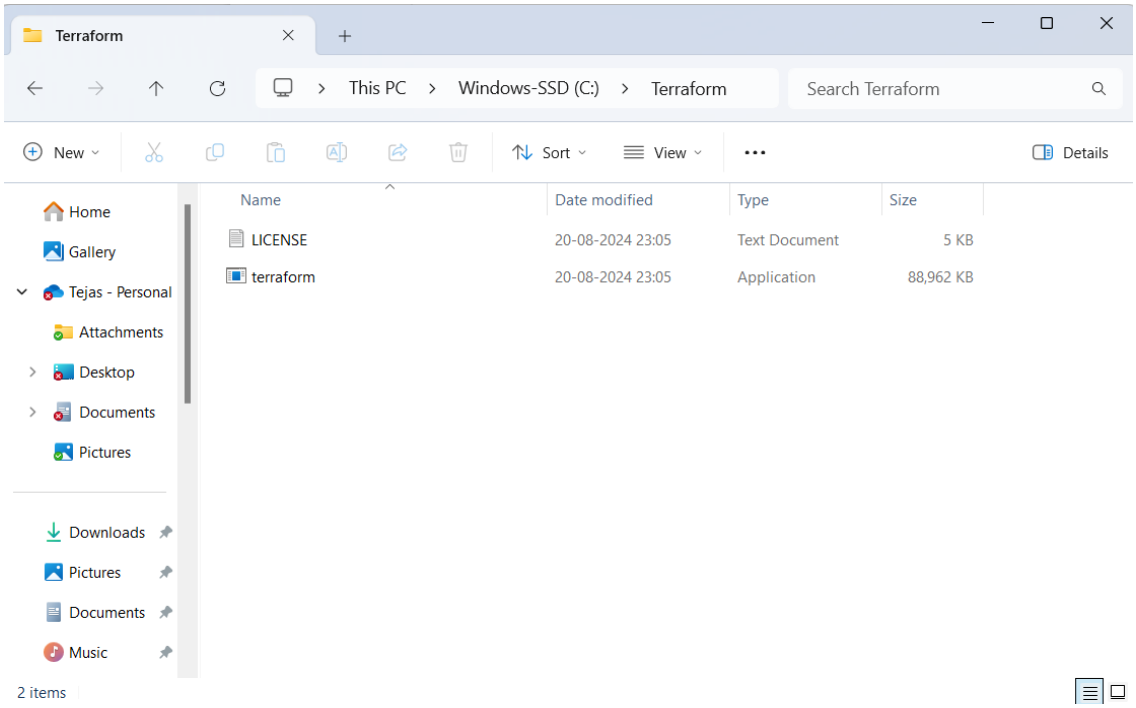
➢ Terraform installation on Windows



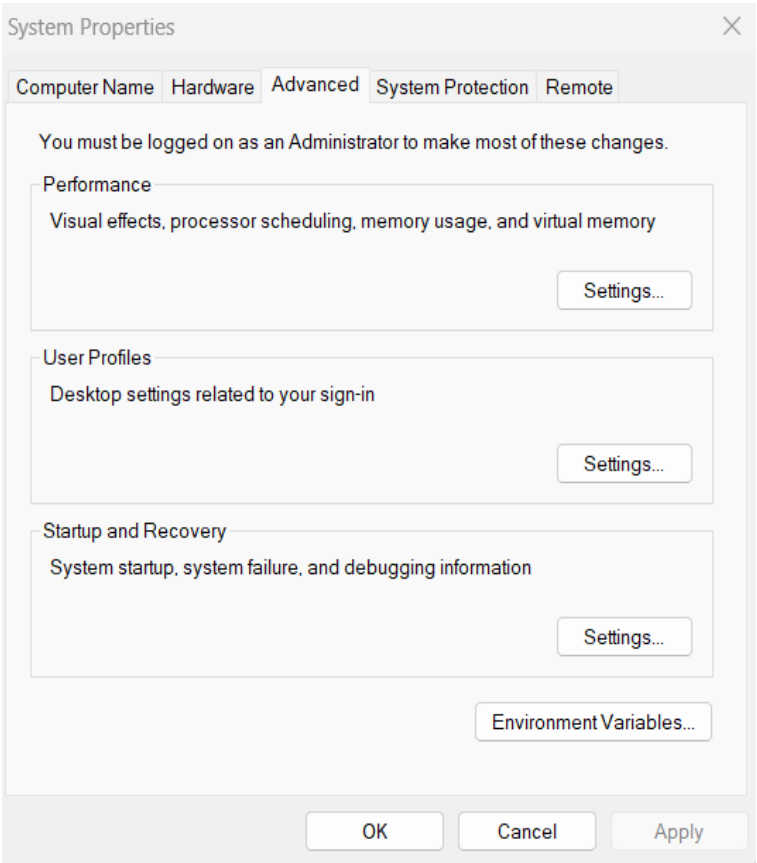


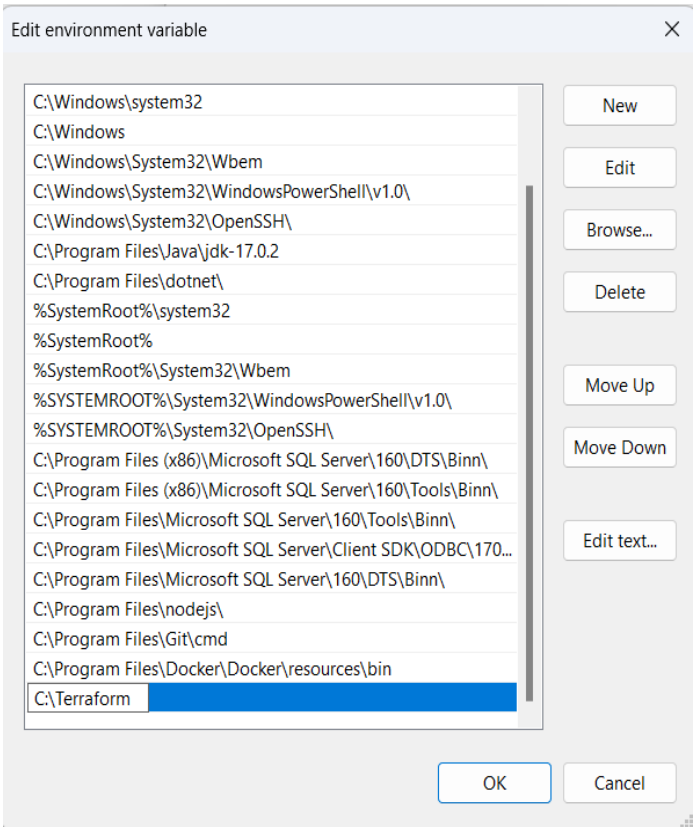
➢ Extract the downloaded setup file Terraform.exe in C:\Terraform directory

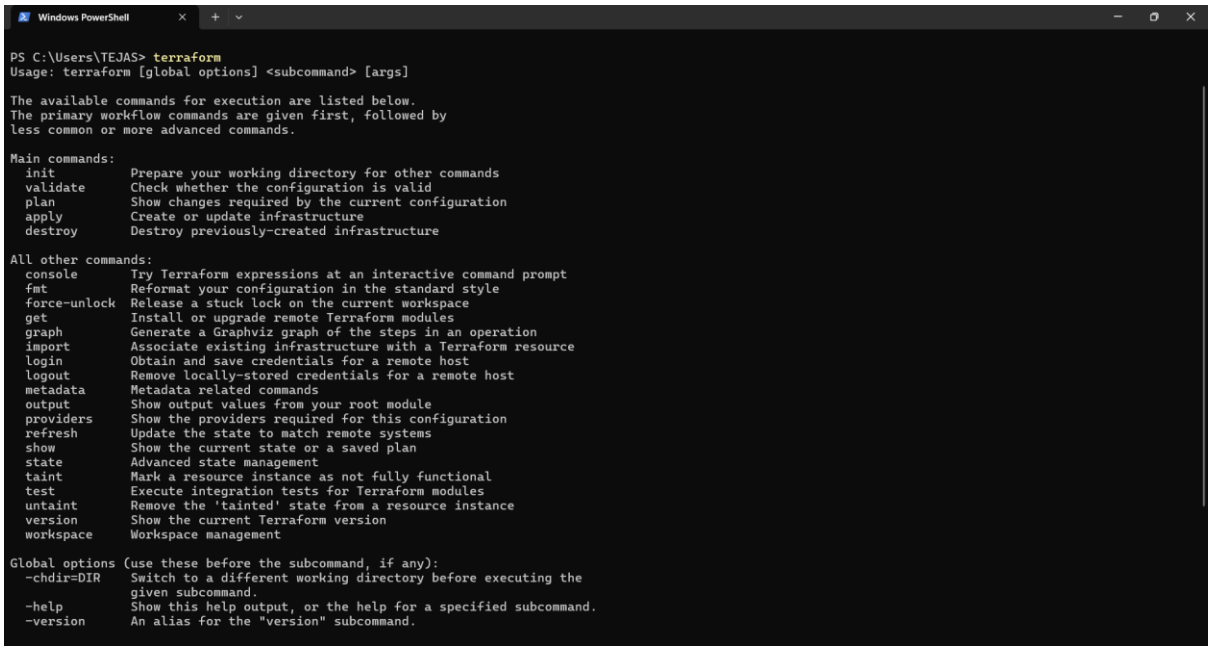


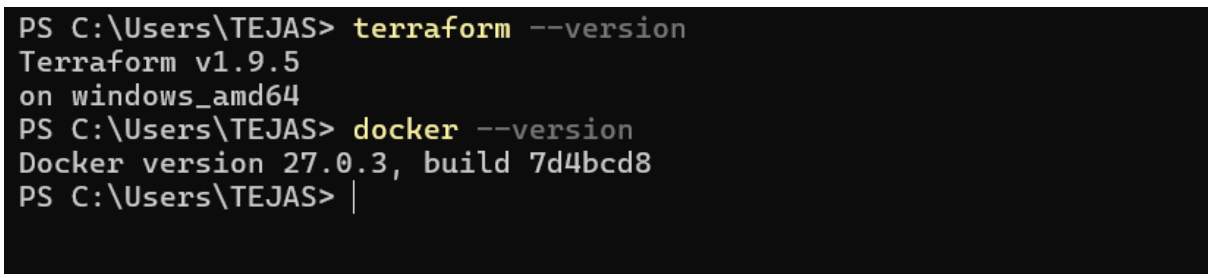


- Set the System path for Terraform in Environment Variables




Edit environment variable
C:\Windows\system32
C:\Windows
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
C:\Windows\System32\OpenSSH\
C:\Program Files\Java\jdk-17.0.2
C:\Program Files\dotnet\
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files (x86)\Microsoft SQL Server\160\DTS\Binn\
C:\Program Files (x86)\Microsoft SQL Server\160\Tools\Binn\
C:\Program Files\Microsoft SQL Server\160\Tools\Binn\
C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\170...
C:\Program Files\Microsoft SQL Server\160\DTS\Binn\
C:\Program Files\nodejs\
C:\Program Files\Git\cmd
C:\Program Files\Docker\Docker\resources\bin
C:\Terraform
New
Edit
Browse...
Delete
Move Up
Move Down
Edit text...
OK
Cancel


Windows PowerShell
PS C:\Users\TEJAS> terraform
Usage: terraform [global options] <subcommand> [args]
The available commands for execution are listed below.
The primary workflow commands are given first, followed by
less common or more advanced commands.
Main commands:
init Prepare your working directory for other commands
validate Check whether the configuration is valid
plan Show changes required by the current configuration
apply Create or update infrastructure
destroy Destroy previously-created infrastructure
All other commands:
console Try Terraform expressions at an interactive command prompt
fmt Reformat your configuration in the standard style
force-unlock Release a stuck lock on the current workspace
get Install or upgrade remote Terraform modules
graph Generate a Graphviz graph of the steps in an operation
import Associate existing infrastructure with a Terraform resource
login Obtain and save credentials for a remote host
logout Remove locally-stored credentials for a remote host
metadata Metadata related commands
output Show output values from your root module
providers Show the providers required for this configuration
refresh Update the state to match remote systems
show Show the current state or a saved plan
state Advanced state management
taint Mark a resource instance as not fully functional
test Execute integration tests for Terraform modules
untaint Remove the 'tainted' state from a resource instance
version Show the current Terraform version
workspace Workspace management
Global options (use these before the subcommand, if any):
-chdir=DIR Switch to a different working directory before executing the
given subcommand.
-help Show this help output, or the help for a specified subcommand.
-version An alias for the "version" subcommand.

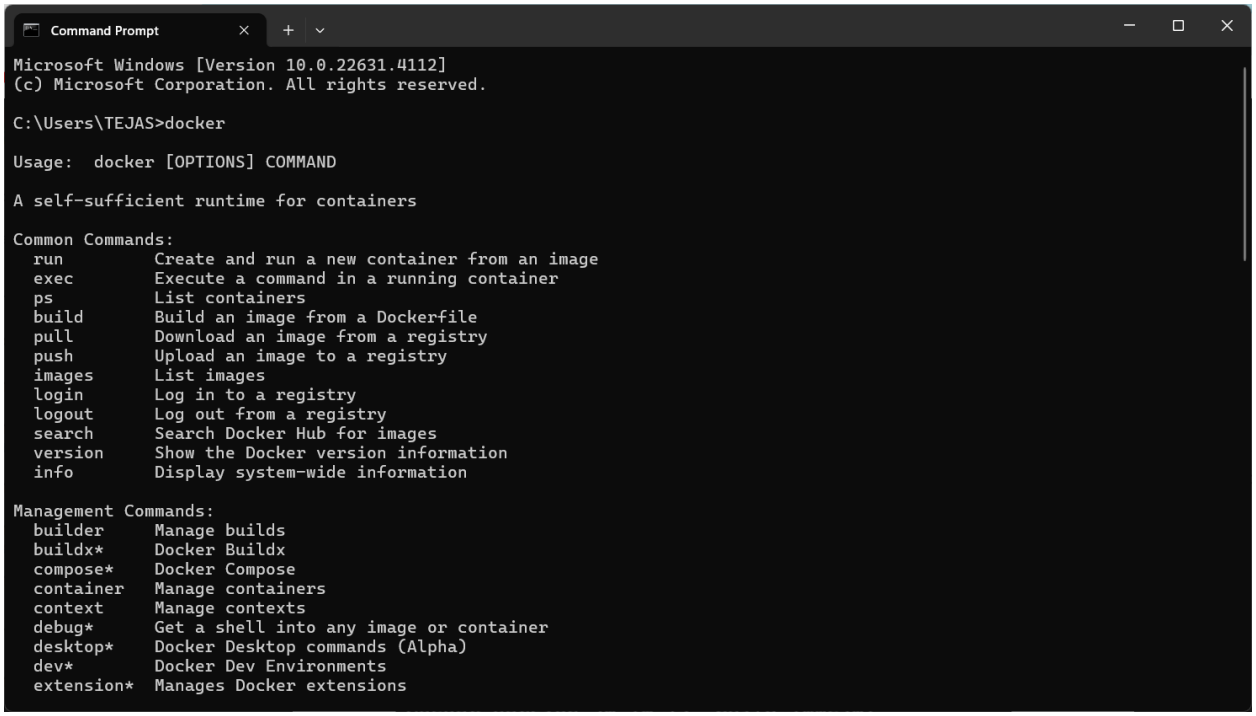

PS C:\Users\TEJAS> terraform --version
Terraform v1.9.5
on windows_amd64
PS C:\Users\TEJAS> docker --version
Docker version 27.0.3, build 7d4bcd8
PS C:\Users\TEJAS>

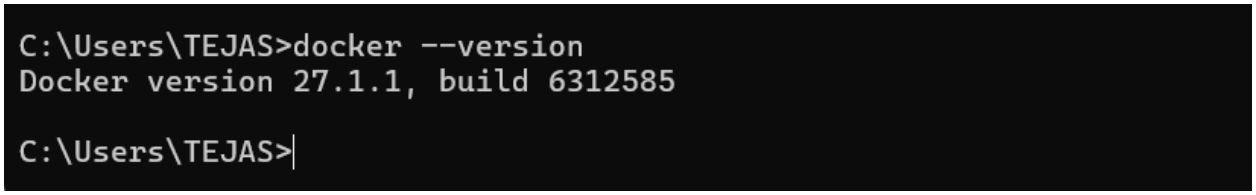
EXPERIMENT NO: - 06

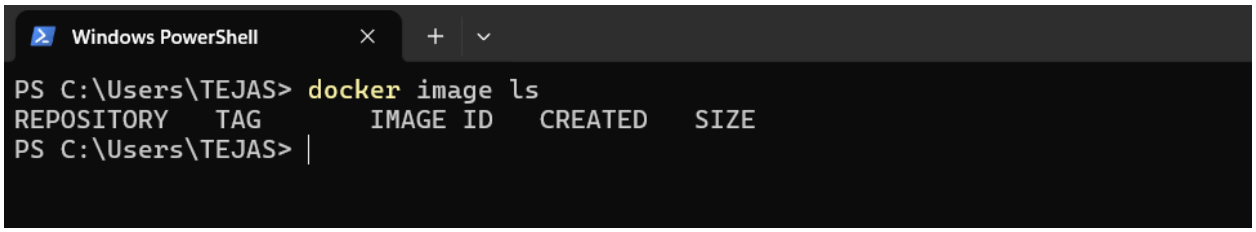
AIM: - To Build, change, and destroy AWS / GCP /Microsoft Azure/ DigitalOcean infrastructure Using Terraform. (S3 bucket or Docker)

A] Create Docker Image using Terraform

- Checking the Docker Functionality







- Creation of docker.tf file

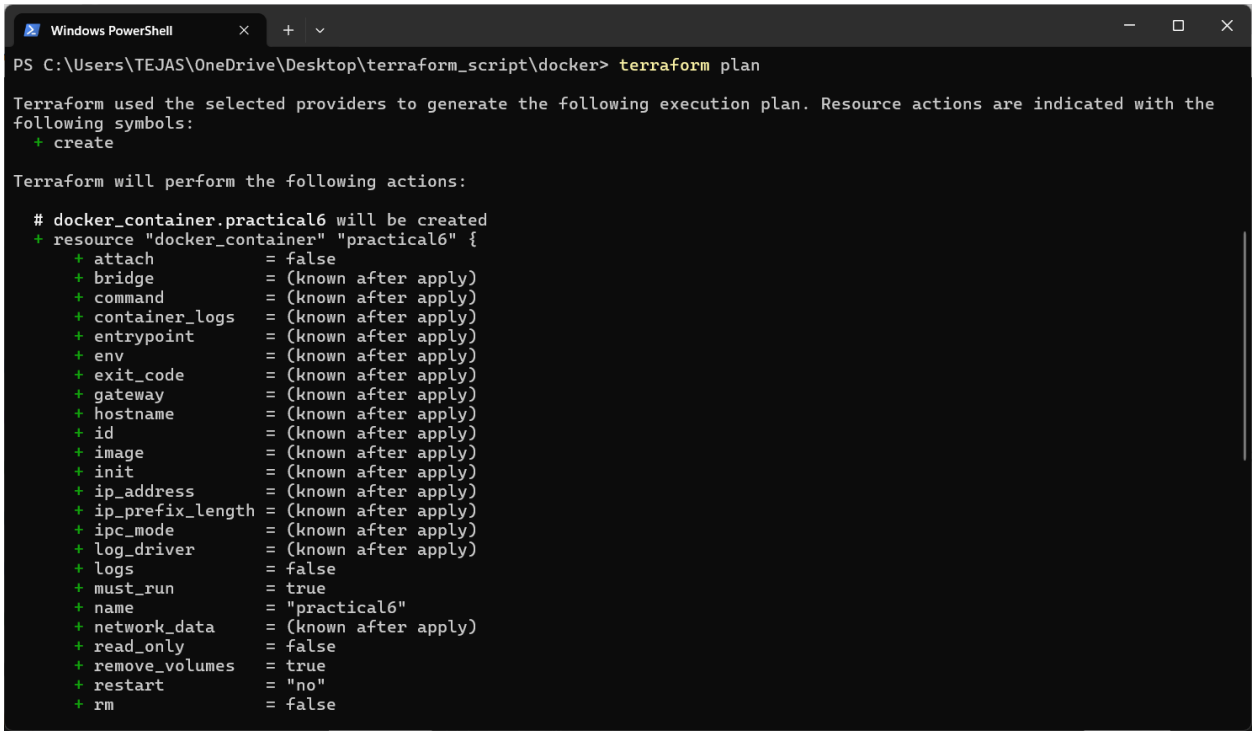
  Create a folder named ‘Terraform Scripts’ in which we save our different types of scripts. Then, create a new folder named ‘Docker’ in the ‘TerraformScripts’ folder.
  Then create a new docker.tf file to create a Ubuntu Linux container.

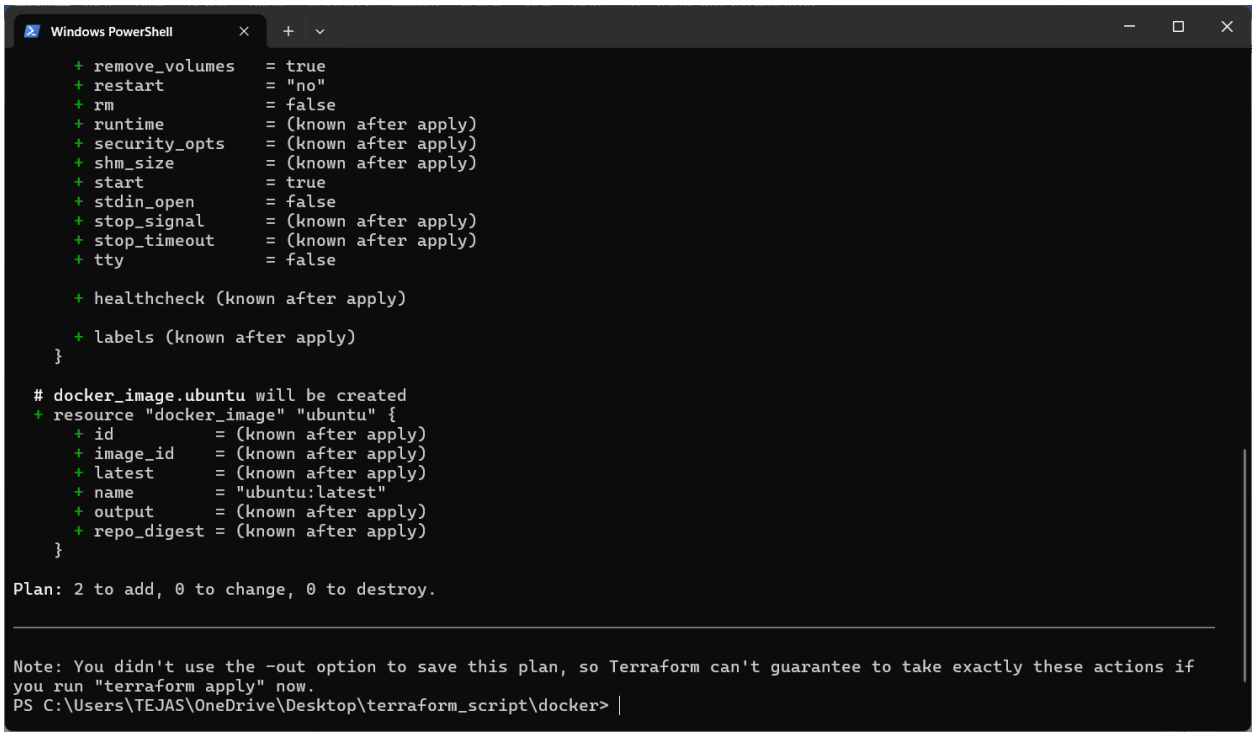
```
docker.tf > ...
1   terraform {
2     required_providers {
3       docker = {
4         source  = "kreuzwerker/docker"
5         version = "2.21.0"
6       }
7     }
8   }
9
10  provider "docker" {
11    host = "npipe:////.//pipe//docker_engine"
12  }
13
14  resource "docker_image" "ubuntu" {
15    name = "ubuntu:latest"
16  }
17
18  resource "docker_container" "practical6" {
19    image = docker_image.ubuntu.image_id
20    name  = "practical6"
21  }
22
```

- Execute Terraform Init command to initialize the resources

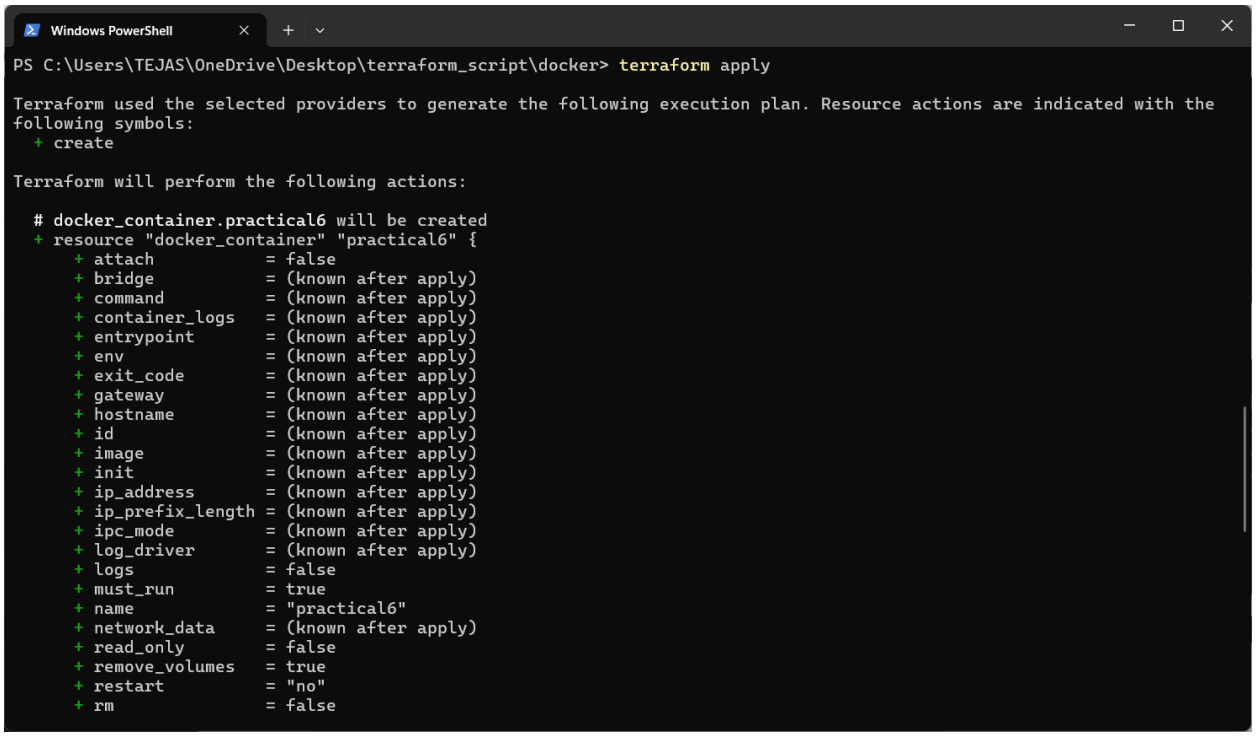
```
Windows PowerShell
PS C:\Users\TEJAS\OneDrive\Desktop\terraform_script\docker> terraform init
Initializing the backend...
Initializing provider plugins...
- Finding kreuzwerker/docker versions matching "2.21.0"...
- Installing kreuzwerker/docker v2.21.0...
- Installed kreuzwerker/docker v2.21.0 (self-signed, key ID 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.
PS C:\Users\TEJAS\OneDrive\Desktop\terraform_script\docker>
```

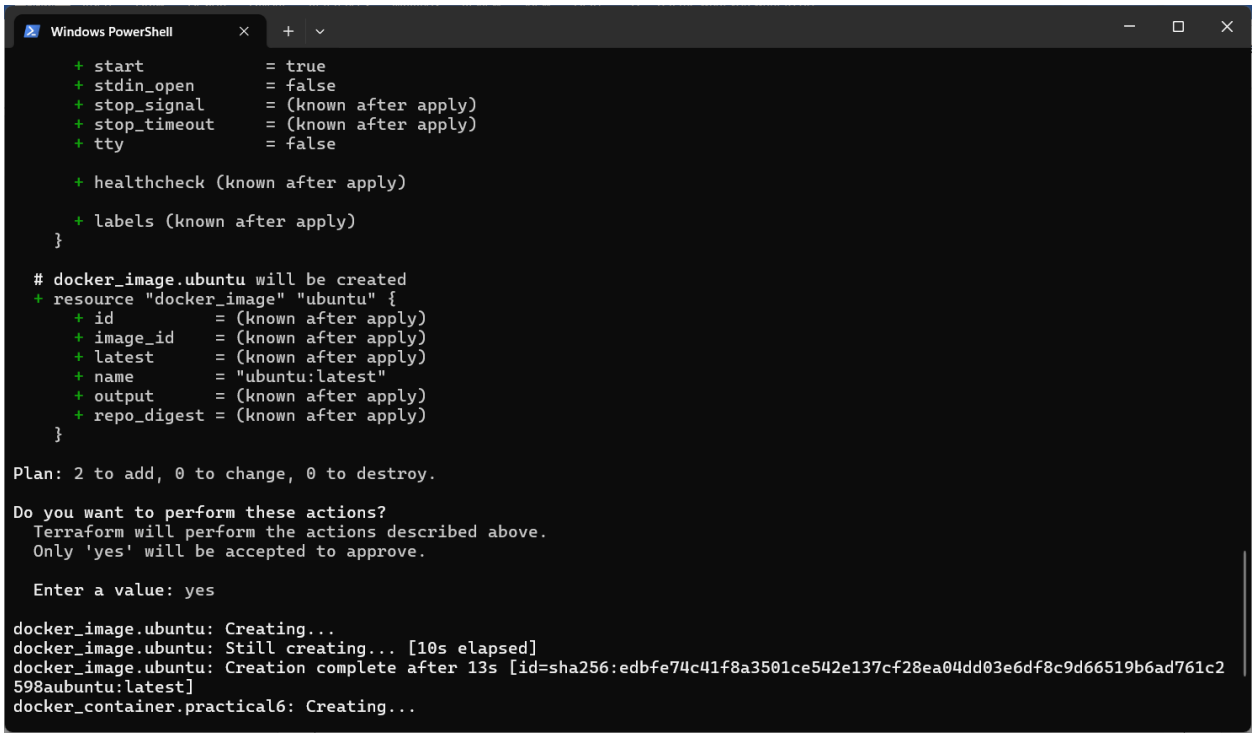
- Execute Terraform plan to see the available resources



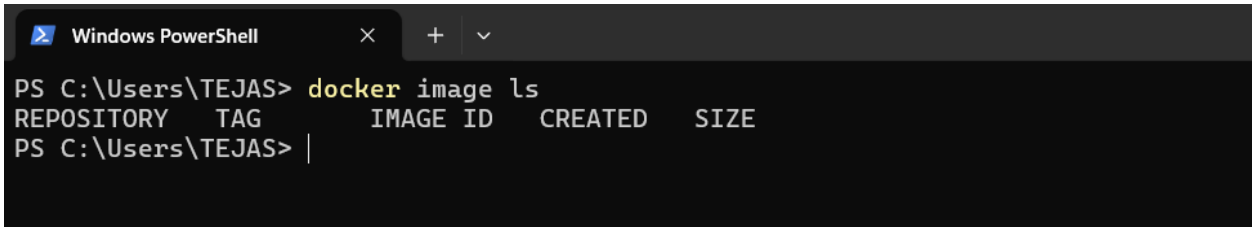


- Execute Terraform apply to apply the configuration, which will automatically create and run the Ubuntu Linux container based on our configuration.

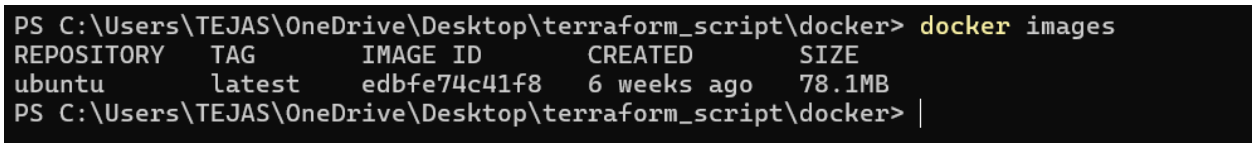




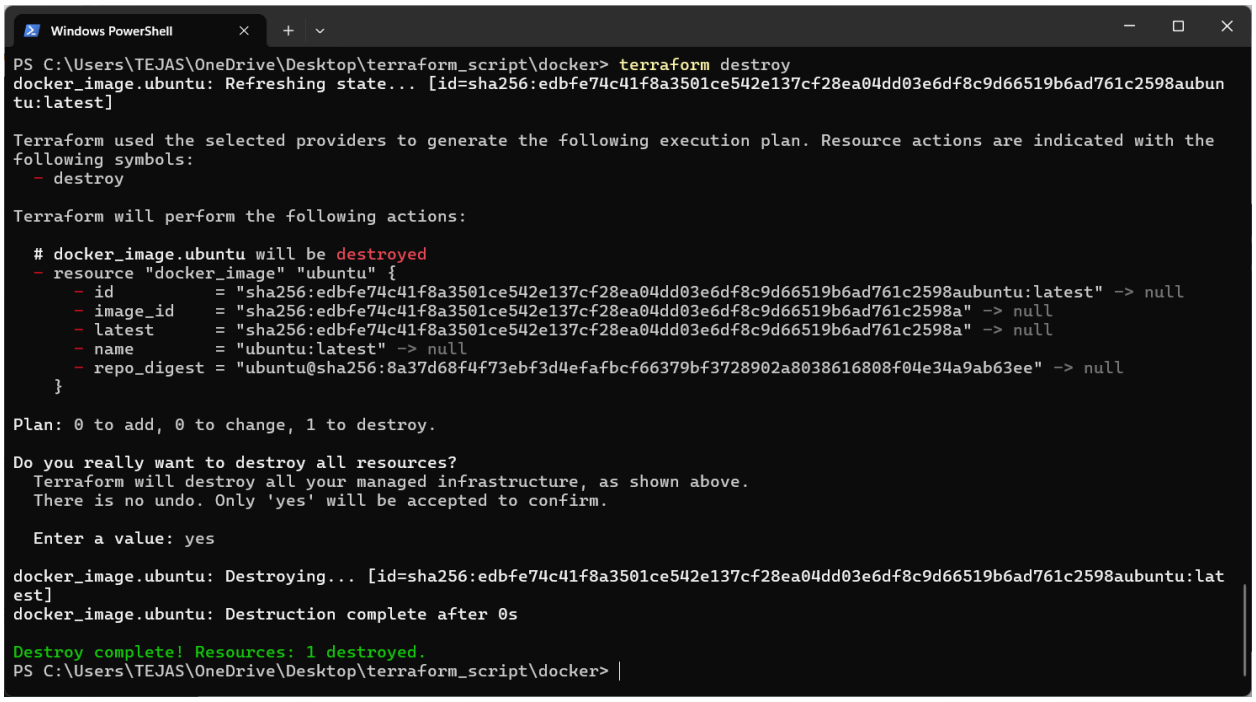
- Docker images, Before Executing Apply step:



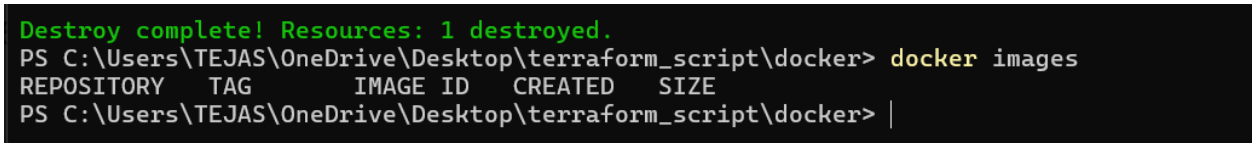
- Docker images, After Executing Apply step:



- Execute Terraform destroy to delete the configuration, which will automatically delete the Ubuntu Container.



- Docker images After Executing Destroy step



B] Creating S3 Bucket using terraform.

- Write a Terraform Script in Atom for creating S3 Bucket on Amazon AWS

aws-s3 > main.tf > ...

```
terraform {
  required_providers {
    aws={
        source="hashicorp/aws"
        version="5.64.0"
    }
    random = {
      source = "hashicorp/random"
      version = "3.6.2"
    }
  }
}

resource "random_id" "rand_id"{
    byte_length = 8
}

resource "aws_s3_bucket" "demo-bucket" {
  bucket = "demo-bucket-${random_id.rand_id.hex}"
}

resource "aws_s3_object" "bucket-data" {
  bucket = aws_s3_bucket.demo-bucket.bucket
  source = "./myfile.txt"
  key="newfile.txt"
}
```

- Create a new provider.tf file

aws-s3 > provider.tf > ...

```
provider "aws" {
  access_key="ASIAXDVAWUCTAHPBYALZ"
  secret_key="kdP24c5nfLcCbuWQe49jBthzJc4RefZBns6t6J8y"
  token="IQoJb3JpZ2luX2VjEJf//////////wEaCXVzLXdlc3QtMiJIMEYCIQCSJ3i5FhAaZU/xRQHF5L2DoNBiAuOG0PAemX7fCnC2bwIhAPIRX10U0Af6Bz3Qs7oX8Cnpn62XK7LhDo3/
  u4vx66juKsACCMD/////////wEQABoMNDg4ODg5NTU3MTU4IgzPDJgqgUUyZ9njLWoqlAKlv6HhXOV2Vhev/w51dDaiTqa9/
  lTgri1VOfAFoZhGHRcK5yuOAho5IXYG0thTErBx8N1n0i8fQFbqnI/dB/pEMSk02xh/5PYBrdulaHVSWUvV86LPqZzJ0wTS4iTn+l6fi6PyhM4kZtbPTPixhAjZGJuvNYzrPbe6t6lhGsw/
  Rsv51B9lHEjyVyW7LDQsXvz+YX4ie5DtO9+Xbxh+WMWAp2+GucNkQOKRg+IbkdRPf5njQHJ818h3xQYD72tIGmWwXZgBHrobKI5zVkKFzMseofXImgj1hlZQ8zGof9KOUlxdivnRpFzH
  +FhYPV/cCEM1jE8M1euRcDlRnaD5E7PdAg3k7F8FnMAfCu4qck9uhQCJOo0wqZyRtwY6nAGhyr0X3yVQrC1EWK3j6HbY5MOJC4lThme7fNTs5HyKfDVSFUGZkRzoJspESepbn/
  YnNlOAadFZCEbY4ayUfSSEhMnKHyZdl85xewRnp8kZTKSdxWzqyuTjoi/dL1vfq0B5jPSdJ4J0v453YIsyZmXRnWdh8AnYg2zAyw3kC5CdipQLDEti37zP20nciYcA1Oy+iF+e3Cw/
  vcgFR08="
  region = "us-east-1"
}
\end{code}
```

- Execute Terraform Init command to initialize the resources

```
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical>
cd aws-s3
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical\aws-s3> terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/random from the dependency lock file
- Using previously-installed hashicorp/aws v5.64.0
- Using previously-installed hashicorp/random v3.6.2

Terraform has been successfully initialized!

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

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical\aws-s3>
```

- Execute Terraform plan to see the available resources

```
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical\aws-s3> terraform plan
random_id.rand_id: Refreshing state... [id=mMJ9L9tLf-c]
aws_s3_bucket.demo-bucket: Refreshing state... [id=demo-bucket-98c27d2fdb4b7fe7]
aws_s3_object.bucket-data: Refreshing state... [id=newfile.txt]

Note: Objects have changed outside of Terraform

Terraform detected the following changes made outside of Terraform since the last "terraform apply" which may have affected this plan:

  # aws_s3_bucket.demo-bucket has been deleted
  - resource "aws_s3_bucket" "demo-bucket" {
      - bucket                      = "demo-bucket-98c27d2fdb4b7fe7" -> null
        id                          = "demo-bucket-98c27d2fdb4b7fe7"
        # (12 unchanged attributes hidden)

        # (3 unchanged blocks hidden)
    }


Unless you have made equivalent changes to your configuration, or ignored the relevant attributes using ignore_changes, the following plan may include actions to
undo or respond to these changes.

─────────────────────────────────────────────────────────────────────────────


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

Terraform will perform the following actions:

  # aws_s3_bucket.demo-bucket will be created
  + resource "aws_s3_bucket" "demo-bucket" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "demo-bucket-98c27d2fdb4b7fe7"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
```

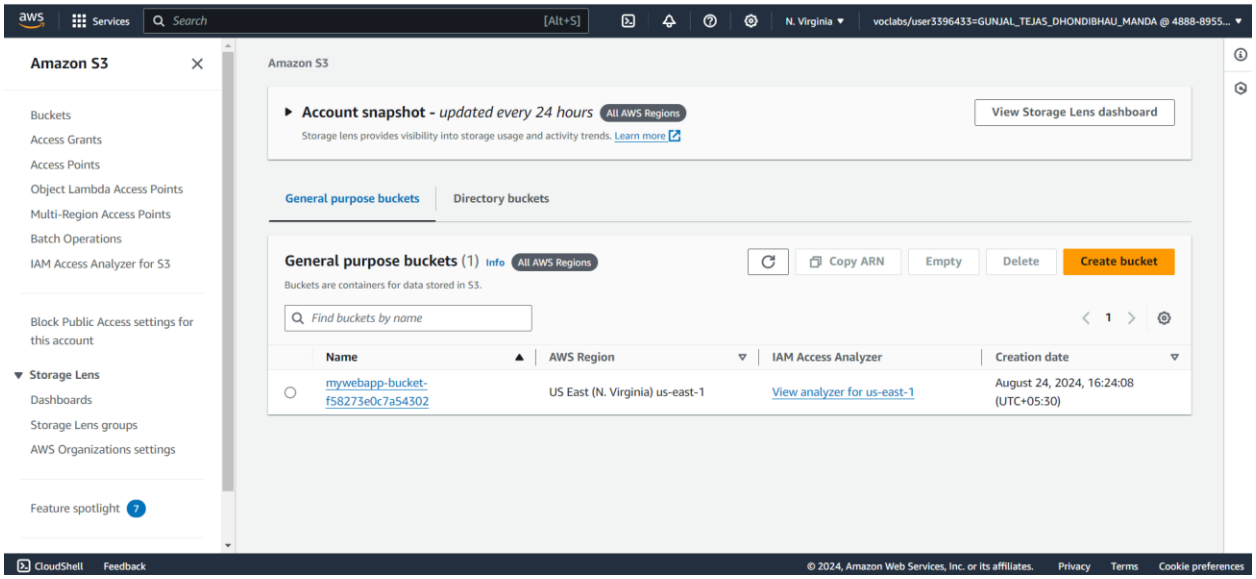
```
      + replication_configuration (known after apply)

      + server_side_encryption_configuration (known after apply)

      + versioning (known after apply)

      + website (known after apply)
    }

  # aws_s3_object.bucket-data will be created
  + resource "aws_s3_object" "bucket-data" {
      + acl                    = (known after apply)
      + arn                    = (known after apply)
      + bucket                 = "demo-bucket-98c27d2fdb4b7fe7"
      + bucket_key_enabled     = (known after apply)
      + checksum_crc32         = (known after apply)
      + checksum_crc32c        = (known after apply)
      + checksum_sha1          = (known after apply)
      + checksum_sha256        = (known after apply)
      + content_type           = (known after apply)
      + etag                   = (known after apply)
      + force_destroy          = false
      + id                     = (known after apply)
      + key                    = "newfile.txt"
      + kms_key_id             = (known after apply)
      + server_side_encryption = (known after apply)
      + source                 = "./myfile.txt"
      + storage_class          = (known after apply)
      + tags_all               = (known after apply)
      + version_id             = (known after apply)
    }

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

─────────────────────────────────────────────────────────────────────────────

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical\aws-s3>
```

- Execute Terraform apply to apply the configuration, which will automatically create an S3 bucket based on our configuration.

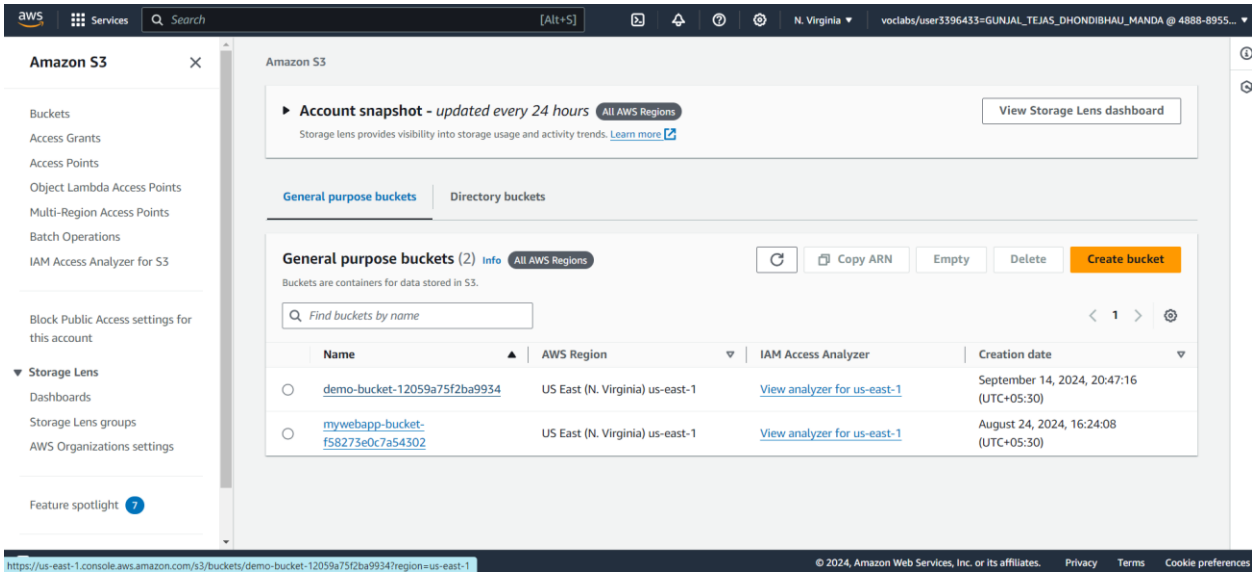
```
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical\aws-s3> terraform apply
random_id.rand_id: Refreshing state... [id=mMJ9L9tLf-c]
aws_s3_bucket.demo-bucket: Refreshing state... [id=demo-bucket-98c27d2fdb4b7fe7]
aws_s3_object.bucket-data: Refreshing state... [id=newfile.txt]

Note: Objects have changed outside of Terraform

Terraform detected the following changes made outside of Terraform since the last "terraform apply" which may have affected this plan:

  # aws_s3_bucket.demo-bucket has been deleted
  - resource "aws_s3_bucket" "demo-bucket" {
      - bucket                      = "demo-bucket-98c27d2fdb4b7fe7" -> null
        id                          = "demo-bucket-98c27d2fdb4b7fe7"
        # (12 unchanged attributes hidden)

        # (3 unchanged blocks hidden)
    }


Unless you have made equivalent changes to your configuration, or ignored the relevant attributes using ignore_changes, the following plan may include actions to
undo or respond to these changes.

─────────────────────────────────────────────────────────────────────────────

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

Terraform will perform the following actions:

  # aws_s3_bucket.demo-bucket will be created
  + resource "aws_s3_bucket" "demo-bucket" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "demo-bucket-98c27d2fdb4b7fe7"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
```

```
  # aws_s3_object.bucket-data will be created
  + resource "aws_s3_object" "bucket-data" {
      + acl                    = (known after apply)
      + arn                    = (known after apply)
      + bucket                 = "demo-bucket-98c27d2fdb4b7fe7"
      + bucket_key_enabled     = (known after apply)
      + checksum_crc32         = (known after apply)
      + checksum_crc32c        = (known after apply)
      + checksum_sha1          = (known after apply)
      + checksum_sha256        = (known after apply)
      + content_type           = (known after apply)
      + etag                   = (known after apply)
      + force_destroy          = false
      + id                     = (known after apply)
      + key                    = "newfile.txt"
      + kms_key_id             = (known after apply)
      + server_side_encryption = (known after apply)
      + source                 = "./myfile.txt"
      + storage_class          = (known after apply)
      + tags_all               = (known after apply)
      + version_id             = (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

aws_s3_bucket.demo-bucket: Creating...
aws_s3_bucket.demo-bucket: Creation complete after 7s [id=demo-bucket-98c27d2fdb4b7fe7]
aws_s3_object.bucket-data: Creating...
aws_s3_object.bucket-data: Creation complete after 2s [id=newfile.txt]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical\aws-s3>
```

- AWS S3bucket dashboard, Before Executing Apply command:



- AWS S3 Bucket dashboard, After Executing Apply step:

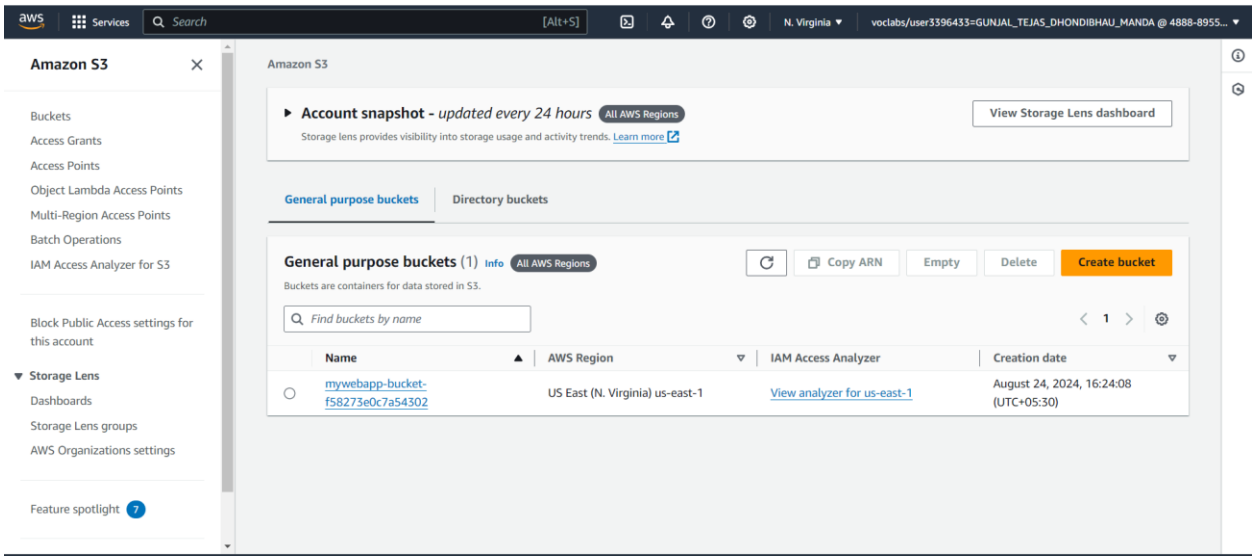


- Execute Terraform destroy to delete the configuration, which will automatically delete the bucket

```
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical\aws-s3> terraform destroy
random_id.rand_id: Refreshing state... [id=mMJ9L9tLf-c]
aws_s3_bucket.demo-bucket: Refreshing state... [id=demo-bucket-98c27d2fdb4b7fe7]
aws_s3_object.bucket-data: Refreshing state... [id=newfile.txt]

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

Terraform will perform the following actions:

  # aws_s3_bucket.demo-bucket will be destroyed
  - resource "aws_s3_bucket" "demo-bucket" {
      - arn                         = "arn:aws:s3:::demo-bucket-98c27d2fdb4b7fe7" -> null
      - bucket                      = "demo-bucket-98c27d2fdb4b7fe7" -> null
      - bucket_domain_name          = "demo-bucket-98c27d2fdb4b7fe7.s3.amazonaws.com" -> null
      - bucket_regional_domain_name = "demo-bucket-98c27d2fdb4b7fe7.s3.us-east-1.amazonaws.com" -> null
      - force_destroy               = false -> null
      - hosted_zone_id              = "Z3AQBSTGFYJSTF" -> null
      - id                          = "demo-bucket-98c27d2fdb4b7fe7" -> null
      - object_lock_enabled         = false -> null
      - region                      = "us-east-1" -> null
      - request_payer               = "BucketOwner" -> null
      - tags                        = {} -> null
      - tags_all                    = {} -> null
        # (3 unchanged attributes hidden)

      - grant {
          - id          = "89f1916699ac713a2d0a157aa4176d8ce16adda216e3bafb6e7f0811cb559dca" -> null
          - permissions = [
              - "FULL_CONTROL",
            ] -> null
          - type        = "CanonicalUser" -> null
            # (1 unchanged attribute hidden)
        }

      - server_side_encryption_configuration {
          - rule {
              - bucket_key_enabled = false -> null

              - apply_server_side_encryption_by_default {
                  - sse_algorithm     = "AES256" -> null
```

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

  Enter a value: yes

aws_s3_object.bucket-data: Destroying... [id=newfile.txt]
aws_s3_object.bucket-data: Destruction complete after 1s
aws_s3_bucket.demo-bucket: Destroying... [id=demo-bucket-98c27d2fdb4b7fe7]
aws_s3_bucket.demo-bucket: Destruction complete after 1s
random_id.rand_id: Destroying... [id=mMJ9L9tLf-c]
random_id.rand_id: Destruction complete after 0s

Destroy complete! Resources: 3 destroyed.
PS C:\Users\TEJAS\OneDrive\Desktop\SEM 5 VESIT\Teraform practical\aws-s3>
```

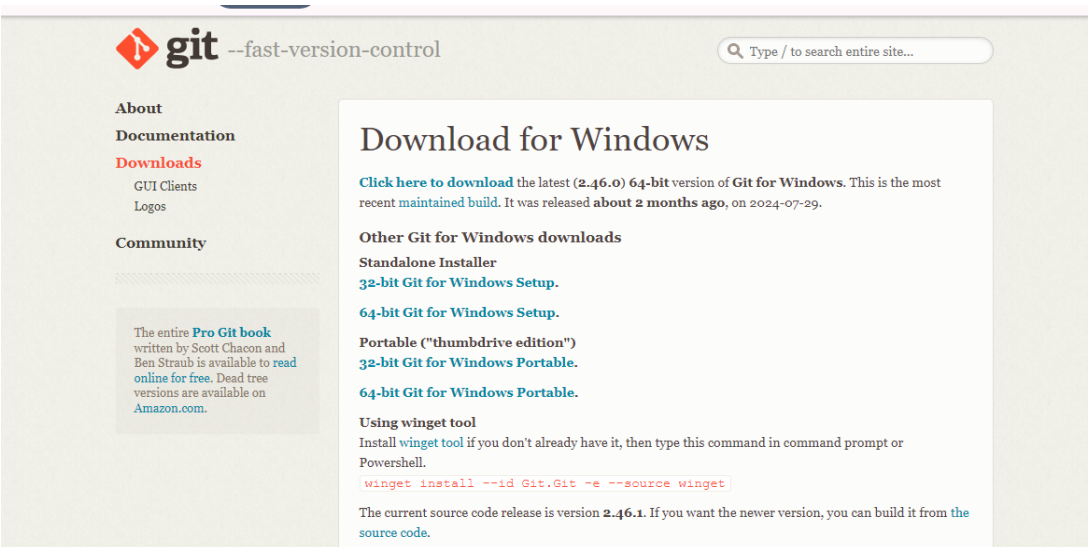
- AWS S3 Bucket dashboard, After Executing Destroy step:



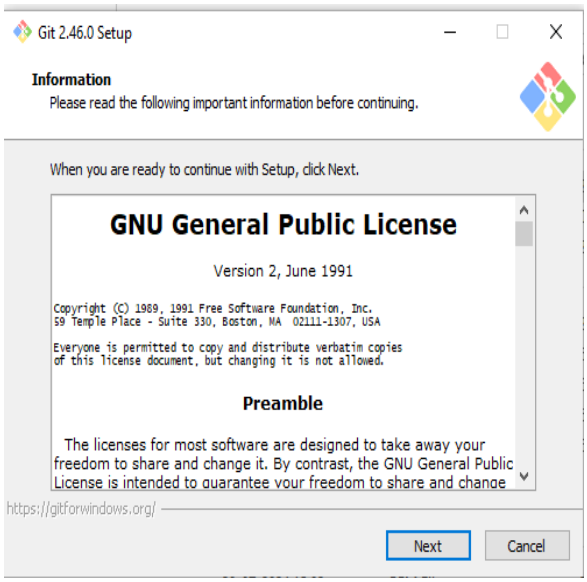
EXPERIMENT NO: - 07

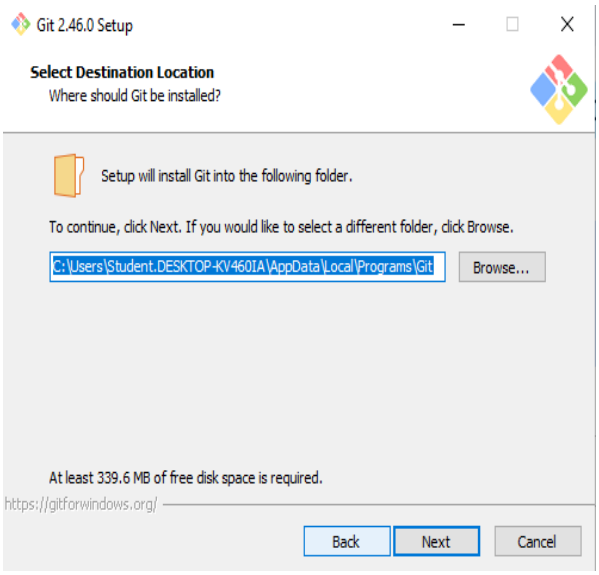
AIM:- To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab

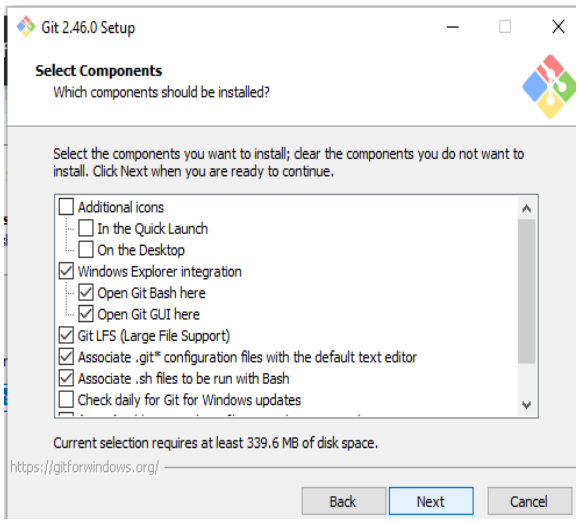
➢ GIT Installation

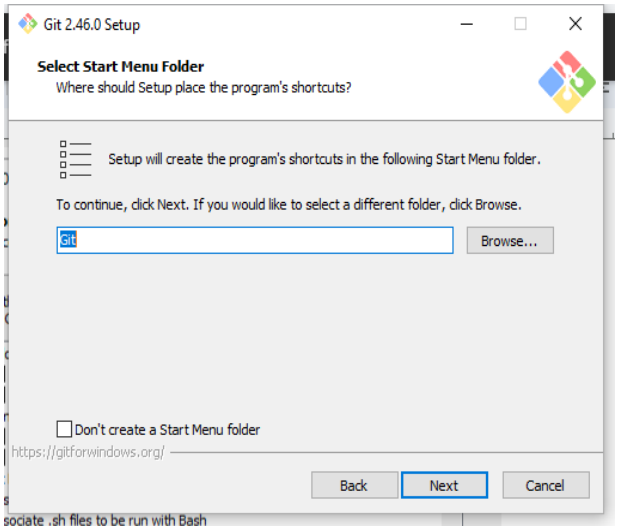


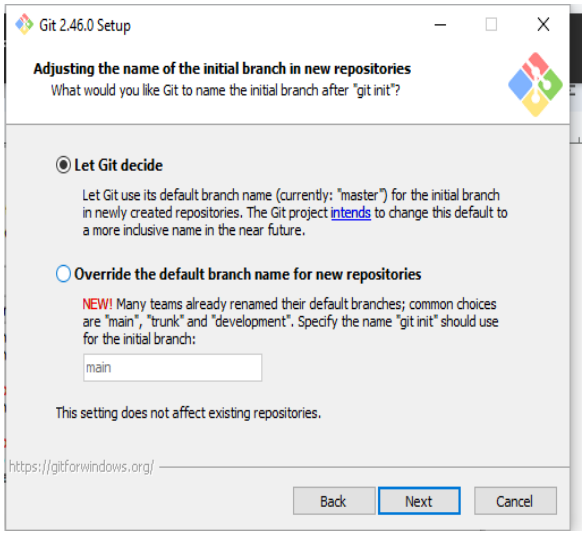
Select Installation for Windows setup and set up the destination for GIT installation.

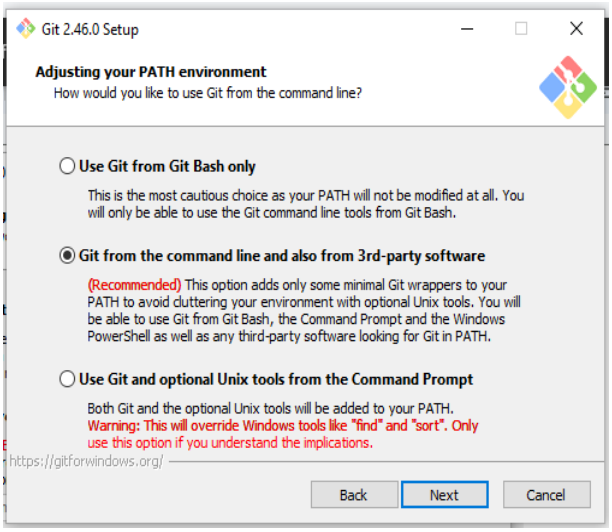


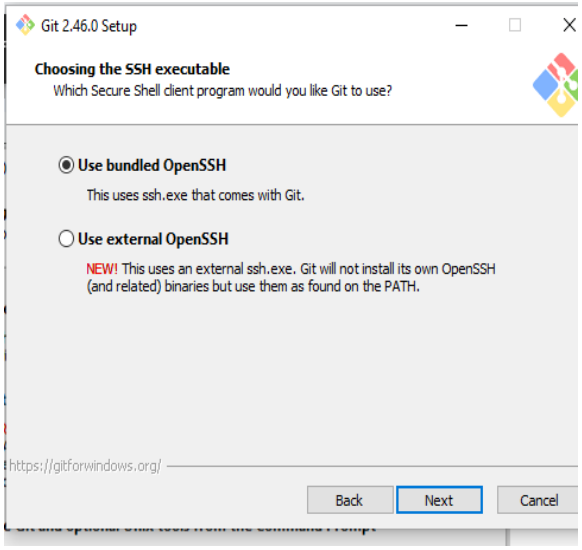


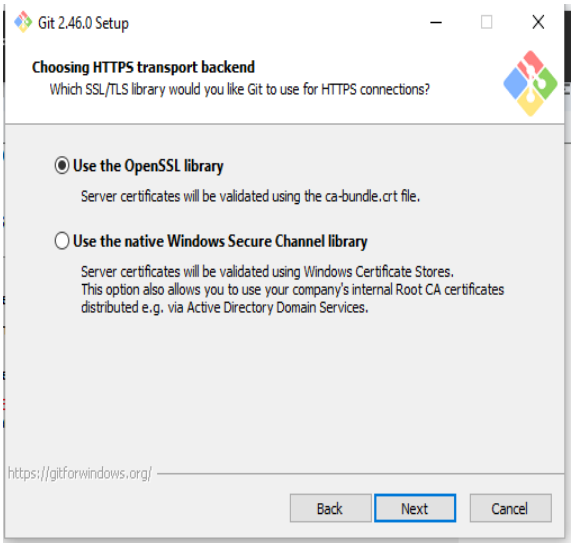

Git 2.46.0 Setup
Select Components
Which components should be installed?
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
Additional icons
In the Quick Launch
On the Desktop
Windows Explorer integration
Open Git Bash here
Open Git GUI here
Git LFS (Large File Support)
Associate .git* configuration files with the default text editor
Associate .sh files to be run with Bash
Check daily for Git for Windows updates
Current selection requires at least 339.6 MB of disk space.
https://gitforwindows.org/
Back
Next
Cancel


Git 2.46.0 Setup
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
Git
Browse...
Don't create a Start Menu folder
https://gitforwindows.org/
Back
Next
Cancel

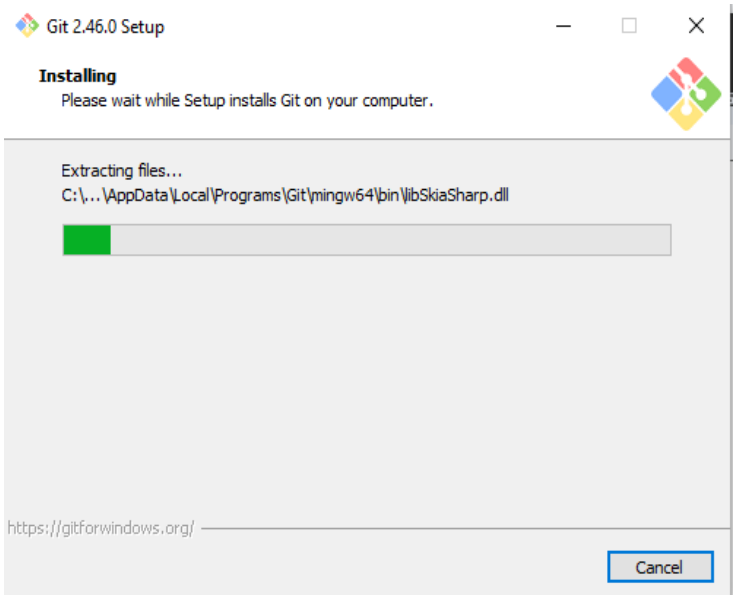

Git 2.46.0 Setup
Adjusting the name of the initial branch in new repositories
What would you like Git to name the initial branch after "git init"?
Let Git decide
Let Git use its default branch name (currently: "master") for the initial branch in newly created repositories. The Git project intends to change this default to a more inclusive name in the near future.
Override the default branch name for new repositories
NEW! Many teams already renamed their default branches; common choices are "main", "trunk" and "development". Specify the name "git init" should use for the initial branch:
main
This setting does not affect existing repositories.
https://gitforwindows.org/
Back
Next
Cancel

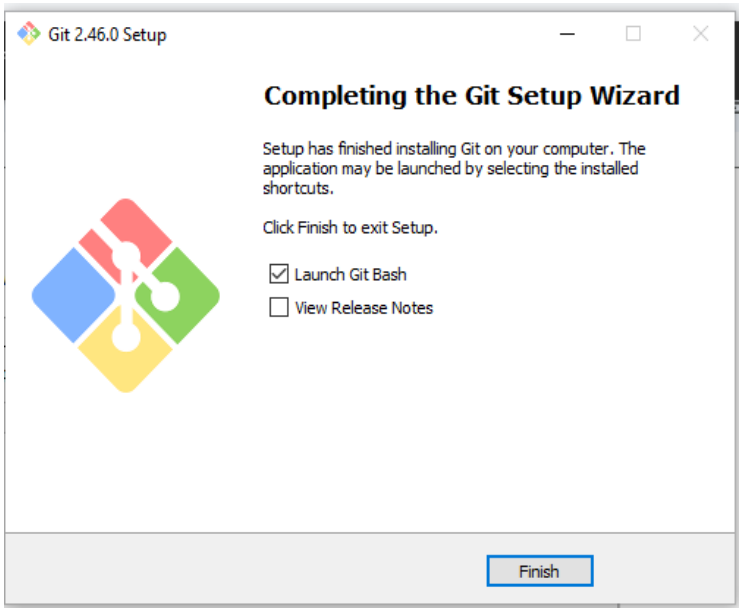

Git 2.46.0 Setup
Adjusting your PATH environment
How would you like to use Git from the command line?
Use Git from Git Bash only
This is the most cautious choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.
Git from the command line and also from 3rd-party software
(Recommended) This option adds only some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools. You will be able to use Git from Git Bash, the Command Prompt and the Windows PowerShell as well as any third-party software looking for Git in PATH.
Use Git and optional Unix tools from the Command Prompt
Both Git and the optional Unix tools will be added to your PATH.
Warning: This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.
https://gitforwindows.org/
Back
Next
Cancel


Git 2.46.0 Setup
Choosing the SSH executable
Which Secure Shell client program would you like Git to use?
Use bundled OpenSSH
This uses ssh.exe that comes with Git.
Use external OpenSSH
NEW! This uses an external ssh.exe. Git will not install its own OpenSSH (and related) binaries but use them as found on the PATH.
https://gitforwindows.org/
Back
Next
Cancel

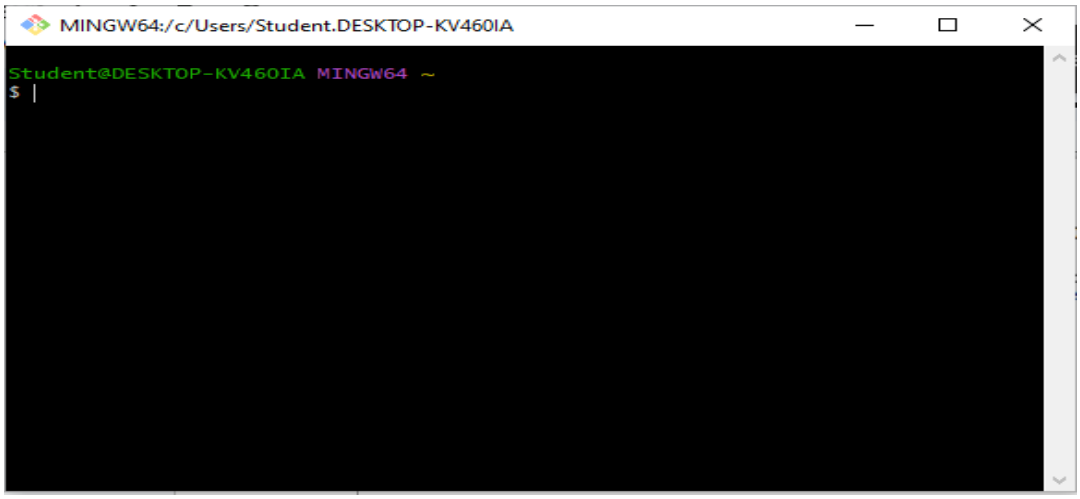

Git 2.46.0 Setup
Choosing HTTPS transport backend
Which SSL/TLS library would you like Git to use for HTTPS connections?
Use the OpenSSL library
Server certificates will be validated using the ca-bundle.crt file.
Use the native Windows Secure Channel library
Server certificates will be validated using Windows Certificate Stores.
This option also allows you to use your company's internal Root CA certificates distributed e.g. via Active Directory Domain Services.
https://gitforwindows.org/
Back
Next
Cancel

After all such configurations, Git installation is started



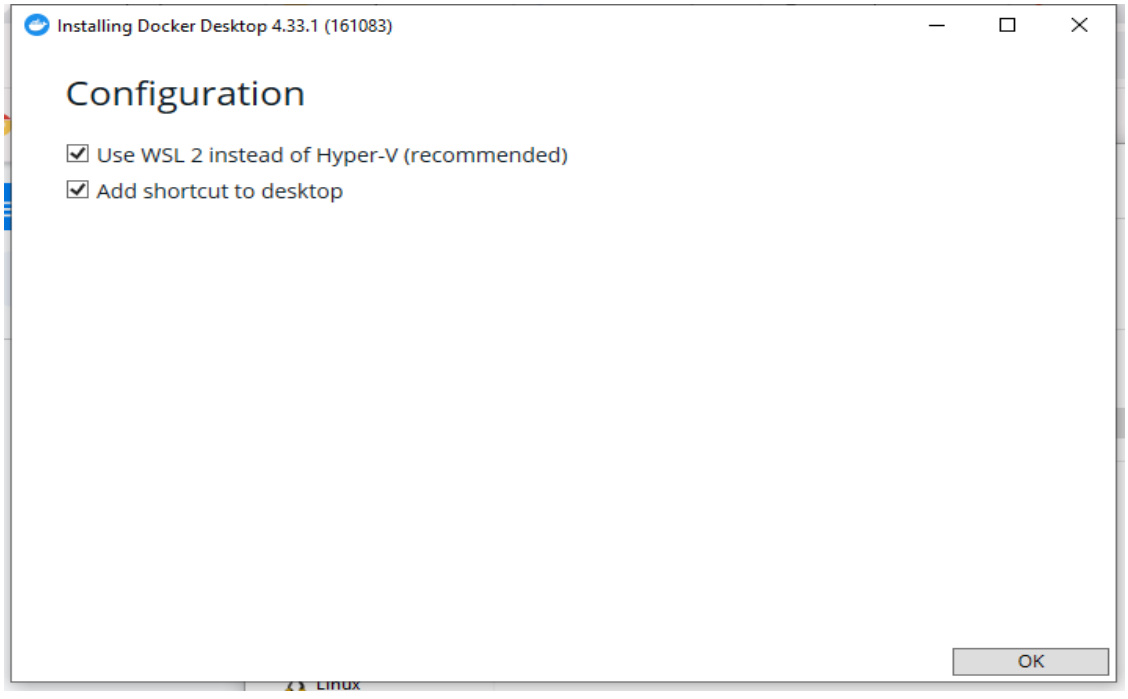


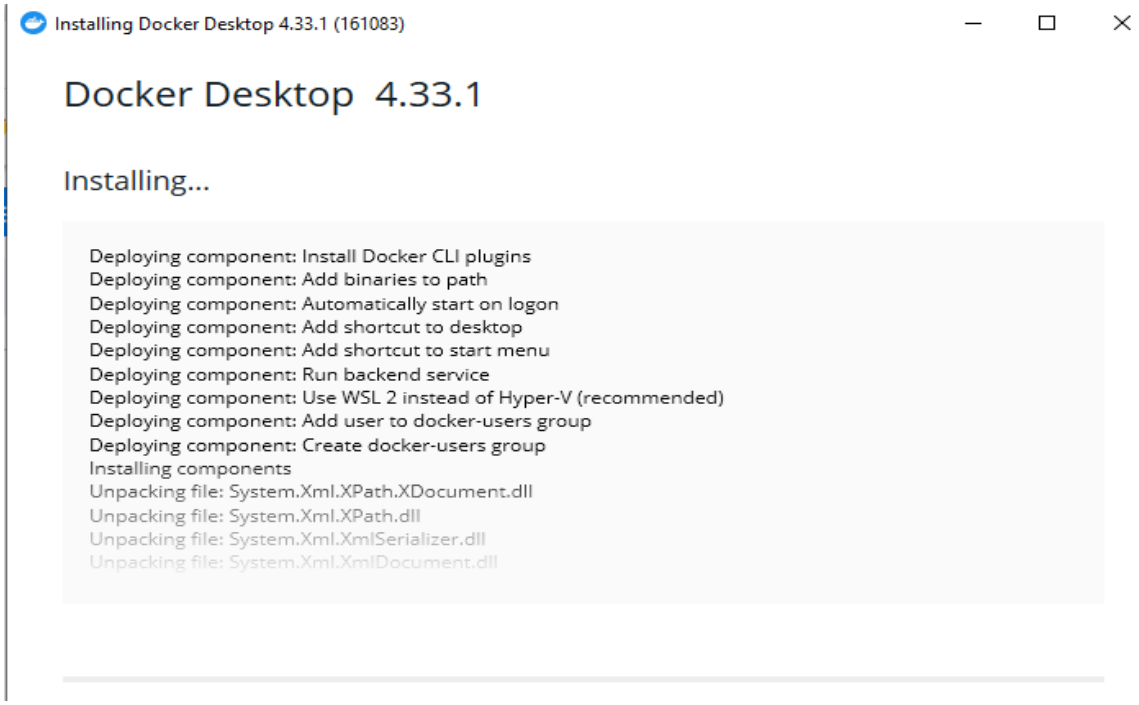
After Installation, Git Bash is launched

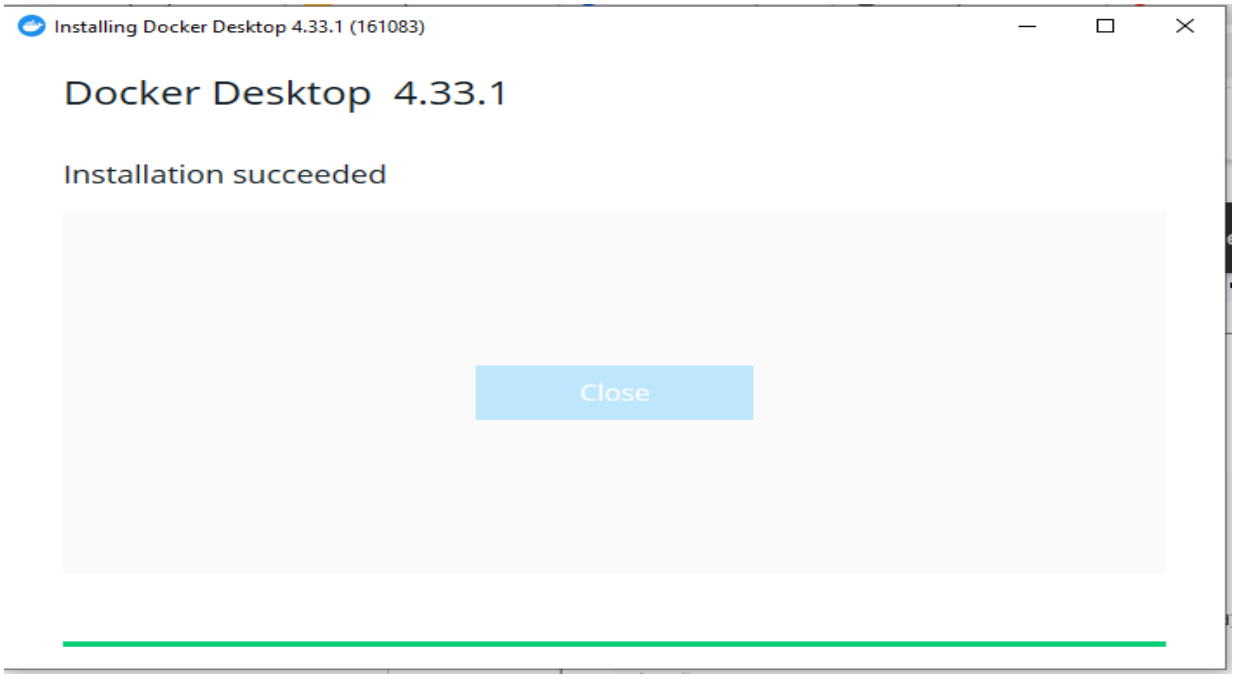


➢ Docker Installation

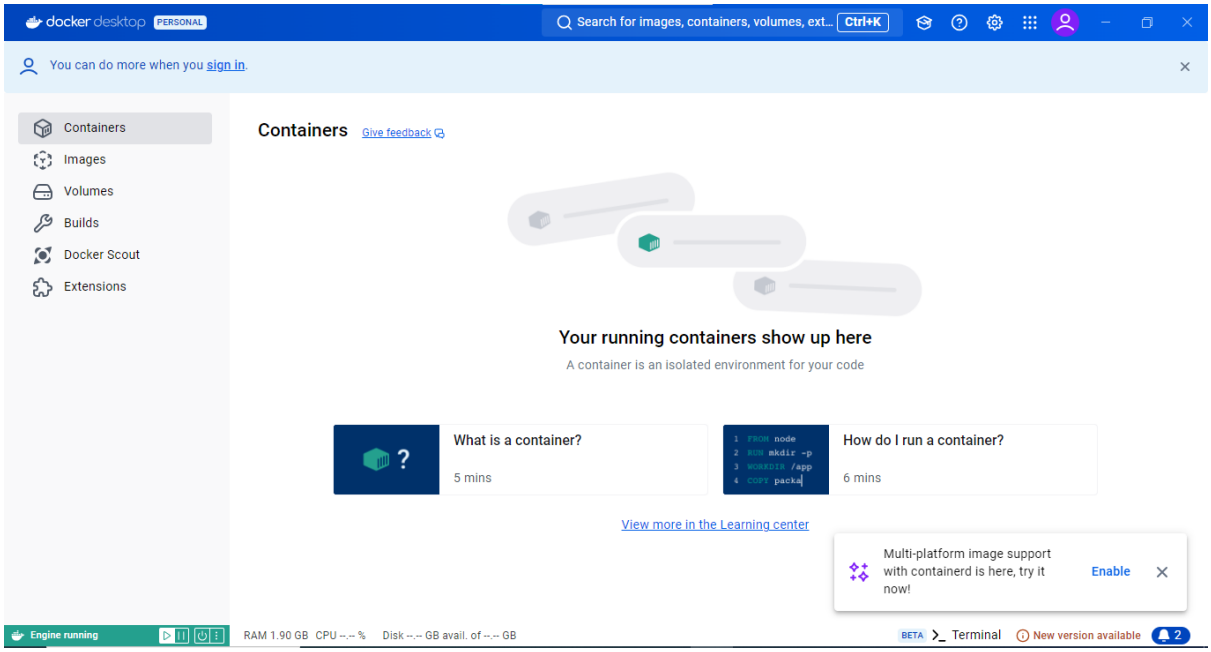
Enable WSL2 in order to work the docker correctly



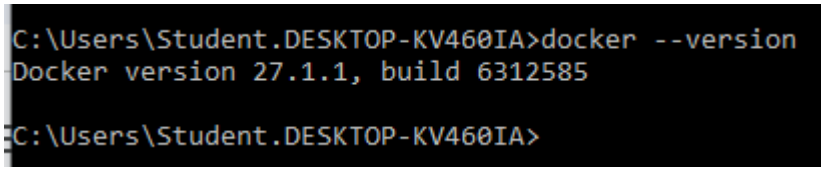

Installing Docker Desktop 4.33.1 (161083)
Docker Desktop 4.33.1
Installing...
Deploying component: Install Docker CLI plugins
Deploying component: Add binaries to path
Deploying component: Automatically start on logon
Deploying component: Add shortcut to desktop
Deploying component: Add shortcut to start menu
Deploying component: Run backend service
Deploying component: Use WSL 2 instead of Hyper-V (recommended)
Deploying component: Add user to docker-users group
Deploying component: Create docker-users group
Installing components
Unpacking file: System.Xml.XPath.XDocument.dll
Unpacking file: System.Xml.XPath.dll
Unpacking file: System.Xml.XmlSerializer.dll
Unpacking file: System.Xml.XmlDocument.dll


Installing Docker Desktop 4.33.1 (161083)
Docker Desktop 4.33.1
Installation succeeded
Close

After Docker is installed, homepage will look like these

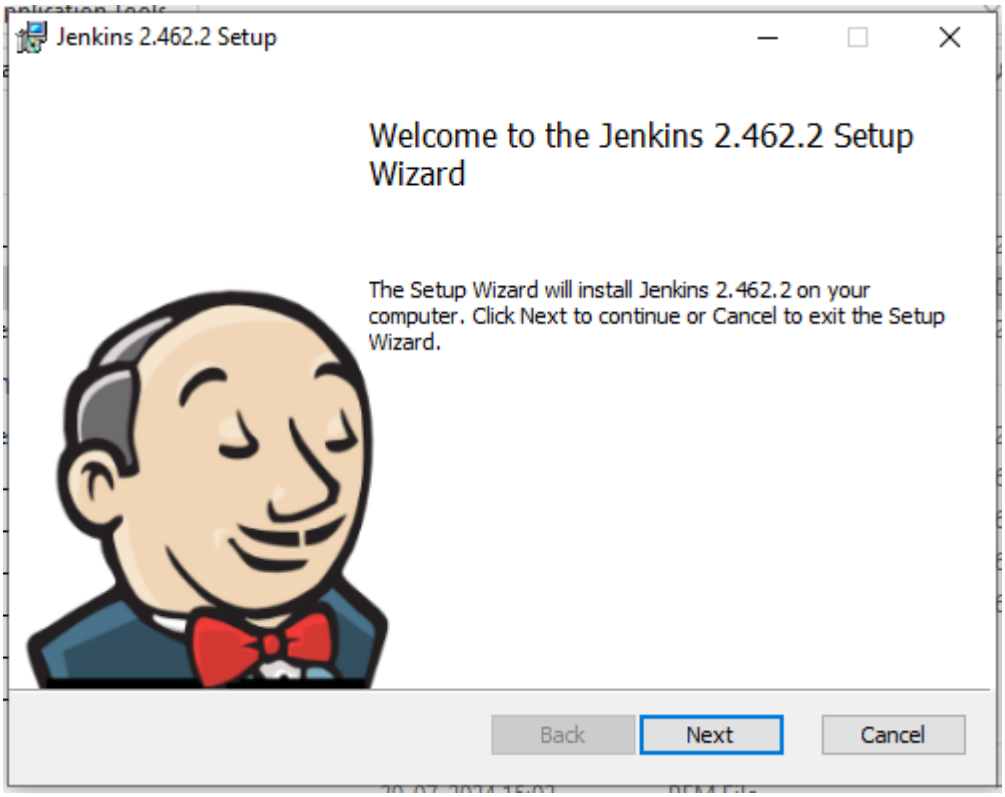


We can ensure whether docker is downloaded successfully or not by command “docker –version”

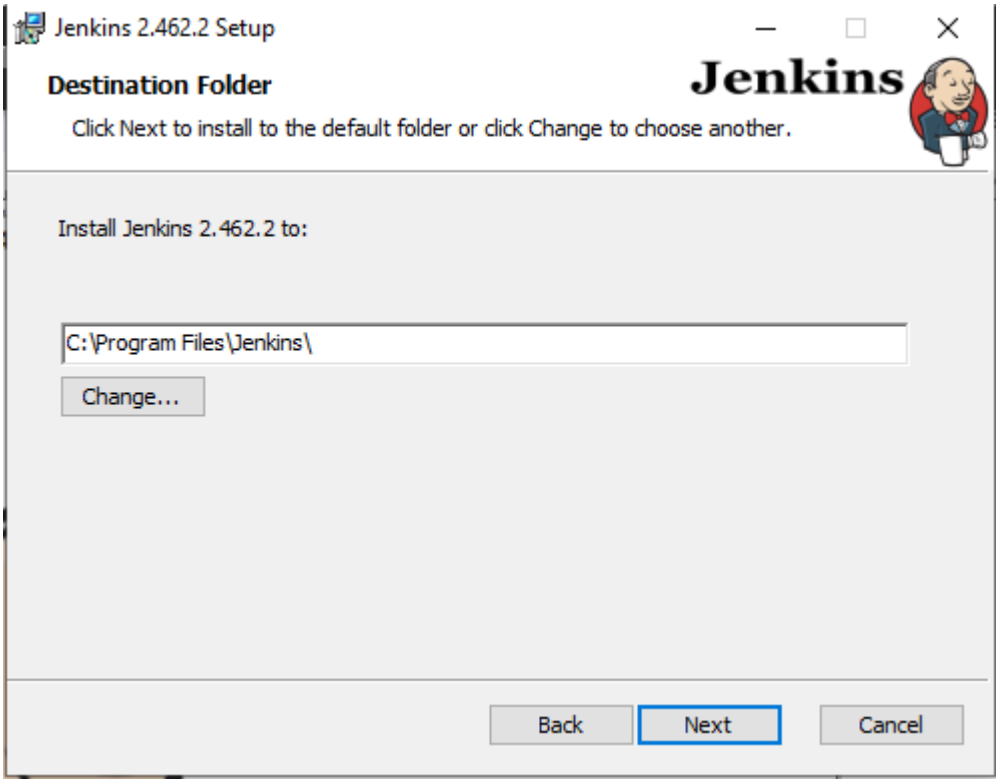


- **Jenkins Installation**

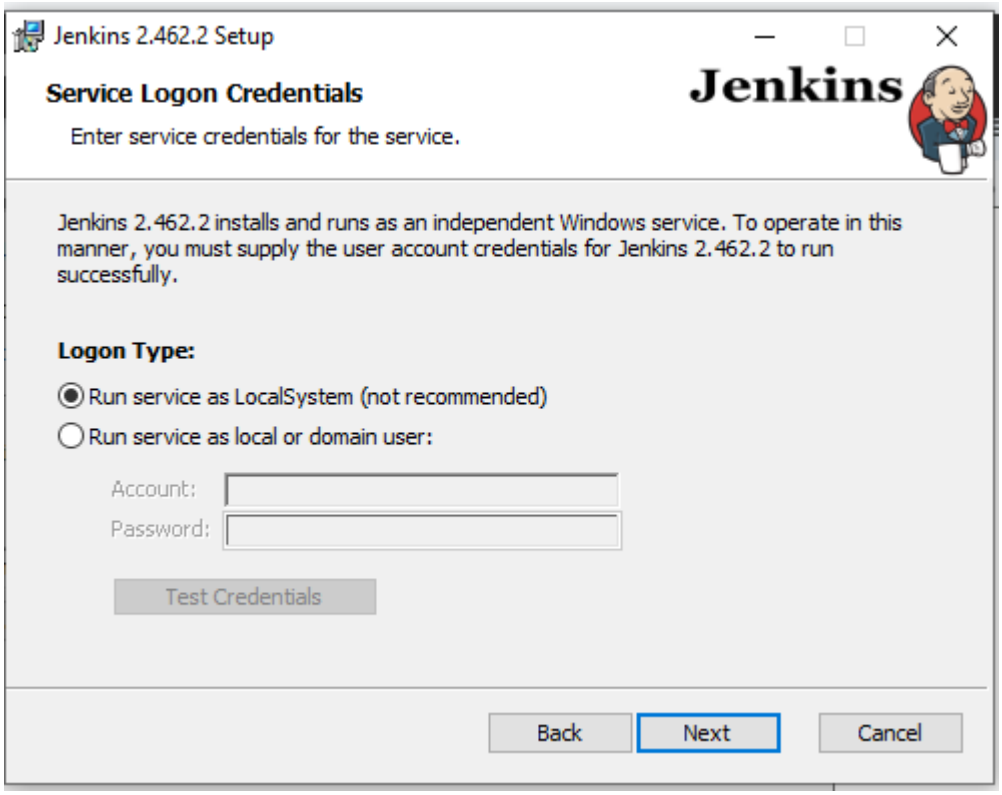
Launch the Jenkins exe file downloaded



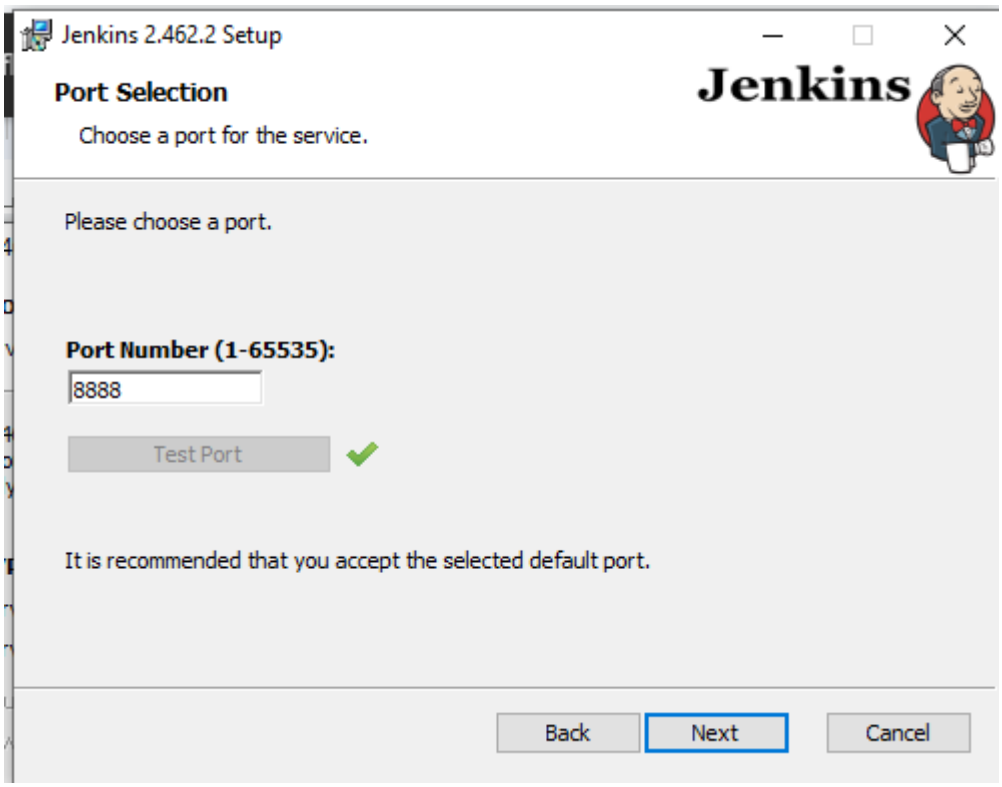
Set the path for Jenkins all workspaces and jobs



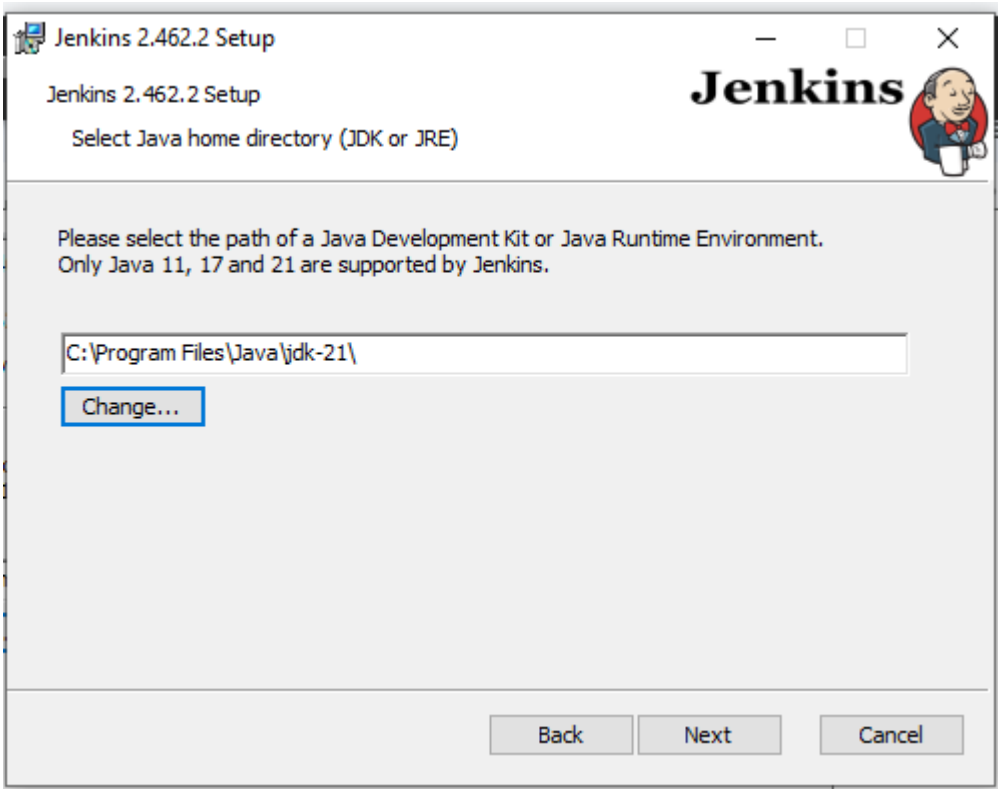
Select “Run service as a LocalSystem” to start Jenkins locally



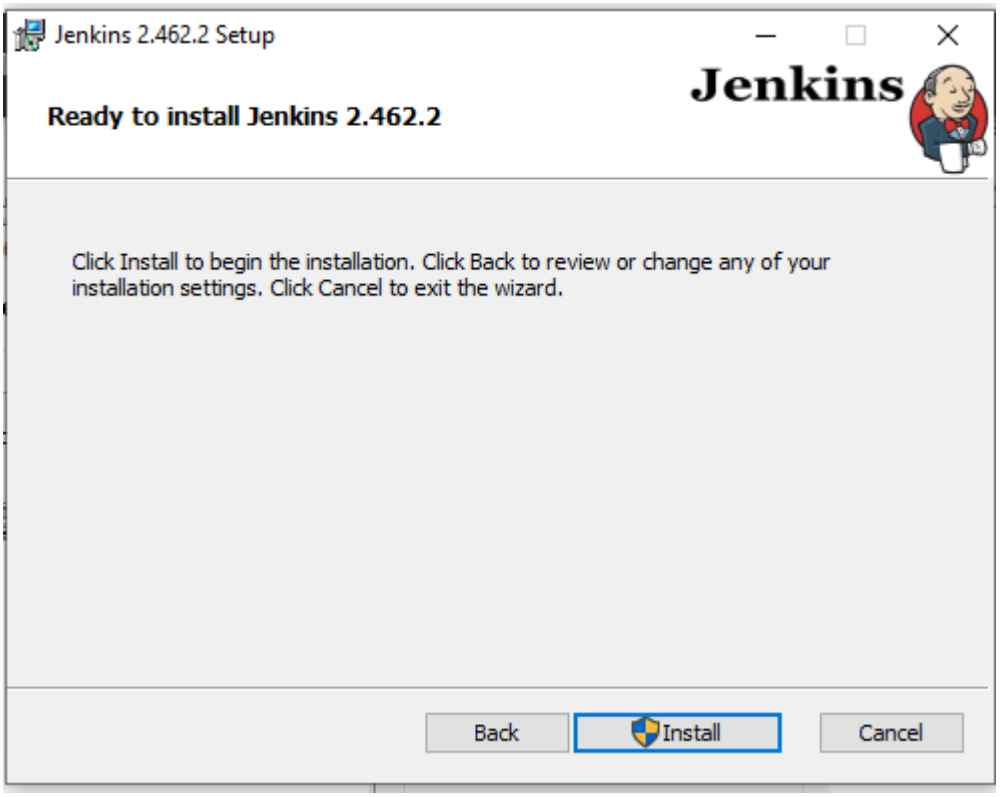
Choose a port on which you want Jenkins to work.(By default it is 8080)

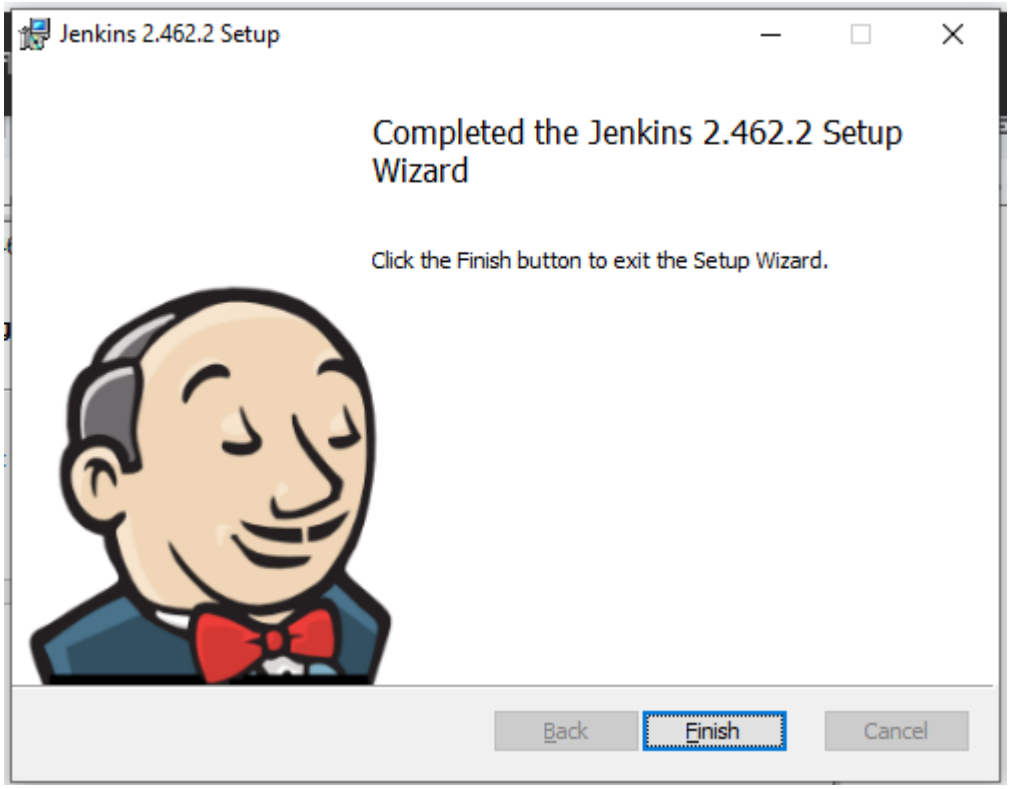


Set up the latest Java directory

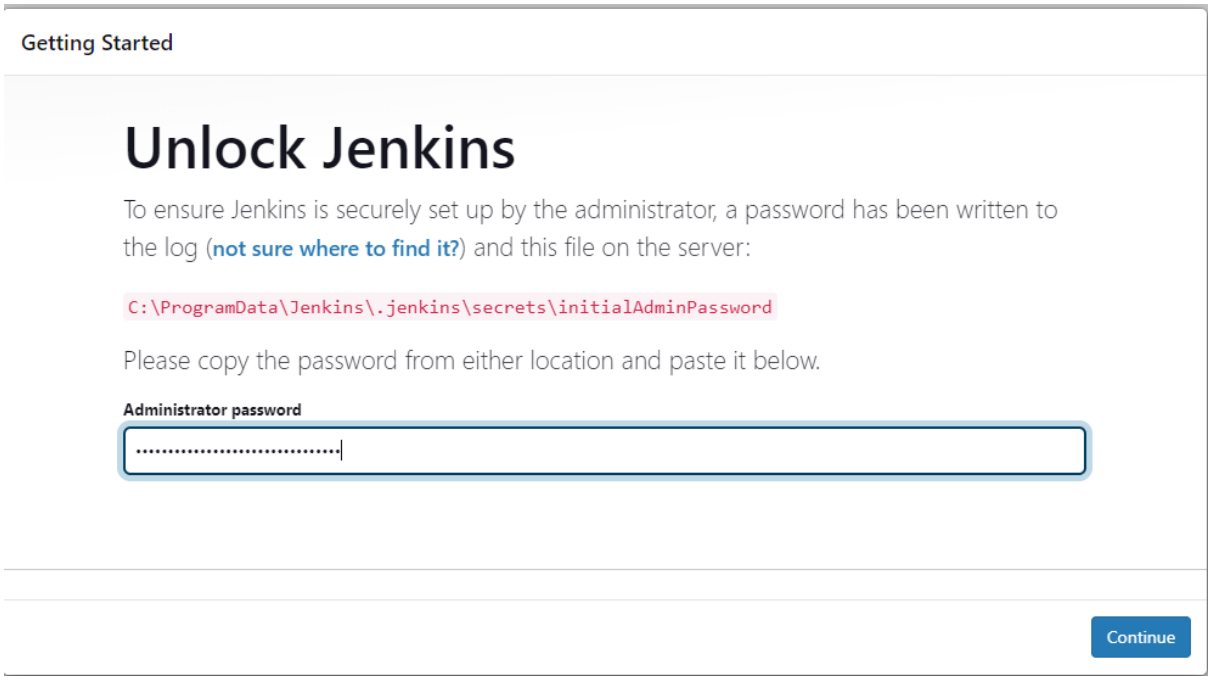


Click Install to begin the installation process

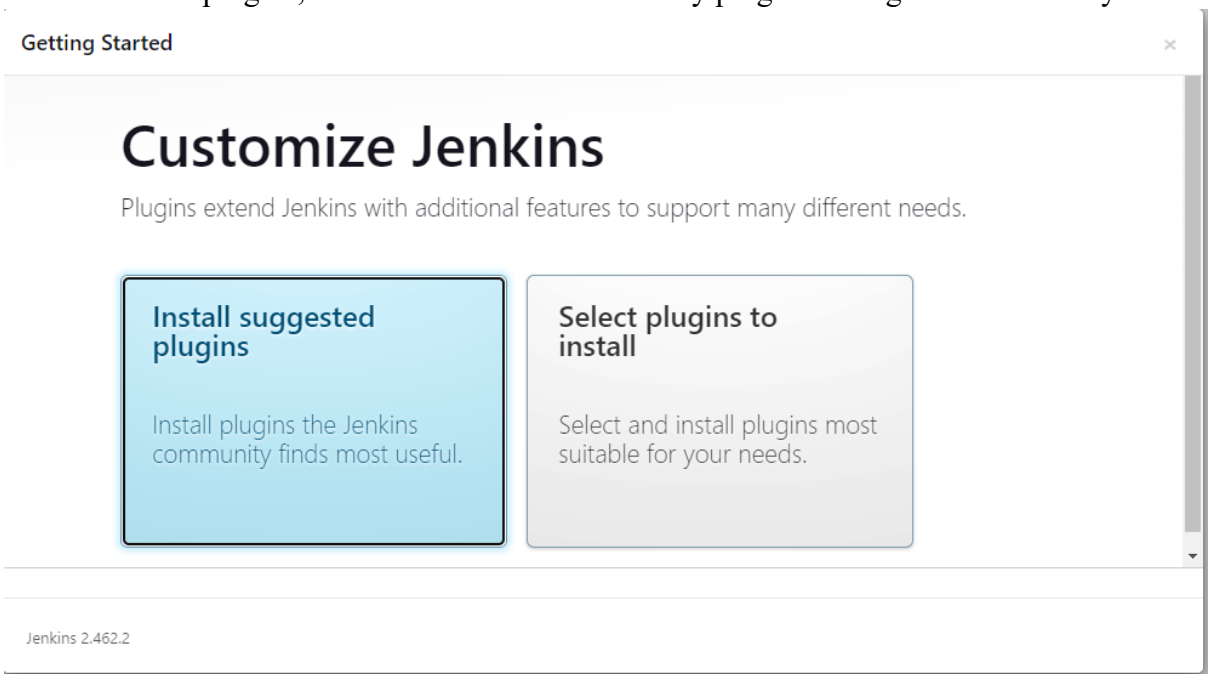




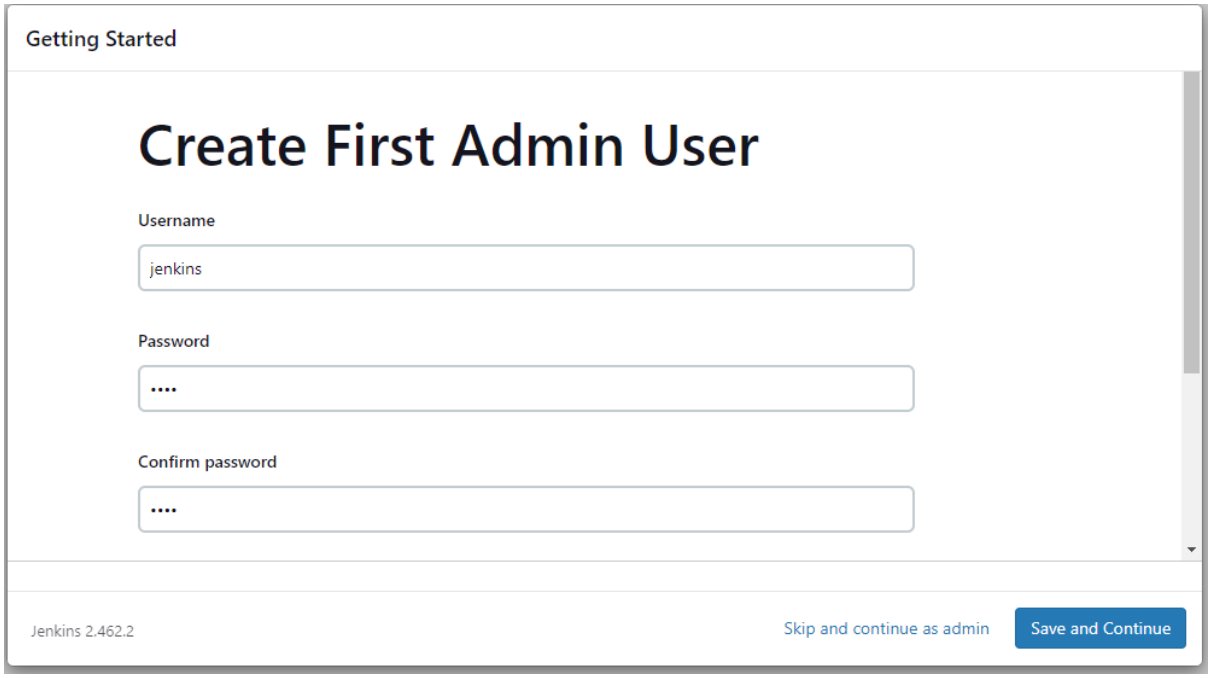
Once it is installed, go to http://localhost:8888 , where we first need to setup the Jenkins

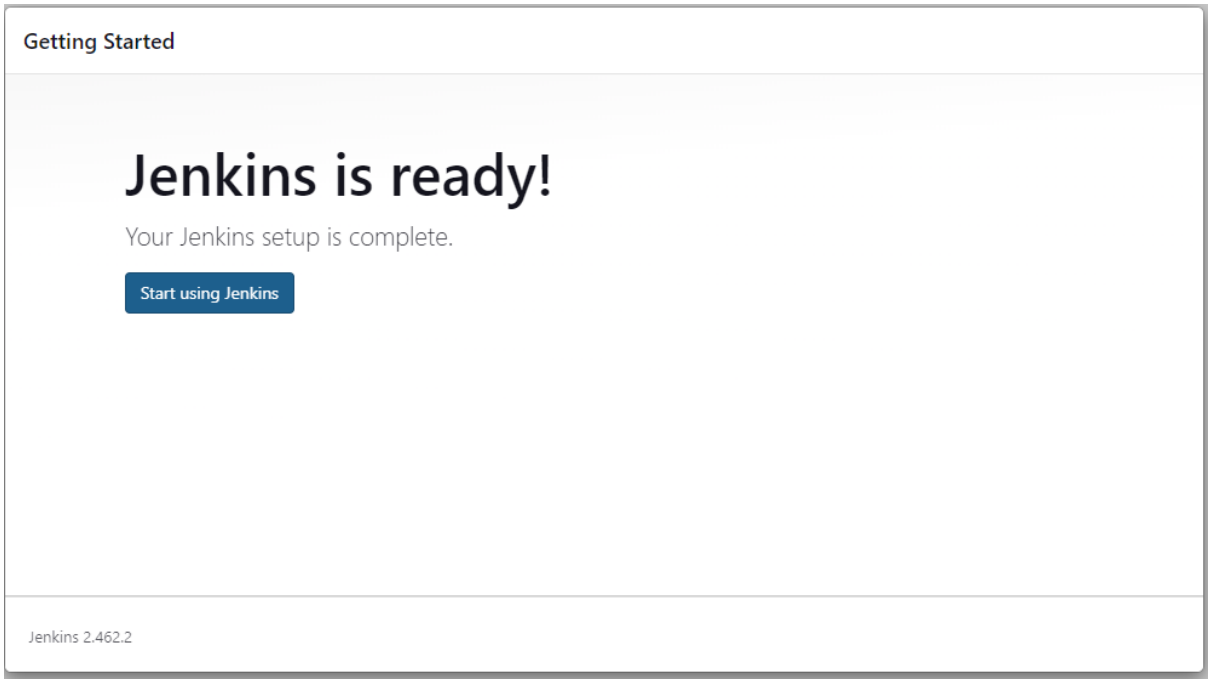


Install selected plugins, which will install all necessary plugins during installation only

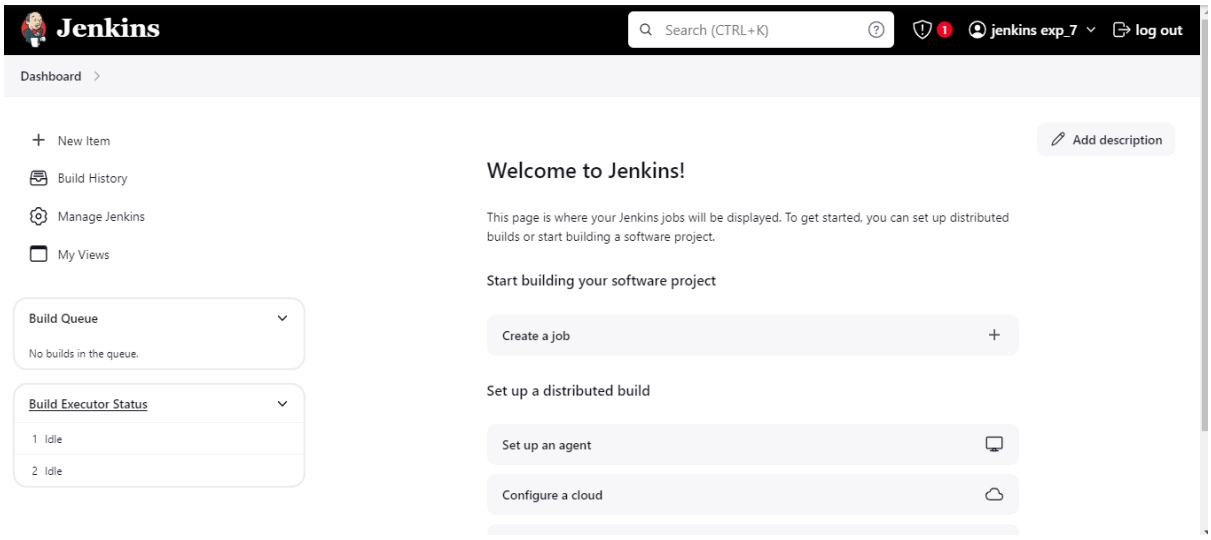


Set up the configuration details





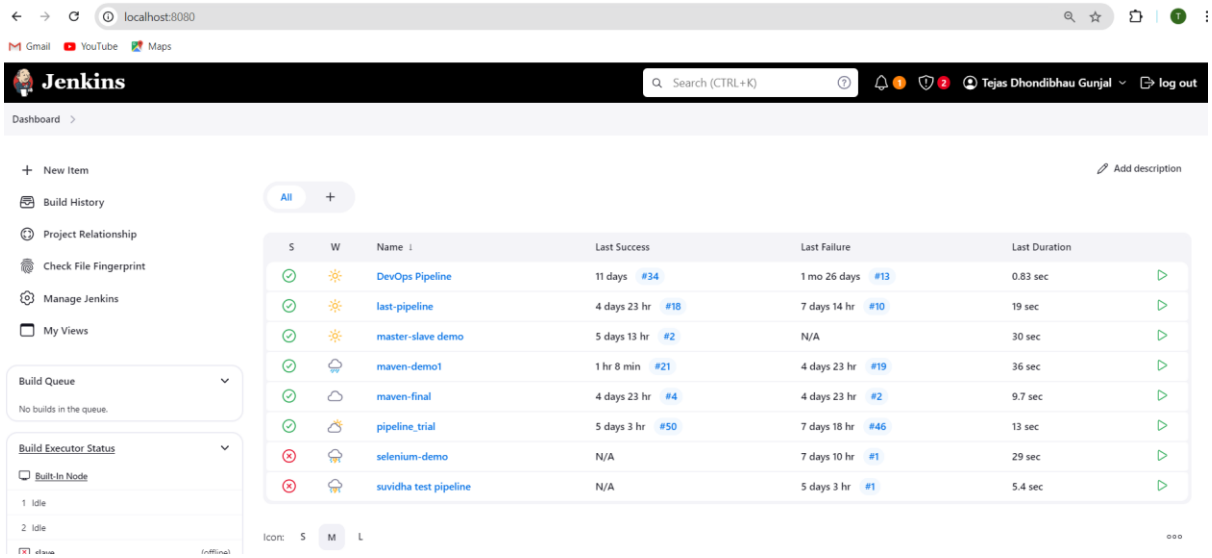
Once, all the setup is ready, you can login with your credentials



EXPERIMENT NO: - 08

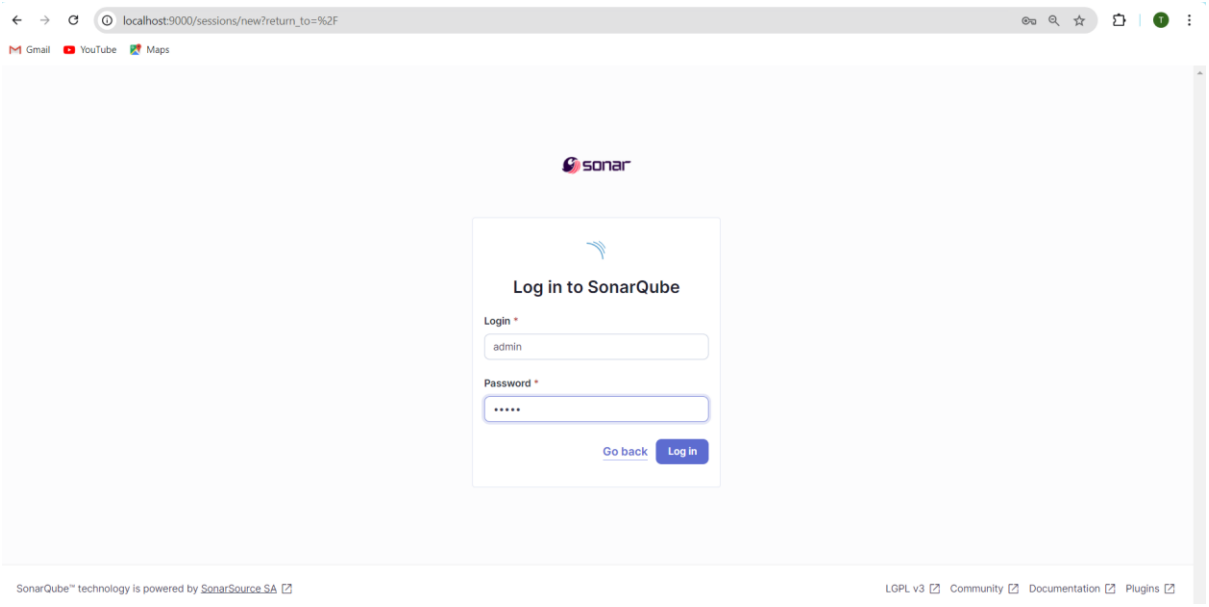
AIM: - Create a Jenkins CICD Pipeline with SonarQube / GitLab Integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web / Java / Python application.

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

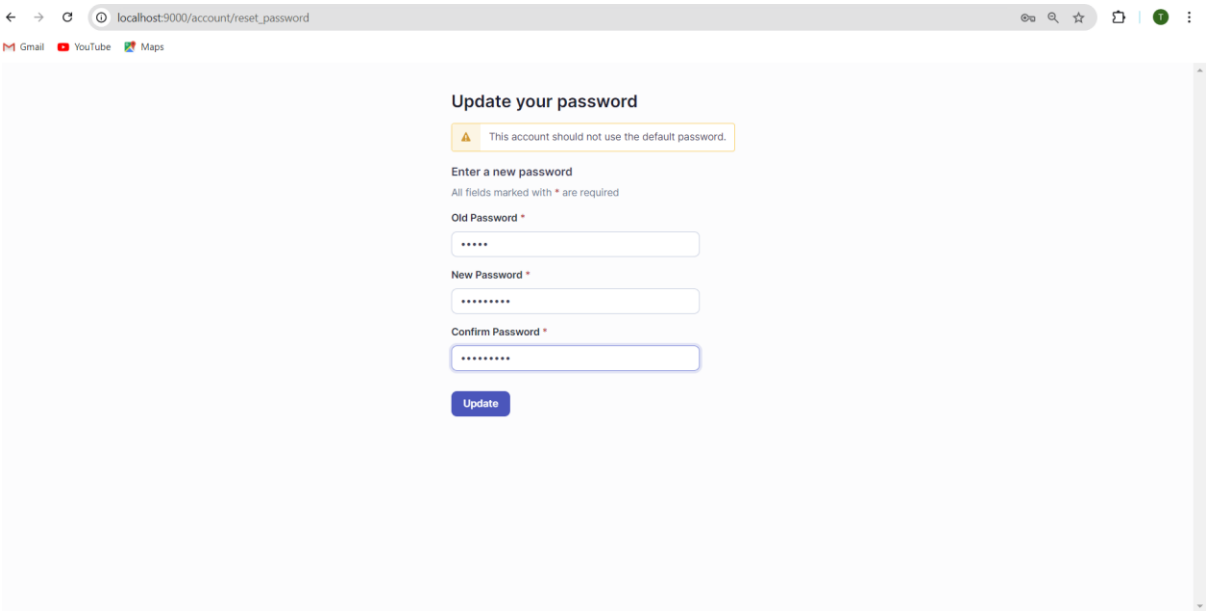


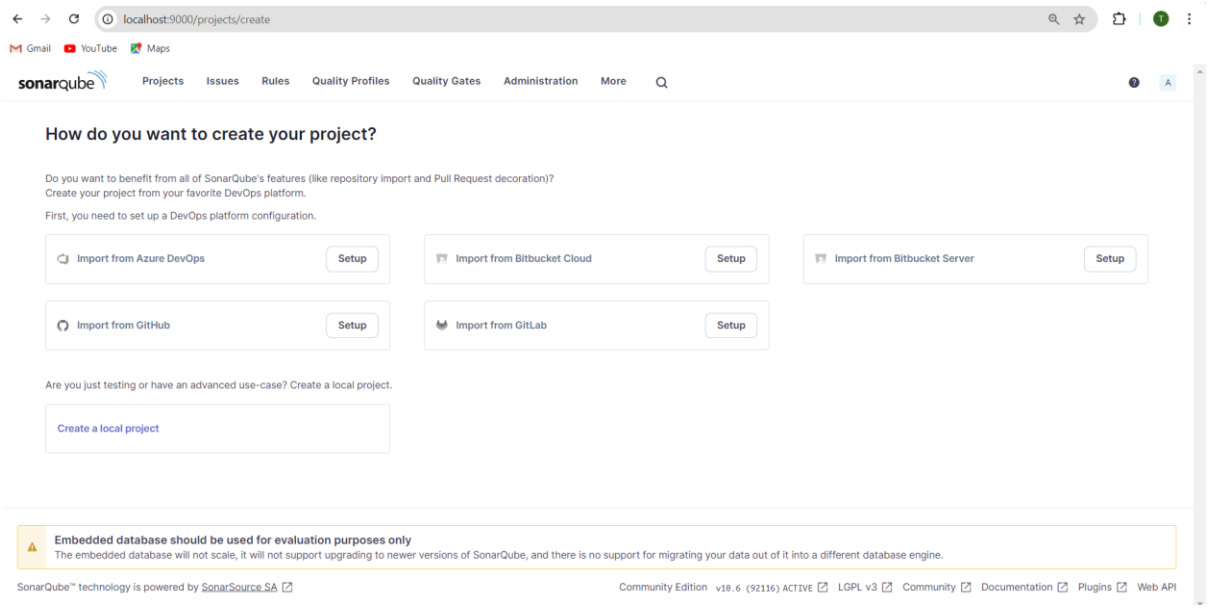
- Run SonarQube in a Docker container using this command –
  docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest

- Once the container is up and running, you can check the status of SonarQube at localhost port 9000. Login to SonarQube using username admin and password admin.

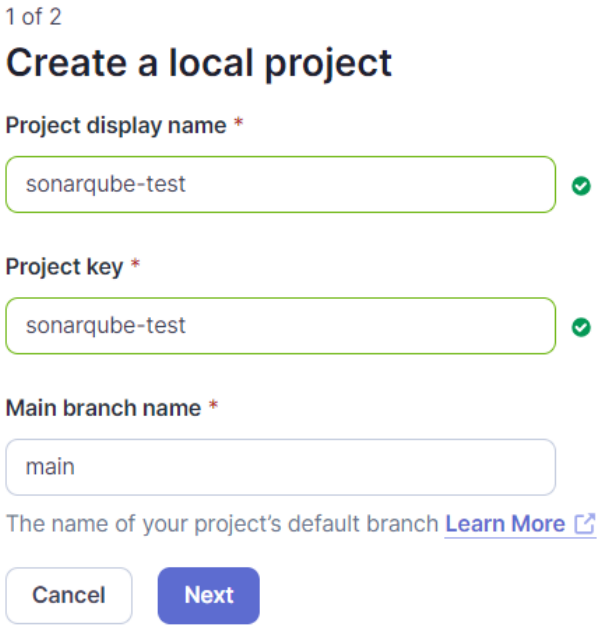


- It also gives an option to update the password, for the security purposes

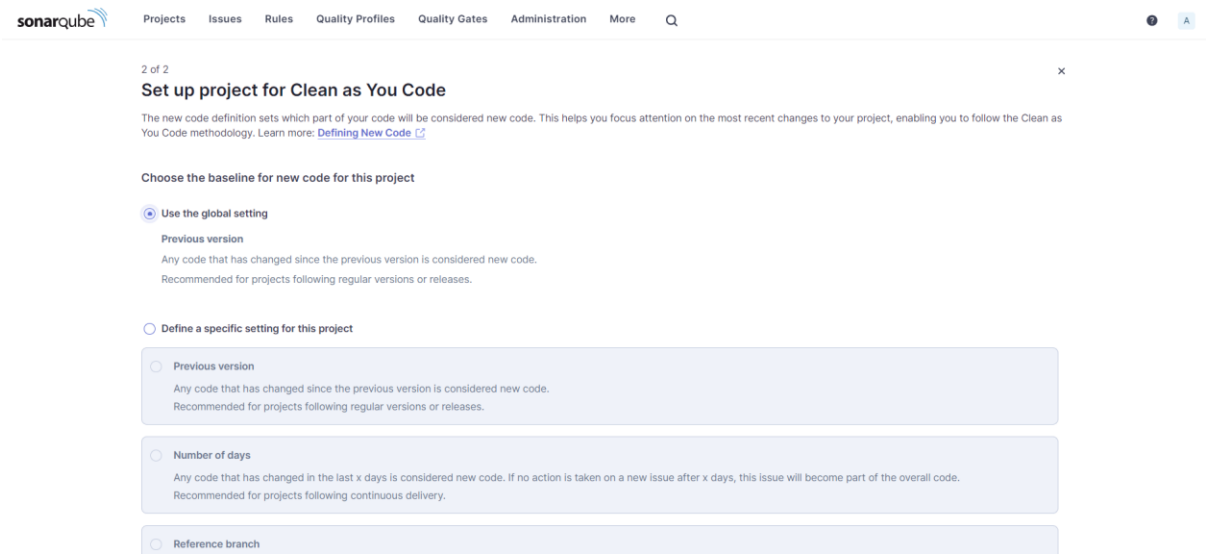




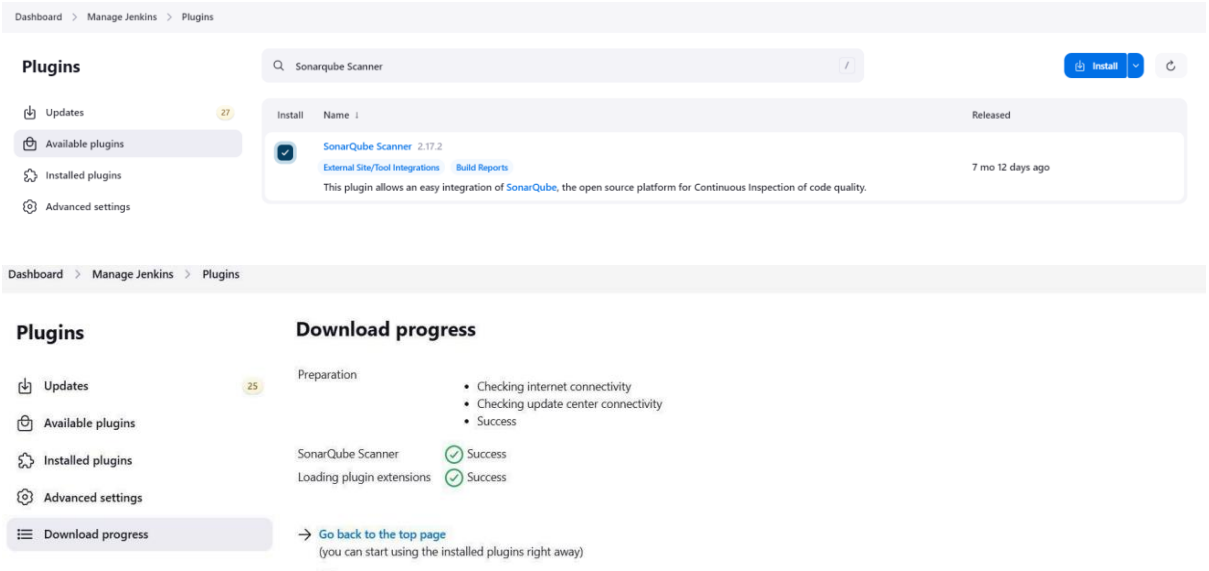
➢ Create a manual project in SonarQube with the name sonarqube-test



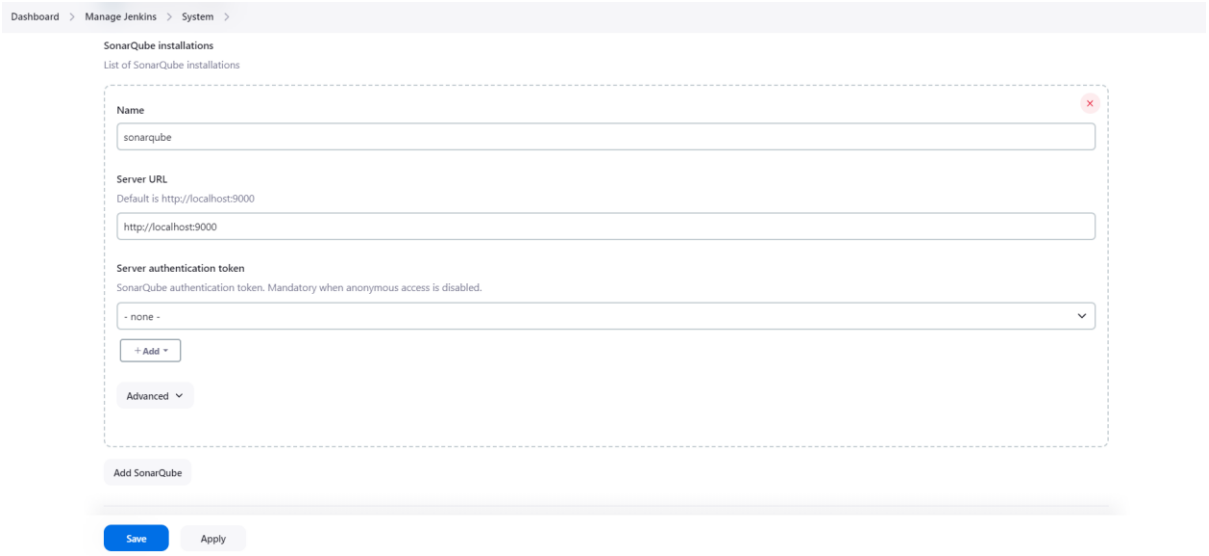
- Setup the project and come back to Jenkins Dashboard.



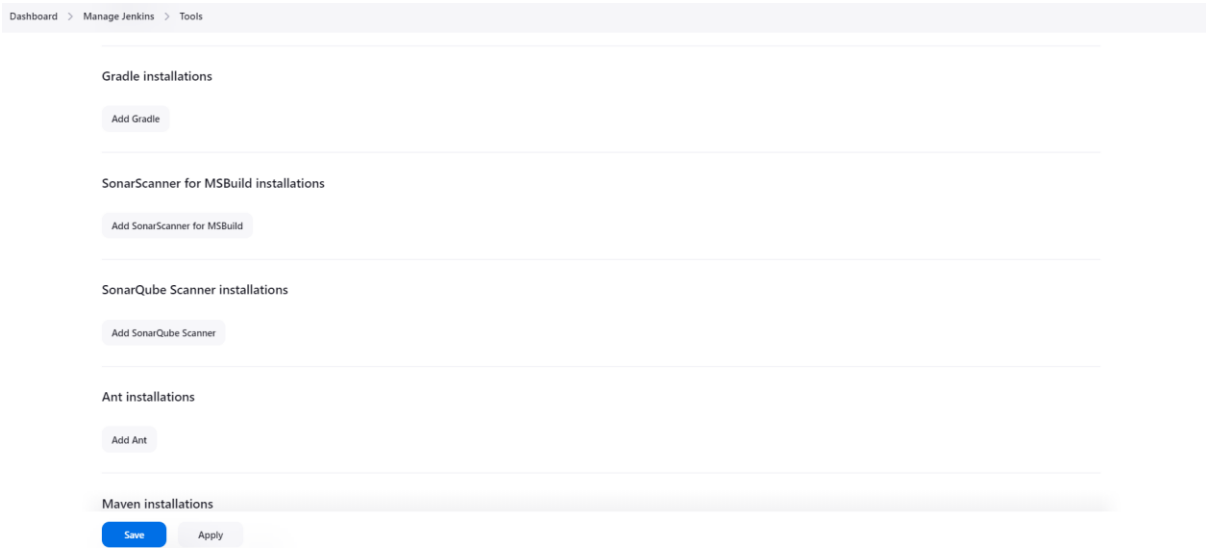
- Go to Manage Jenkins and search for SonarQube Scanner for Jenkins and install it.



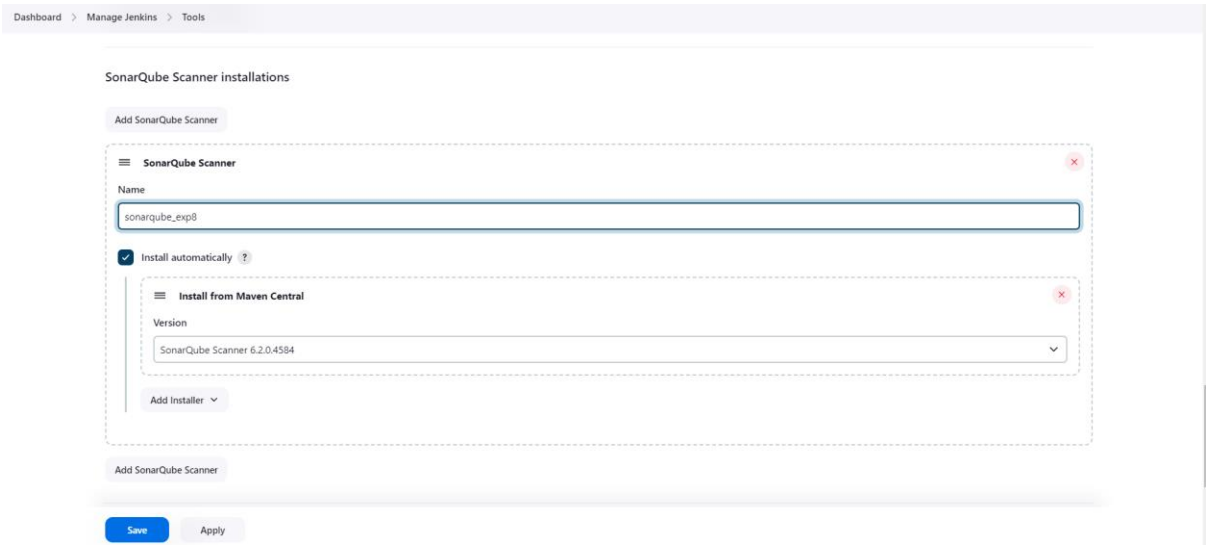
- Under Jenkins ‘Manage Jenkins’ then go to ‘system’, scroll and look for SonarQube Servers and enter the details. Enter the Server Authentication token if needed.



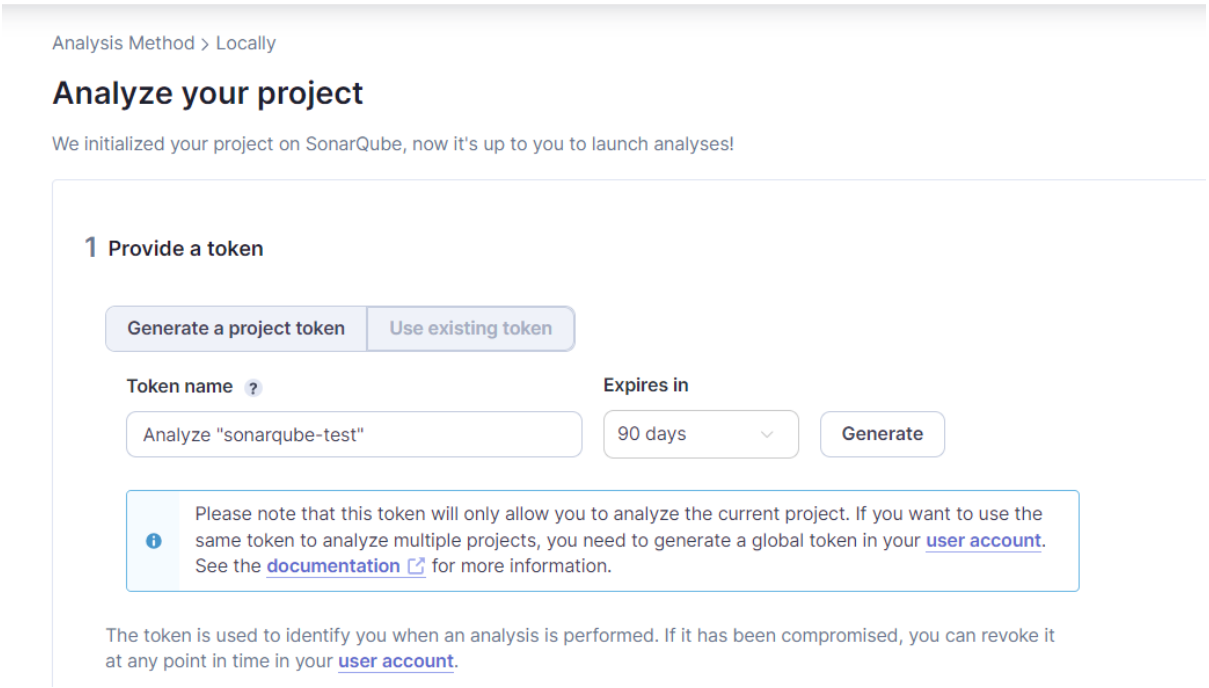
- Search for SonarQube Scanner under Global Tool Configuration. Choose the latest configuration and choose Install automatically. Dashboard > Manage Jenkins > Tools



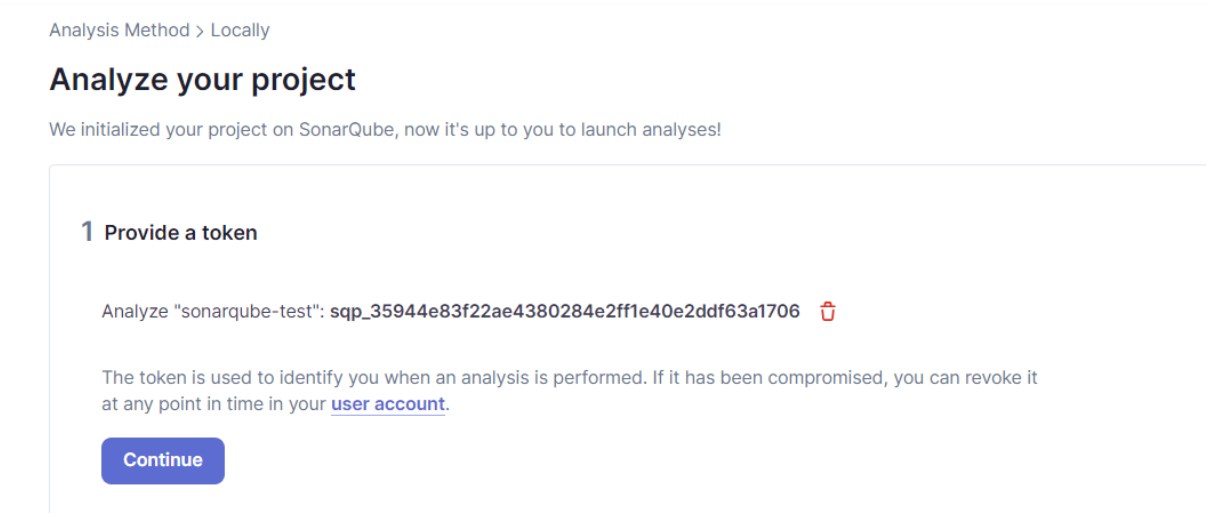
- Check the “Install automatically” option. → Under name any name as identifier → Check the “Install automatically” option.



- Generating a token on sonarqubes for authentication [optional]

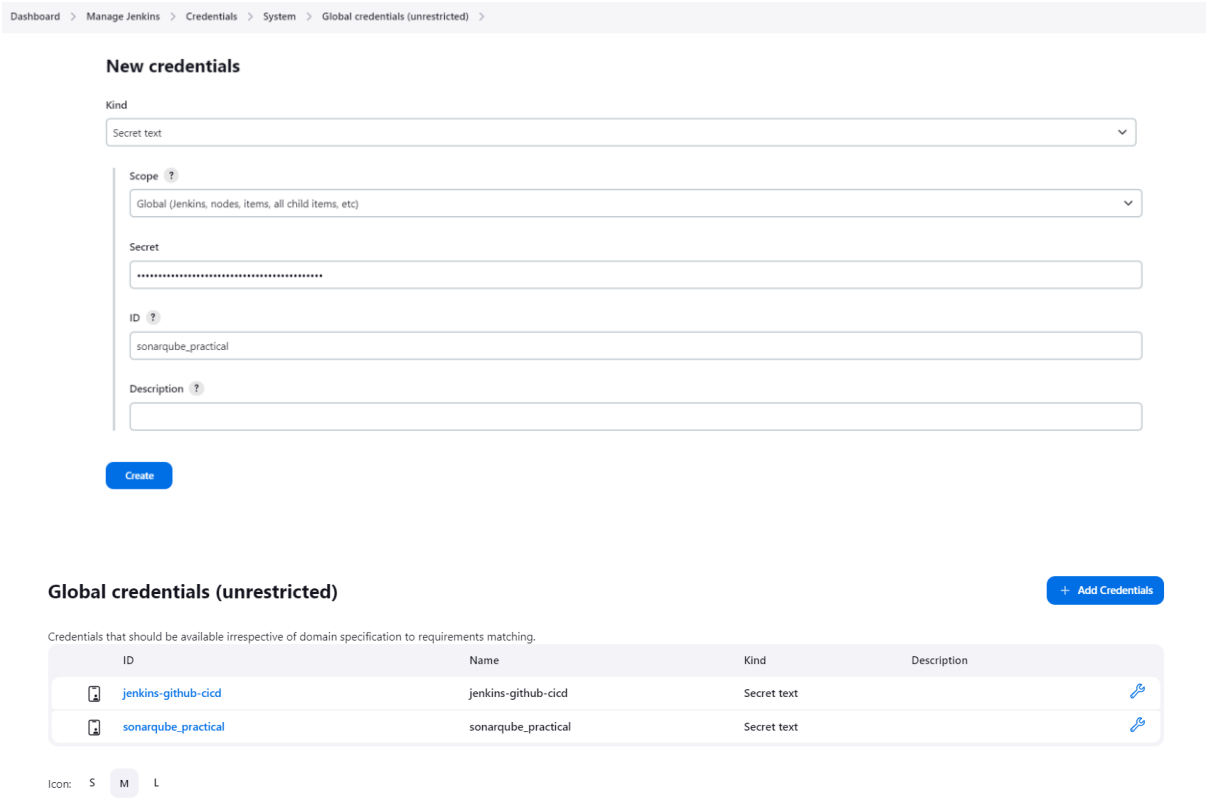


- Copy the token you get here and save it securely as we would need it in Jenkins.



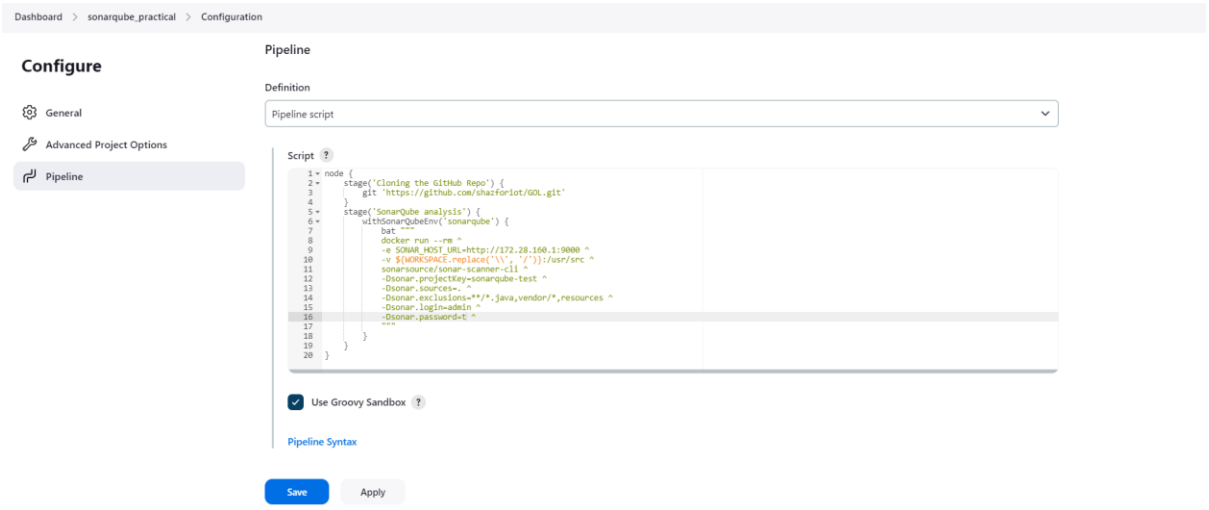
- Update the Token in Jenkins

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

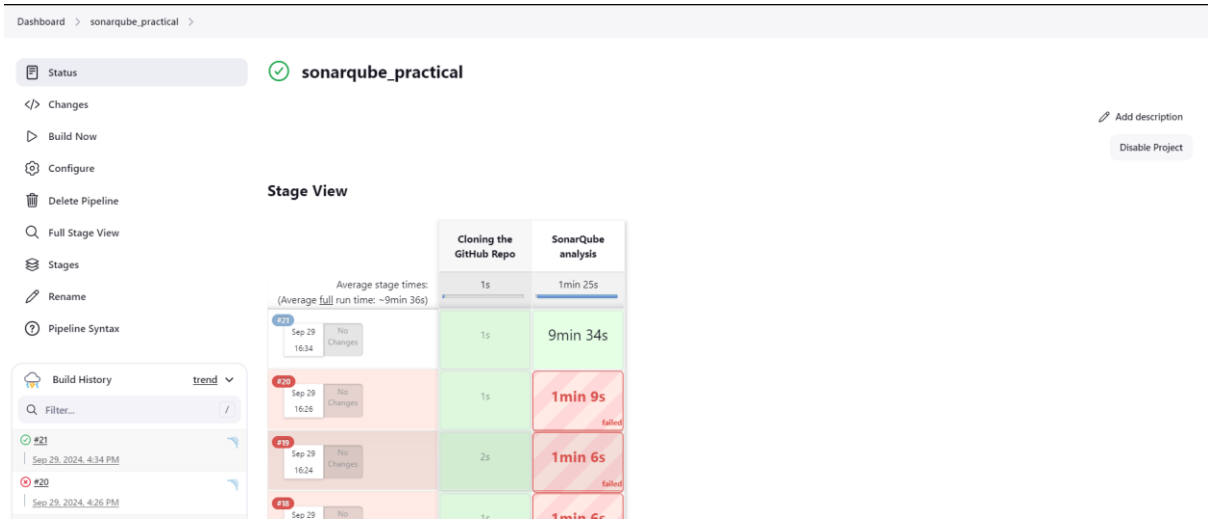


- Now go to Manage Jenkins—>System—>SonarQube servers and proceed with the following details. Reference the authentication token generated in the previous step.

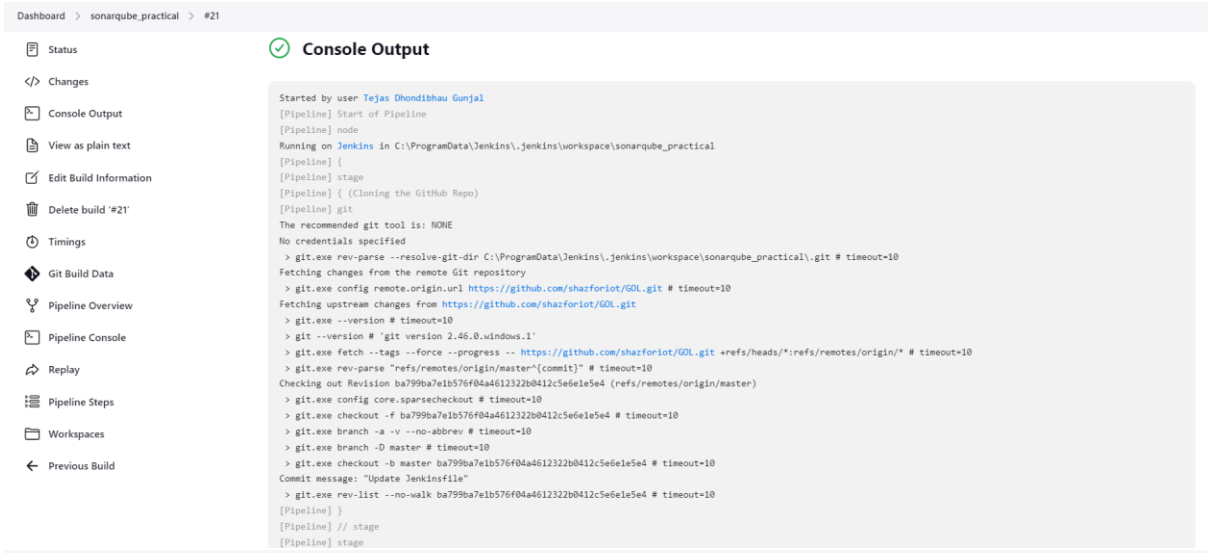
- After configuration, create a New Item → choose a pipeline project
  Under Pipeline script, enter the following:
  It is a java sample project which has a lot of repetitions and issues that will be detected by SonarQube.

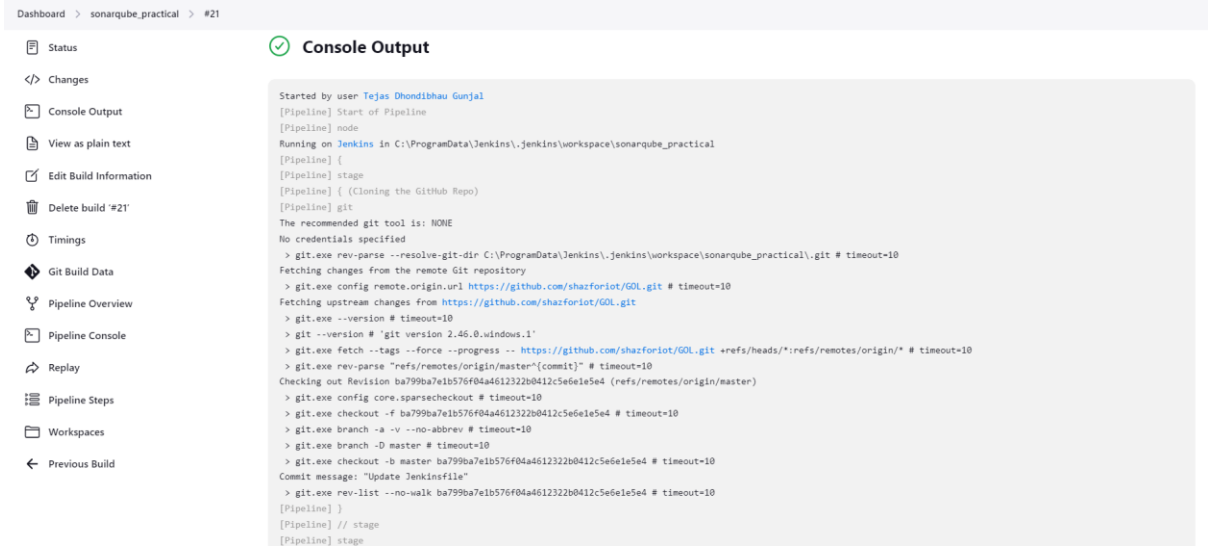


- Build project

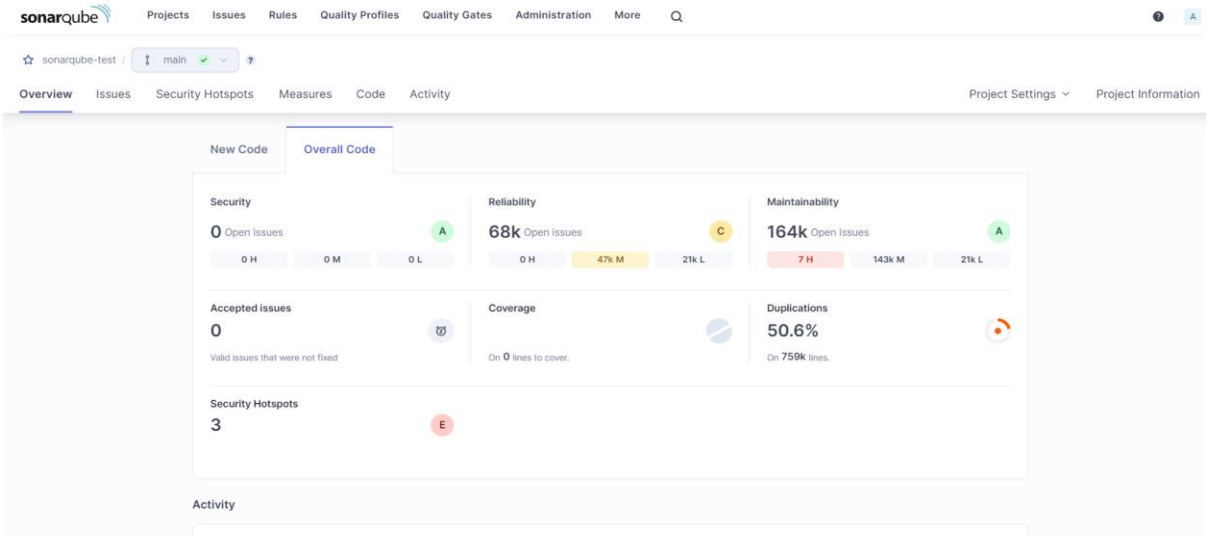


- Check Console



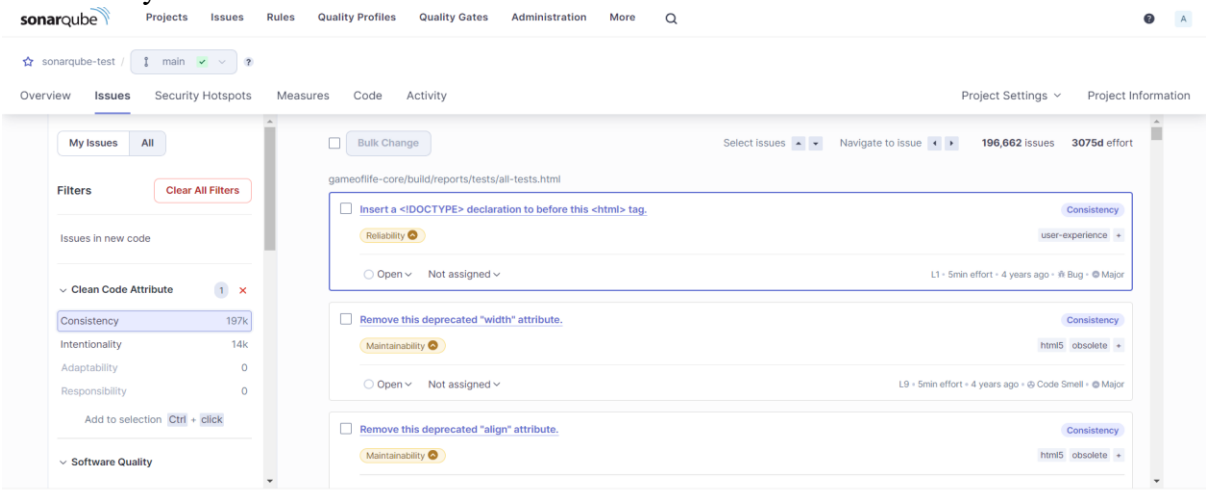


- Now, check the project in SonarQube

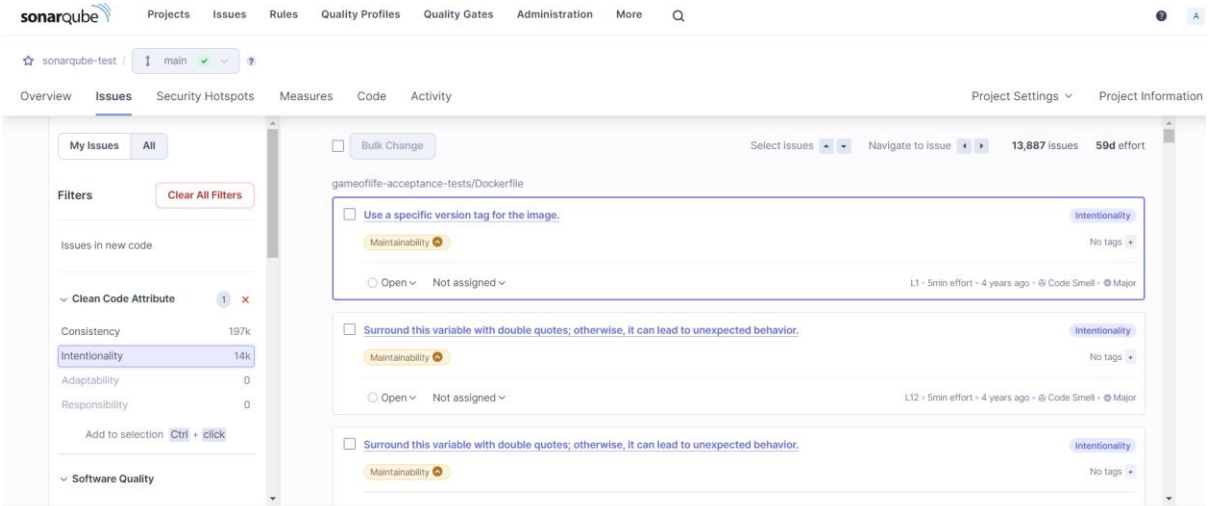


- Code Problems

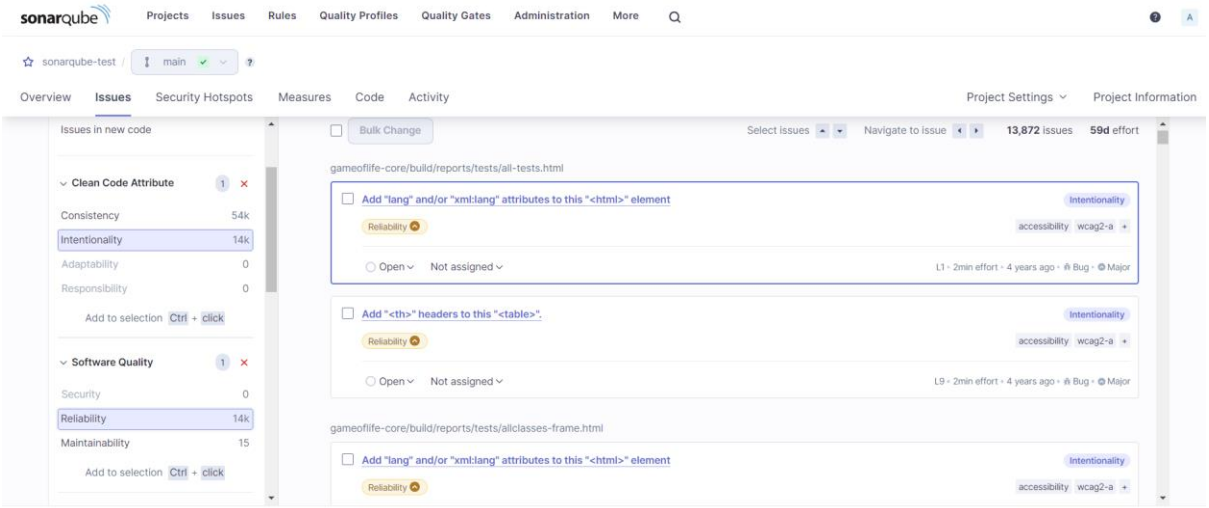
Consistency



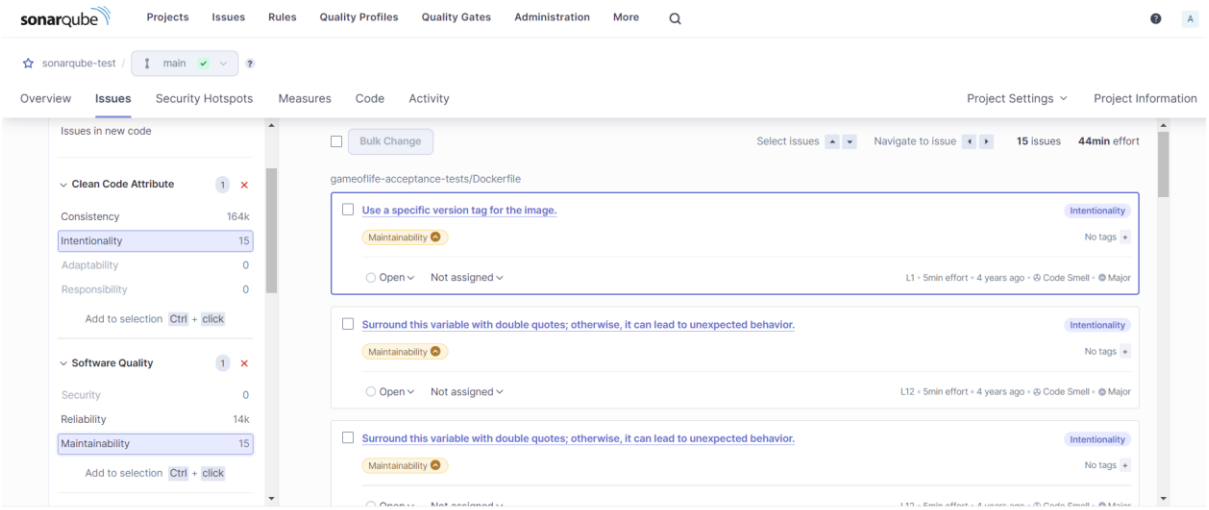
Intentionality



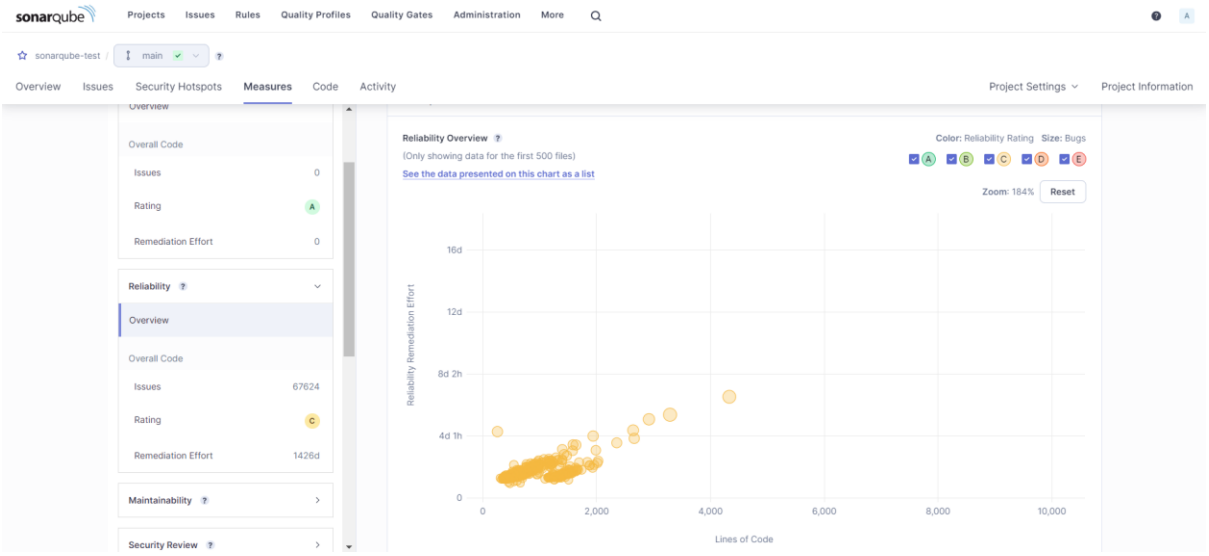
Bugs



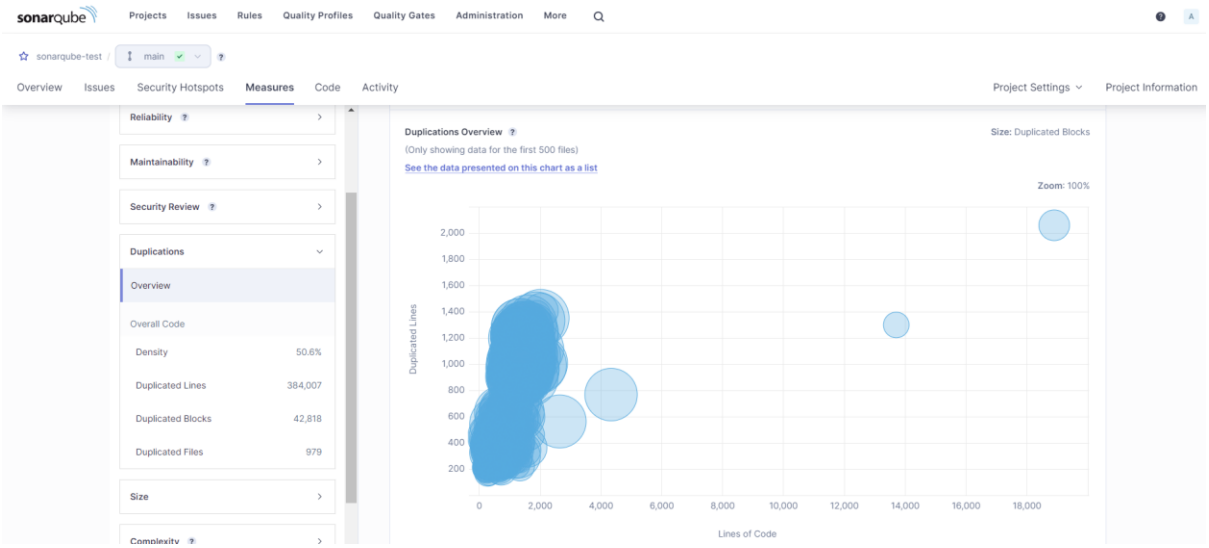
Code Smells



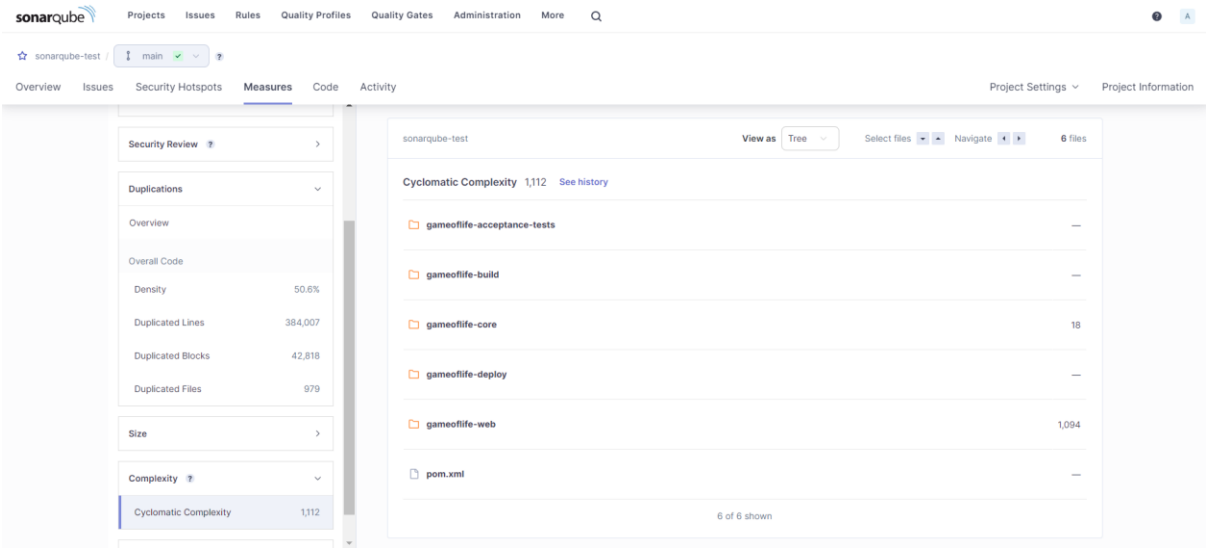
Reliability



Duplications



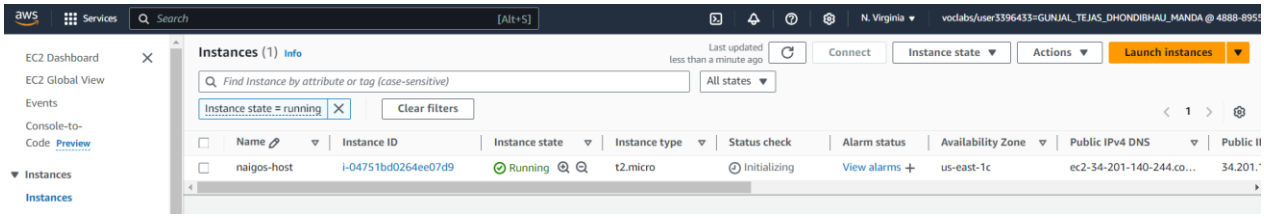
Cyclomatic Complexities



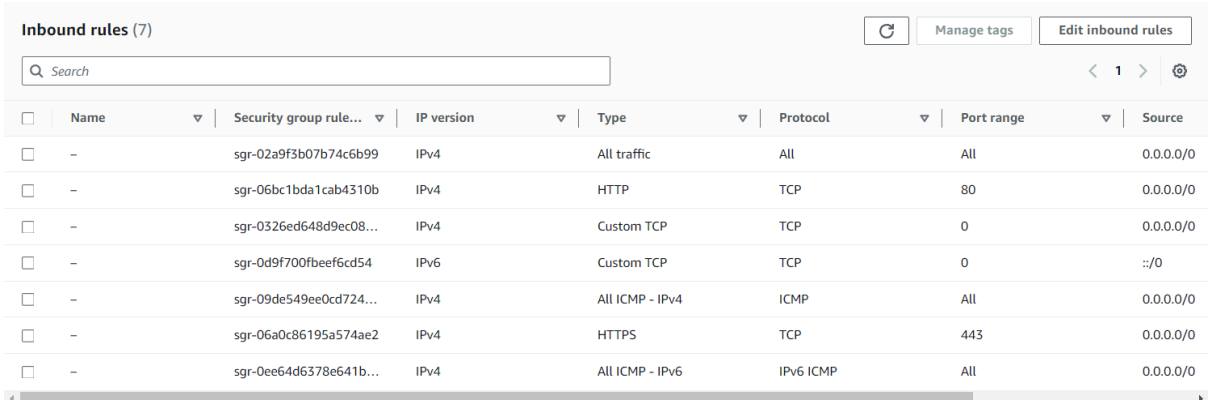
EXPERIMENT NO :- 09

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

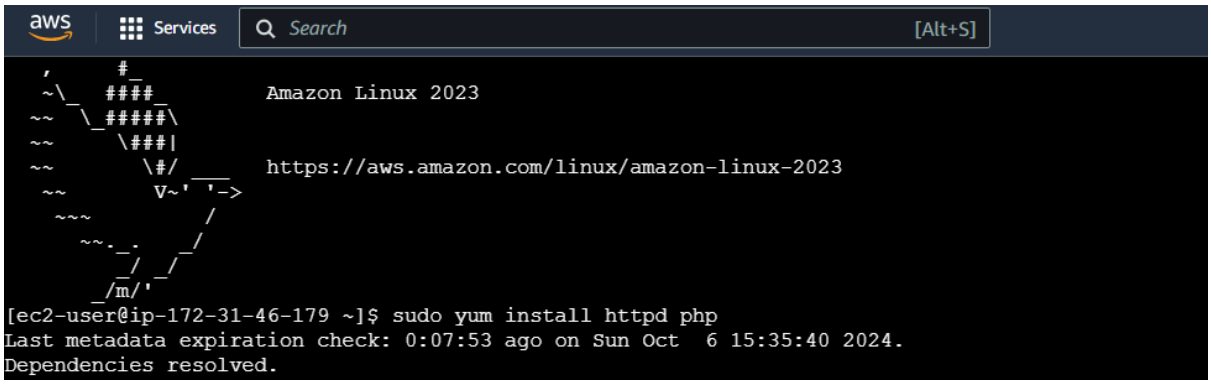
- Create an Amazon Linux EC2 instance and name it as nagios-host



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



- Connect to your EC2 instance via the connect option available in EC2 instances menu. After that command prompt will be started.



- Update and install the required packages

sudo yum install httpd php

```
[ec2-user@ip-172-31-46-179 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:07:53 ago on Sun Oct  6 15:35:40 2024.
Dependencies resolved.
================================================================================
 Package                 Architecture          Version                    Repository                S
================================================================================
Installing:
 httpd                   x86_64                2.4.62-1.amzn2023          amazonlinux               4
 php8.3                  x86_64                8.3.10-1.amzn2023.0.1      amazonlinux               1
Installing dependencies:
 apr                     x86_64                1.7.2-2.amzn2023.0.2       amazonlinux              12
 apr-util                x86_64                1.6.3-1.amzn2023.0.1       amazonlinux               9
 generic-logos-httpd     noarch                18.0.0-12.amzn2023.0.3     amazonlinux               1
 httpd-core              x86_64                2.4.62-1.amzn2023          amazonlinux              1.
 httpd-filesystem        noarch                2.4.62-1.amzn2023          amazonlinux               1
 httpd-tools             x86_64                2.4.62-1.amzn2023          amazonlinux               8
 libbrotli               x86_64                1.0.9-4.amzn2023.0.2       amazonlinux              31
 libsodium               x86_64                1.0.19-4.amzn2023          amazonlinux              17
 libxslt                 x86_64                1.1.34-5.amzn2023.0.2      amazonlinux              24
 mailcap                 noarch                2.1.49-3.amzn2023.0.3      amazonlinux               3
 nginx-filesystem        noarch                1:1.24.0-1.amzn2023.0.4    amazonlinux              9.
 php8.3-cli              x86_64                8.3.10-1.amzn2023.0.1      amazonlinux              3.
 php8.3-common           x86_64                8.3.10-1.amzn2023.0.1      amazonlinux              73
 php8.3-process          x86_64                8.3.10-1.amzn2023.0.1      amazonlinux               4
 php8.3-xml              x86_64                8.3.10-1.amzn2023.0.1      amazonlinux              15
Installing weak dependencies:
 apr-util-openssl        x86_64                1.6.3-1.amzn2023.0.1       amazonlinux               1
 mod_http2               x86_64                2.0.27-1.amzn2023.0.3      amazonlinux              16
 mod_lua                 x86_64                2.4.62-1.amzn2023          amazonlinux               6
 php8.3-fpm              x86_64                8.3.10-1.amzn2023.0.1      amazonlinux              1.
```

sudo yum install gcc glibc glibc-common

```
[ec2-user@ip-172-31-46-179 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:09:46 ago on Sun Oct  6 15:35:40 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                S
================================================================================
Installing:
 gcc                     x86_64                11.4.1-2.amzn2023.0.2      amazonlinux               3
Installing dependencies:
 annobin-docs            noarch                10.93-1.amzn2023.0.1       amazonlinux               9
 annobin-plugin-gcc      x86_64                10.93-1.amzn2023.0.1       amazonlinux              88
 cpp                     x86_64                11.4.1-2.amzn2023.0.2      amazonlinux               1
 gc                      x86_64                8.0.4-5.amzn2023.0.2       amazonlinux              10
 glibc-devel             x86_64                2.34-52.amzn2023.0.11      amazonlinux               2
 glibc-headers-x86       noarch                2.34-52.amzn2023.0.11      amazonlinux              42
 guile22                 x86_64                2.2.7-2.amzn2023.0.3       amazonlinux              6.
 kernel-headers          x86_64                6.1.109-118.189.amzn2023   amazonlinux              1.
 libmpc                  x86_64                1.2.1-2.amzn2023.0.2       amazonlinux               6
 libtool-ltdl            x86_64                2.4.7-1.amzn2023.0.3       amazonlinux               3
 libxcrypt-devel         x86_64                4.4.33-7.amzn2023          amazonlinux               3
 make                    x86_64                1:4.3-5.amzn2023.0.2       amazonlinux              53

Transaction Summary
================================================================================
Install  13 Packages

Total download size: 52 M
Installed size: 160 M
Is this ok [y/N]: y
```

sudo yum install gd gd-devel

```
[ec2-user@ip-172-31-46-179 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:10:57 ago on Sun Oct  6 15:35:40 2024.
Dependencies resolved.
================================================================================
 Package                 Architecture          Version                    Repository                S
================================================================================
Installing:
 gd                      x86_64                2.3.3-5.amzn2023.0.3       amazonlinux              13
 gd-devel                x86_64                2.3.3-5.amzn2023.0.3       amazonlinux               3
Installing dependencies:
 brotli                  x86_64                1.0.9-4.amzn2023.0.2       amazonlinux              31
 brotli-devel            x86_64                1.0.9-4.amzn2023.0.2       amazonlinux               3
 bzip2-devel             x86_64                1.0.8-6.amzn2023.0.2       amazonlinux              21
 cairo                   x86_64                1.17.6-2.amzn2023.0.1      amazonlinux              68
 cmake-filesystem        x86_64                3.22.2-1.amzn2023.0.4      amazonlinux               1
 fontconfig              x86_64                2.13.94-2.amzn2023.0.2     amazonlinux              27
 fontconfig-devel        x86_64                2.13.94-2.amzn2023.0.2     amazonlinux              12
 fonts-filesystem        noarch                1:2.0.5-12.amzn2023.0.2    amazonlinux              9.
 freetype                x86_64                2.13.2-5.amzn2023.0.1      amazonlinux              42
 freetype-devel          x86_64                2.13.2-5.amzn2023.0.1      amazonlinux              91
 glib2-devel             x86_64                2.74.7-689.amzn2023.0.2    amazonlinux              48
 google-noto-fonts-common noarch               20201206-2.amzn2023.0.2    amazonlinux               1
 google-noto-sans-vf-fonts noarch              20201206-2.amzn2023.0.2    amazonlinux              49
 graphite2               x86_64                1.3.14-7.amzn2023.0.2      amazonlinux               9
 graphite2-devel         x86_64                1.3.14-7.amzn2023.0.2      amazonlinux               2
 harfbuzz                x86_64                7.0.0-2.amzn2023.0.1       amazonlinux              86
 harfbuzz-devel          x86_64                7.0.0-2.amzn2023.0.1       amazonlinux              40
 harfbuzz-icu            x86_64                7.0.0-2.amzn2023.0.1       amazonlinux               1
 jbigkit-libs            x86_64                2.1-21.amzn2023.0.2        amazonlinux               5
 langpacks-core-font-en  noarch                3.0-21.amzn2023.0.4        amazonlinux               1
 libICE                  x86_64                1.0.10-6.amzn2023.0.2      amazonlinux               7
```

```
  Verifying       : sysprof-capture-devel-3.40.1-2.amzn2023.0.2.x86_64                     58
  Verifying       : xml-common-0.6.3-56.amzn2023.0.2.noarch                                59
  Verifying       : xorg-x11-proto-devel-2021.4-1.amzn2023.0.2.noarch                      60
  Verifying       : xz-devel-5.2.5-9.amzn2023.0.2.x86_64                                   61
  Verifying       : zlib-devel-1.2.11-33.amzn2023.0.5.x86_64                               62

Installed:
  brotli-1.0.9-4.amzn2023.0.2.x86_64                    brotli-devel-1.0.9-4.amzn2023.0.2.x86_64                      bzip2-devel-1.0.8-6.amzn2023.0.2.x86_64
  cairo-1.17.6-2.amzn2023.0.1.x86_64                    cmake-filesystem-3.22.2-1.amzn2023.0.4.x86_64                 fontconfig-2.13.94-2.amzn2023.0.2.x86_64
  fontconfig-devel-2.13.94-2.amzn2023.0.2.x86_64        fonts-filesystem-1:2.0.5-12.amzn2023.0.2.noarch               freetype-2.13.2-5.amzn2023.0.1.x86_64
  freetype-devel-2.13.2-5.amzn2023.0.1.x86_64           gd-2.3.3-5.amzn2023.0.3.x86_64                                gd-devel-2.3.3-5.amzn2023.0.3.x86_64
  glib2-devel-2.74.7-689.amzn2023.0.2.x86_64            google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch       google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch
  graphite2-1.3.14-7.amzn2023.0.2.x86_64                graphite2-devel-1.3.14-7.amzn2023.0.2.x86_64                  harfbuzz-7.0.0-2.amzn2023.0.1.x86_64
  harfbuzz-devel-7.0.0-2.amzn2023.0.1.x86_64            harfbuzz-icu-7.0.0-2.amzn2023.0.1.x86_64                      jbigkit-libs-2.1-21.amzn2023.0.2.x86_64
  langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch     libICE-1.0.10-6.amzn2023.0.2.x86_64                           libSM-1.2.3-8.amzn2023.0.2.x86_64
  libX11-1.7.2-3.amzn2023.0.4.x86_64                    libX11-common-1.7.2-3.amzn2023.0.4.noarch                     libX11-devel-1.7.2-3.amzn2023.0.4.x86_64
  libX11-xcb-1.7.2-3.amzn2023.0.4.x86_64                libXau-1.0.9-6.amzn2023.0.2.x86_64                            libXau-devel-1.0.9-6.amzn2023.0.2.x86_64
  libXext-1.3.4-6.amzn2023.0.2.x86_64                   libXpm-3.5.15-2.amzn2023.0.3.x86_64                           libXpm-devel-3.5.15-2.amzn2023.0.3.x86_64
  libXrender-0.9.10-14.amzn2023.0.2.x86_64              libXt-1.2.0-4.amzn2023.0.2.x86_64                             libblkid-devel-2.37.4-1.amzn2023.0.4.x86_64
  libffi-devel-3.4.4-1.amzn2023.0.1.x86_64              libicu-67.1-7.amzn2023.0.3.x86_64                             libicu-devel-67.1-7.amzn2023.0.3.x86_64
  libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64             libjpeg-turbo-devel-2.1.4-2.amzn2023.0.5.x86_64               libmount-devel-2.37.4-1.amzn2023.0.4.x86_64
  libpng-2:1.6.37-10.amzn2023.0.6.x86_64                libpng-devel-2:1.6.37-10.amzn2023.0.6.x86_64                  libselinux-devel-3.4-5.amzn2023.0.2.x86_64
  libsepol-devel-3.4-3.amzn2023.0.3.x86_64              libtiff-4.4.0-4.amzn2023.0.18.x86_64                          libtiff-devel-4.4.0-4.amzn2023.0.18.x86_64
  libwebp-1.2.4-1.amzn2023.0.6.x86_64                   libwebp-devel-1.2.4-1.amzn2023.0.6.x86_64                     libxcb-1.13.1-7.amzn2023.0.2.x86_64
  libxcb-devel-1.13.1-7.amzn2023.0.2.x86_64             libxml2-devel-2.10.4-1.amzn2023.0.6.x86_64                    pcre2-devel-10.40-1.amzn2023.0.3.x86_64
  pcre2-utf16-10.40-1.amzn2023.0.3.x86_64               pcre2-utf32-10.40-1.amzn2023.0.3.x86_64                       pixman-0.40.0-3.amzn2023.0.3.x86_64
  sysprof-capture-devel-3.40.1-2.amzn2023.0.2.x86_64    xml-common-0.6.3-56.amzn2023.0.2.noarch                       xorg-x11-proto-devel-2021.4-1.amzn2023.0.2.noarch
  xz-devel-5.2.5-9.amzn2023.0.2.x86_64                  zlib-devel-1.2.11-33.amzn2023.0.5.x86_64

Complete!
[ec2-user@ip-172-31-46-179 ~]$
```

- Create a new nagios user by writing the following commands
  sudo adduser -m nagios
  sudo passwd nagios

```
[ec2-user@ip-172-31-46-179 ~]$ sudo adduser -m naigos
[ec2-user@ip-172-31-46-179 ~]$ sudo passwd naigos
Changing password for user naigos.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-46-179 ~]$
```

- Create a new user group using sudo groupadd nagcmd and add users to the group using the following commands

```
[ec2-user@ip-172-31-46-179 ~]$ sudo usermod -a -G nagcmd naigos
usermod: group 'nagcmd' does not exist
[ec2-user@ip-172-31-46-179 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-46-179 ~]$ sudo usermod -a -G nagcmd naigod
usermod: user 'naigod' does not exist
[ec2-user@ip-172-31-46-179 ~]$ sudo usermod -a -G nagcmd naigos
[ec2-user@ip-172-31-46-179 ~]$ sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-46-179 ~]$
```

- Create a directory for Nagios downloads using the following commands and Also download Nagios and plugin source files
  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

```
[ec2-user@ip-172-31-46-179 ~]$ mkdir downloads
[ec2-user@ip-172-31-46-179 ~]$ cd downloads
[ec2-user@ip-172-31-46-179 downloads]$ 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
--2024-10-06 16:03:26--  https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00::f03c:92ff:fef7:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11333414 (11M) [application/x-gzip]
Saving to: 'nagios-4.4.6.tar.gz'

nagios-4.4.6.tar.gz                100%[==========================================================>]  10.81M  11.0MB/s    in 1.0s

2024-10-06 16:03:27 (11.0 MB/s) - 'nagios-4.4.6.tar.gz' saved [11333414/11333414]

--2024-10-06 16:03:27--  https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2782610 (2.7M) [application/x-gzip]
Saving to: 'nagios-plugins-2.3.3.tar.gz'

nagios-plugins-2.3.3.tar.gz        100%[==========================================================>]   2.65M  7.08MB/s    in 0.4s

2024-10-06 16:03:28 (7.08 MB/s) - 'nagios-plugins-2.3.3.tar.gz' saved [2782610/2782610]

[ec2-user@ip-172-31-46-179 downloads]$
```

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

```
[ec2-user@ip-172-31-46-179 downloads]$ tar zxvf nagios-4.4.6.tar.gz
nagios-4.4.6/
nagios-4.4.6/.gitignore
nagios-4.4.6/.travis.yml
nagios-4.4.6/CONTRIBUTING.md
nagios-4.4.6/Changelog
nagios-4.4.6/INSTALLING
nagios-4.4.6/LEGAL
nagios-4.4.6/LICENSE
nagios-4.4.6/Makefile.in
nagios-4.4.6/README.md
nagios-4.4.6/THANKS
nagios-4.4.6/UPGRADING
nagios-4.4.6/aclocal.m4
nagios-4.4.6/autoconf-macros/
nagios-4.4.6/autoconf-macros/.gitignore
nagios-4.4.6/autoconf-macros/CHANGELOG.md
nagios-4.4.6/autoconf-macros/LICENSE
nagios-4.4.6/autoconf-macros/LICENSE.md
nagios-4.4.6/autoconf-macros/README.md
nagios-4.4.6/autoconf-macros/add_group_user
nagios-4.4.6/autoconf-macros/ax_nagios_get_distrib
nagios-4.4.6/autoconf-macros/ax_nagios_get_files
nagios-4.4.6/autoconf-macros/ax_nagios_get_inetd
nagios-4.4.6/autoconf-macros/ax_nagios_get_init
nagios-4.4.6/autoconf-macros/ax_nagios_get_os
nagios-4.4.6/autoconf-macros/ax_nagios_get_paths
nagios-4.4.6/autoconf-macros/ax_nagios_get_ssl
nagios-4.4.6/base/
nagios-4.4.6/base/.gitignore
nagios-4.4.6/base/Makefile.in
```

- Listing out all the files present in the nagios directory

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ ls -l
total 704
-rw-r--r--.  1 ec2-user ec2-user   6291 Apr 28  2020 CONTRIBUTING.md
-rw-r--r--.  1 ec2-user ec2-user  32590 Apr 28  2020 Changelog
-rw-r--r--.  1 ec2-user ec2-user    422 Apr 28  2020 INSTALLING
-rw-r--r--.  1 ec2-user ec2-user    841 Apr 28  2020 LEGAL
-rw-r--r--.  1 ec2-user ec2-user  18002 Apr 28  2020 LICENSE
-rw-r--r--.  1 ec2-user ec2-user  16506 Apr 28  2020 Makefile.in
-rw-r--r--.  1 ec2-user ec2-user   3395 Apr 28  2020 README.md
-rw-r--r--.  1 ec2-user ec2-user   5832 Apr 28  2020 THANKS
-rw-r--r--.  1 ec2-user ec2-user    643 Apr 28  2020 UPGRADING
-rwxr-xr-x.  1 ec2-user ec2-user    346 Apr 28  2020 aclocal.m4
drwxr-xr-x.  2 ec2-user ec2-user  16384 Apr 28  2020 autoconf-macros
drwxr-xr-x.  2 ec2-user ec2-user  16384 Apr 28  2020 base
drwxr-xr-x.  2 ec2-user ec2-user  16384 Apr 28  2020 cgi
drwxr-xr-x.  2 ec2-user ec2-user    148 Apr 28  2020 common
-rwxr-xr-x.  1 ec2-user ec2-user  43765 Apr 28  2020 config.guess
-rwxr-xr-x.  1 ec2-user ec2-user  36345 Apr 28  2020 config.sub
-rwxr-xr-x.  1 ec2-user ec2-user 246354 Apr 28  2020 configure
-rw-r--r--.  1 ec2-user ec2-user  29812 Apr 28  2020 configure.ac
drwxr-xr-x.  5 ec2-user ec2-user  16384 Apr 28  2020 contrib
drwxr-xr-x.  2 ec2-user ec2-user    129 Apr 28  2020 docs
-rw-r--r--.  1 ec2-user ec2-user    886 Apr 28  2020 doxy.conf
-rwxr-xr-x.  1 ec2-user ec2-user   7025 Apr 28  2020 functions
drwxr-xr-x. 11 ec2-user ec2-user  16384 Apr 28  2020 html
drwxr-xr-x.  2 ec2-user ec2-user  16384 Apr 28  2020 include
-rwxr-xr-x.  1 ec2-user ec2-user     77 Apr 28  2020 indent-all.sh
-rwxr-xr-x.  1 ec2-user ec2-user    161 Apr 28  2020 indent.sh
-rwxr-xr-x.  1 ec2-user ec2-user   5869 Apr 28  2020 install-sh
drwxr-xr-x.  2 ec2-user ec2-user  16384 Apr 28  2020 lib
-rwxr-xr-x.  1 ec2-user ec2-user    461 Apr 28  2020 make-tarball
```

- Then run the configuration script with the following command
  ./configure --with-command-group=nagcmd

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ ./configure --with-command-group=nagcmd
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking whether make sets $(MAKE)... yes
checking whether ln -s works... yes
checking for strip... /usr/bin/strip
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for ANSI C header files... yes
checking whether time.h and sys/time.h may both be included... yes
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for sys/types.h... yes
checking for sys/stat.h... yes
checking for stdlib.h... yes
checking for string.h... yes
checking for memory.h... yes
checking for strings.h... yes
checking for inttypes.h... yes
checking for stdint.h... yes
checking for unistd.h... yes
```

```
*** Configuration summary for nagios 4.4.6 2020-04-28 ***:

 General Options:
 -------------------------
        Nagios executable:  nagios
        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:  /run/nagios.lock
   Check result directory:  /usr/local/nagios/var/spool/checkresults
          Init directory:  /lib/systemd/system
  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-172-31-46-179 nagios-4.4.6]$
```

- Compile the source code with the following commands make all

```
  inflating: dist/css/bootstrap-theme.css
  inflating: dist/css/bootstrap-theme.min.css.map
  inflating: dist/css/bootstrap-theme.css.map
  inflating: dist/css/bootstrap.min.css
  inflating: dist/css/bootstrap-theme.min.css
  inflating: dist/css/bootstrap.css
mkdir -p d3
(cd d3 && unzip -u ../d3-3.5.17.zip)
Archive:  ../d3-3.5.17.zip
  inflating: bower.json
  inflating: d3.js
  inflating: d3.min.js
  inflating: LICENSE
  inflating: README.md
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/html'
if [ xyes = xyes ]; then \
        cd ./module && make; \
fi
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/module'
gcc -I.. -fPIC -g -O2 -DHAVE_CONFIG_H -o helloworld.o helloworld.c -shared
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/module'
cd ./worker && make all
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/worker'
cd ./ping && make all
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/worker/ping'
gcc -I.. -I ../../include -I ../.. -g -O2 -DHAVE_CONFIG_H -o worker-ping worker-ping.c -L ../../lib -l nagios
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/worker/ping'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/worker'

*** Compile finished ***
```

- Install binaries,init script and sample config files

sudo make install

```
nagios:x:1002:1003::/home/nagios:/bin/bash
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ sudo make install
cd ./base && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagiostats /usr/local/nagios/bin
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/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 -s -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin; \
done
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/cgi'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.6/cgi'
cd ./html && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.6/html'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/media
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/stylesheets
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/contexthelp
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/docs
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/docs/images
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/js
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/images
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/images/logos
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/includes
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/ssi
```

sudo make install-init

sudo make install-config

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ sudo make install-config
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg /usr/local/nagios/etc/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg

*** Config files installed ***

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

[ec2-user@ip-172-31-46-179 nagios-4.4.6]$
```

sudo make install-commandmode

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

[ec2-user@ip-172-31-46-179 nagios-4.4.6]$
```

- Edit the Config File to Change the Email Address

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



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

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

define contact {
                                                            [ Read 51 lines ]
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo        M-A Set Mark     M-] To
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo        M-6 Copy         ^Q Wher
```

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

```
##############################################################################
#
# 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                   tejasgunja1021@gmail.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS ******
}



##############################################################################
#
# CONTACT GROUPS
#
##############################################################################
                                                            [ Wrote 51 lines ]
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo        M-A Set Mark
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line    M-E Redo        M-6 Copy
```

- Extract the Plugins Source File

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ cd ..
[ec2-user@ip-172-31-46-179 downloads]$ tar zxvf nagios-plugins-2.3.3.tar.gz
nagios-plugins-2.3.3/
nagios-plugins-2.3.3/perlmods/
nagios-plugins-2.3.3/perlmods/Config-Tiny-2.14.tar.gz
nagios-plugins-2.3.3/perlmods/parent-0.226.tar.gz
nagios-plugins-2.3.3/perlmods/Test-Simple-0.98.tar.gz
nagios-plugins-2.3.3/perlmods/Makefile.in
nagios-plugins-2.3.3/perlmods/version-0.9903.tar.gz
nagios-plugins-2.3.3/perlmods/Makefile.am
nagios-plugins-2.3.3/perlmods/Module-Runtime-0.013.tar.gz
nagios-plugins-2.3.3/perlmods/Module-Metadata-1.000014.tar.gz
nagios-plugins-2.3.3/perlmods/Params-Validate-1.08.tar.gz
nagios-plugins-2.3.3/perlmods/Class-Accessor-0.34.tar.gz
nagios-plugins-2.3.3/perlmods/Try-Tiny-0.18.tar.gz
nagios-plugins-2.3.3/perlmods/Module-Implementation-0.07.tar.gz
nagios-plugins-2.3.3/perlmods/Makefile
nagios-plugins-2.3.3/perlmods/Perl-OSType-1.003.tar.gz
nagios-plugins-2.3.3/perlmods/install_order
nagios-plugins-2.3.3/perlmods/Nagios-Plugin-0.36.tar.gz
nagios-plugins-2.3.3/perlmods/Math-Calc-Units-1.07.tar.gz
nagios-plugins-2.3.3/perlmods/Module-Build-0.4007.tar.gz
nagios-plugins-2.3.3/ABOUT-NLS
nagios-plugins-2.3.3/configure.ac
nagios-plugins-2.3.3/Makefile.in
nagios-plugins-2.3.3/config.h.in
nagios-plugins-2.3.3/ChangeLog
nagios-plugins-2.3.3/AUTHORS
nagios-plugins-2.3.3/lib/
nagios-plugins-2.3.3/lib/parse_ini.h
```

- Create a Nagios Admin Account

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$
```

- Compile and Install Plugins
  ./configure --with-nagios-user=nagios --with-nagios-group=nagios make

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ configure --with-nagios-user=nagios --with-nagios-group=nagios
make
-bash: configure: command not found
Please supply a command line argument (i.e. 'make all').  Other targets are:
   nagios cgis contrib modules workers
   test
   install                         install-base
   install-cgis                    install-html
   install-webconf                 install-config
   install-init                    install-daemoninit
   install-commandmode             install-groups-users
   install-exfoliation             install-classicui
   install-basic                   install-unstripped
   fullinstall
   clean
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$
```

- Start Nagios

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

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ 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
error reading information on service nagios: No such file or directory
Note: Forwarding request to 'systemctl enable nagios.service'.
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /usr/lib/systemd/system/nagios.service.

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...
   Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
       Checked 8 services.
       Checked 1 hosts.
       Checked 1 host groups.
       Checked 0 service groups.
       Checked 1 contacts.
       Checked 1 contact groups.
       Checked 24 commands.
       Checked 5 time periods.
       Checked 0 host escalations.
```

```
Reading configuration data...
   Read main config file okay...
   Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
        Checked 8 services.
        Checked 1 hosts.
        Checked 1 host groups.
        Checked 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 1 hosts
        Checked 0 service dependencies
        Checked 0 host dependencies
        Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors:   0

Things look okay - No serious problems were detected during the pre-flight check
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$
```

- Verify the sample configuration files

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ 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...
   Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
        Checked 8 services.
        Checked 1 hosts.
        Checked 1 host groups.
        Checked 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 1 hosts
        Checked 0 service dependencies
        Checked 0 host dependencies
```

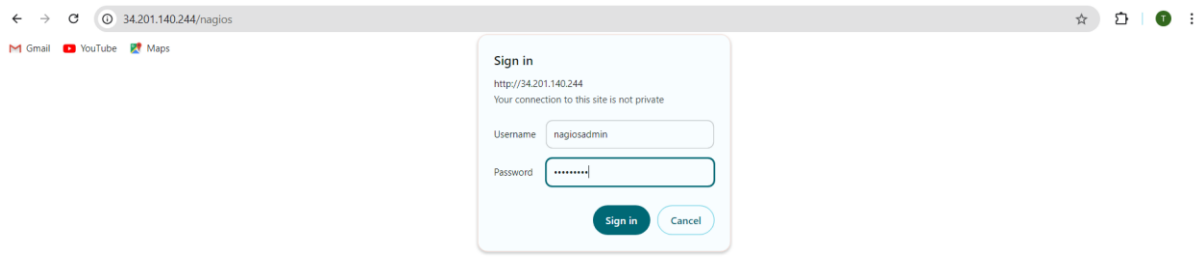
- If there are no errors, you can go ahead and start Nagios and Check the status of Nagios

```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
     Active: active (running) since Sun 2024-10-06 16:37:12 UTC; 1h 47min ago
       Docs: https://www.nagios.org/documentation
   Main PID: 49199 (nagios)
      Tasks: 8 (limit: 1112)
     Memory: 3.7M
        CPU: 1.379s
     CGroup: /system.slice/nagios.service
             ├─49199 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─49200 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─49201 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─49202 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─49203 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─49204 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─73692 /usr/local/nagios/libexec/check_ping -H 127.0.0.1 -w 100.0,20% -c 500.0,60% -p 5
             └─73693 /usr/bin/ping -n -U -w 10 -c 5 127.0.0.1

Oct 06 16:58:27 ip-172-31-46-179.ec2.internal nagios[49199]: SERVICE ALERT: localhost;Current Users;OK;SOFT;1;USERS OK - 1 users currently logged in
Oct 06 16:59:46 ip-172-31-46-179.ec2.internal nagios[49199]: SERVICE ALERT: localhost;PING;OK;SOFT;1;PING OK - Packet loss = 0%, RTA = 0.04 ms
Oct 06 17:37:12 ip-172-31-46-179.ec2.internal nagios[49199]: Auto-save of retention data completed successfully.
Oct 06 17:56:34 ip-172-31-46-179.ec2.internal nagios[49199]: SERVICE NOTIFICATION: nagiosadmin;localhost;Swap Usage;CRITICAL;notify-service-by-email;SWA
Oct 06 17:56:34 ip-172-31-46-179.ec2.internal nagios[49199]: wproc: NOTIFY job 38 from worker Core Worker 49202 is a non-check helper but exited with re
Oct 06 17:56:34 ip-172-31-46-179.ec2.internal nagios[49199]: wproc:   host=localhost; service=Swap Usage; contact=nagiosadmin
Oct 06 17:56:34 ip-172-31-46-179.ec2.internal nagios[49199]: wproc:   early_timeout=0; exited_ok=1; wait_status=32512; error_code=0;
Oct 06 17:56:34 ip-172-31-46-179.ec2.internal nagios[49199]: wproc:   stderr line 01: /bin/sh: line 1: /bin/mail: No such file or directory
```

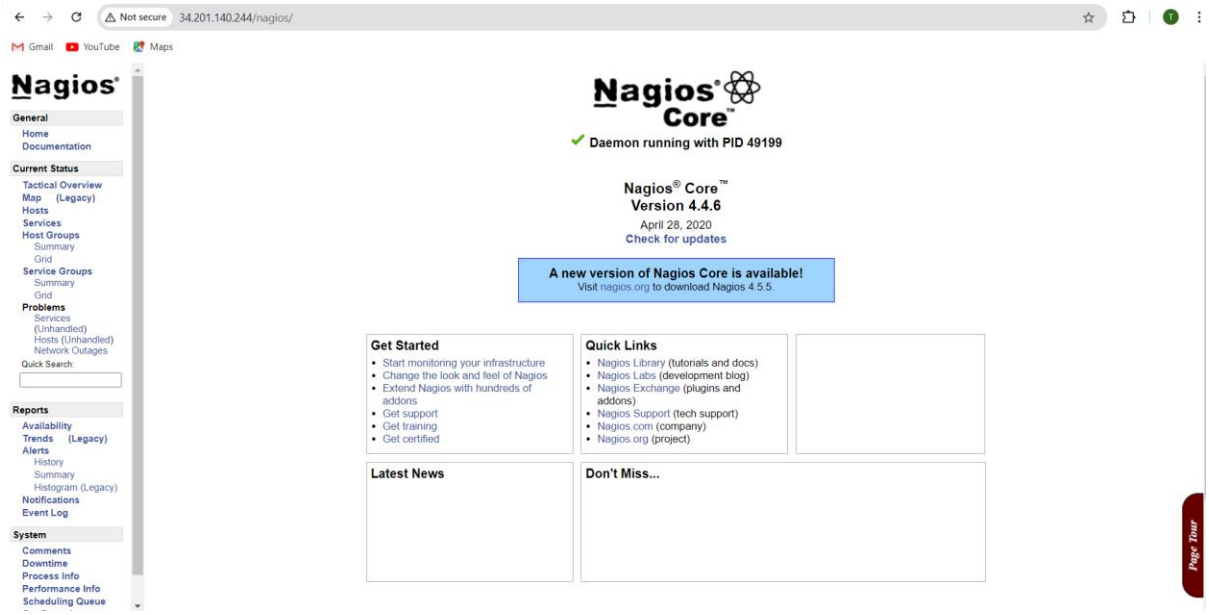
- Access Nagios Web Interface

  Copy the Public IP address of your EC2 instance.
  Open your browser and navigate to http://nagios.

- Enter username as nagiosadmin and password you set



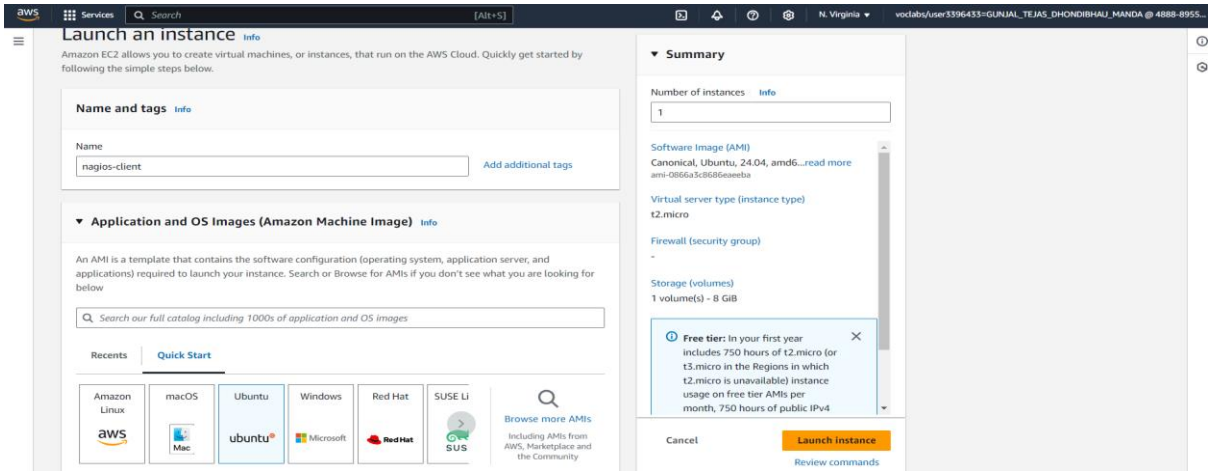
- After sign in, this page will be displayed.
  This means that Nagios was correctly installed and configured with its plugins so far.



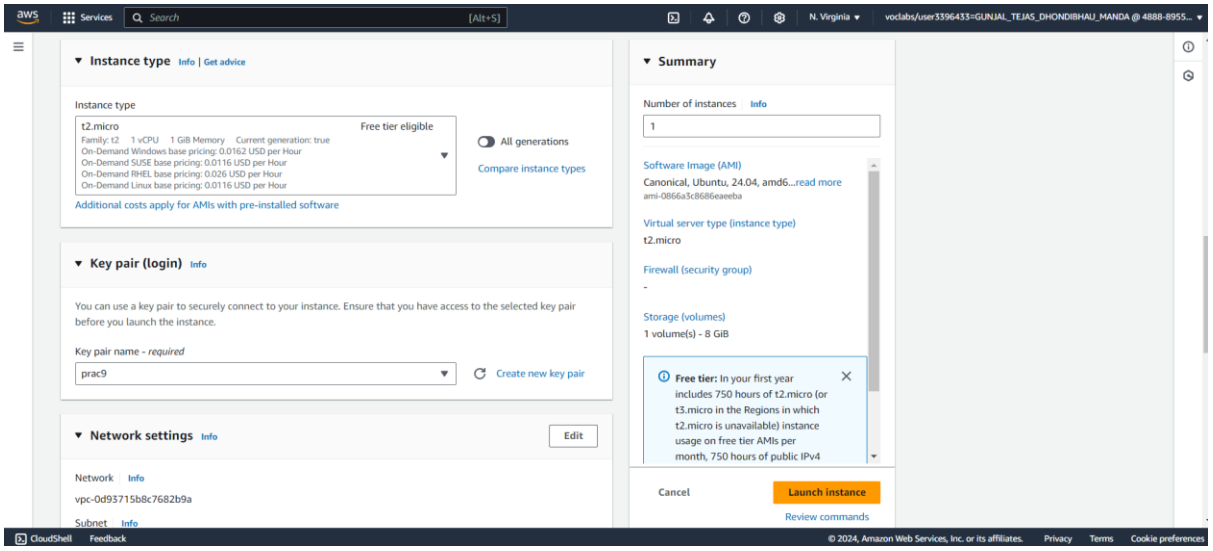
EXPERIMENT NO: - 10

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

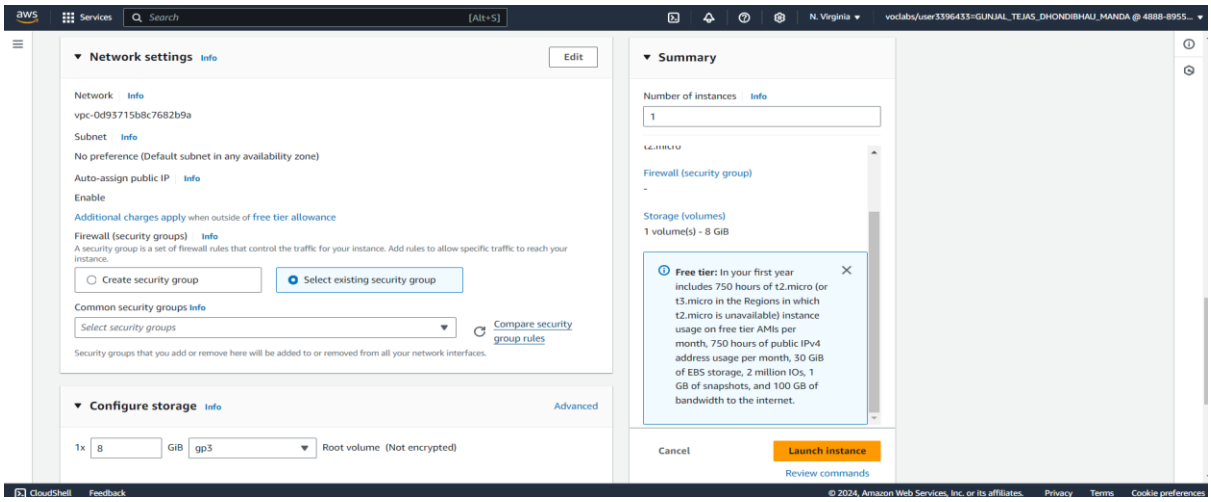
- Launch an instance

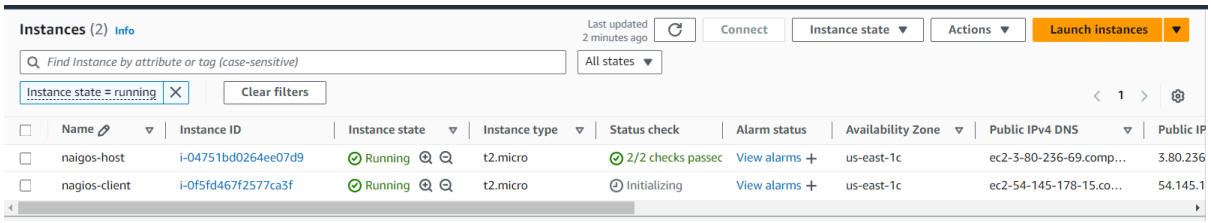


- Select Ubuntu as the os give a meaningful name of the instance.
  Make sure to select the same key-pair login used in the prac9 machine

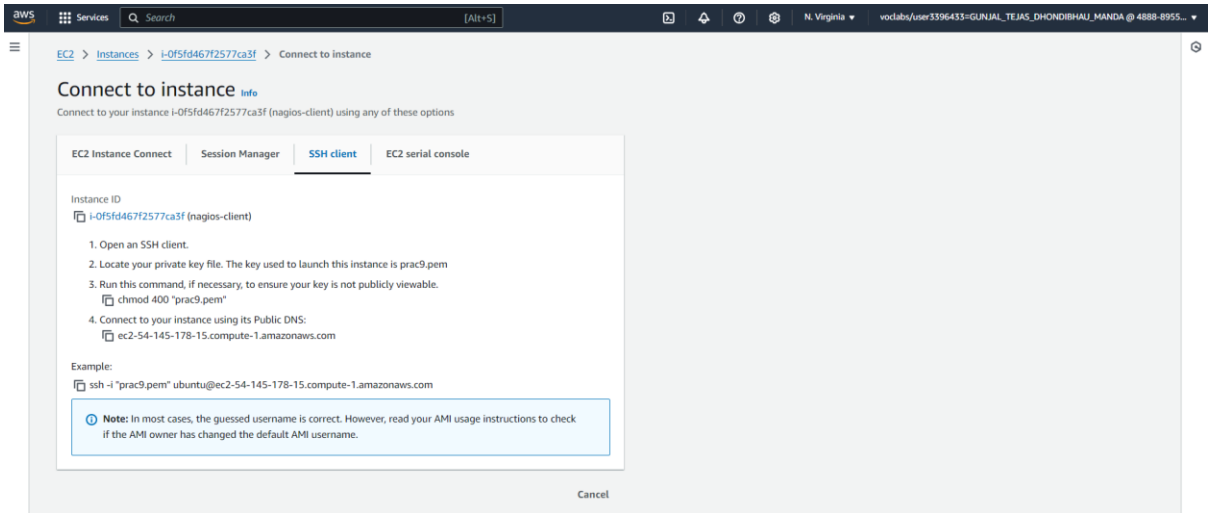


- Click on launch instance.



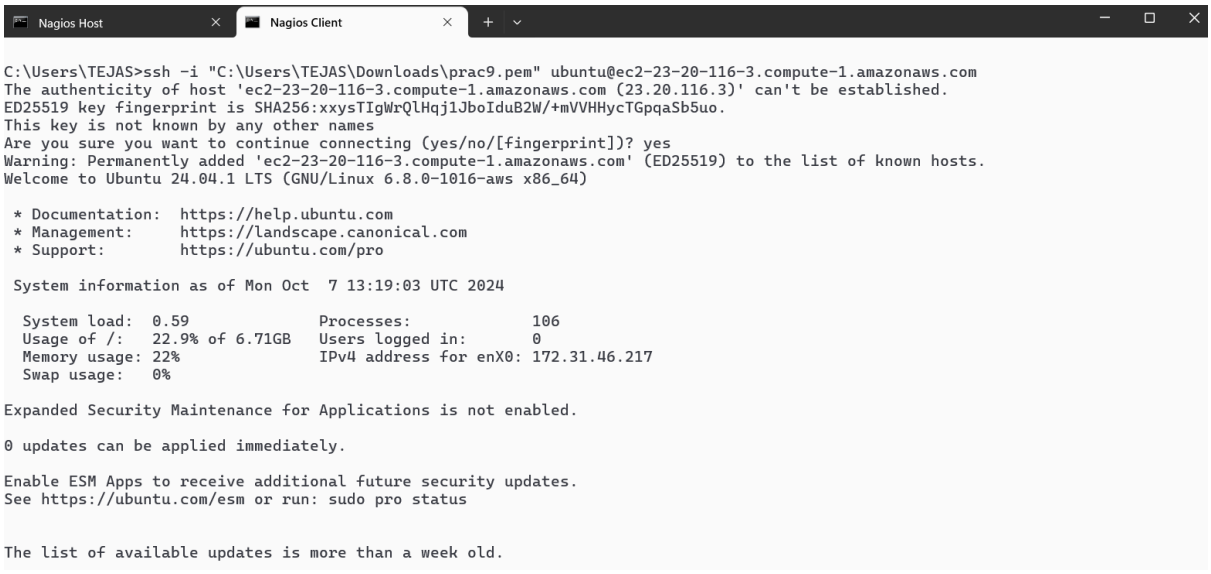


- Now connect with this client machine using the ssh through your terminal(open a new terminal in your local machine and we will need both of the terminals open)



- Nagios Host

- Note to change the path of the .pem file.



- Go to nagios host machine (Host machine) & Perform the following commands

ps -ef | grep nagios

```
[ec2-user@ip-172-31-46-179 ~]$ ps -ef | grep nagios
nagios      3420       1  0 12:40 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios      3421    3420  0 12:40 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      3422    3420  0 12:40 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      3423    3420  0 12:40 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      3424    3420  0 12:40 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      3425    3420  0 12:40 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user   10340    9541  0 13:20 pts/0    00:00:00 grep --color=auto nagios
```

sudo su

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

```
[ec2-user@ip-172-31-46-179 ~]$ sudo su
[root@ip-172-31-46-179 ec2-user]# mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-46-179 ec2-user]# ls
downloads  nagios-4.4.6  nagios-4.4.6.tar.gz  nagios-plugins-2.3.3.tar.gz
```

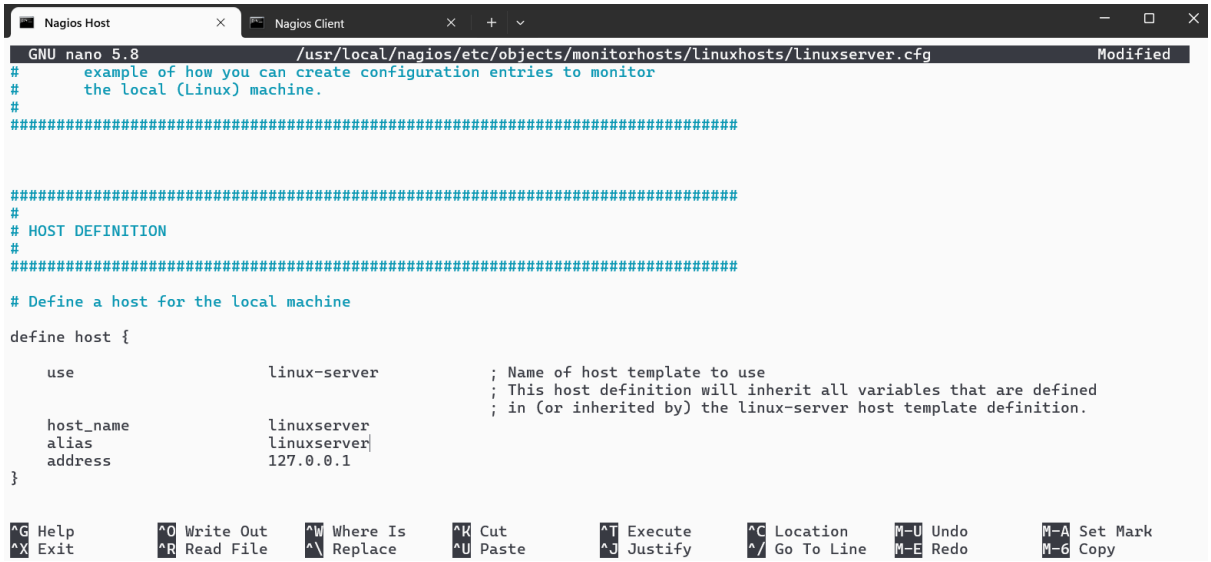
cp /usr/local/nagios/etc/objects/localhost.cfg
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

```
[root@ip-172-31-46-179 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linu
xhosts/linuxserver.cfg
```

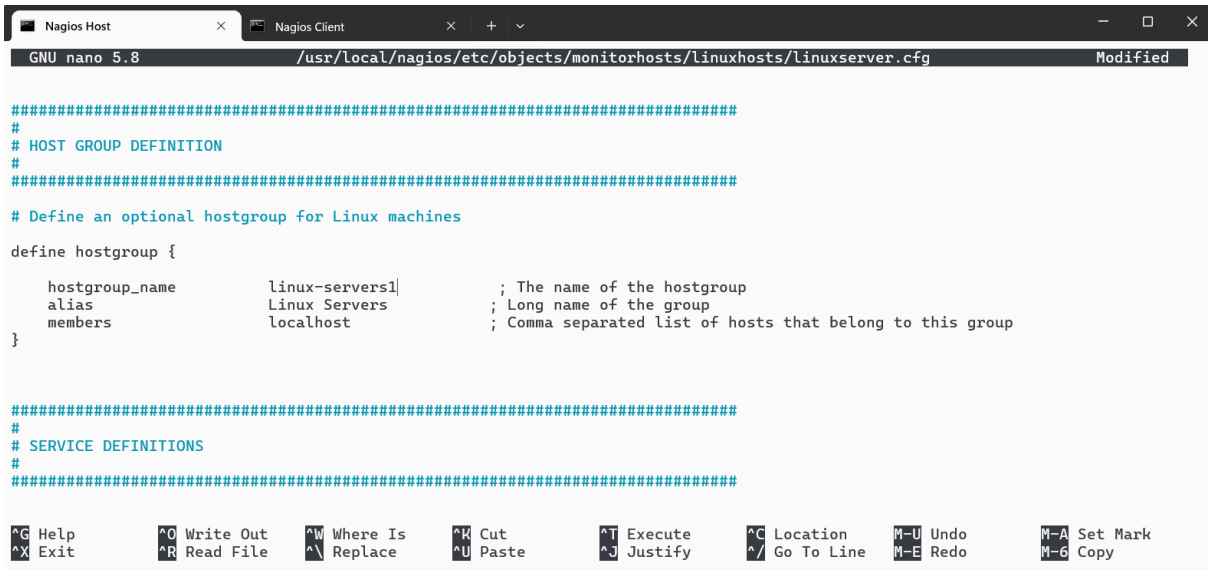
nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

```
[root@ip-172-31-46-179 ec2-user]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-46-179 ec2-user]#
```

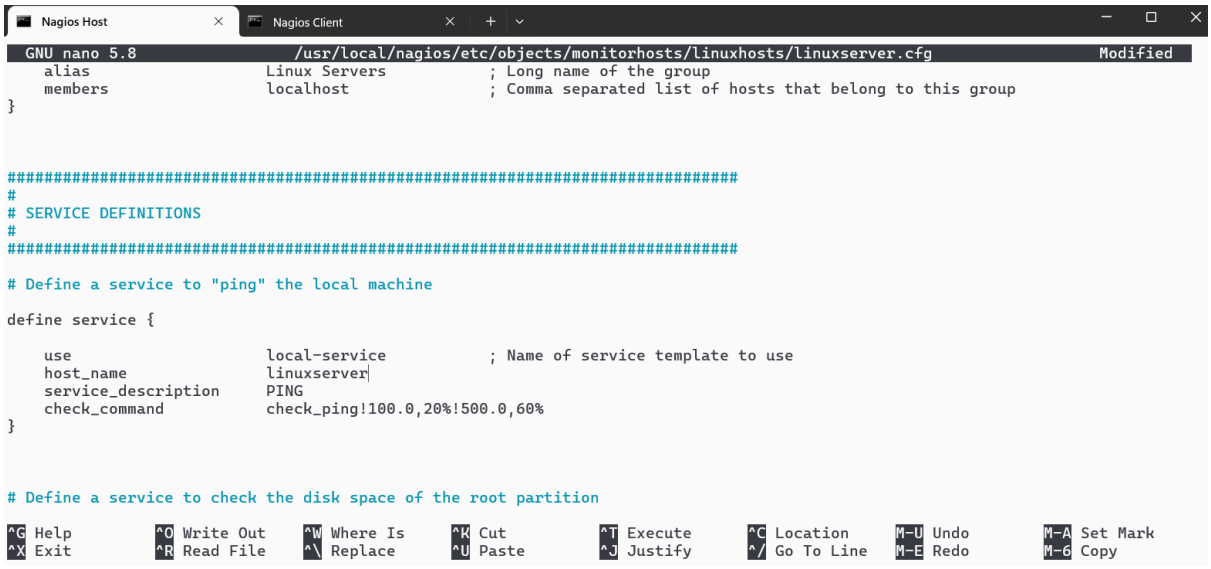
- Change hostname and alias to linuxserver Change address to public ip address of client instance (Ubuntu instance)

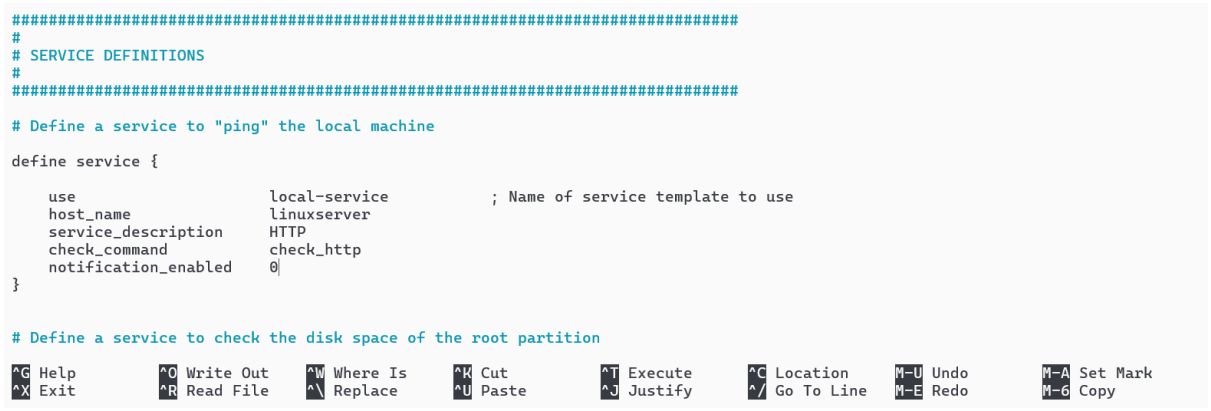


- Change hostgroup_name to linux-servers1

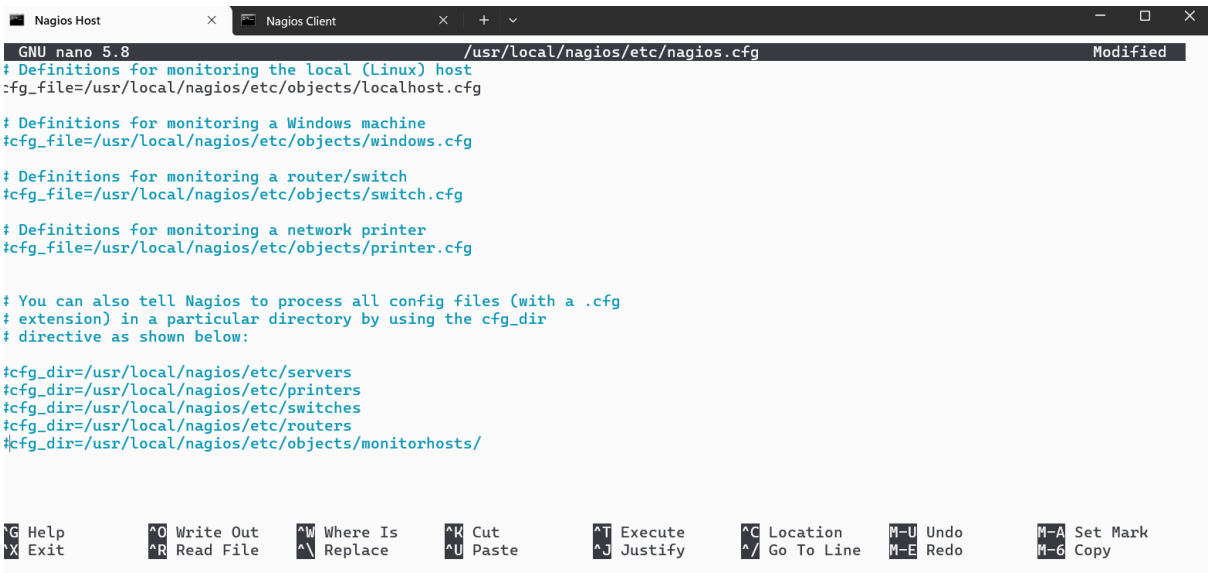


- Change the occurrences of hostname further in the document from localhost to linuxserver

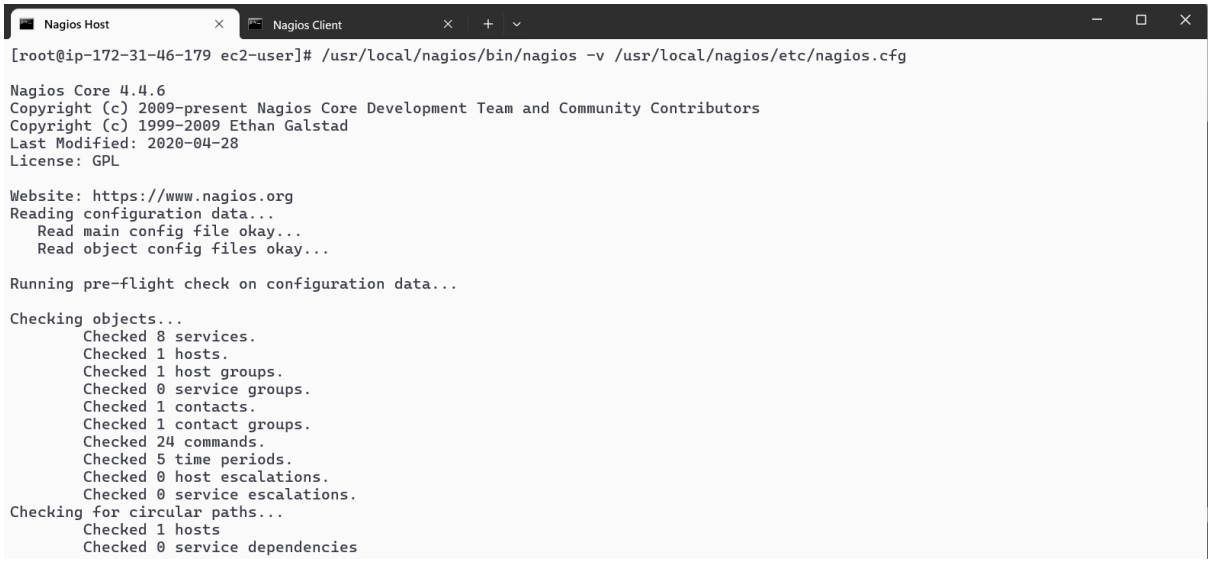


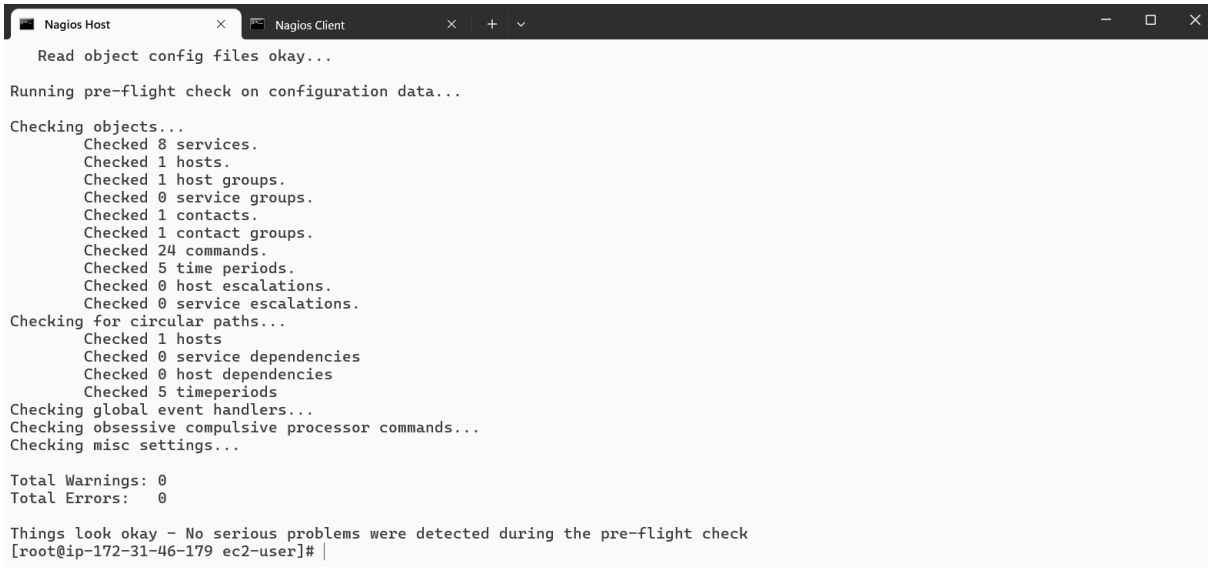


- Open nagios configuration file and add the line shown below
  cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/

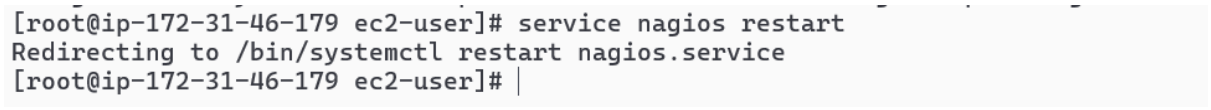


- Verify configuration files
  /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

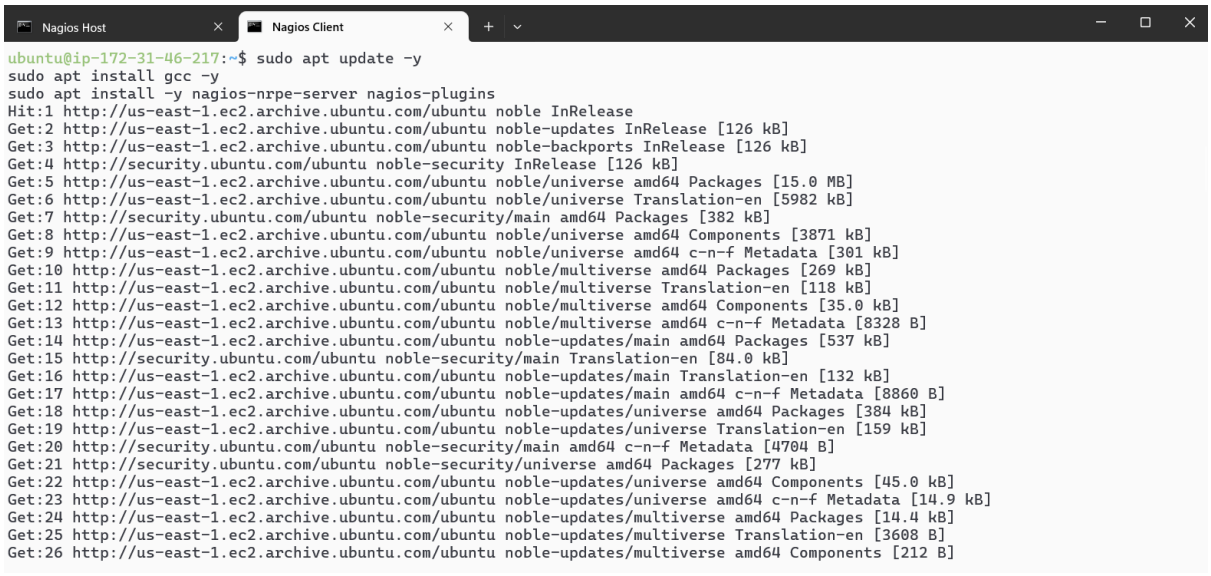


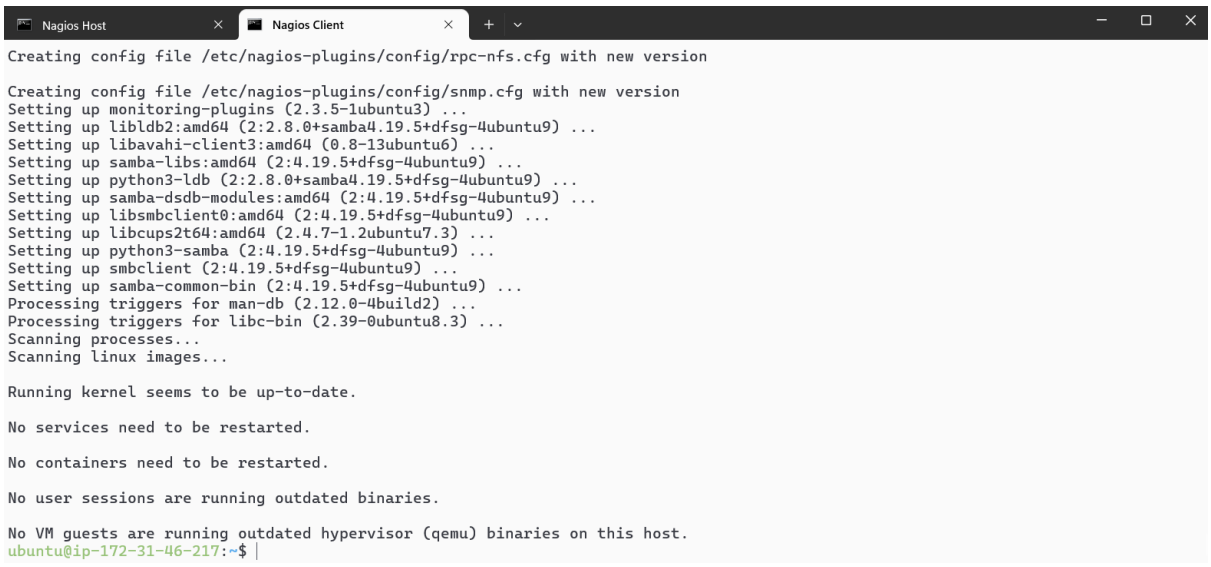


- Restart nagios service.
  service nagios restart

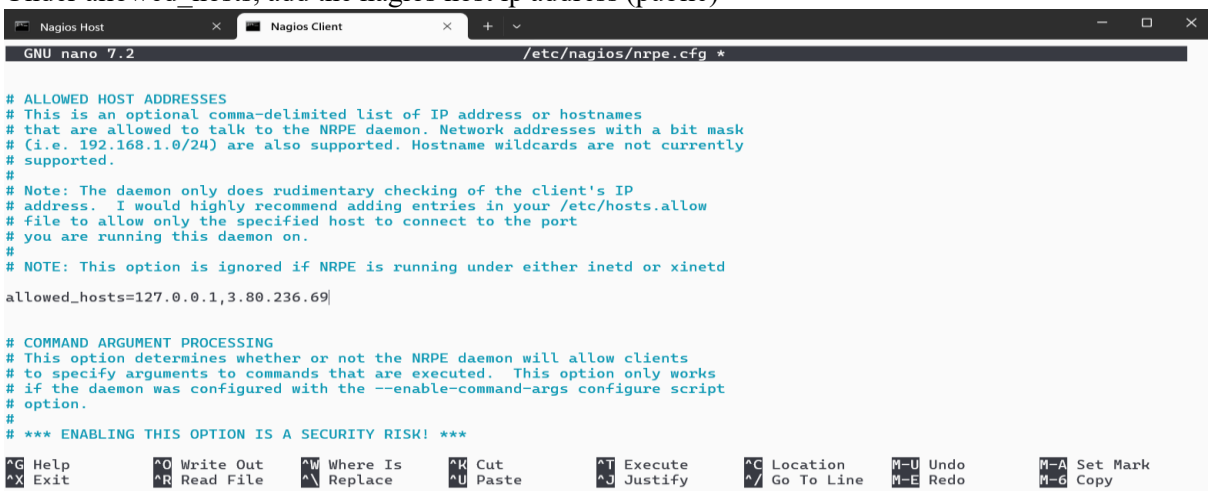


- Go to client machine (ubuntu machine) & Perform the following commands
  sudo apt update -y
  sudo apt install gcc -y
  sudo apt install -y nagios-nrpe-server nagios-plugins

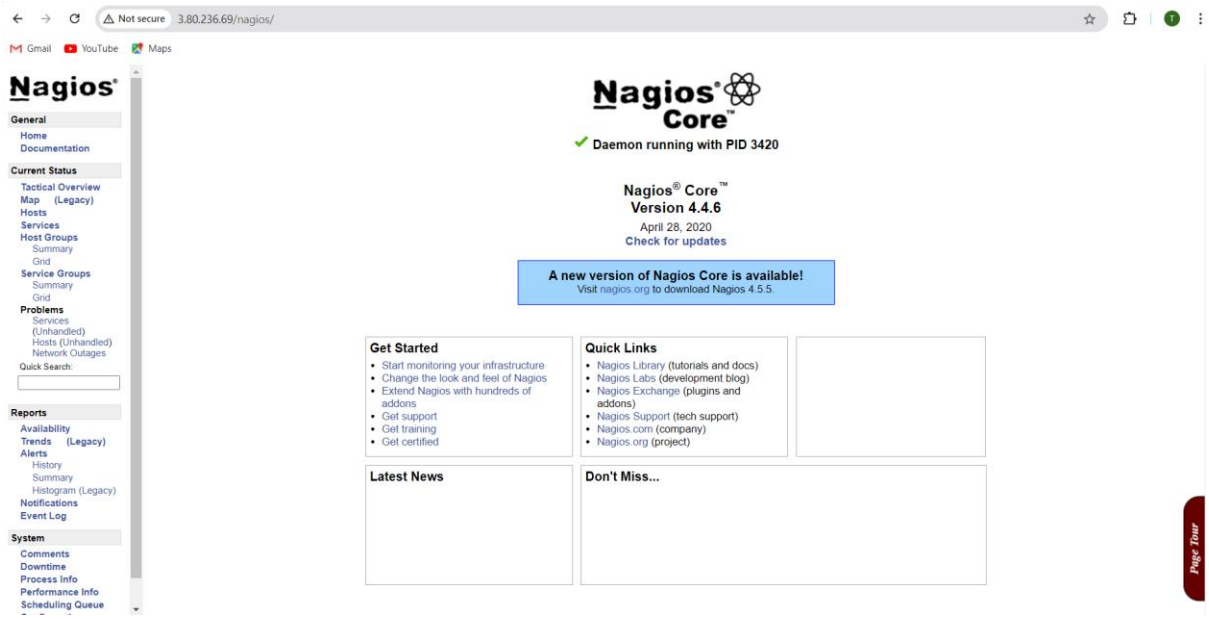




- Open the nrpe.cfg file in nano editor
  sudo nano /etc/nagios/nrpe.cfg
  Under allowed_hosts, add the nagios host ip address (public)

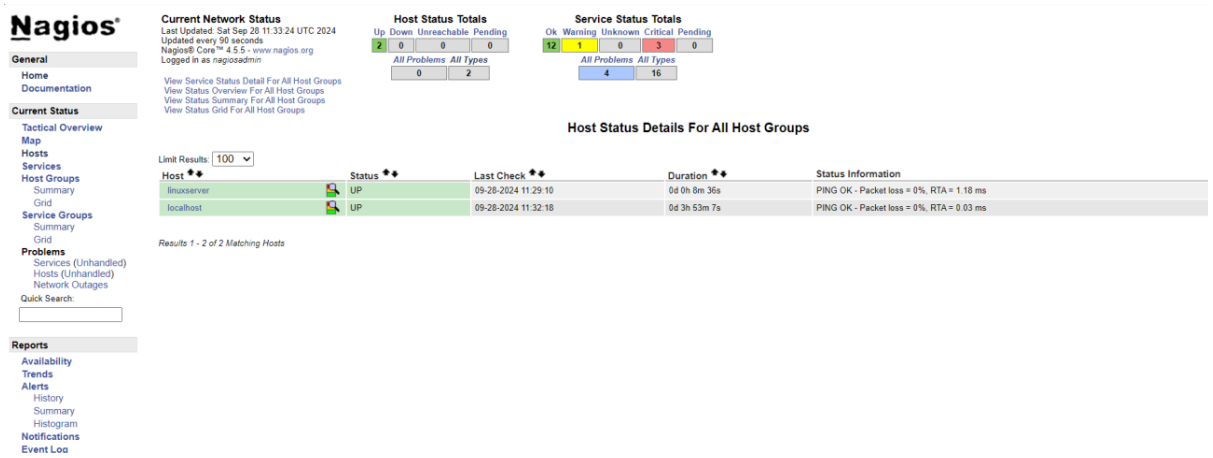


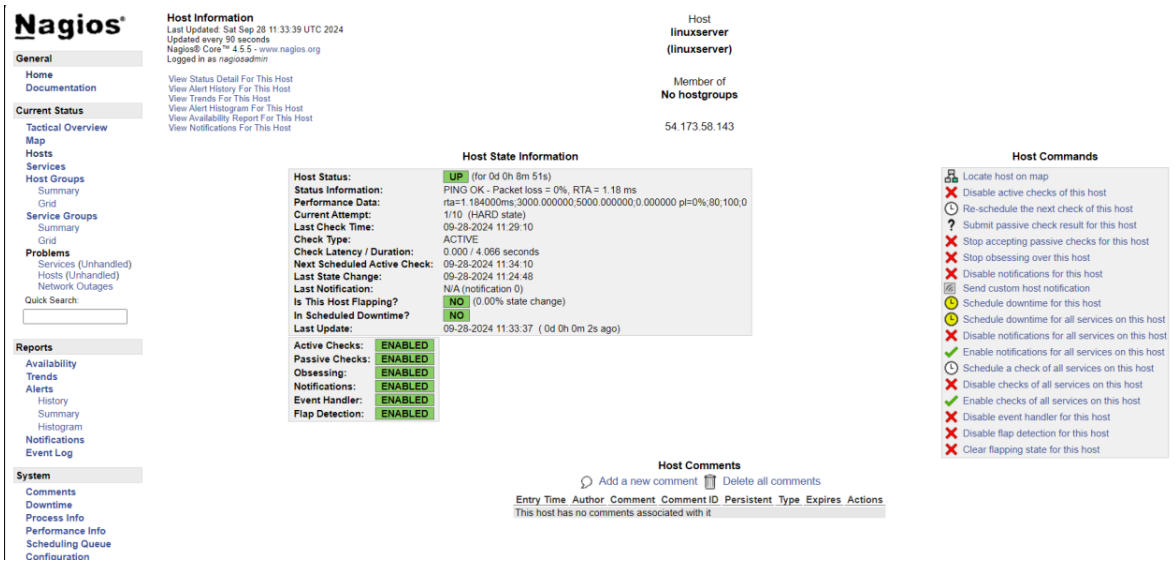
- Go to nagios dashboard



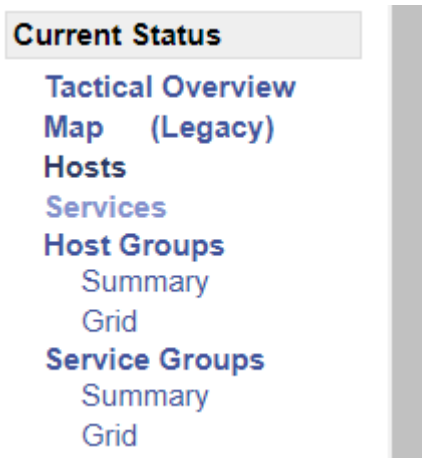
- Click on hosts

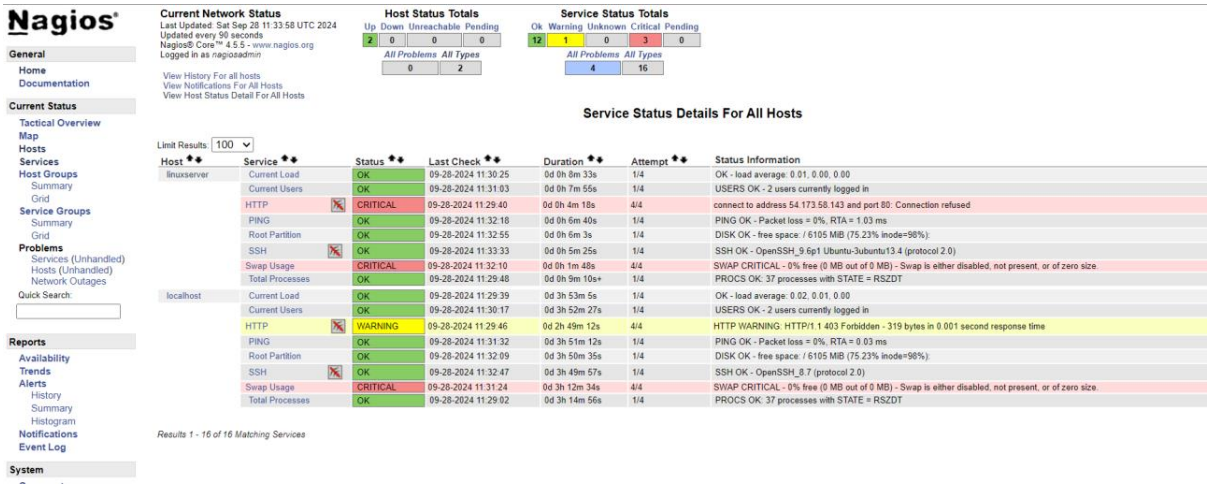
- Click on linux server





- Click on nagios services





Nagios

General
- Home
- Documentation

Current Status
- Tactical Overview
- Map
- Hosts
- Services
- Host Groups
  - Summary
  - Grid
- Service Groups
  - Summary
  - Grid
- Problems
  - Services (Unhandled)
  - Hosts (Unhandled)
  - Network Outages

Quick Search:

Reports
- Availability
- Trends
- Alerts
  - History
  - Summary
  - Histogram
- Notifications
- Event Log

System

Current Network Status
Last Updated: Sat Sep 28 11:33:58 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged in as *nagiosadmin*

View History For all hosts
View Notifications For All Hosts
View Host Status Detail For All Hosts

Host Status Totals

Up	Down	Unreachable	Pending
2	0	0	0

All Problems	All Types
0	2

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
12	1	0	3	0

All Problems	All Types
4	16

Service Status Details For All Hosts

Limit Results: 100

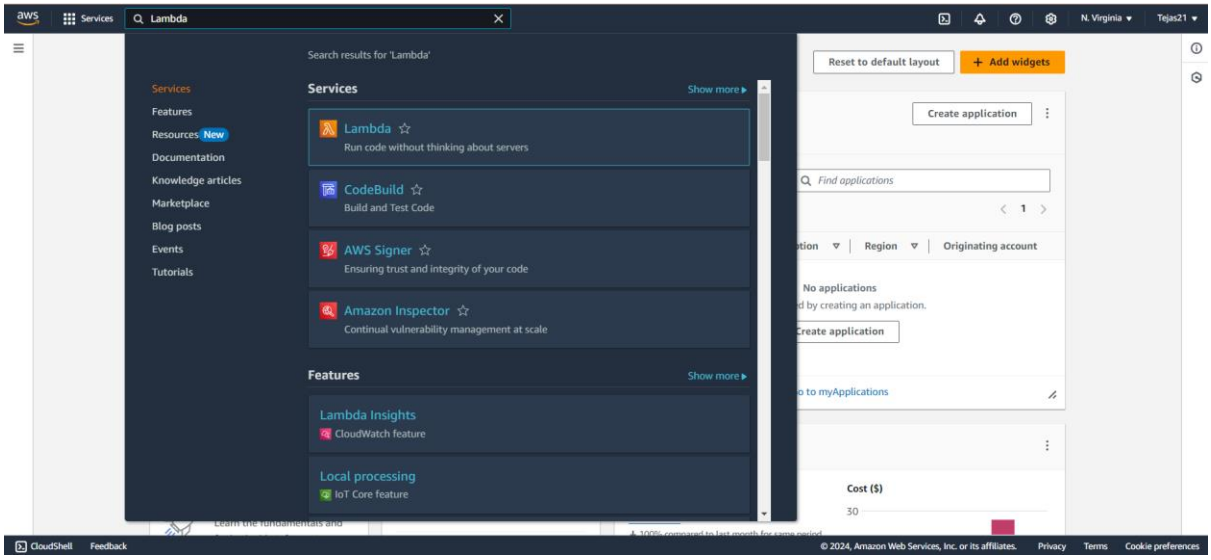
Host	Service	Status	Last Check	Duration	Attempt	Status Information
linuxserver	Current Load	OK	09-28-2024 11:30:25	0d 0h 8m 33s	1/4	OK - load average: 0.01, 0.00, 0.00
	Current Users	OK	09-28-2024 11:31:03	0d 0h 7m 55s	1/4	USERS OK - 2 users currently logged in
	HTTP	CRITICAL	09-28-2024 11:29:40	0d 0h 4m 18s	4/4	connect to address 54.173.58.143 and port 80: Connection refused
	PING	OK	09-28-2024 11:32:18	0d 0h 6m 40s	1/4	PING OK - Packet loss = 0%, RTA = 1.03 ms
	Root Partition	OK	09-28-2024 11:32:55	0d 0h 6m 3s	1/4	DISK OK - free space: / 6105 MiB (75.23% inode=98%):
	SSH	OK	09-28-2024 11:33:33	0d 0h 5m 25s	1/4	SSH OK - OpenSSH_9.6p1 Ubuntu-3ubuntu13.4 (protocol 2.0)
	Swap Usage	CRITICAL	09-28-2024 11:32:10	0d 0h 1m 48s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
	Total Processes	OK	09-28-2024 11:29:48	0d 0h 9m 10s+	1/4	PROCS OK: 37 processes with STATE = RSZDT
localhost	Current Load	OK	09-28-2024 11:29:39	0d 3h 53m 5s	1/4	OK - load average: 0.02, 0.01, 0.00
	Current Users	OK	09-28-2024 11:30:17	0d 3h 52m 27s	1/4	USERS OK - 2 users currently logged in
	HTTP	WARNING	09-28-2024 11:29:46	0d 2h 49m 12s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.001 second response time
	PING	OK	09-28-2024 11:31:32	0d 3h 51m 12s	1/4	PING OK - Packet loss = 0%, RTA = 0.03 ms
	Root Partition	OK	09-28-2024 11:32:09	0d 3h 50m 35s	1/4	DISK OK - free space: / 6105 MiB (75.23% inode=98%):
	SSH	OK	09-28-2024 11:32:47	0d 3h 49m 57s	1/4	SSH OK - OpenSSH_8.7 (protocol 2.0)
	Swap Usage	CRITICAL	09-28-2024 11:31:24	0d 3h 12m 34s	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	09-28-2024 11:29:02	0d 3h 14m 56s	1/4	PROCS OK: 37 processes with STATE = RSZDT

Results 1 - 16 of 16 Matching Services

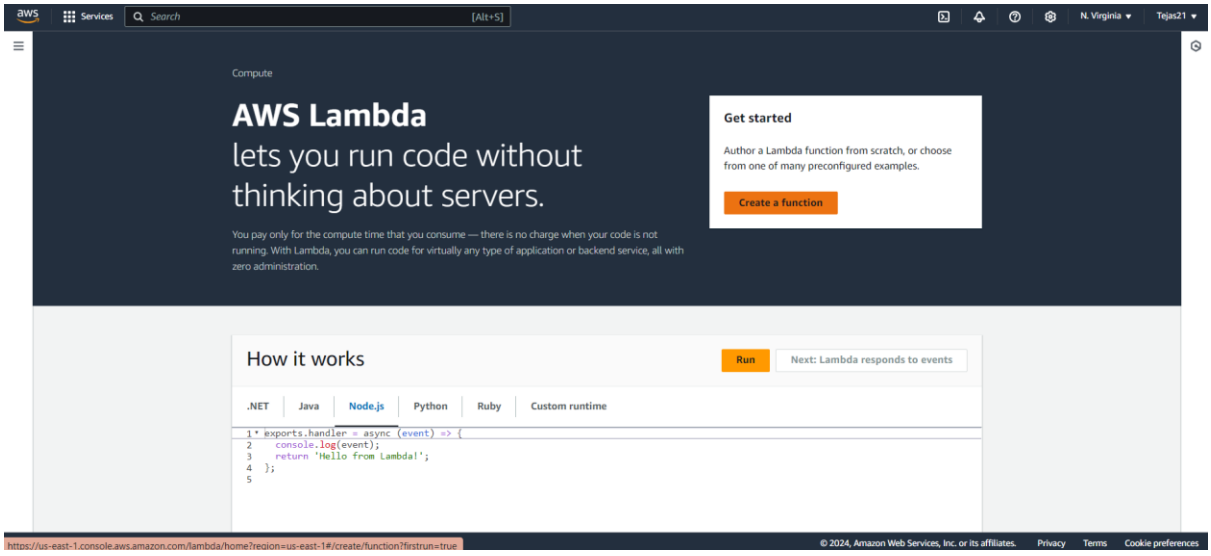
EXPERIMENT NO: - 11

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

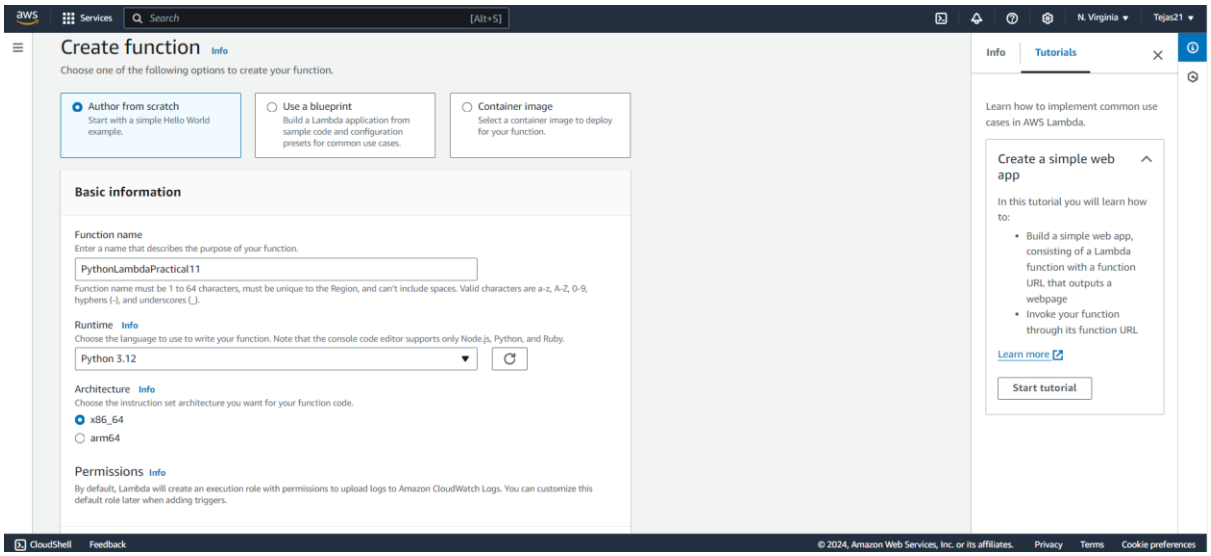
- Open Lambda to create an AWS Lambda function



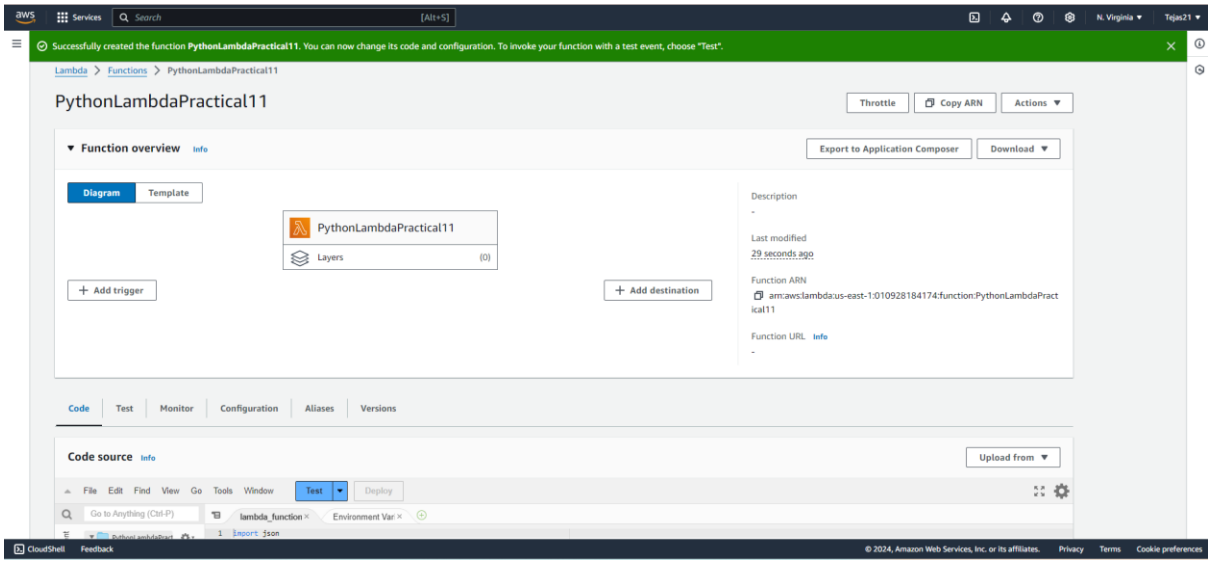
- Open up the Lambda Console and click on the Create button.

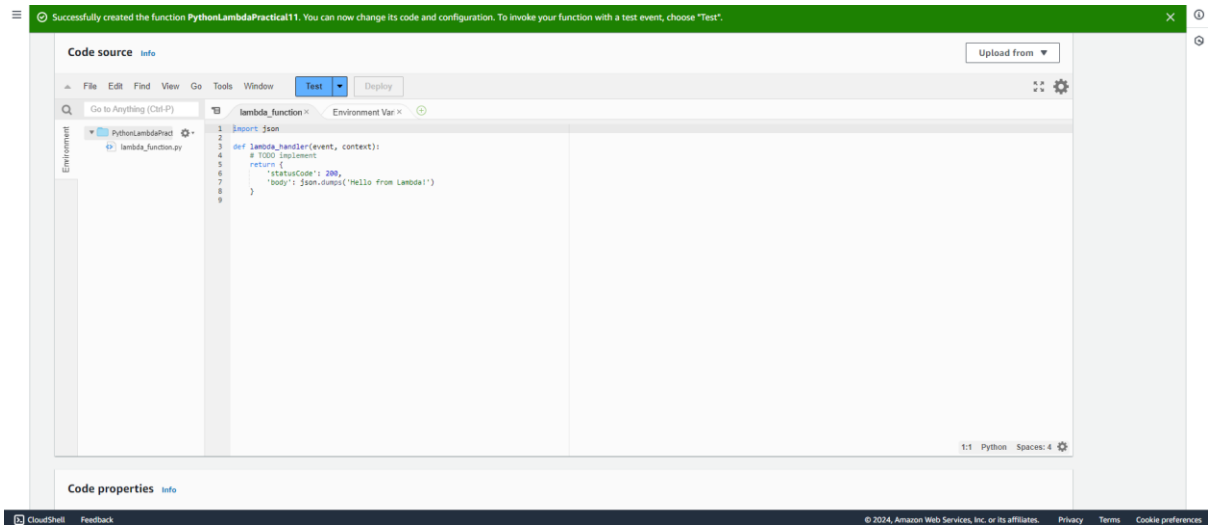


- Choose to create a function from scratch or use a blueprint, i.e templates defined by AWS for you with all configuration presets required for the most common use cases. Then, choose a runtime env for your function, under the dropdown, you can see all the options AWS supports, Python, Nodejs, .NET and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

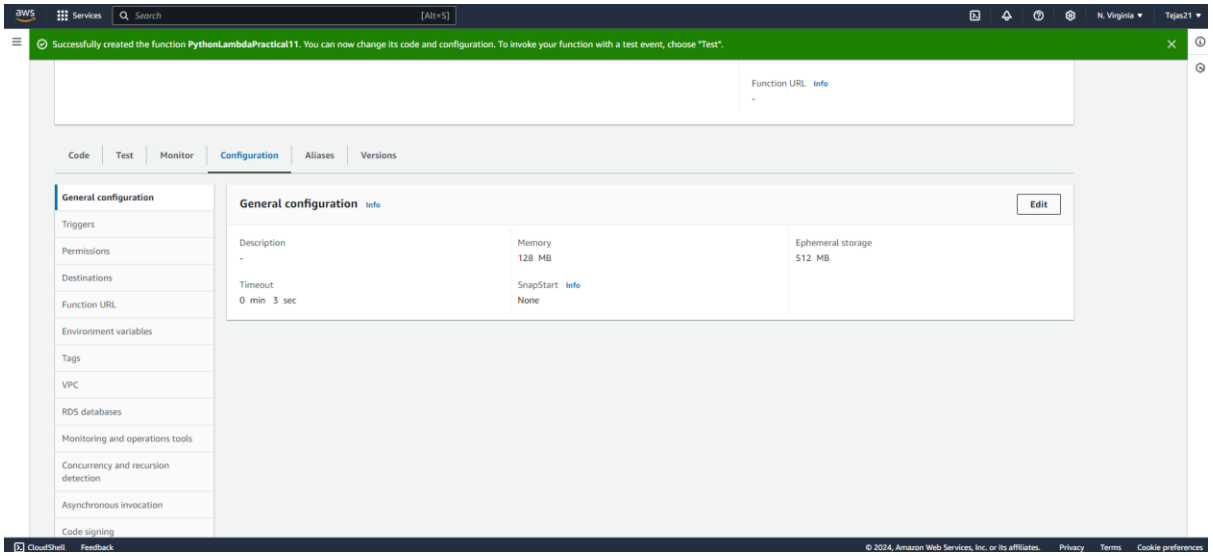


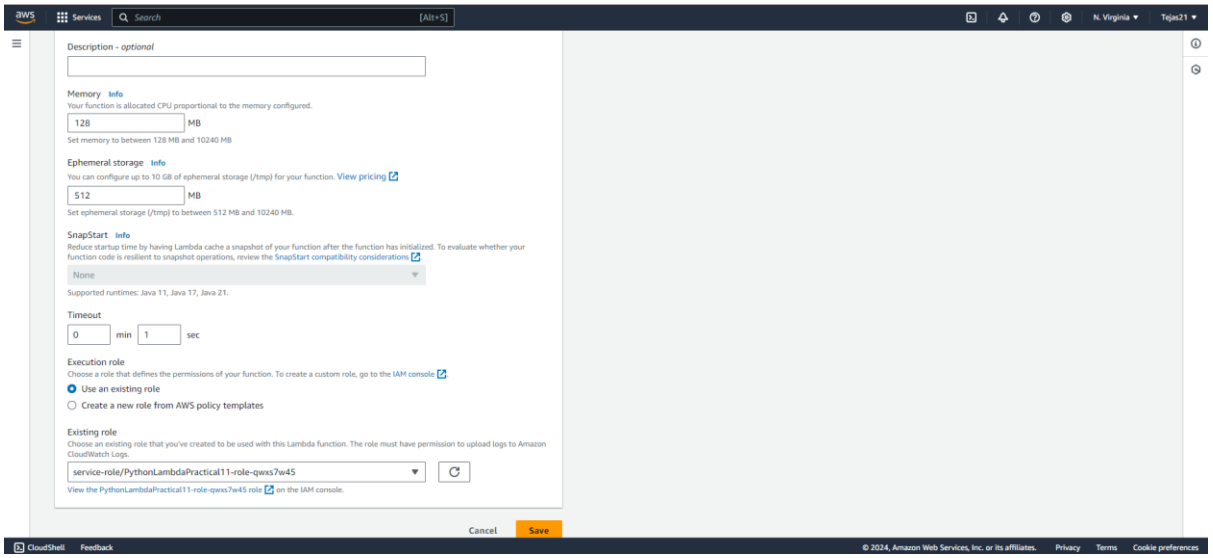
- Function is successfully created



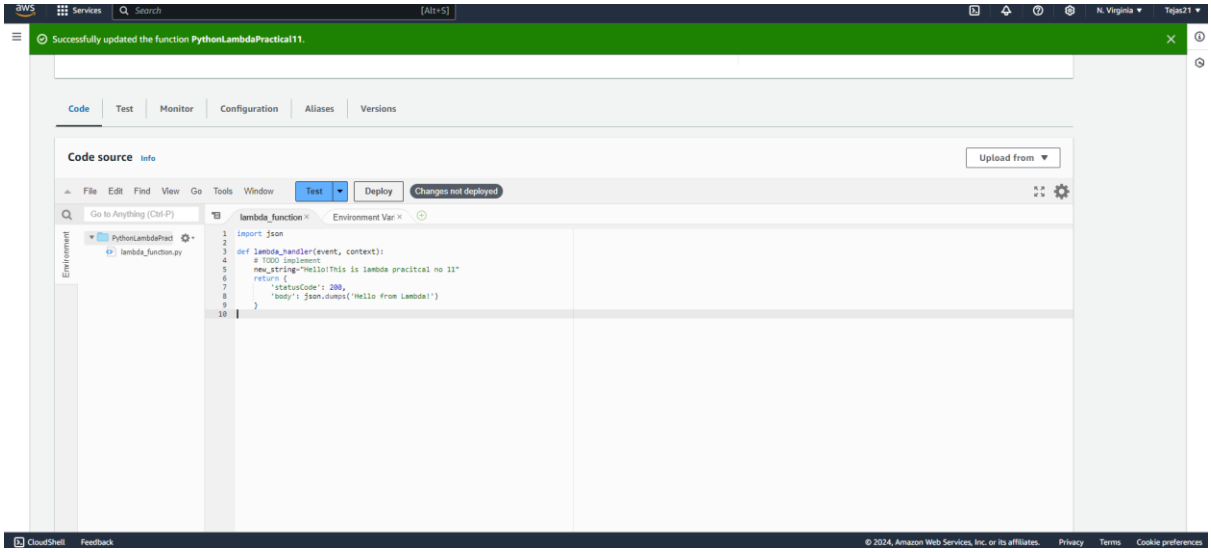


- To change the configuration, open up the Configuration tab and under General Configuration, choose Edit. Here, you can enter a description and change Memory and Timeout. I've changed the Timeout period to 1 sec since that is sufficient for now.

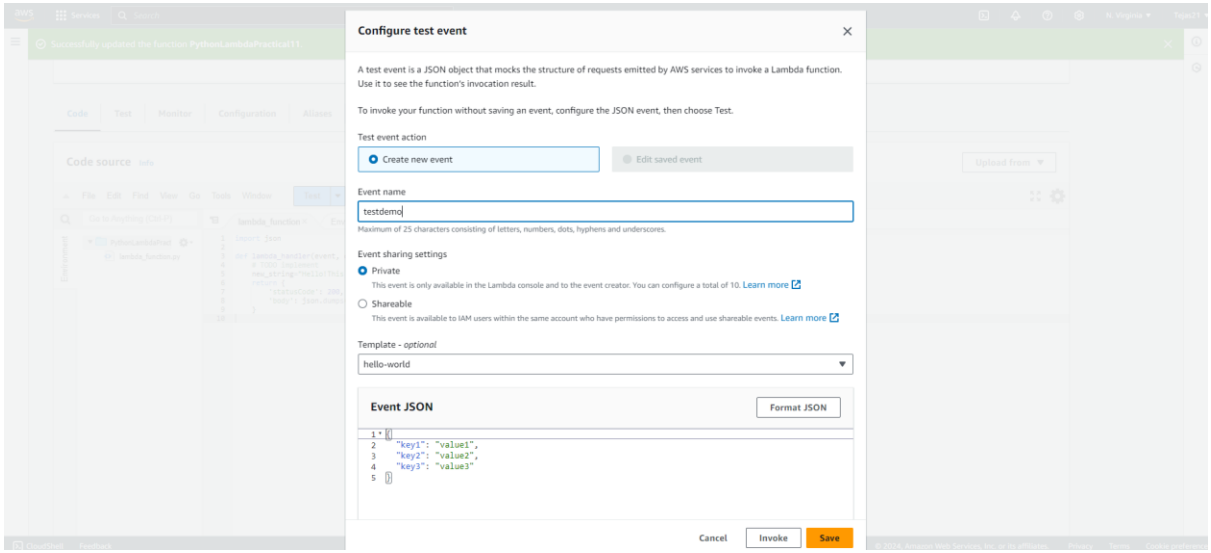




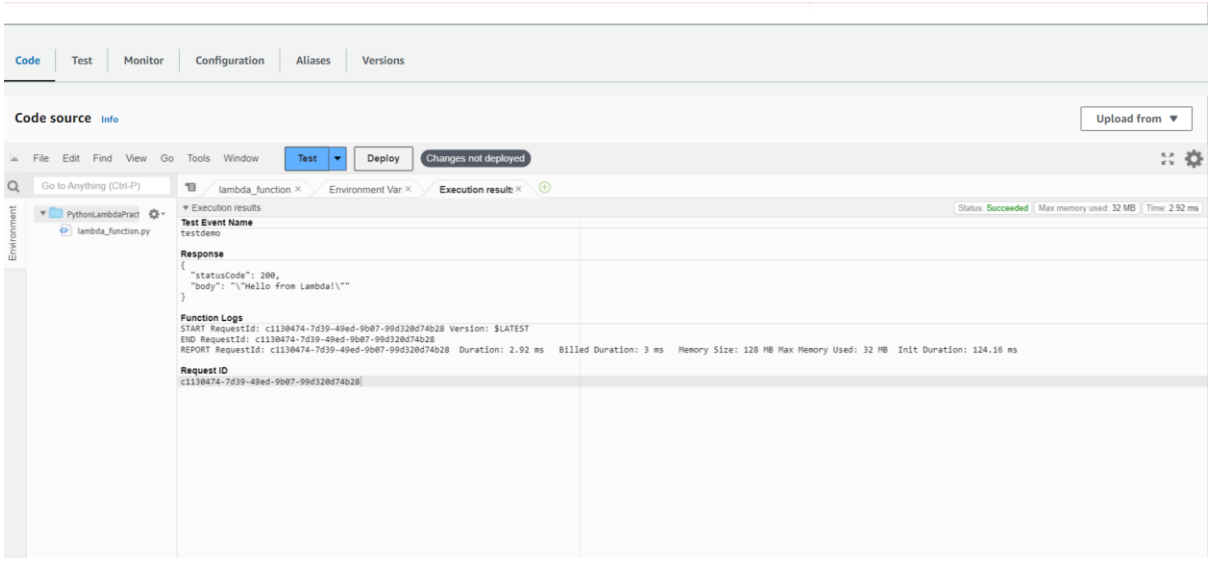
- You can make changes to your function inside the code editor. You can also upload a zip file of your function or upload one from an S3 bucket if needed. Press Ctrl + S to save the file and click Deploy to deploy the changes.



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



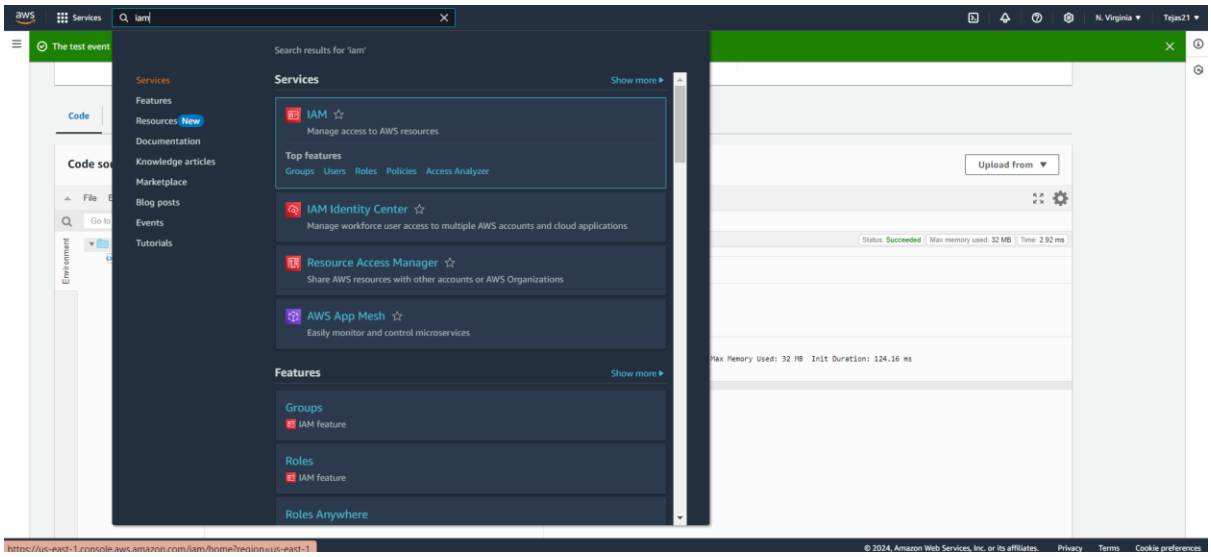
- Click on Test and you should be able to see the results



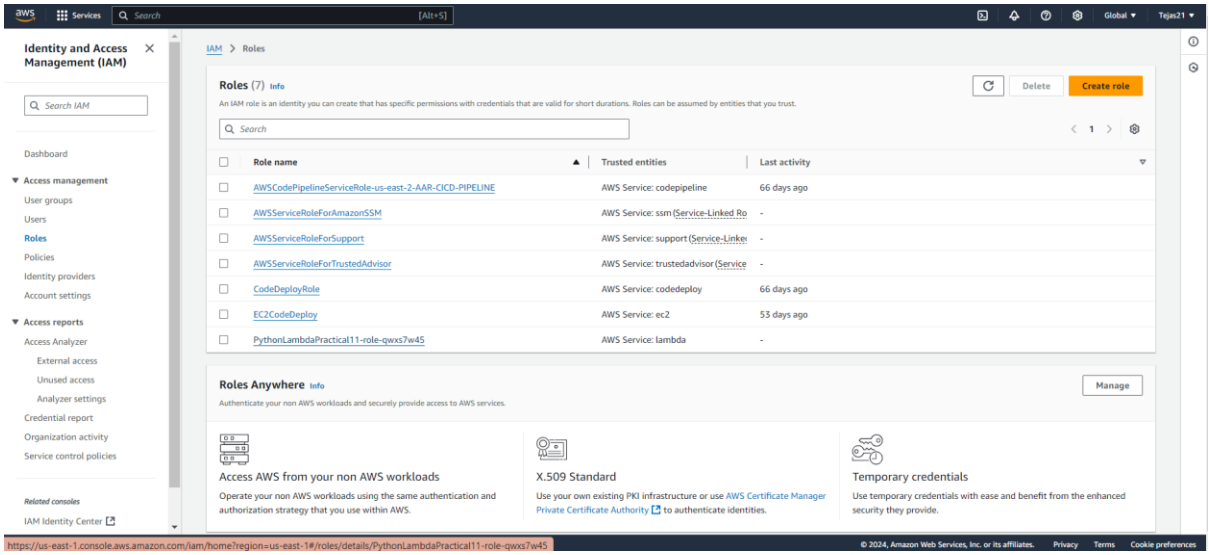
EXPERIMENT NO: - 12

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

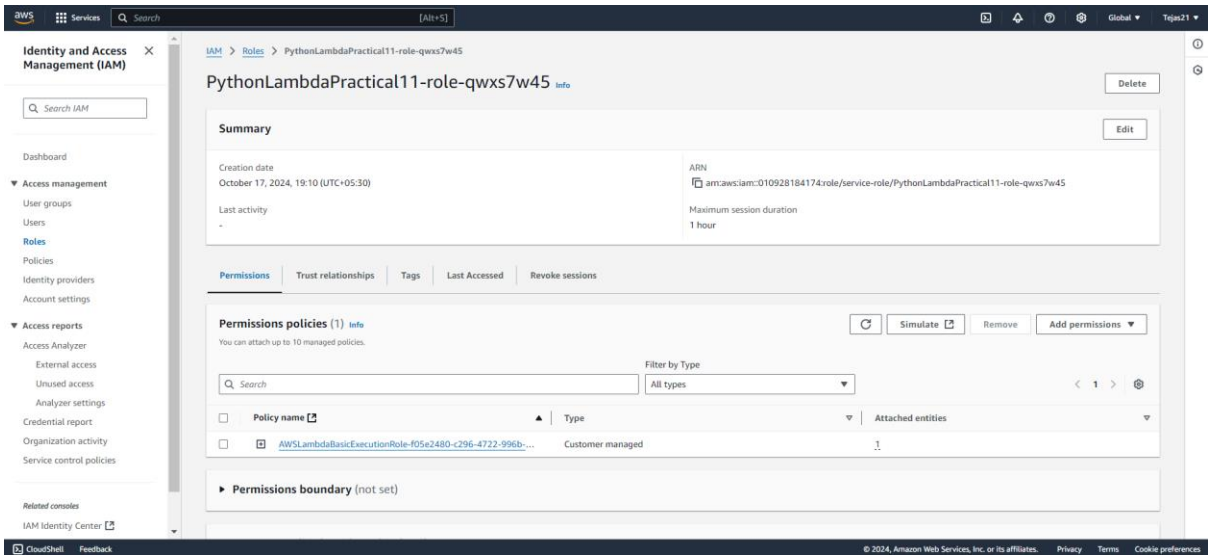
- Select an IAM services



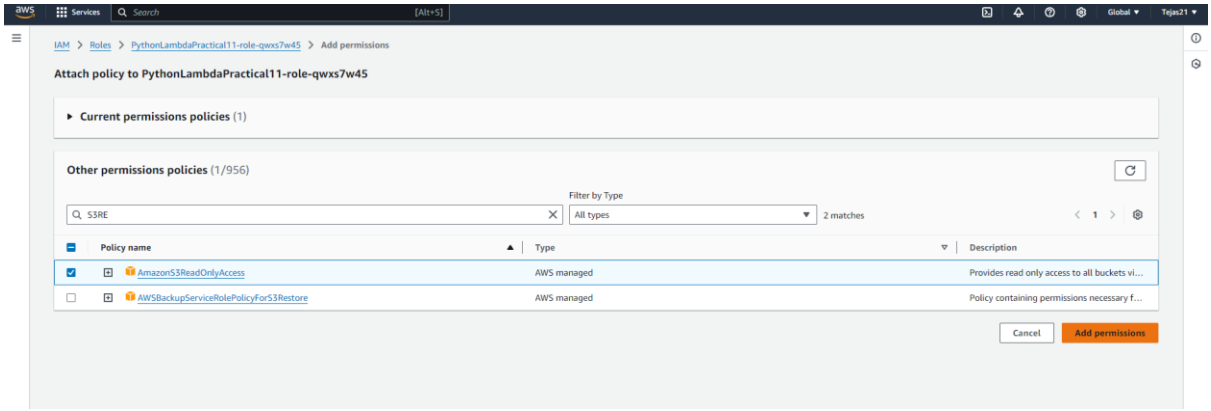
- Open up the IAM Console and under Roles, choose the Role we previously created for the Python Lambda Function (You can find your role name configuration of your Lambda function).



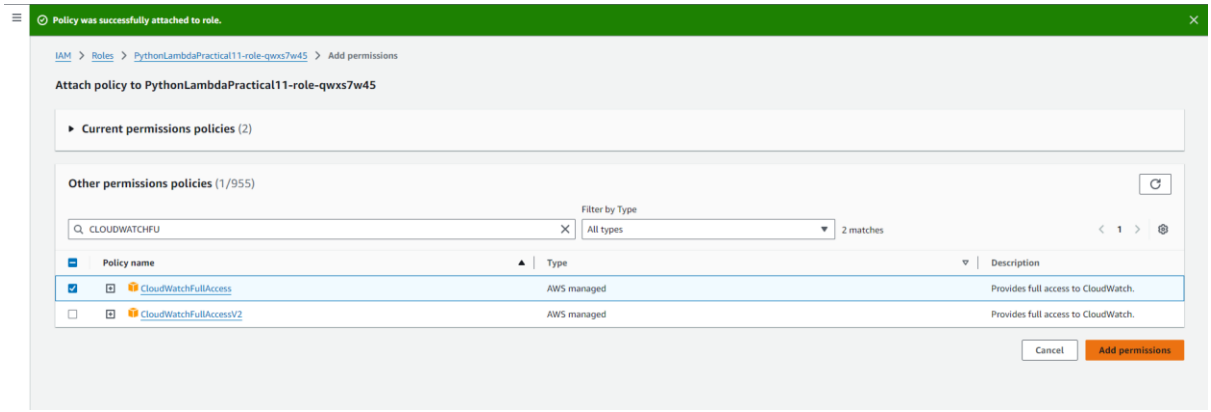
- Under Attach Policies, add S3-ReadOnly and CloudWatchFull permissions to this role.



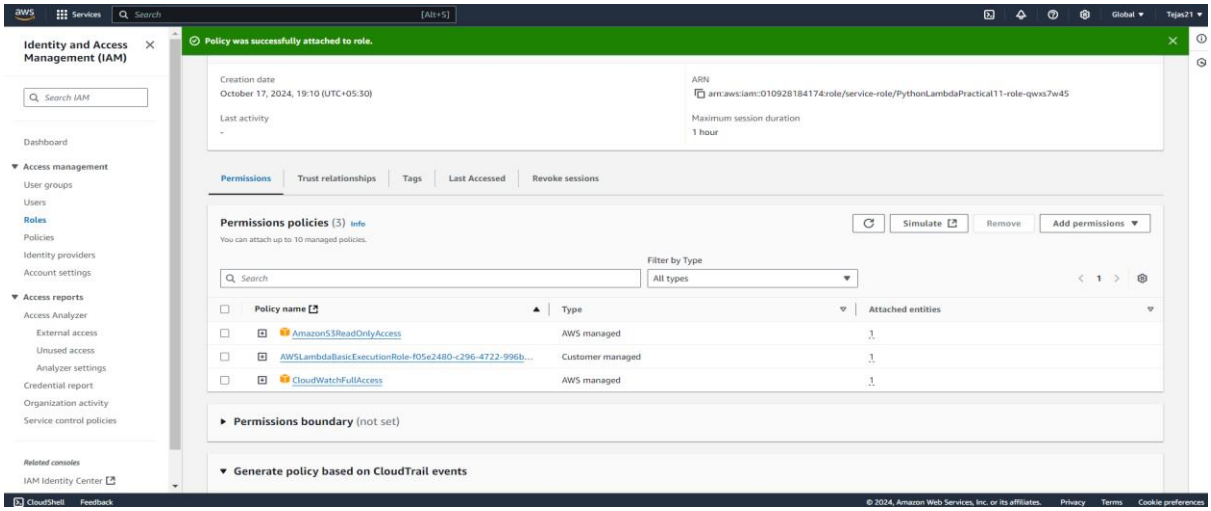
- S3-ReadOnly



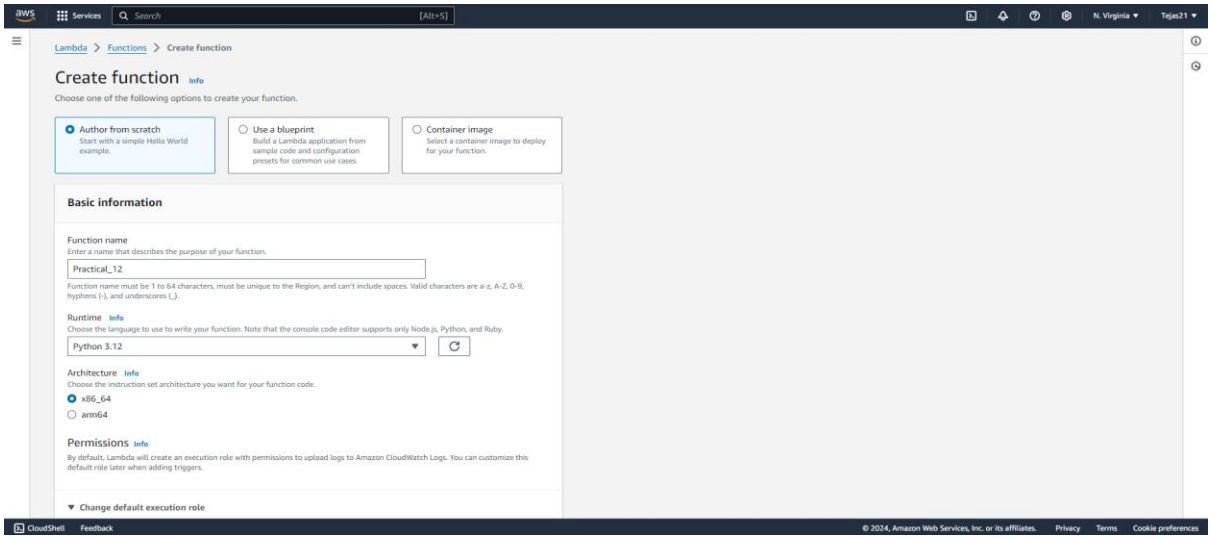
- CloudWatchFull

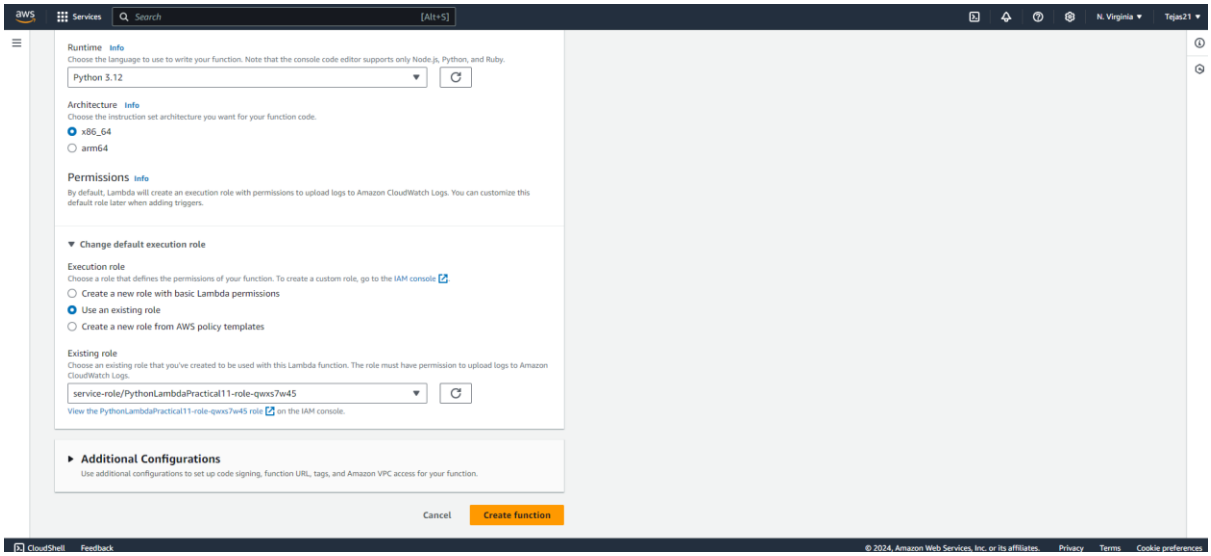


- After successful attachment of policy you will see something like this you will be able to see the updated policies.

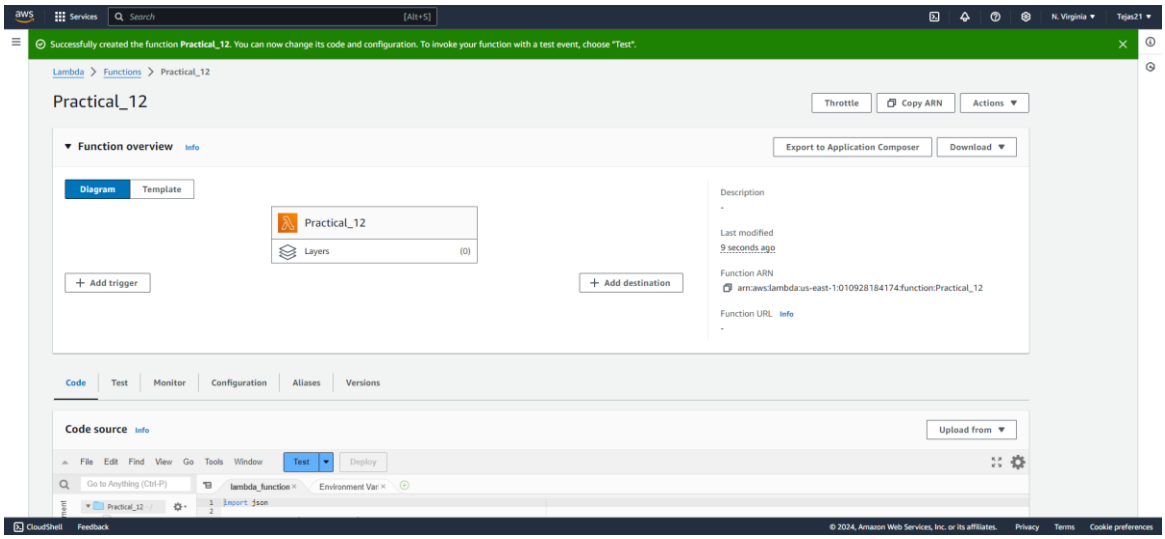


- Open up AWS Lambda and create a new Python function. Under Execution Role, choose the existing role, then select the one which was previously created and to which we just added permissions.

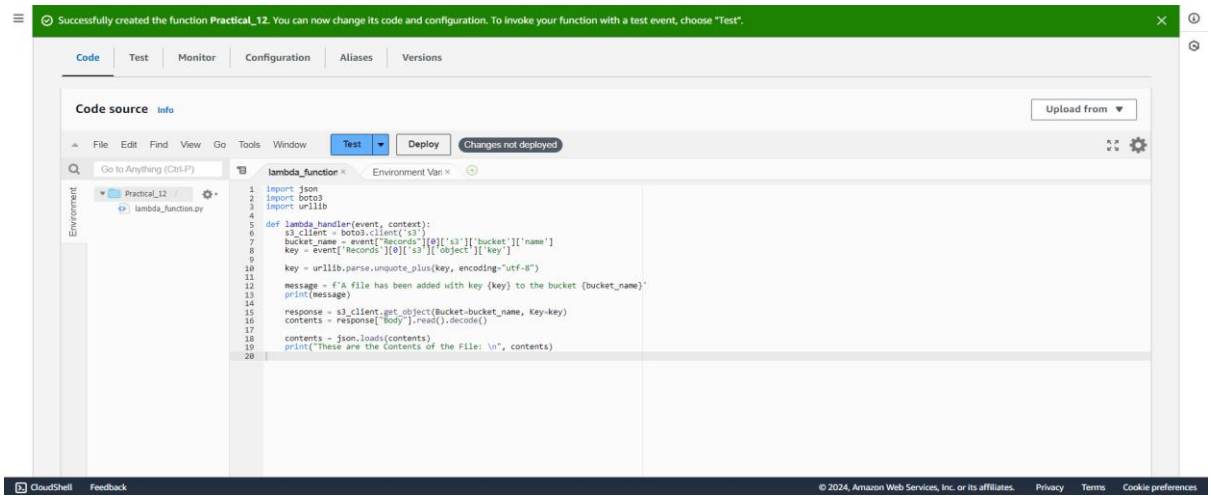




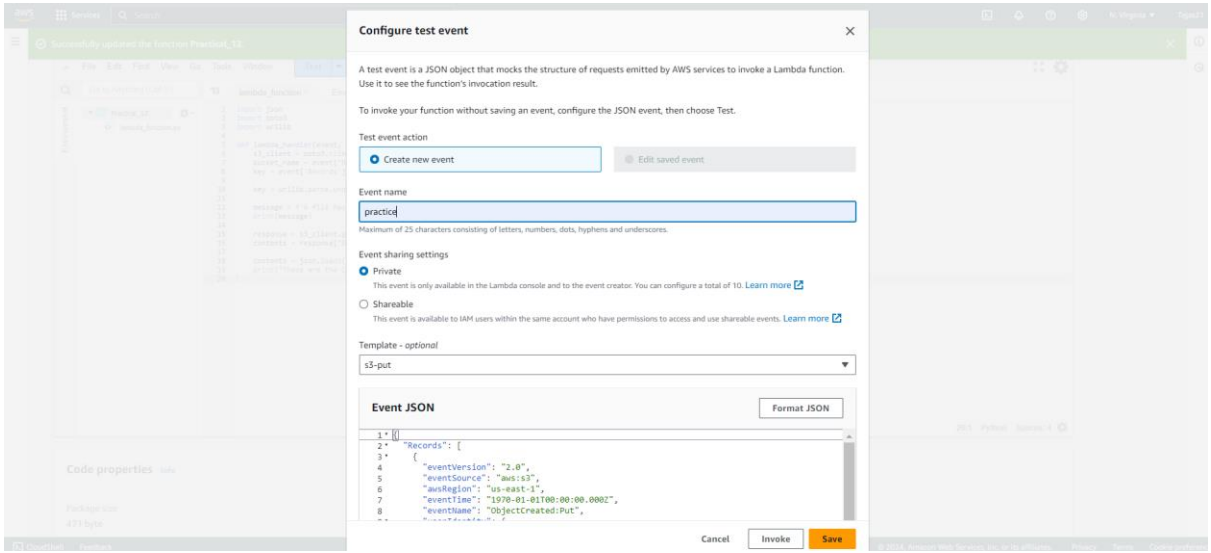
- The function is now successfully created and running



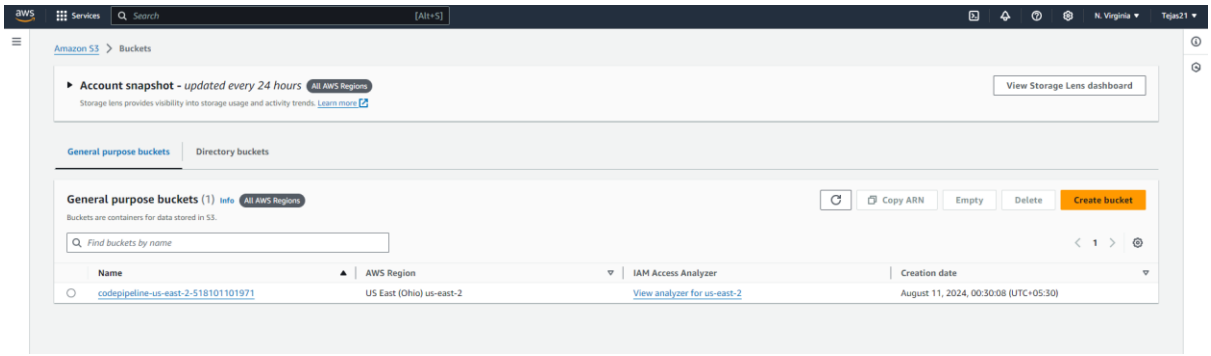
- Make the following changes to the function and click on the deploy button. This code basically logs a message and logs the contents of a JSON file which is uploaded to an S3 Bucket and then deploy the code.



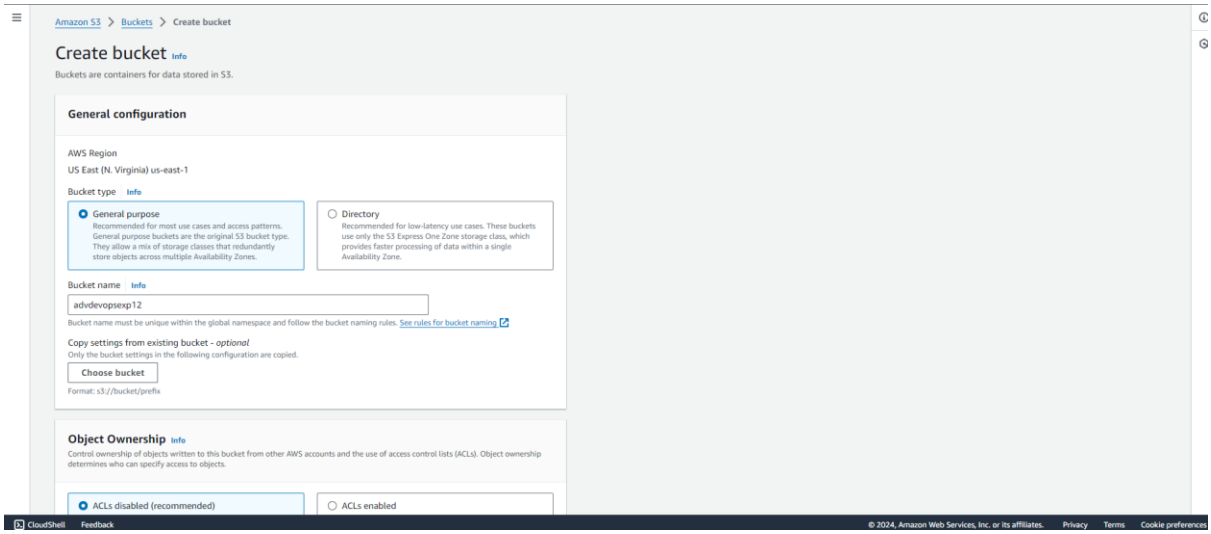
- Click on Test and choose the 'S3 Put' Template.



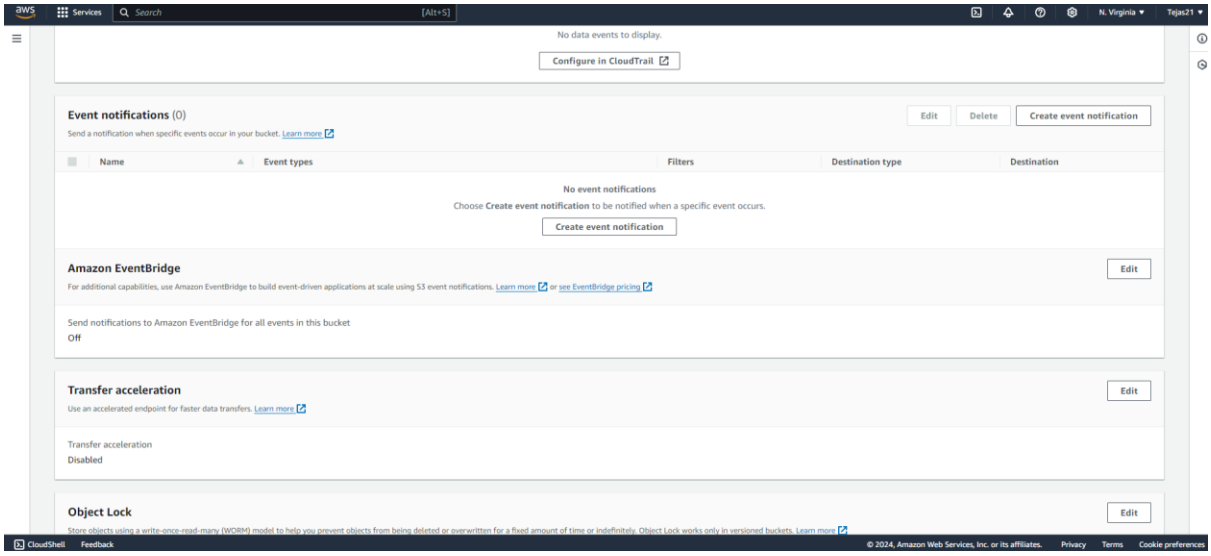
- Open up the S3 Console and create a new bucket.



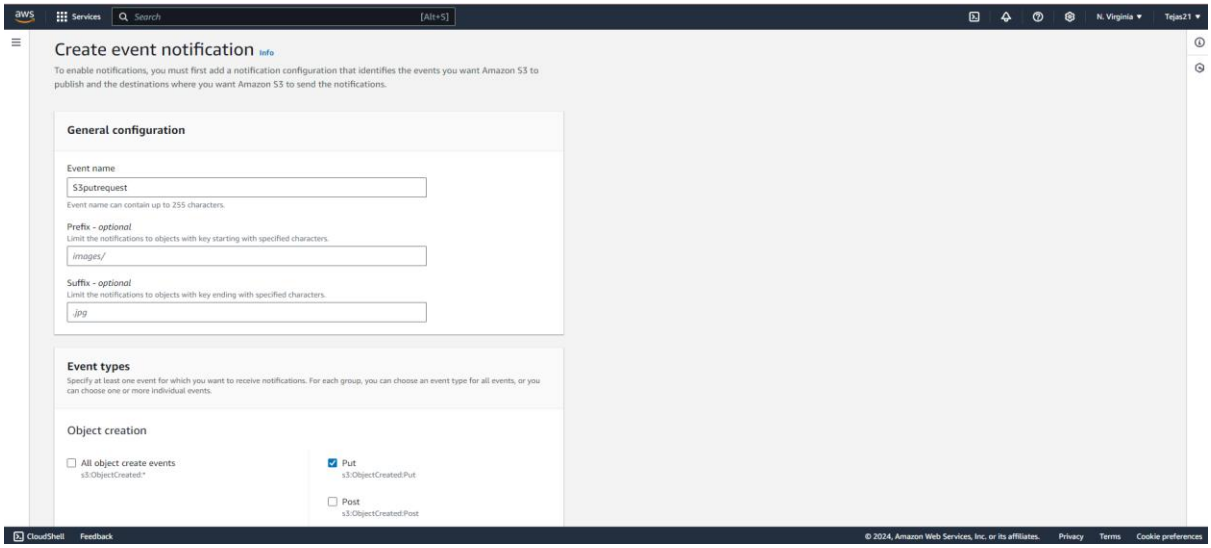
- With all general settings, create the bucket in the same region as the function.



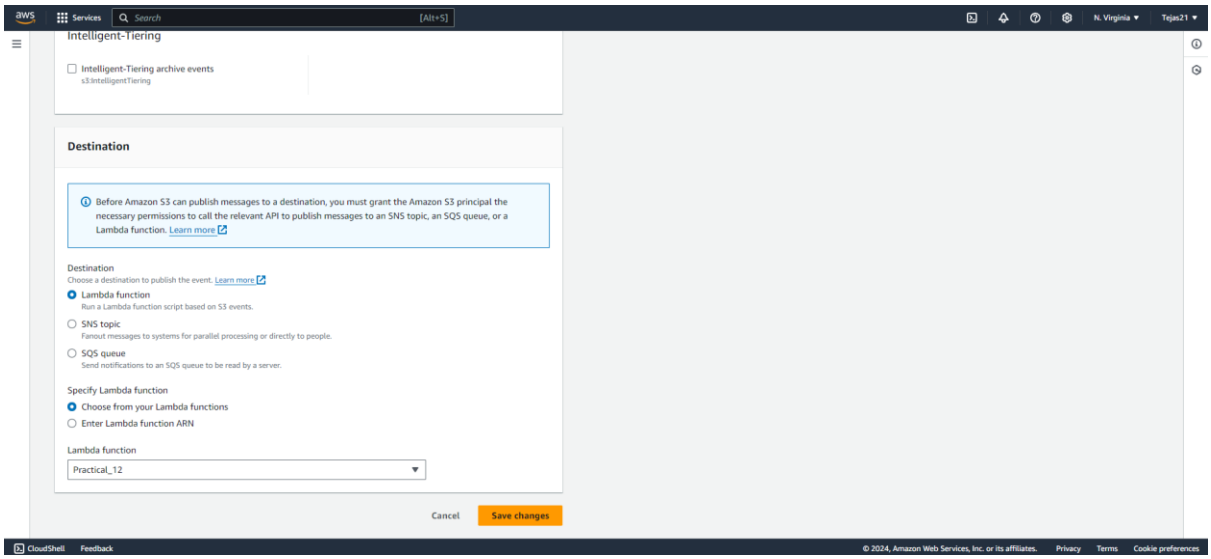
- Click on the created bucket and under properties, look for events. Click on Create Event Notification.



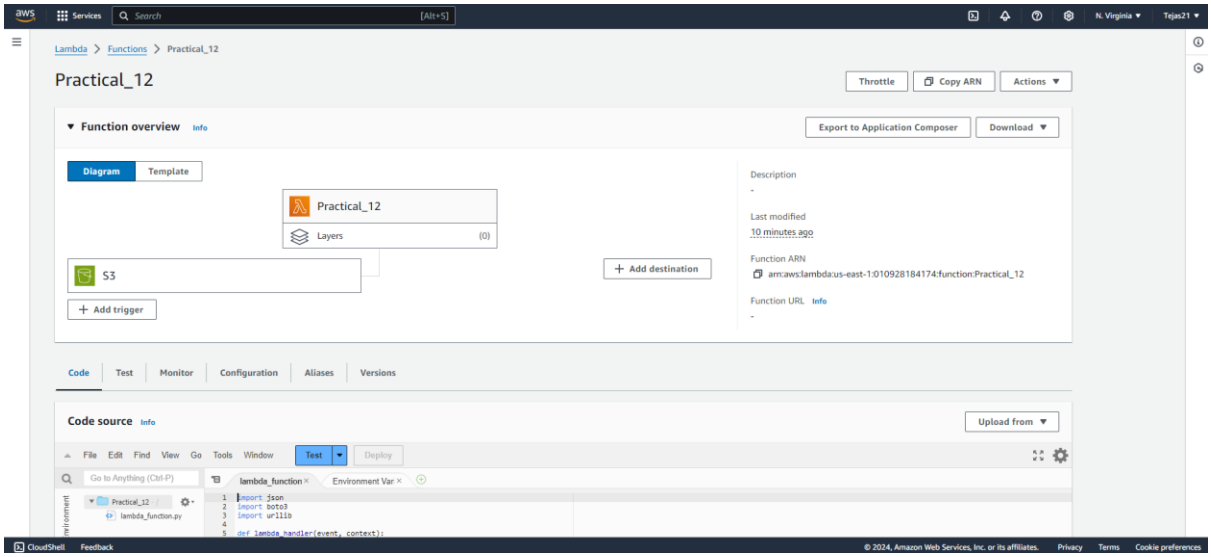
- Mention an event name and check Put under event types.



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

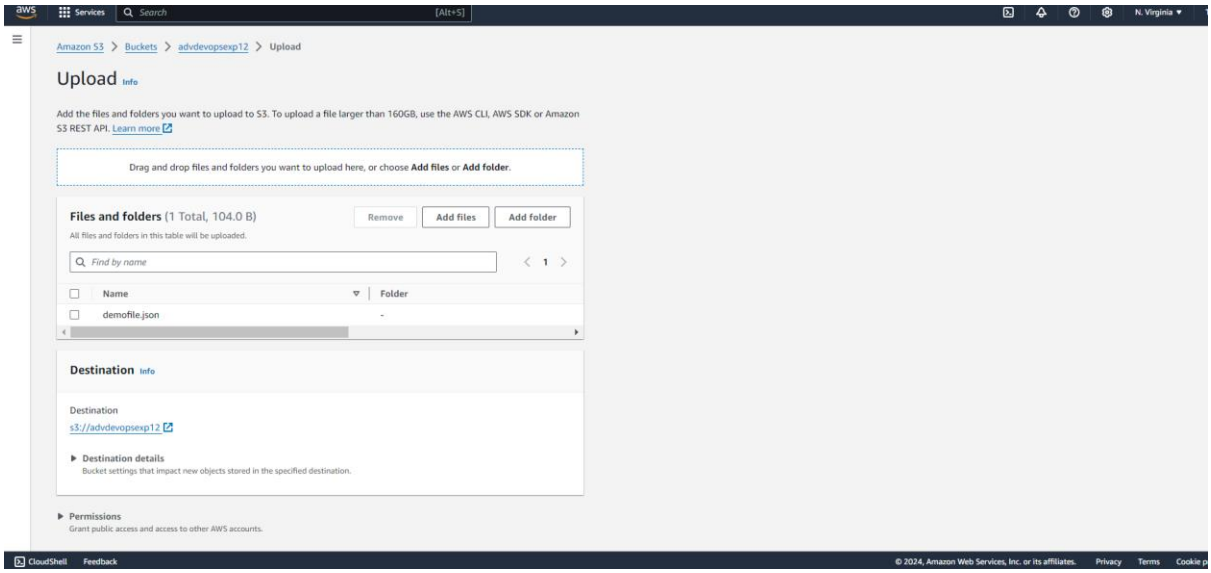


- Refresh the Lambda function console and you should be able to see an S3 Trigger in the overview.

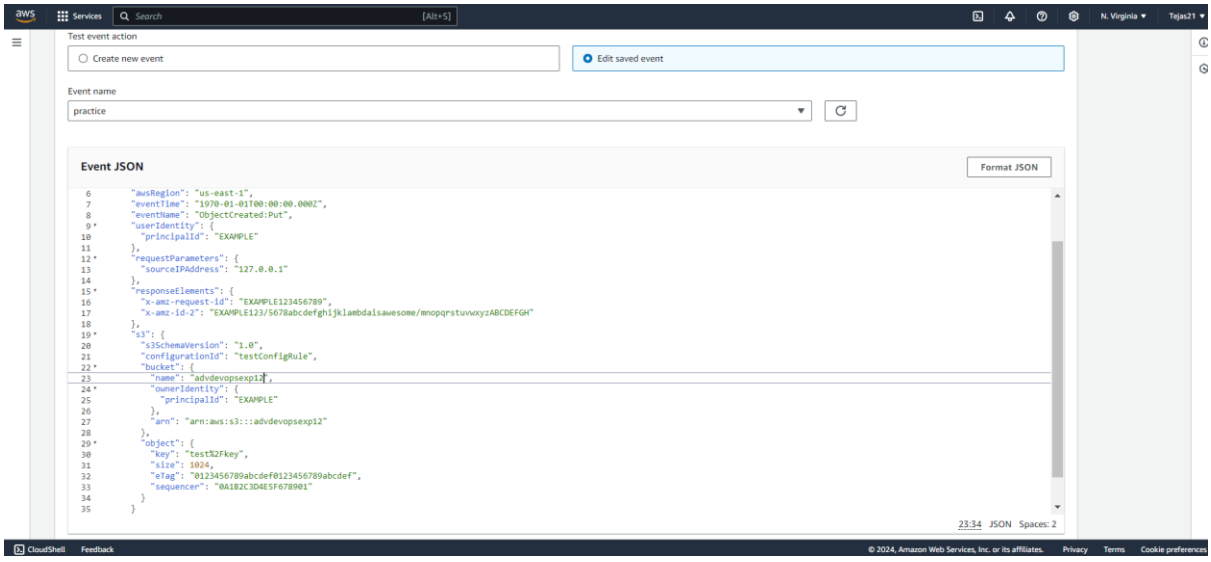


- Now, create a demofile JSON file locally.

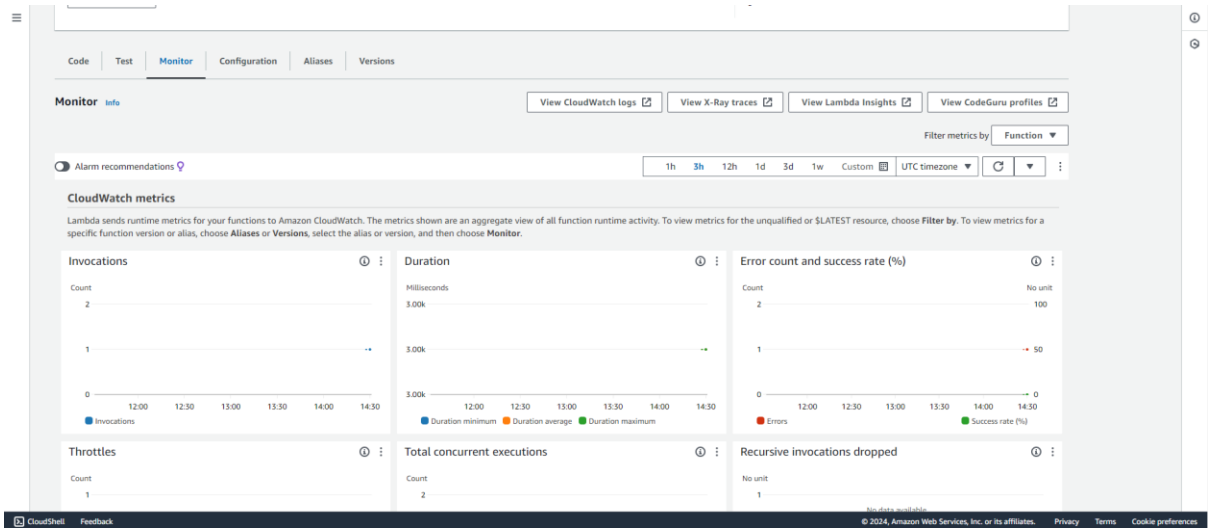
- Go back to your S3 Bucket and click on Add Files to upload a new file. Select the demofile data file from your computer and click Upload.



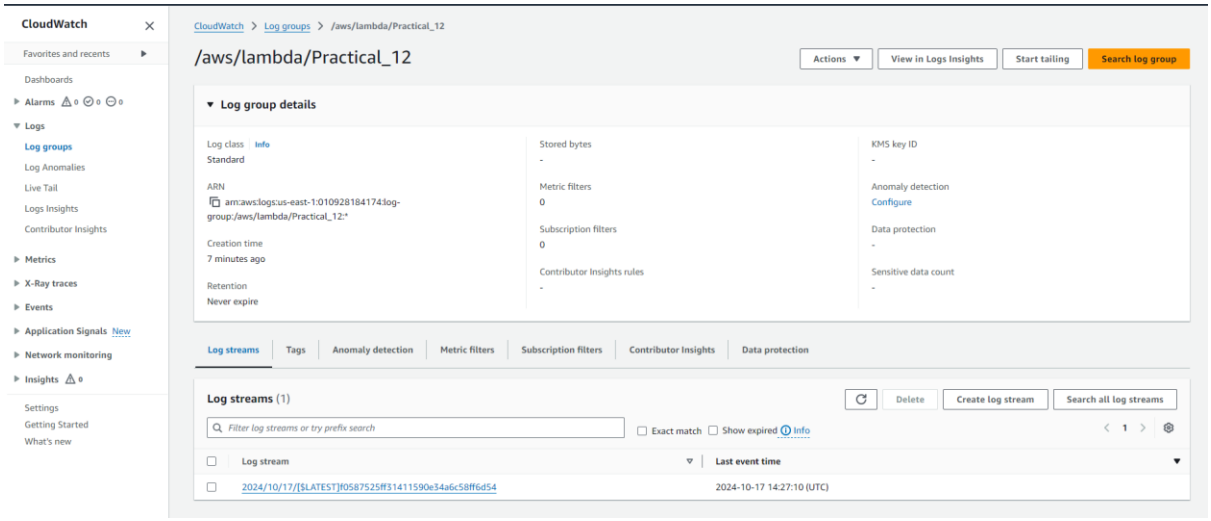
- After this make the necessary changes in the Test configuration file which we created it previously by replacing the Bucket Name and the ARN of Bucket.



➢ Go back to your Lambda function , Refresh it and check the Monitor tab.



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



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