

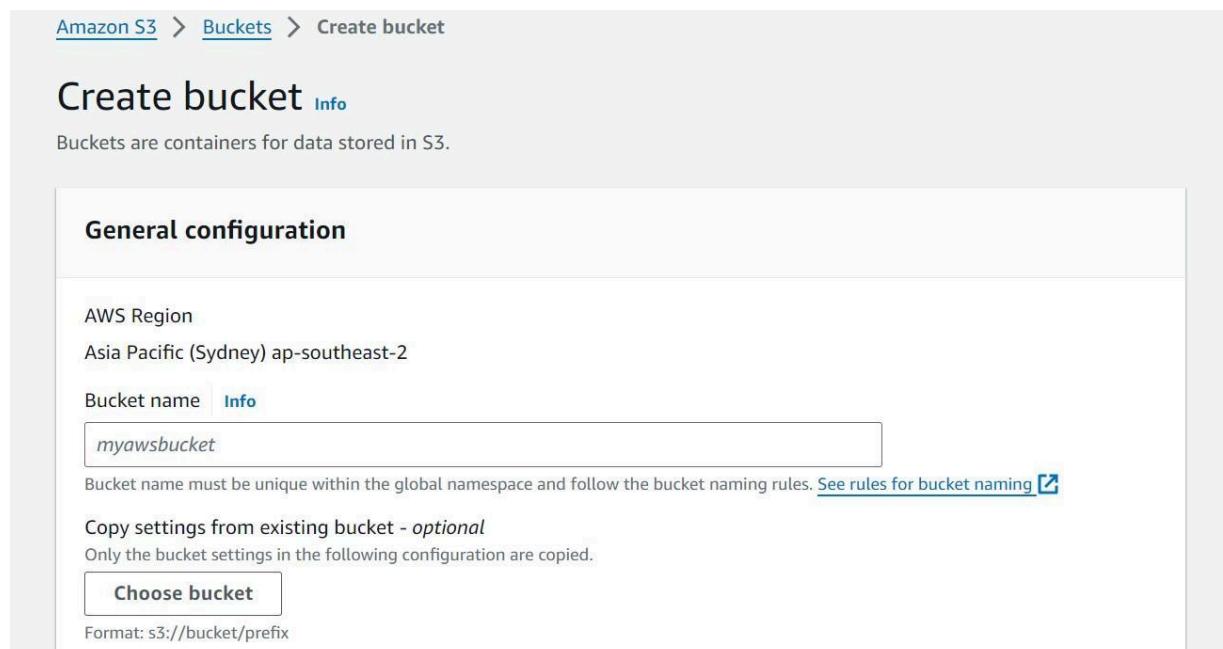
EXPERIMENT NO 1



```
p1.php  x
C: > xampp > htdocs > Adv Devops > p1.php
1: <html>
2: <head>
3: <title>First PHP Program </title>
4: </head>
5: <body>
6: <?php
7: echo "MY NAME IS SPANDAN DEB";
8: ?>
```



Hi! My name is Aryan Dangat



The screenshot shows the "Create bucket" wizard in the AWS S3 console. The top navigation bar shows "Amazon S3 > Buckets > Create bucket". The main title is "Create bucket Info". A sub-instruction says "Buckets are containers for data stored in S3." Below this is a "General configuration" section. Under "AWS Region", it shows "Asia Pacific (Sydney) ap-southeast-2". The "Bucket name" field contains "myawsbucket". A note states "Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)". There is an "Info" link next to the bucket name field. Below the bucket name field is a "Copy settings from existing bucket - optional" section, which says "Only the bucket settings in the following configuration are copied." It includes a "Choose bucket" button and a note about the format: "Format: s3://bucket/prefix".

Aryan Dangat

Screenshot of the AWS S3 console showing the Objects tab for a bucket named 'spandan1'. The bucket contains 0 objects.

Bucket Summary:

- Region: us-east-1
- Storage Class: Standard
- Owner: Aryan Dangat (arn:aws:s3:::spandan1)
- Last modified: 1 hour ago
- Approximate Size: 0 B
- Number of objects: 0

Actions:

- Upload (Orange)
- Create folder
- Open
- Delete
- Actions ▾
- Copy S3 URI
- Copy URL
- Download

Search bar: Find objects by prefix

Table Headers: Name, Type, Last modified, Size, Storage class

No objects message: You don't have any objects in this bucket.

Upload button: Upload

Success Message:

☰ **Upload succeeded**
View details below.

Destination	Succeeded	Failed
s3://spandan1	1 file, 300.0 B (100.00%)	0 files, 0 B (0%)

File and Folders Tab: Configuration

File and Folders List:

- Files and folders (1 Total, 300.0 B)

Search bar: Find by name

Table Headers: Name, Folder, Type, Size, Status, Error

index.html	AdvD/	text/html	300.0 B	Succeeded	-
------------	-------	-----------	---------	-----------	---

Aryan Dangat

EXPERIMENT NO 2

NAME-Aryan Dangat

CLASS-D15A

ROLLNO- 12

AIM-

To Build Your Application using AWS Code Build and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS Code Deploy.

Trusted entity type

<input checked="" type="radio"/> AWS service Allow AWS services like EC2, Lambda, or others to perform actions in this account.	<input type="radio"/> AWS account Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.	<input type="radio"/> Web identity Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
<input type="radio"/> SAML 2.0 federation Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.	<input type="radio"/> 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

Trusted entity type

<input checked="" type="radio"/> AWS service Allow AWS services like EC2, Lambda, or others to perform actions in this account.	<input type="radio"/> AWS account Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.	<input type="radio"/> Web identity Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
<input type="radio"/> SAML 2.0 federation Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.	<input type="radio"/> 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

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**

Role details

Role name

Enter a meaningful name to identify this role.

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.

Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following char

Step 1: Select trusted entities

⌚ Role CodeDeployRole1 created.

[IAM](#) > [Roles](#) > [CodeDeployRole1](#)

CodeDeployRole1 [Info](#)

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

Summary

Creation date

August 11, 2024, 22:43 (UTC+05:30)

ARN

arn:aws:iam::0109281845:

Last activity

-

Maximum session duration

1 hour

[EC2](#) > [Instances](#) > Launch an instance

⌚ Success

Successfully initiated launch of instance ([i-083ec36bb1e861e50](#))

[Launch log](#)

Next Steps

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

< 1 2 3 4 5 6 >

[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](#)

[Create EBS snapshot policy](#)

Create a policy that automates the creation, retention, and deletion of EBS snapshots

[Create EBS snapshot policy](#)

Create application

Application configuration

Application name

Enter an application name

100 character limit

Compute platform

Choose a compute platform



Tags

Rate this service console?

[Alt+S]



[Developer Tools](#) > [CodeDeploy](#) > [Applications](#) > AD-CICD

AD-CICD

[Notify](#) [Delete](#)

Application details

Name

AD-CICD

Compute platform

EC2/On-premises

[Deployments](#)

[Deployment groups](#)

[Revisions](#)

Deployment groups

[View details](#)

[Edit](#)

[Create deployment](#)



Name	Status	Last attempted de...	Last successful de...	Trigger count
AD-CICD-DP	-	-	-	0

S | Services | Search [Alt+S] | X | A | ? | 🌐 | Sydney ▾ | /

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

Success Deployment group created

Developer Tools > CodeDeploy > Applications > AD-CICD > AD-CICD-DP

AD-CICD-DP

Edit Delete Create deployment

Deployment group details

Deployment group name	Application name	Compute platform
AD-CICD-DP	AD-CICD	EC2/On-premises
Deployment type	Service role ARN	Deployment configuration
In-place	arn:aws:iam::637423278915:role/CodeDeploy	CodeDeployDefault.AllAtOnce
Rollback enabled	Agent update scheduler	
False	Learn to schedule update in AWS Systems Manager	

Environment configuration: Amazon EC2 instances

Step 1: Choose pipeline settings

Pipeline settings

Pipeline name
AD-CICD

Pipeline type
V2

Execution mode
QUEUED

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

Service role name
AWSCodePipelineServiceRole-ap-southeast-2-AD-CICD

Step 2: Add source stage

Source action provider

Source action provider
GitHub (Version 2)
OutputArtifactFormat
CODE_ZIP
DetectChanges
false
ConnectionArn
arn:aws:codeconnections:ap-southeast-2:637423278915:connection/376a6087-e520-40b3-99e9-dbd5e655ddbc
FullRepositoryId
Aryan2052/aws_cicd_pipeline_codedeploy
Default branch
main

Step 3: Add build stage

Build action provider

Build stage
No build

Step 4: Add deploy stage

Deploy action provider

Deploy action provider
AWS CodeDeploy
ApplicationName
AD-CICD
DeploymentGroupName
AD-CICD-DP
Configure automatic rollback on stage failure
Disabled

Success
Congratulations! The pipeline AD-CICD has been created.

Create a notification rule for this pipeline

Developer Tools > CodePipeline > Pipelines > AD-CICD

AD-CICD

Pipeline type: V2 Execution mode: QUEUED

Source In progress

Source
In progress - Just now
View details

Disable transition

Deploy Didn't Run
Start rollback

Notify ▾ Edit Stop execution Clone pipeline Release change

Inbound security group rules successfully modified on security group (sg-0d1a8bb976dd19e86 | default)

Details

EC2 > Security Groups > sg-0d1a8bb976dd19e86 - default

sg-0d1a8bb976dd19e86 - default

Action

Details			
Security group name	Security group ID	Description	VPC ID
default	sg-0d1a8bb976dd19e86	default VPC security group	ypc-097d5c311487d2b
Owner	Inbound rules count	Outbound rules count	
637423278915	3 Permission entries	1 Permission entry	

Inbound rules Outbound rules Tags

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

Connect to GitHub

GitHub connection settings Info

Connection name

GitHub Apps

GitHub Apps create a link for your connection with GitHub. Install a new app and save this connection.



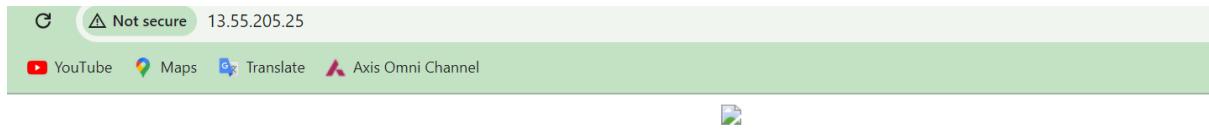
or

[Install a new app](#)

► Tags - optional

[Connect](#)

Edit inbound rules <small>Info</small>						
Inbound rules <small>Info</small>						
Security group rule ID	Type <small>Info</small>	Protocol <small>Info</small>	Port range	Source <small>Info</small>	Description - optional <small>Info</small>	
sgr-0a518631208bbdfc1	All traffic	All	All	Custom	<input type="text"/> sg- 0fa33ce36b0186adb	Delete
-	HTTP	TCP	80	Any...	<input type="text"/> 0.0.0.0	Delete
-	SSH	TCP	22	Any...	<input type="text"/> 0.0.0.0	Delete



Congratulations!!!

Welcome to AAR SOURCE| Ram Hemareddy

This application was deployed using AWS CodeDeploy.

For next steps, read the [AWS CodeDeploy Documentation](#).

Aryan Dangat

D15A 12

EXPERIMENT NO 3

Create 3 EC2 instances

Name: master

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 24.04, amd64...read more
ami-0e86e20dae9224db8

Virtual server type (instance type): t2.medium

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 30 GiB

Launch instance

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
<input type="checkbox"/>	w1	i-0ea763696bab632be	Running	t2.medium	2/2 checks passed	View alarms	us-east-1a
<input type="checkbox"/>	w2	i-0c2ee507c438a918e	Running	t2.medium	Initializing	View alarms	us-east-1a
<input checked="" type="checkbox"/>	master	i-006b722fdf7abf434	Running	t2.medium	2/2 checks passed	View alarms	us-east-1a
<input type="checkbox"/>	aryandangat	i-00dd34f0bbf8c456f	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a

Connect to instance Info

Connect to your instance i-03de8ab0e6260426b (master) using any of these options

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

Instance ID
i-03de8ab0e6260426b (master)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is advd.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 "advd.pem"
4. Connect to your instance using its Public DNS:
ec2-52-86-6-228.compute-1.amazonaws.com

Example:
ssh -i "advd.pem" ubuntu@ec2-52-86-6-228.compute-1.amazonaws.com

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

```
ubuntu@ip-172-31-21-159: ~
Student@VESIT512-21 MINGW64 ~ (student)
$ cd Downloads/
Student@VESIT512-21 MINGW64 ~/Downloads (student)
$ ssh -i "xyz.pem" ubuntu@ec2-54-145-14-165.compute-1.amazonaws.com
The authenticity of host 'ec2-54-145-14-165.compute-1.amazonaws.com (54.145.14.165)' can't be established.
ED25519 key fingerprint is SHA256:9KsUnnHJd5y00WcRLpn7gwA3F6qL6BjecyhwpbpeUB4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-145-14-165.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 Mon Sep 16 07:36:01 UTC 2024
System load: 0.35          Processes:           116
Usage of /: 20.7% of 7.57GB   Users logged in:      0
Memory usage: 5%            IPv4 address for eth0: 172.31.21.159
```

Prepare nodes

Use sudo apt-get update && sudo apt-get upgrade -y to update the package manager

Disable Swap

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

Load necessary kernel modules

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

```
ubuntu@ip-172-31-21-159:~$ sudo apt-get upgrade -y  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
Calculating upgrade... Done  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
ubuntu@ip-172-31-21-159:~$ sudo swapoff -a  
ubuntu@ip-172-31-21-159:~$ sudo swapoff -a  
  sudo sed -i '/ swap / s/^/#/' /etc/fstab  
ubuntu@ip-172-31-21-159:~$ cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf  
overlay  
br_netfilter  
EOF  
sudo modprobe overlay  
sudo modprobe br_netfilter  
overlay  
br_netfilter
```

Configure systemctl settings for Kuber

```
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf  
net.bridge.bridge-nf-call-ip6tables = 1  
net.bridge.bridge-nf-call-iptables = 1  
EOF
```

```
sudo sysctl -- system
```

```
ubuntu@ip-172-31-23-85:~$ cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf  
net.bridge.bridge-nf-call-ip6tables = 1  
net.bridge.bridge-nf-call-iptables = 1  
EOF  
sudo sysctl --system  
net.bridge.bridge-nf-call-ip6tables = 1  
net.bridge.bridge-nf-call-iptables = 1  
* Applying /usr/lib/sysctl.d/10-apparmor.conf ...  
* Applying /etc/sysctl.d/10-console-messages.conf ...  
* Applying /etc/sysctl.d/10-ipv6-privacy.conf ...  
* Applying /etc/sysctl.d/10-kernel-hardening.conf ...  
* Applying /etc/sysctl.d/10-magic-sysrq.conf ...  
* Applying /etc/sysctl.d/10-map-count.conf ...  
* Applying /etc/sysctl.d/10-network-security.conf ...  
* Applying /etc/sysctl.d/10-ptrace.conf ...  
* Applying /etc/sysctl.d/10-zero-page.conf ...  
* Applying /etc/sysctl.d/50-cloudimg-settings.conf ...  
* Applying /usr/lib/sysctl.d/50-pid-max.conf ...
```

Installing Docker

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

```
ubuntu@ip-172-31-23-85:~$ sudo apt-get update
sudo apt-get install -y apt-transport-https ca-certificates
curl software-properties-common curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu
$(lsb_release -cs) stable"
sudo apt-get update
sudo apt-get install -y docker-ce docker-ce-cli containerd.io
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Fetched 126 kB in 0s (358 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
```

Configure Docker For Kubernetes

```
cat <<EOF | sudo tee /etc/docker/daemon.json
```

```
{
  "exec-opts": ["native.cgroupdriver=systemd"], "log-driver": "json-file",
  "log-opt": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
```

```
EOF
```

```
sudo systemctl restart docker
```

```

ubuntu@ip-172-31-23-85:~$ cat <<EOF | sudo tee /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"], "log-driver": "json-file",
  "log-opt": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF
sudo systemctl restart docker

{
  "exec-opts": ["native.cgroupdriver=systemd"], "log-driver": "json-file",
  "log-opt": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
ubuntu@ip-172-31-23-85:~$ |

```

Add Kubernetes APT repository

```

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

```

```

ubuntu@ip-172-31-27-12:~$ curl -fssl https://pkgs.k8s.io/core:/stable:/v1.29/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-archive-keyring.gpg
echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-archive-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /' | sudo tee /etc/apt/sources.list.d/kubernetes.list
sudo apt-get update -y
sudo apt-get install -y kubelet="1.29.0-*" kubectl="1.29.0-*" kubeadm="1.29.0-*"
sudo apt-get update -y
sudo apt-get install -y jq
deb [signed-by=/etc/apt/keyrings/kubernetes-archive-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:6 https://prod-cdn.packages.k8s.io/repositories/isv/kubernetes:/addons:/cri-o:/prerelease:/main/deb InRelease
Get:7 https://prod-cdn.packages.k8s.io/repositories/isv/kubernetes:/core:/stable:/v1.29/deb InRelease [1189 B]
Get:8 https://prod-cdn.packages.k8s.io/repositories/isv/kubernetes:/core:/stable:/v1.29/deb Packages [14.0 kB]
Fetched 15.1 kB in 1s (20.2 kB/s)
Reading package lists... Done
W: https://download.docker.com/linux/ubuntu/dists/jammy/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the
y(8) for details.
Reading package lists... Done
Building dependency tree... Done

```

Install kubeadm, kubelet, and kubectl:

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

```
sudo apt-mark hold kubelet kubeadm kubectl
```

```
ubuntu@ip-172-31-27-12:~$ sudo apt-get install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  ebttables socat
Use 'sudo apt autoremove' to remove them.
The following packages will be upgraded:
  kubeadm kubectl kubelet
3 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
Need to get 40.7 MB of archives.
After this operation, 1758 kB of additional disk space will be used.
Get:1 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1
```

Initialize the Kubernetes cluster on master node

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

```
sudo kubeadm init --cri-socket unix:///var/run/crio/crio.sock
```

```
ubuntu@ip-172-31-27-12:~$ nano kubeadm-config.yaml
ubuntu@ip-172-31-27-12:~$ sudo kubeadm init --config kubeadm-config.yaml
W0915 10:18:14.121956    23054 initconfiguration.go:312] error unmarshaling configuration schema.GroupVersionKind{Group:"k8s.io/kubernetes", Version:"v1", Kind:"NodeConfig"}: strict decoding error: unknown field "criSocket"
Found multiple CRI endpoints on the host. Please define which one do you wish to use by setting the 'criSocket' field in the NodeConfig.
To see the stack trace of this error execute with --v=5 or higher
ubuntu@ip-172-31-27-12:~$ sudo kubeadm init --cri-socket unix:///var/run/crio/crio.sock
I0915 10:19:09.235316    23090 version.go:256] remote version is much newer: v1.31.0; falling back to: stable
[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'
```

To setup kubectl on master node

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

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

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

```
ubuntu@ip-172-31-27-12:~$ 
  mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

Install pod network add-on

```
kubectl apply -f  
https://raw.githubusercontent.com/coreos/flannel/master/Documentation/ku  
be-flannel.yml
```

```
ubuntu@ip-172-31-27-12:~$ kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.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-172-31-27-12:~$ client [local] send disconnect: Connection reset by peer
```

Join Worker Nodes

```
sudo kubeadm join 172.31.27.12:6443 --token zerrkd.zyzasxm2e69grelo \  
--discovery-token-ca-cert-hash  
sha256:c7e56b4e0a3eaf45f6c55c19e338a0e68ad04a2cab623808ddebcbd74e7bbce  
d \  
--cri-socket unix:///var/run/crio/crio.sock --v=5
```

```
ubuntu@ip-172-31-30-123:~$ sudo kubeadm join 172.31.27.12:6443 --token zerrkd.zyzasxm2e69grelo \  
--discovery-token-ca-cert-hash sha256:c7e56b4e0a3eaf45f6c55c19e338a0e68ad04a2cab623808ddebcbd74e7bbced \  
--cri-socket unix:///var/run/crio/crio.sock --v=5  
I0915 10:57:30.867146 17284 join.go:413] [preflight] found NodeName empty; using OS hostname as NodeName  
[preflight] Running pre-flight checks  
I0915 10:57:30.867225 17284 preflight.go:93] [preflight] Running general checks  
I0915 10:57:30.867256 17284 checks.go:280] validating the existence of file /etc/kubernetes/kubelet.conf  
I0915 10:57:30.867262 17284 checks.go:280] validating the existence of file /etc/kubernetes/bootstrap-kubelet.conf  
I0915 10:57:30.867269 17284 checks.go:104] validating the container runtime  
I0915 10:57:30.885423 17284 checks.go:639] validating whether swap is enabled or not  
I0915 10:57:30.885480 17284 checks.go:370] validating the presence of executable crictl  
I0915 10:57:30.885597 17284 checks.go:370] validating the presence of executable conntrack  
I0915 10:57:30.885622 17284 checks.go:370] validating the presence of executable ip  
I0915 10:57:30.886766 17284 checks.go:370] validating the presence of executable iptables  
I0915 10:57:30.886801 17284 checks.go:370] validating the presence of executable mount  
I0915 10:57:30.886900 17284 checks.go:370] validating the presence of executable nsenter  
I0915 10:57:30.886925 17284 checks.go:370] validating the presence of executable ebttables  
I0915 10:57:30.886973 17284 checks.go:370] validating the presence of executable ethtool  
I0915 10:57:30.887017 17284 checks.go:370] validating the presence of executable socat  
I0915 10:57:30.887117 17284 checks.go:370] validating the presence of executable tc  
I0915 10:57:30.887131 17284 checks.go:370] validating the presence of executable touch  
I0915 10:57:30.887147 17284 checks.go:516] running all checks  
I0915 10:57:30.888356 17284 checks.go:401] checking whether the given node name is valid and reachable using net.Lookup
```

Verify the cluster

```
ubuntu@ip-172-31-27-12:~$ kubectl get nodes
NAME           STATUS    ROLES      AGE   VERSION
ip-172-31-27-12  Ready    control-plane  38m   v1.29.9
ip-172-31-27-123 Ready    <none>     59s   v1.29.9
ip-172-31-30-123 Ready    <none>     45s   v1.29.9
ubuntu@ip-172-31-27-12:~$ |
```

EXPERIMENT 4

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

- **Running the application on the cluster**

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

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

- **Verifying the deployment using command**

```
kubectl get deployments
```

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

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

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

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

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

```
kubectl get services
```

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

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

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

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

1. sudo su

Aryan Dangat 12D15A
2. curl worker:30816

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

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

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

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



Conclusion:

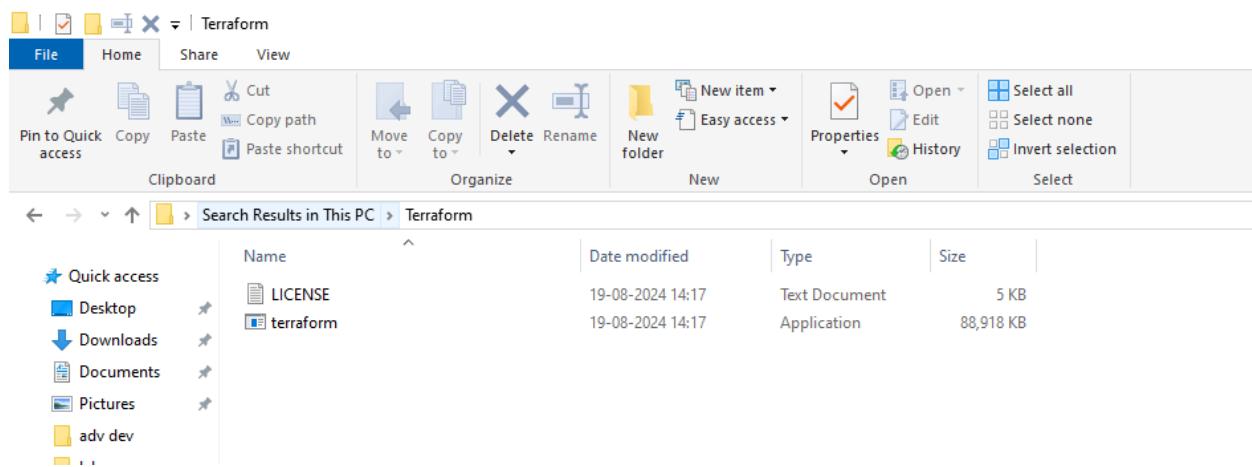
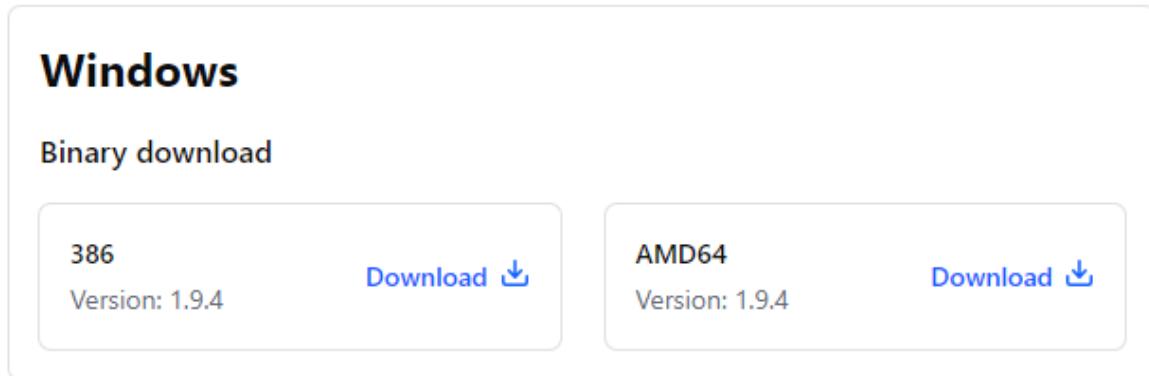
Thus, we have studied and implemented how to install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy.

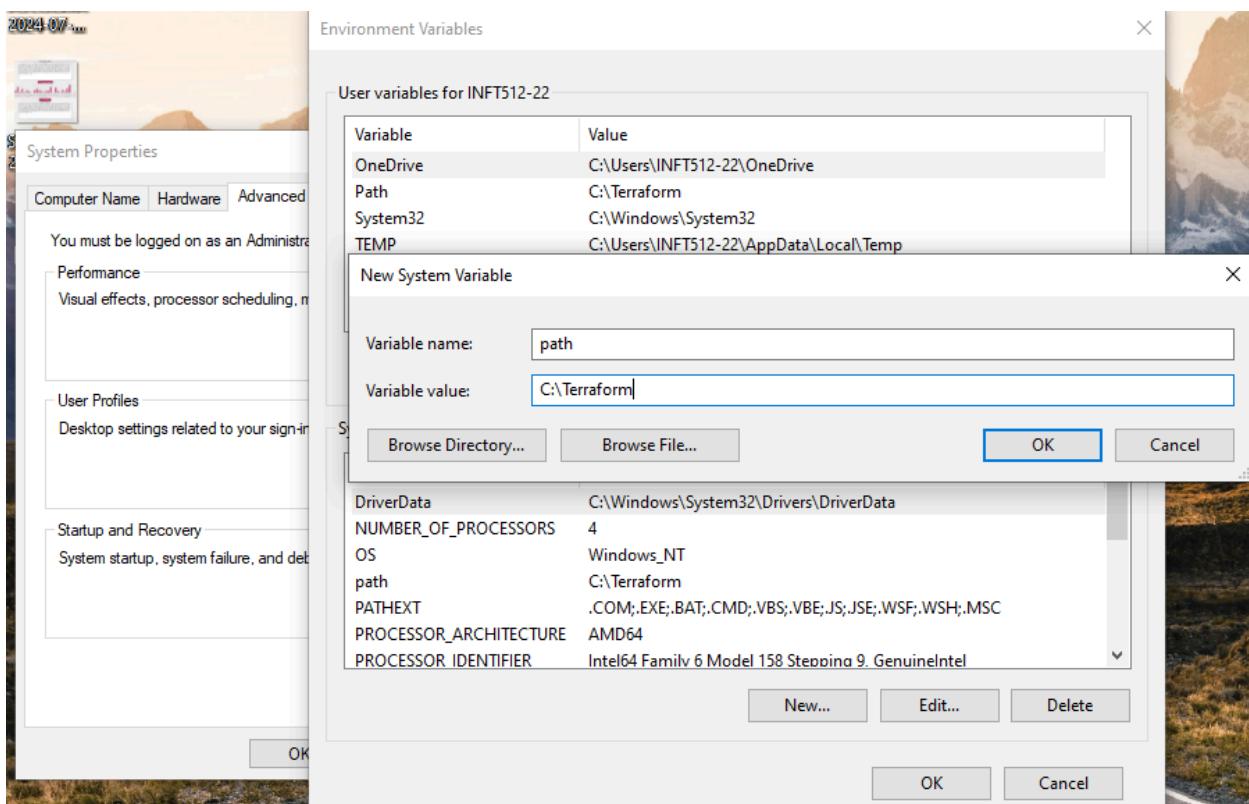
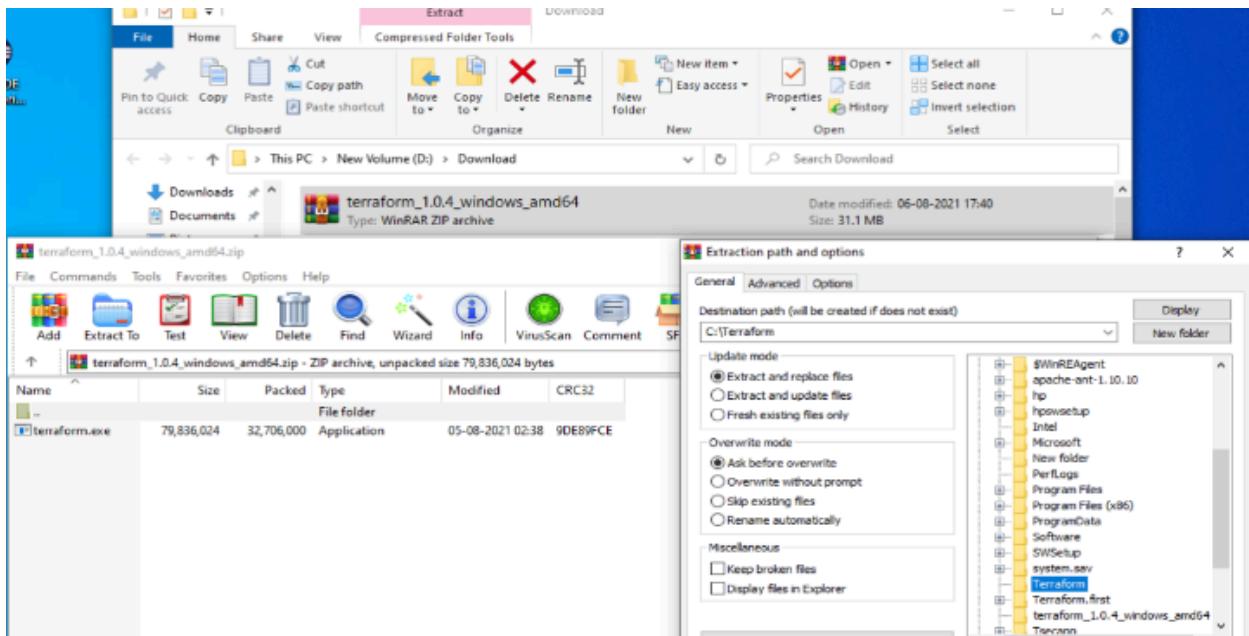
Errors:

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

EXPERIMENT NO 5:

Aryan Dangat
D15A
12





```
PS C:\Windows\system32> 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:\Windows\system32>
```

Aryan Dangat

D15A - 12

EXPERIMENT NO 6

```
PS C:\Users\HOME> docker

Usage: docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

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

Management Commands:
  builder  Manage builds
  buildx*  Docker Buildx
  checkpoint  Manage checkpoints
  compose*  Docker Compose
  container  Manage containers
  context    Manage contexts
  debug*    Get a shell into any image or container
  desktop*  Docker Desktop commands (Alpha)
  dev*     Docker Dev Environments
  extension* Manages Docker extensions
  feedback* Provide feedback, right in your terminal!
  image     Manage images
  init*    Creates Docker-related starter files for your project
  manifest  Manage Docker image manifests and manifest lists
  network   Manage networks
  plugin    Manage plugins
```

```
PS C:\Users\HOME> docker --version
Docker version 27.0.3, build 7d4bcd8
PS C:\Users\HOME>
```

The screenshot shows a code editor interface with a dark theme. On the left is an Explorer sidebar titled 'DOCKER' containing files: '.terraform', '.terraform.lock.hcl', '.terraform.state.lock.info', 'docker.tf' (which is selected), 'terraform.state', and 'terraform.state.backup'. The main pane displays the 'docker.tf' file content:

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

```
PS C:\Users\admin\Downloads\TerraformScript\Docker> terraform init  
Initializing the backend...  
Initializing provider plugins...  
- Reusing previous version of kreuzwerker/docker from the dependency lock file
```

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\admin\Downloads\TerraformScript\Docker> terraform plan
```

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

Terraform will perform the following actions:

```
# docker_container.foo will be created  
+ resource "docker_container" "foo" {  
    + attach          = false  
    + bridge          = (known after apply)  
    + command         = (known after apply)  
    + container_logs = (known after apply)  
    + entrypoint      = (known after apply)  
    + env             = (known after apply)  
    + exit_code       = (known after apply)  
    + gateway         = (known after apply)
```

```

+ exit_code      = (known after apply)
+ gateway        = (known after apply)
+ hostname       = (known after apply)
+ id             = (known after apply)
+ image          = (known after apply)
+ init           = (known after apply)
+ ip_address     = (known after apply)
+ ip_prefix_length = (known after apply)
+ ipc_mode       = (known after apply)
+ log_driver     = (known after apply)
+ logs           = false
+ must_run       = true
+ name           = "foo"
+ network_data   = (known after apply)
+ read_only      = false
+ remove_volumes = true
+ restart         = "no"
+ rm              = false
+ runtime         = (known after apply)
+ security_opts  = (known after apply)
+ shm_size        = (known after apply)
+ start           = true
+ stdin_open      = false
+ stop_signal     = (known after apply)
+ stop_timeout    = (known after apply)
+ tty              = false

+ healthcheck (known after apply)

+ labels (known after apply)
}

```

```

# docker_image.ubuntu will be created
+ resource "docker_image" "ubuntu" {
  + id      = (known after apply)
  + image_id = (known after apply)
  + latest   = (known after apply)
  + output    = (known after apply)
  + repo_digest = (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\admin\Downloads\TerraformScript\Docker> `terraform apply`

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

+ create

Terraform will perform the following actions:

```

# docker_container.foo will be created
+ resource "docker_container" "foo" {
  + attach      = false
  + bridge      = (known after apply)
  + command     = (known after apply)
  + container_logs = (known after apply)
  + entrypoint  = (known after apply)
  + env         = (known after apply)
  + exit_code    = (known after apply)

```

```
PS C:\Users\admin\Downloads\TerraformScript\Docker> terraform apply
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
```

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

```
+ create
```

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

```
# docker_container.foo will be created
+ resource "docker_container" "foo" {
  + attach          = false
  + bridge          = (known after apply)
  + command         = [
    + "sleep",
    + "infinity",
  ]
  + container_logs = (known after apply)
  + entrypoint      = (known after apply)
  + env             = (known after apply)
  + exit_code       = (known after apply)
  + gateway         = (known after apply)
  + hostname        = (known after apply)
  + id              = (known after apply)
  + image           = "sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a"
  + init            = (known after apply)
  + ip_address      = (known after apply)
  + ip_prefix_length = (known after apply)
  + ipc_mode        = (known after apply)
  + log_driver      = (known after apply)
  + logs            = false
}
```

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

```
PS C:\Users\admin\Downloads\TerraformScript\Docker> docker images
```

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

```
PS C:\Users\admin\Downloads\TerraformScript\Docker> terraform destroy
```

```
docker_image.ubuntu: Refreshing state... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598aubuntu:latest]
docker_container.foo: Refreshing state... [id=6f8c9cd2310ff1a47f534aab06dee78efe8e30d9444a6fc8fd65025b0789b78a]
```

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

```
- destroy
```

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

```
# docker_container.foo will be destroyed
- resource "docker_container" "foo" {
  - attach          = false -> null
  - command         = [
    - "sleep",
    - "infinity",
  ] -> null
  - cpu_shares     = 0 -> null
  - dns             = [] -> null
  - dns_opts        = [] -> null
  - dns_search      = [] -> null
  - entrypoint      = [] -> null
  - env             = [] -> null
  - gateway         = "172.17.0.1" -> null
  - group_add       = [] -> null
  - hostname        = "6f8c9cd2310f" -> null
}
```

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

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

```
Terraform will destroy all your managed infrastructure, as shown above.
```

```
There is no undo. Only 'yes' will be accepted to confirm.
```

```
Enter a value: yes
```

```
docker_container.foo: Destroying... [id=49376784d30a5434da697a13c16793bd7e7b89cae005d54167cedfe80]
docker_container.foo: Destruction complete after 0s
docker_image.ubuntu: Destroying... [id=sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519]
docker_image.ubuntu: Destruction complete after 1s
```

```
Destroy complete! Resources: 2 destroyed.
```



Jenkins 2.462.2 Setup



Welcome to the Jenkins 2.462.2 Setup Wizard

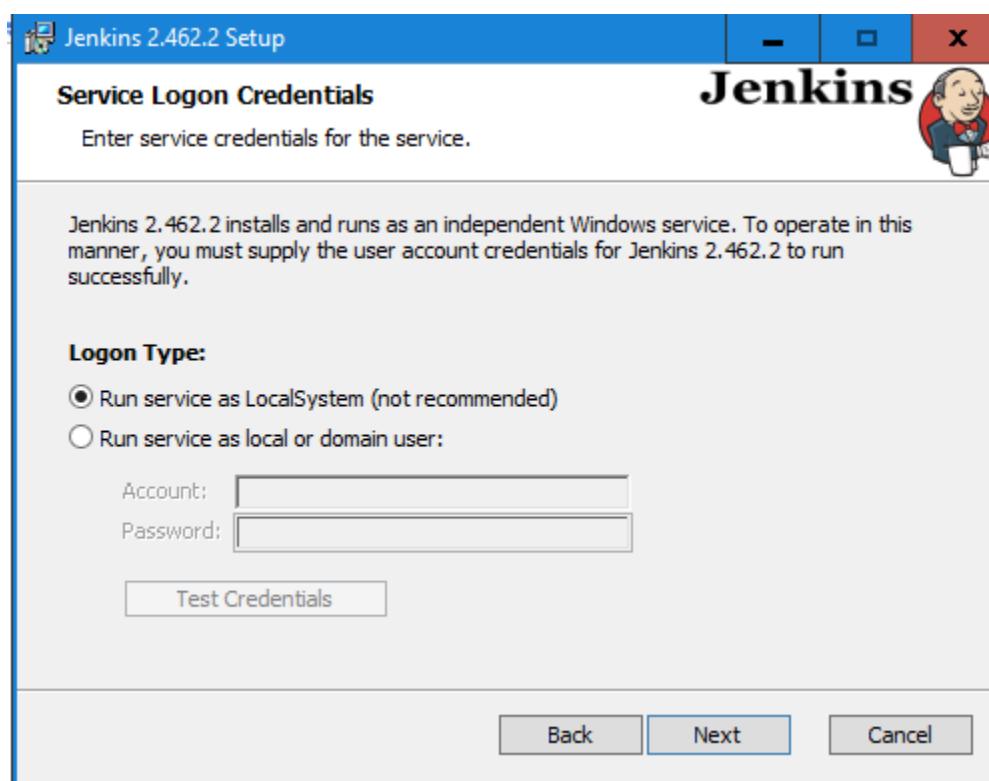
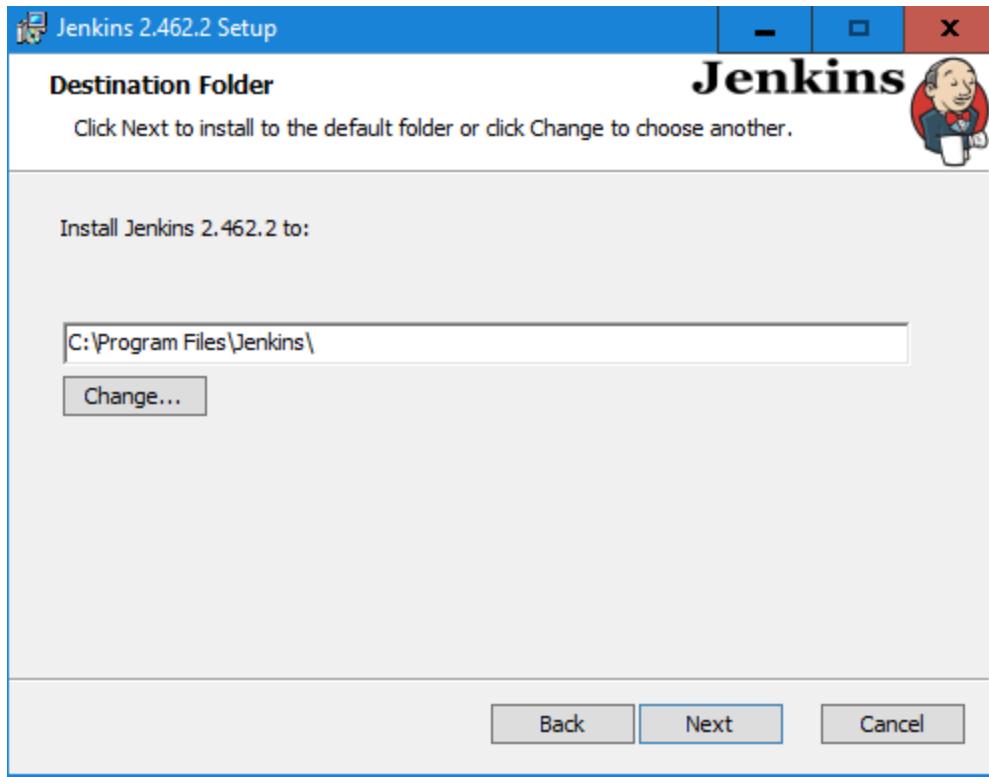
The Setup Wizard will install Jenkins 2.462.2 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

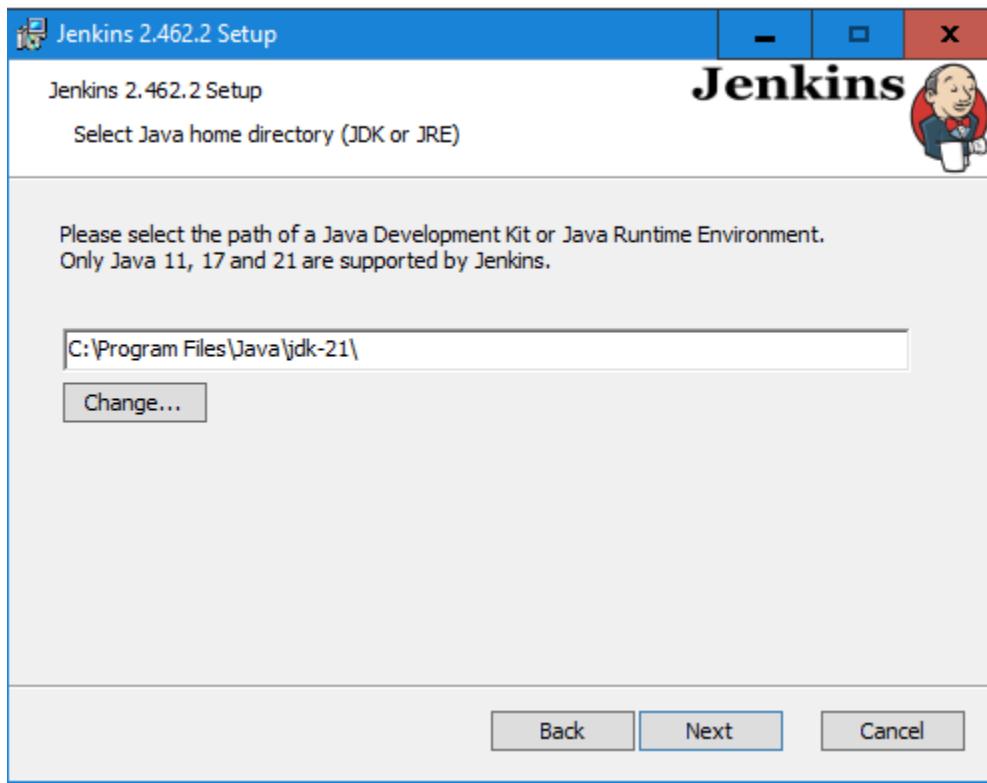
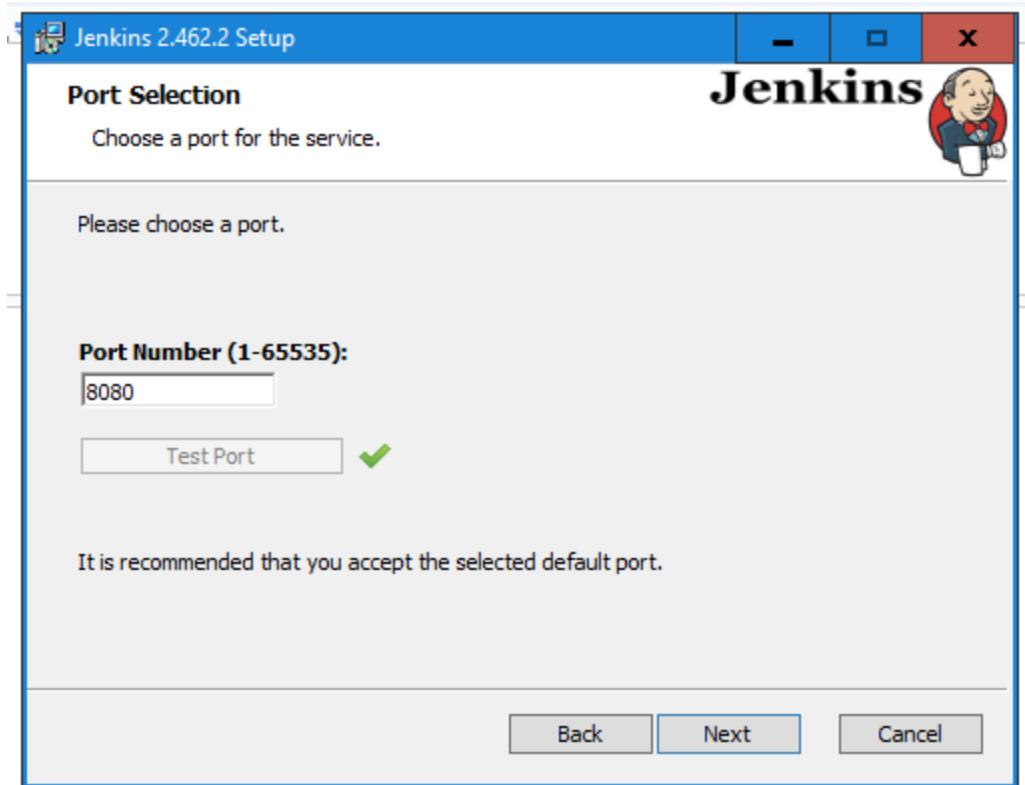


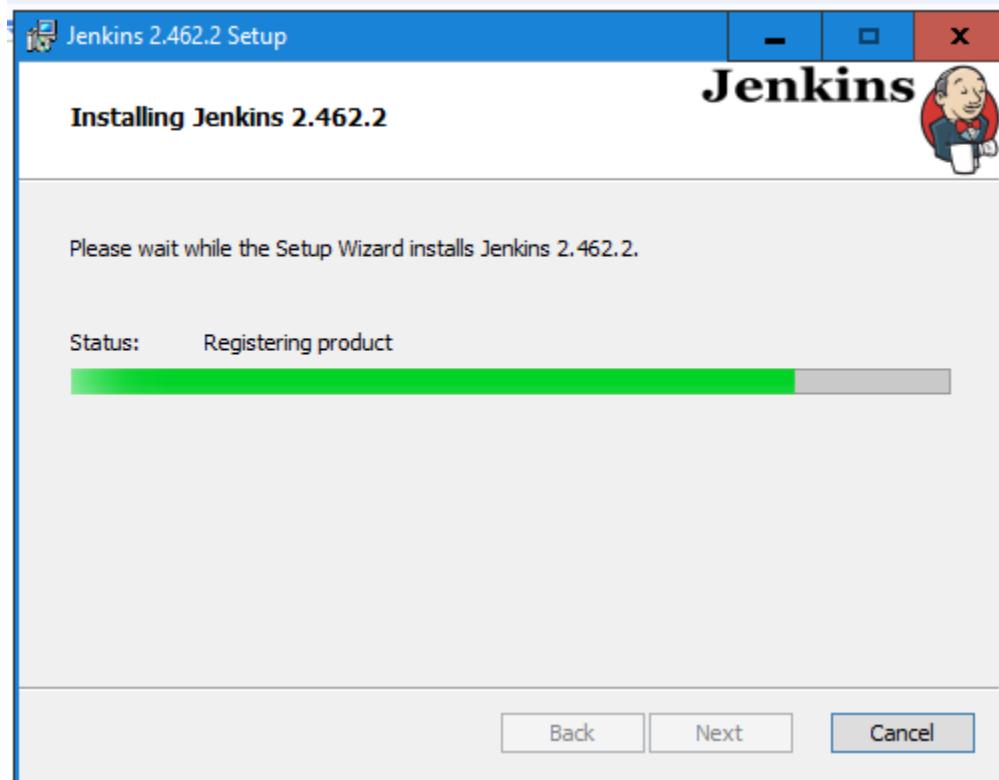
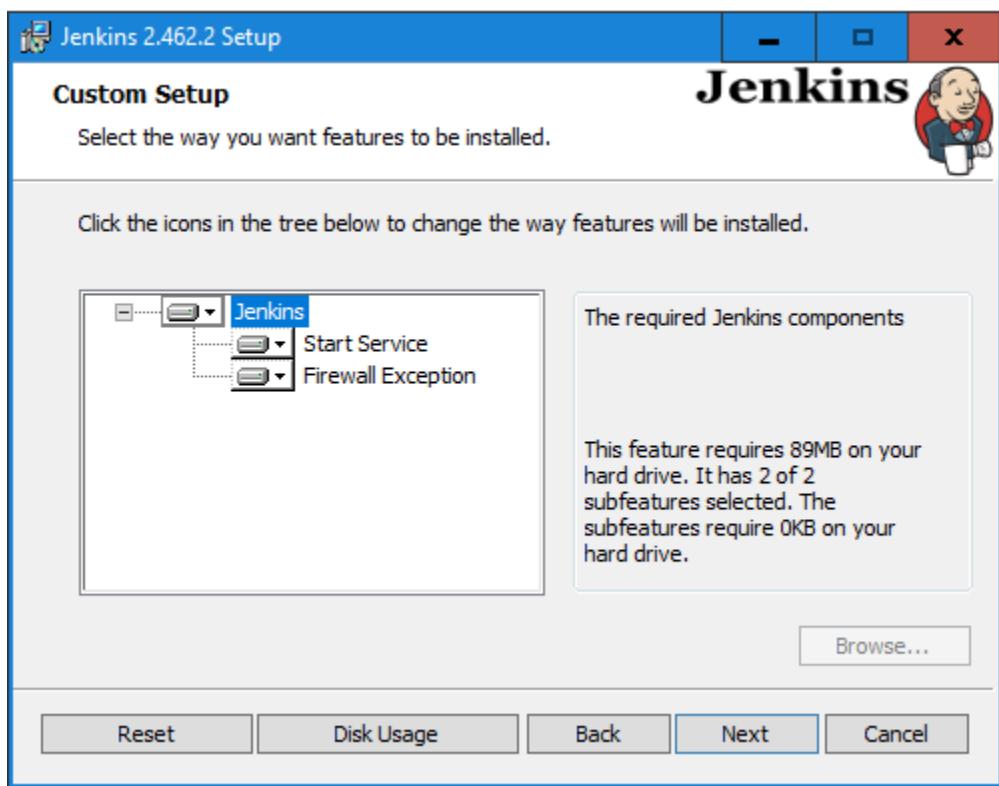
Back

Next

Cancel







Getting Started

Getting Started

✓ Folders	✓ OWASP Markup Formatter	↻ Build Timeout	○ Credentials Binding	** Ionicons API Folders OWASP Markup Formatter ** ASM API ** JSON Path API
⌚ Timestamper	⌚ Workspace Cleanup	○ Ant	⌚ Gradle	
⌚ Pipeline	⌚ GitHub Branch Source	⌚ Pipeline: GitHub Groovy Libraries	⌚ Pipeline Graph View	
⌚ Git	○ SSH Build Agents	⌚ Matrix Authorization Strategy	○ PAM Authentication	
○ LDAP	⌚ Email Extension	○ Mailer	○ Dark Theme	

Jenkins 2.462.2

The screenshot shows the Jenkins dashboard. At the top, there is a navigation bar with the Jenkins logo, a search bar labeled "Search (CTRL+K)", and user information for "aryan dangat". On the right side of the top bar are icons for help, status, and log out.

The main content area starts with a "Welcome to Jenkins!" message. Below it, a note says: "This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project." There is a "Start building your software project" button.

On the left, there are two collapsed sections: "Build Queue" (which says "No builds in the queue.") and "Build Executor Status" (which lists "1 Idle" and "2 Idle").

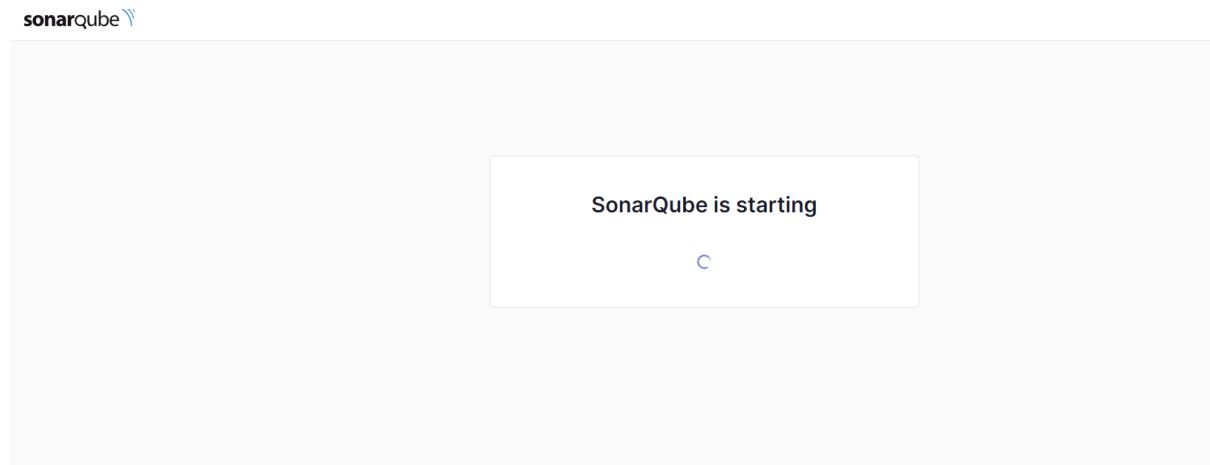
On the right, there are three buttons under the heading "Set up a distributed build": "Create a job" (with a plus sign icon), "Set up an agent" (with a monitor icon), and "Configure a cloud" (with a cloud icon). Below these is a link "Learn more about distributed builds" with a help icon.

EXPERIMENT NO 8

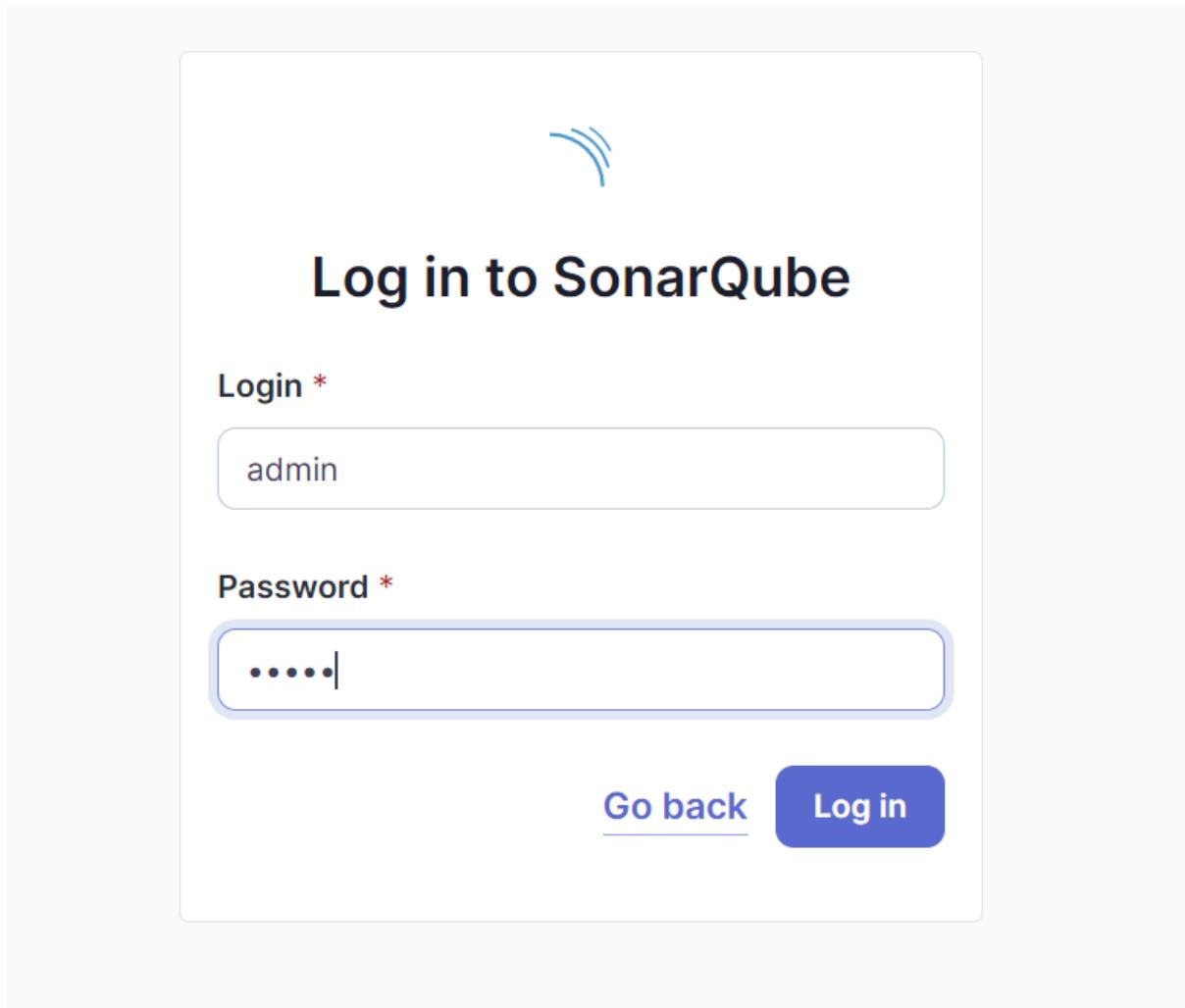
Run SonarQube in a Docker container using this command – docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest

```
PS C:\Users\HOME> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
Unable to find image 'sonarqube:latest' locally
latest: Pulling from library/sonarqube
762bedf4b1b7: Pull complete
95f9bd9906fa: Pull complete
a32d681e6b99: Pull complete
aabdd0a18314: Pull complete
5161e45ecd8d: Pull complete
ae0020dfa06: Pull complete
01548d361aea: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:bb444c58c1e04d8a147a3bb12af941c57e0100a5b21d10e599384d59bed36c86
Status: Downloaded newer image for sonarqube:latest
60e160dc38e9d08738e29dfab68329dbff2aa0ffe9db57fec825ee41a9d1608c
PS C:\Users\HOME>
```

At localhost:9000



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.



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

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

1 of 2

Create a local project

Project display name *

sonarqube-test



Project key *

sonarqube-test



Main branch name *

main

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

[Cancel](#)

[Next](#)

2 of 2

Set up project for Clean as You Code

x

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

Choose the baseline for new code for this project

Use the global setting

Previous version

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

Recommended for projects following regular versions or releases.

Define a specific setting for this project

Previous version

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

Recommended for projects following regular versions or releases.

Number of days

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

The screenshot shows the Jenkins 'Plugins' page. At the top, there is a breadcrumb navigation: Dashboard > Manage Jenkins > Plugins. Below this, the left sidebar has four items: 'Updates' (with a '29' badge), 'Available plugins', 'Installed plugins', and 'Advanced settings'. The main area is titled 'Download progress' and shows the 'Preparation' step completed successfully. It lists two tasks: 'SonarQube Scanner' (Success) and 'Loading plugin extensions' (Success). A progress bar at the bottom of the sidebar indicates the download progress.

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.

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

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



Generate a token

Analyze your project

We initialized your project on SonarQube, now it's up to you to launch analyses!

1 Provide a token

[Generate a project token](#) [Use existing token](#)

Token name [?](#)

Analyze "sonarqube-test"

Expires in

90 days

[Generate](#)

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

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

Analyze your project

We initialized your project on SonarQube, now it's up to you to launch analyses!

1 Provide a token

Analyze "sonarqube-test": **sqp_def32b923cf4d6ba1a713fbf2980a90a95752daf** 

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

[Continue](#)

2 Run analysis on your project

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.

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

New credentials

Kind

Secret text

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Secret

.....

ID ?

sonarqube_practical

 Unacceptable characters

Description ?

[Create](#)

SonarQube installations

List of SonarQube installations

Name	<input type="text" value="sonarqube"/>
Server URL	Default is http://localhost:9000 <input type="text" value="http://localhost:9000"/>
Server authentication token	SonarQube authentication token. Mandatory when anonymous access is disabled. <input type="text" value="sonarqube_practical"/>
+ Add	
Advanced ▼	
<input type="button" value="Save"/> <input type="button" value="Apply"/>	

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.

Dashboard > sonarqube8 > Configuration

Definition

Configure

General

Advanced Project Options

Pipeline

Pipeline script

```

1 node {
2   stage('Cloning the GitHub Repo') {
3     git 'https://github.com/shazforiot/GOL.git'
4   }
5   stage('SonarQube analysis') {
6     wrap([$class: 'ajarsonwrap.AJARsonWrap'] {
7       sh """
8           PATH_TO SONARQUBE FOLDER/bin/sonar-scanner \
9           -D sonar.login=<SonarQube_USERNAME> \
10          -D sonar.password=<SonarQube_PASSWORD> \
11          -D sonar.projectKey=<Project KEY> \
12          -D sonar.exclusions=vendor/**,resources/**/**.java \
13          -D sonar.host.url=http://127.0.0.1:9000/
14      """
15   })
}
try sample Pipeline... ▾

```

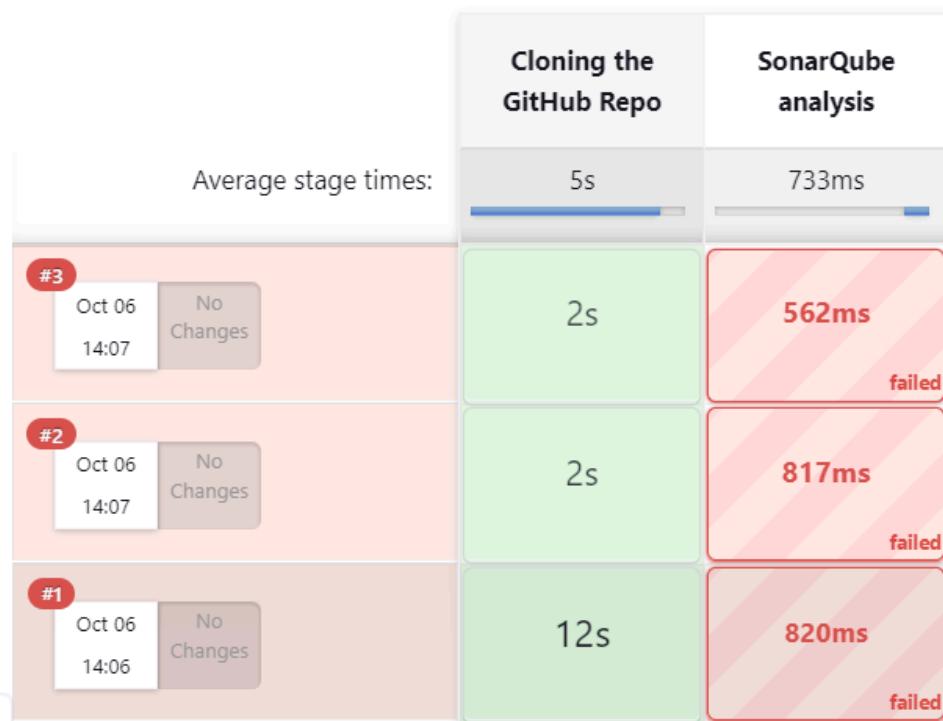
Use Groovy Sandbox

Pipeline Syntax

Build Project

sonarqube8

Stage View

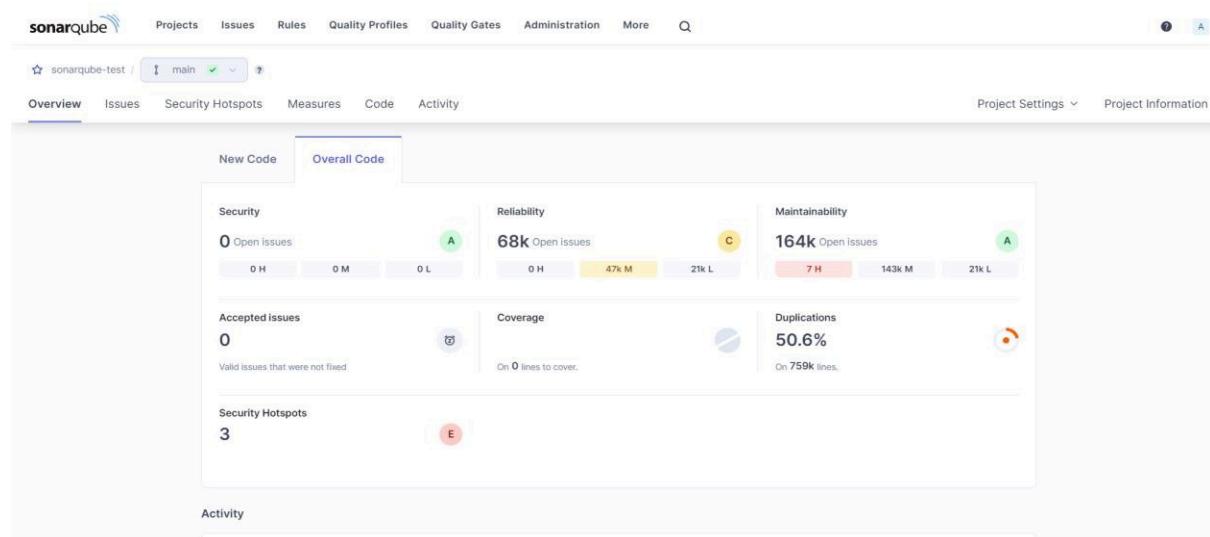


Permalinks

Console Output

```
Started by user Spandan Deb
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\ProgramData\Jenkins\.jenkins\workspace\sonarqube8
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Cloning the GitHub Repo)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\sonarqube8\.git # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/shazforiot/GOL.git # timeout=10
Fetching upstream changes from https://github.com/shazforiot/GOL.git
> git.exe --version # timeout=10
> git --version # 'git version 2.43.0.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/shazforiot/GOL.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision ba799ba7e1b576f04a4612322b0412c5e6e1e5e4 (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f ba799ba7e1b576f04a4612322b0412c5e6e1e5e4 # timeout=10
> git.exe branch -a -v --no-abbrev # timeout=10
> git.exe branch -D master # timeout=10
> git.exe checkout -b master ba799ba7e1b576f04a4612322b0412c5e6e1e5e4 # timeout=10
Commit message: "Update Jenkinsfile"
> git.exe rev-list --no-walk ba799ba7e1b576f04a4612322b0412c5e6e1e5e4 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (SonarQube analysis)
```

Project in Sonarqube



Code Problems

Consistency

The screenshot shows the SonarQube interface for the project 'sonarqube-test'. The 'Issues' tab is selected. On the left, there's a sidebar with 'My Issues' and 'All' buttons, and a 'Filters' section with a 'Clear All Filters' button. Below that is a 'Clean Code Attribute' section with 'Consistency' selected, showing 197k issues. Other categories like Intentionality, Adaptability, and Responsibility have 14k, 0, and 0 issues respectively. A 'Software Quality' section is also present. The main panel displays three specific consistency-related issues:

- Insert a <!DOCTYPE> declaration to before this <html> tag. (Reliability) Consistency user-experience +
L1 - 5min effort - 4 years ago - ⚡ Bug - ⚡ Major
- Remove this deprecated "width" attribute. (Maintainability) Consistency html5 obsolete +
L9 - 5min effort - 4 years ago - ⚡ Code Smell - ⚡ Major
- Remove this deprecated "align" attribute. (Maintainability) Consistency html5 obsolete +
L1 - 5min effort - 4 years ago - ⚡ Code Smell - ⚡ Major

At the top right, it says '196,662 issues' and '3075d effort'. Project settings and information are also visible.

Intentionality

This screenshot shows the SonarQube interface for the project 'sonarqube-test'. The 'Issues' tab is selected. The sidebar and issue list are identical to the previous screenshot. The main panel displays three intentionality-related issues:

- Use a specific version tag for the image. (Maintainability) Intentionality No tags +
L1 - 5min effort - 4 years ago - ⚡ Code Smell - ⚡ Major
- Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. (Maintainability) Intentionality No tags +
L12 - 5min effort - 4 years ago - ⚡ Code Smell - ⚡ Major
- Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. (Maintainability) Intentionality No tags +
L12 - 5min effort - 4 years ago - ⚡ Code Smell - ⚡ Major

Bugs

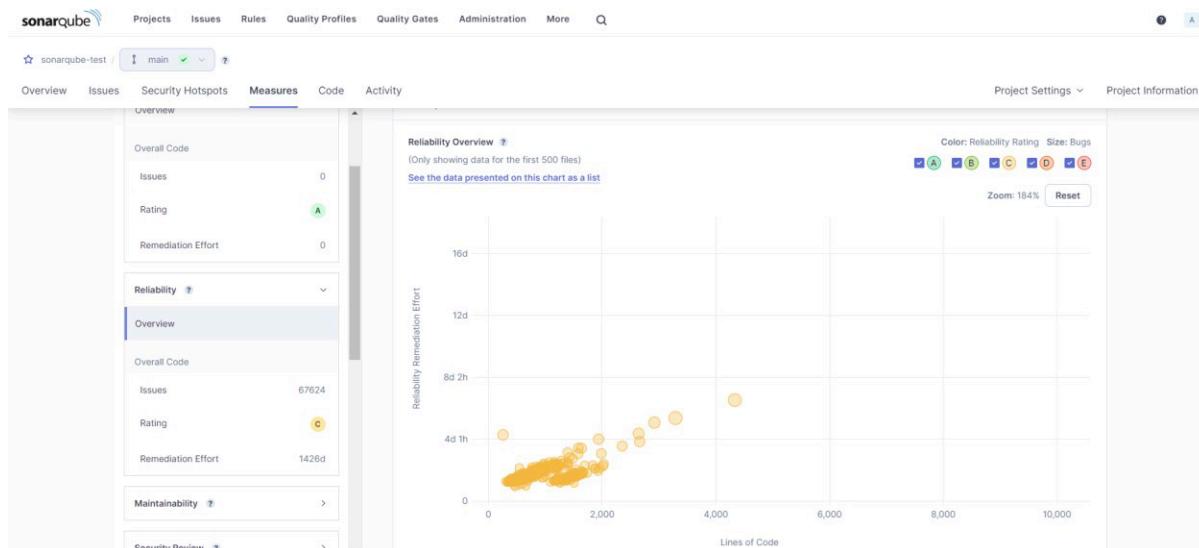
This screenshot shows the SonarQube interface for the project 'sonarqube-test'. The 'Issues' tab is selected. The sidebar and issue list are identical to the previous screenshots. The main panel displays two bug-related issues:

- Add "lang" and/or "xml:lang" attributes to this "<html>" element. (Reliability) Intentionality accessibility wcag2-a +
L1 - 2min effort - 4 years ago - ⚡ Bug - ⚡ Major
- Add "<th>" headers to this "<table>". (Reliability) Intentionality accessibility wcag2-a +
L9 - 2min effort - 4 years ago - ⚡ Bug - ⚡ Major

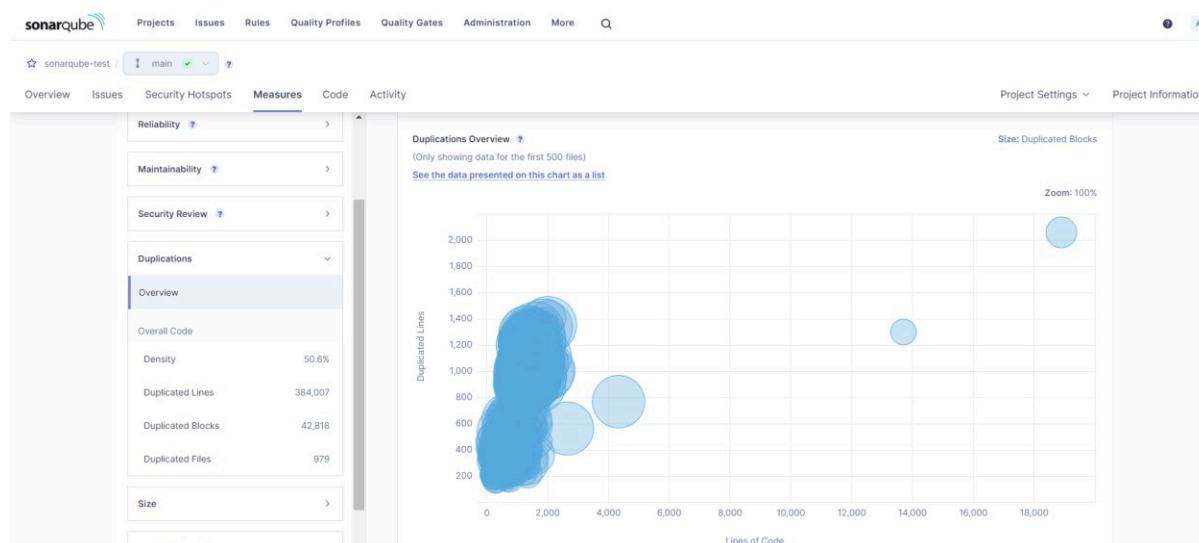
Code Smells

The screenshot shows the SonarQube interface for the project 'sonarqube-test'. The 'Issues' tab is selected. On the left, there's a sidebar with sections for 'Clean Code Attribute' and 'Software Quality', each containing metrics like Consistency, Intentionality, Adaptability, Responsibility, and Maintainability. The 'Intentionality' section under 'Clean Code Attribute' has a value of 15. The main panel displays a list of code smell issues found in the file 'gameoflife-acceptance-tests/Dockerfile'. One issue is highlighted: 'Use a specific version tag for the image.' This issue is categorized under 'Intentionality' and has a status of 'Open'. Other issues listed include 'Surround this variable with double quotes; otherwise, it can lead to unexpected behavior.' These also have 'Intentionality' categories and 'Open' status. The top right of the screen shows '15 Issues' and '44min effort'.

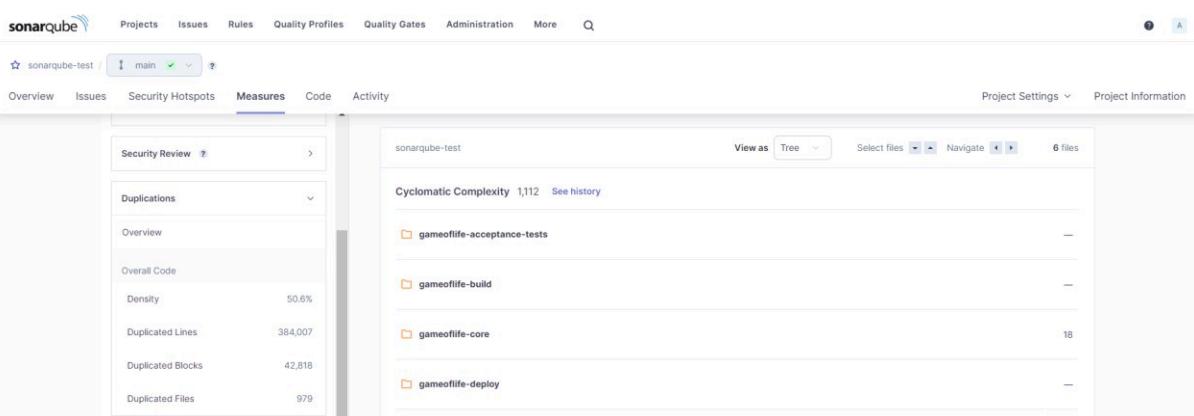
Reliability



Duplications



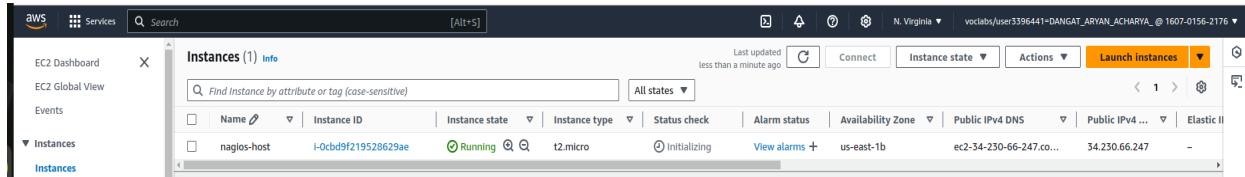
Cyclomatic Complexities



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

Inbound rules (7)							
	Name	Security group rule...	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sgr-02a9f3b07b74ccb99	IPv4	All traffic	All	All	0.0.0.0/0
<input type="checkbox"/>	-	sgr-06bc1bda1cab4310b	IPv4	HTTP	TCP	80	0.0.0.0/0
<input type="checkbox"/>	-	sgr-0326ed648d9ec08...	IPv4	Custom TCP	TCP	0	0.0.0.0/0
<input type="checkbox"/>	-	sgr-0d9f700fbeef6cd54	IPv6	Custom TCP	TCP	0	::/0
<input type="checkbox"/>	-	sgr-09de549ee0cd724...	IPv4	All ICMP - IPv4	ICMP	All	0.0.0.0/0
<input type="checkbox"/>	-	sgr-06a0c86195a574ae2	IPv4	HTTPS	TCP	443	0.0.0.0/0
<input type="checkbox"/>	-	sgr-0ee64d6378e641b...	IPv4	All ICMP - IPv6	IPv6 ICMP	All	0.0.0.0/0

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

```

aws | Services | Search [Alt+S]
          #
          #\##
          #\#\#\#
          \#\#\#
          \#\#\#
          \#/ \
          V~,'--> https://aws.amazon.com/linux/amazon-linux-2023
          / \
          / \
[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.

```

□ Update and install the required packages

`sudo yum install httpd php`

```

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Mon Oct 14 01:19:01 2024 from 18.206.107.28
[ec2-user@ip-172-31-44-161 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:28:54 ago on Mon Oct 14 00:55:17 2024.
Dependencies resolved.
=====
Package           Architecture Version      Repository   Size
=====
Installing:
httpd            x86_64       2.4.62-1.amzn2023.0.1    amazonlinux  48 k
php8.3           x86_64       8.3.10-1.amzn2023.0.1    amazonlinux  19 k
Installing dependencies:
apr              x86_64       1.7.2-2.amzn2023.0.2    amazonlinux  120 k
apr-util          x86_64       1.6.3-1.amzn2023.0.1    amazonlinux  98 k
generic-logos-httpd noarch     18.0.0-12.amzn2023.0.3   amazonlinux  19 k
httpd-core        x86_64       2.4.62-1.amzn2023.0.2   amazonlinux  1.4 M
httpd-filesystem noarch     2.4.62-1.amzn2023.0.2   amazonlinux  11 k
httpd-tools       x86_64       2.4.62-1.amzn2023.0.2   amazonlinux  61 k
libbrotli         x86_64       1.0.9-4.amzn2023.0.2    amazonlinux  315 k
libsodium         x86_64       1.0.19-4.amzn2023.0.2   amazonlinux  176 k
mod_wsgi          noarch     1.1.34-5.amzn2023.0.2   amazonlinux  241 k
mod_wsgi-filesystem noarch     1.1.24-0.1.amzn2023.0.4   amazonlinux  9.8 k
php8.3-clients   x86_64       8.3.10-1.amzn2023.0.4    amazonlinux  3.7 M
php8.3-common    x86_64       8.3.10-1.amzn2023.0.1    amazonlinux  737 k
php8.3-processes x86_64       8.3.10-1.amzn2023.0.1    amazonlinux  45 k
php8.3-zts        x86_64       8.3.10-1.amzn2023.0.1    amazonlinux  154 k
Installing weak dependencies:
apr-util-openssl  x86_64       1.6.3-1.amzn2023.0.1    amazonlinux  17 k
mod_http          x86_64       2.0.27-1.amzn2023.0.3   amazonlinux  166 k
mod_wsgi          x86_64       2.4.62-1.amzn2023.0.2   amazonlinux  6 k
php8.3-fpm        x86_64       8.3.10-1.amzn2023.0.1    amazonlinux  1.9 M
php8.3-mbstring   x86_64       8.3.10-1.amzn2023.0.1    amazonlinux  528 k
php8.3-oncache   x86_64       8.3.10-1.amzn2023.0.1    amazonlinux  370 k
=====
```

`sudo yum install gcc glibc glibc-common`

```

Complete!
[ec2-user@ip-172-31-44-161 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:30:25 ago on Mon Oct 14 00:55:17 2024.
Package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Dependencies resolved.
=====
Package           Architecture Version      Repository   Size
=====
Installing:
gcc              x86_64       11.4.1-2.amzn2023.0.2    amazonlinux  32 M
Installing dependencies:
annobin-docs      noarch     18.93-1.amzn2023.0.1    amazonlinux  92 k
annobin-plugin-gcc x86_64       19.93-1.amzn2023.0.1    amazonlinux  887 k
cpp              x86_64       11.4.1-2.amzn2023.0.2   amazonlinux  10 M
glibc             x86_64       8.0.4-5.amzn2023.0.2   amazonlinux  105 k
glibc-devel       x86_64       2.34-52.amzn2023.0.11  amazonlinux  27 k
glibc-headers-x86_glibc22
guile22          x86_64       2.2.7-2.amzn2023.0.3   amazonlinux  423 k
kernel-headers   x86_64       6.1.109-118.189.amzn2023  amazonlinux  6.4 M
liblomp          x86_64       1.2.1-2.amzn2023.0.2   amazonlinux  1.4 M
libltool-ltdl    x86_64       2.4.2-1.amzn2023.0.3   amazonlinux  62 k
liblrypt-devel   x86_64       4.1.33-7.amzn2023.0.2  amazonlinux  32 k
make              x86_64       1:4.3-5.amzn2023.0.2   amazonlinux  534 k
=====
Transaction Summary
=====
Install 13 Packages

Total download size: 52 M
Installed size: 12208 M
Is this ok? [y/N]: y
Downloading Packages:
(1/13): annobin-docs-10.93-1.amzn2023.0.1.noarch.rpm 1.5 MB/s | 92 kB  00:00
(2/13): annobin-plugin-gcc-10.93-1.amzn2023.0.1.x86_64.rpm 9.6 MB/s | 887 kB  00:09
(3/13): cpp-11.4.1-2.amzn2023.0.2.x86_64.rpm 2.1 MB/s | 130 kB  00:01
(4/13): glibc-devel-2.34-52.amzn2023.0.11.x86_64.rpm 867 kB/s | 27 kB  00:00
(5/13): glibc-headers-x86_64-2.34-52.amzn2023.0.11.noarch.rpm 12 MB/s | 427 kB  00:00
(6/13): cpp-11.4.1-2.amzn2023.0.2.x86_64.rpm 34 MB/s | 10 MB  00:00
(7/13): kernel-headers-6.1.109-118.189.amzn2023.x86_64.rpm 12 MB/s | 1.4 MB  00:00
(8/13): liblomp-1.2.1-2.amzn2023.0.2.x86_64.rpm 2.3 MB/s | 62 kB  00:00
(9/13): libltool-ltdl-2.4.2-1.amzn2023.0.3.x86_64.rpm 22 MB/s | 6.4 MR  00:00
=====
```

`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-fs	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-fs	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
[Verifying : xml-common-0.6.3-56.amzn2023.0.2.noarch
[Verifying : xorg-x11proto-devel-2021.4-1.amzn2023.0.2.noarch
[Verifying : xz-devel-5.2.5-9.amzn2023.0.2.x86_64
[Verifying : zlib-devel-1.2.11-33.amzn2023.0.5.x86_64
[Installed:
brotli-1.0.9-4.amzn2023.0.2.x86_64
cairo-1.17.6-2.amzn2023.0.1.x86_64
fontconfig-devel-2.13.94-2.amzn2023.0.2.x86_64
freetype-devel-2.13.2-5.amzn2023.0.1.x86_64
glib2-devel-2.74.7-689.amzn2023.0.2.x86_64
graphite2-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz-devel-7.0.0-2.amzn2023.0.1.x86_64
langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch
libX11-1.7.2-3.amzn2023.0.4.x86_64
libX11-xcb-1.7.2-3.amzn2023.0.4.x86_64
libXext-1.3.4-6.amzn2023.0.2.x86_64
libXrender-0.9.10-14.amzn2023.0.2.x86_64
libBf1-devel-3.4.4-1.amzn2023.0.1.x86_64
libBp1eg-turbo-2.1.4-2.amzn2023.0.5.x86_64
libPng-2.1.6.37-10.amzn2023.0.6.x86_64
libSepol-devel-3.4-3.amzn2023.0.3.x86_64
libWebp-1.2.4-1.amzn2023.0.6.x86_64
libxcb-devel-1.13.1-7.amzn2023.0.2.x86_64
pcre2-utf16-10.40-1.amzn2023.0.3.x86_64
sysprof-capture-devel-3.40.1-2.amzn2023.0.2.x86_64
xz-devel-5.2.5-9.amzn2023.0.2.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 nagios
[ec2-user@ip-172-31-46-179 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-46-179 ~]$
```

- 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 nagios
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 nagios /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.
```

sudo make install-commandmode

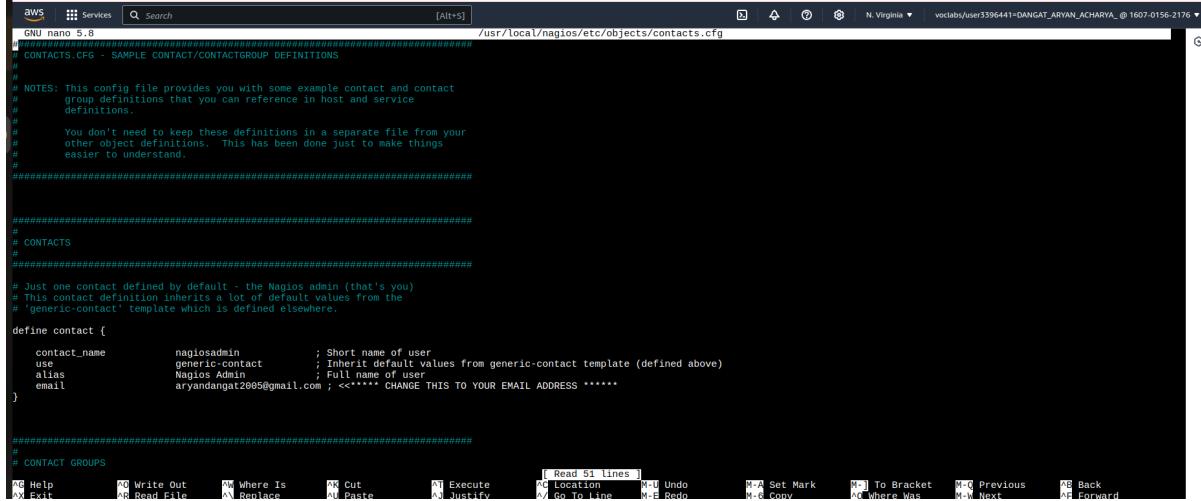
```
[ec2-user@ip-172-31-46-179 nagios-4.4.6]$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

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

□ Edit the Config File to Change the Email Address

Change the email address in the configuration file



```

GNU nano 5.8
[Alt+S] /usr/local/nagios/etc/objects/contacts.cfg N. Virginia vocabs/user:3396441:DANGAT_ARYAN_ACHARYA_ @ 1607.0156-2176
#####
# CONTACTS.CFG - SAMPLE CONTACT/CONTACTGROUP DEFINITIONS
#
# NOTES: This config file provides you with some example contact and contact
# group definitions that you can reference in host and service
# definitions.
#
# You don't need to keep these definitions in a separate file from your
# other object definitions. This has been done just to make things
# easier to understand.
#
#####
#
# CONTACTS
#
#####
#
# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the
# "generic-contact" template which is defined elsewhere.
define contact {
    contact_name      nagiosadmin          ; Short name of user
    use               generic-contact       ; Inherit default values from generic-contact template (defined above)
    alias             Nagios Admin        ; Full name of user
    email             aryandangat2005@gmail.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
}

#####
# CONTACT GROUPS
[Alt+L] [Read 51 lines] [C] Location [U] Undo [M-A] Set Mark [M-S] To Bracket [M-Q] Where Was [M-X] Previous [M-M] Next [M-B] Back
[Alt+F] Help [Alt+R] Write Out [Alt+W] Where Is [Alt+C] Cut [Alt+P] Paste [Alt+E] Execute [Alt+J] Justify [Alt+G] Go To Line [Alt+U] Undo [Alt+A] Set Mark [Alt+S] To Bracket [Alt+Q] Where Was [Alt+X] Previous [Alt+M] Next [Alt+B] Back
[Alt+X] Exit [Alt+R] Read File [Alt+P] Replace [Alt+L] Location [Alt+U] Undo [Alt+A] Set Mark [Alt+Q] Where Was [Alt+X] Next [Alt+B] Back

```

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  cgi 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

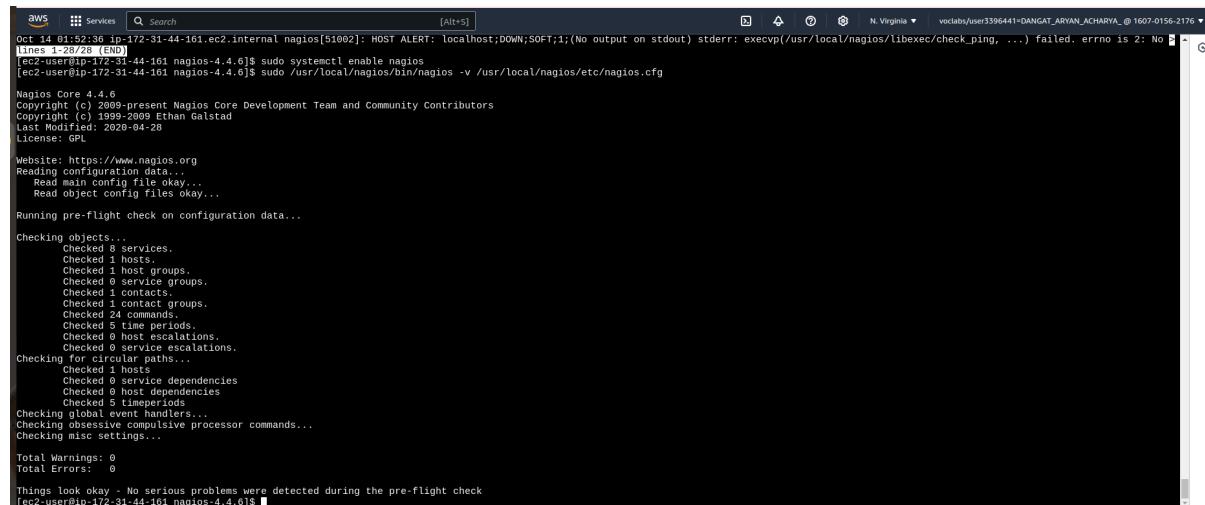
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

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



The screenshot shows a terminal window with the following command history:

```

Oct 14 01:52:36 ip-172-31-44-161.ec2.internal nagios[51002]: HOST ALERT: localhost;DOWN;SOFT;1;(No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_ping, ...) failed. errno is 2: No such file or directory
[ec2-user@ip-172-31-44-161 nagios-4.4.6]$ sudo systemctl enable nagios
[ec2-user@ip-172-31-44-161 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) 1999-2008 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
  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-172-31-44-161 nagios-4.4.6]$ 

```

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

```

AWS Lambda Services Search [Alt+T]
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-172-31-44-161 nagios:~$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-44-161 nagios:~$ sudo systemctl status nagios
● nagios.service - Apache2
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
     Active: active (running) since Mon 2024-10-14 01:52:35 UTC; 2min 8s ago
       Docs: https://www.nagios.org/doc/DOCUMENTATION
   Main PID: 51002 (nagios)
      Tasks: 1 (limit: 1112)
        Memory: 2.3M
          CPU: 26ms
        Group: /system.slice/nagios.service
           └─ 51003 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              ├ 51003 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              ├ 51004 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              ├ 51005 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              ├ 51006 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              └─ 51056 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 14 01:52:35 ip-172-31-44-161.ec2.internal nagios[51002]: wproc: Registry request: name=Core Worker 51005;pid=51005
Oct 14 01:52:35 ip-172-31-44-161.ec2.internal nagios[51002]: wproc: Registry request: name=Core Worker 51006;pid=51006
Oct 14 01:52:35 ip-172-31-44-161.ec2.internal nagios[51002]: wproc: Registry request: name=Core Worker 51003;pid=51003
Oct 14 01:52:36 ip-172-31-44-161.ec2.internal nagios[51002]: wproc: Registry request: name=Core Worker 51004;pid=51004
Oct 14 01:52:36 ip-172-31-44-161.ec2.internal nagios[51002]: HOST ALERT: localhost:DOWN;SOFT;1:(No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_ping, ...) failed. errno is 2: No
Oct 14 01:53:12 ip-172-31-44-161.ec2.internal nagios[51002]: SERVICE ALERT: localhost:Current Load;CRITICAL;HARD;1:(No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_load, ...) failed. errno is 2: No
Oct 14 01:53:36 ip-172-31-44-161.ec2.internal nagios[51002]: HOST ALERT: localhost:DOWN;SOFT;2:(No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_ping, ...) failed. errno is 2: No
Oct 14 01:54:27 ip-172-31-44-161.ec2.internal nagios[51002]: SERVICE ALERT: localhost:HTTP;CRITICAL;HARD;1:(No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_http, ...) failed. errno is 2: No
Oct 14 01:54:36 ip-172-31-44-161.ec2.internal nagios[51002]: HOST ALERT: localhost:DOWN;SOFT;3:(No output on stdout) stderr: execvp(/usr/local/nagios/libexec/check_ping, ...) failed. errno is 2: No
Lines 1-26/26 (END)
[ec2-user@ip-172-31-44-161 nagios:~$ ]

```

□ 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.

The screenshot shows the Nagios Core web interface. The top navigation bar includes links for Launch AWS Academy, Instances | EC2 | us-east-1, Instances | EC2 | us-east-1, EC2 Instance Connect, ChatGPT, and the current URL http://your_public_ip. The main content area displays the Nagios Core dashboard with the following sections:

- General**: Includes links for Home, Documentation, and a Current Status summary.
- Current Status**: Shows a summary of the system's status with a note: "Daemon running with PID 51002".
- Problems**: Lists services with problems, including "Services (Unended)" and "Network Outages".
- Reports**: Includes sections for Alerts, Trends (Legacy), History, and Notifications.
- System**: Includes links for Comments, Downtime, Process Info, Performance Info, Scheduling Queue, and Configuration.
- Quick Links**: Provides links to Nagios Library, Plugins, Exchange, Support, and the official website.
- Latest News**: A section for the latest news and updates.
- Don't Miss...**: A section for other important information.
- User Profile**: On the right side, it shows the user profile for "Aryan Dangat" (aryandangat2005@gmail.com) with options to Sync on, Customize profile, Manage Google Account, and Close this profile.
- Other Chrome profiles**: Lists "ARYAN (ves.ac.in)" and "Open Guest profile".
- Manage Chrome profiles**: Options to Add new profile and Manage Chrome profiles.

ADVANCE DEVOPS EXPERIMENT 10

Name:Aryan Dangat

Class:D15A

Roll No: 12

The screenshot shows the AWS EC2 'Launch an instance' wizard. The top navigation bar indicates the user is at [EC2](#) > [Instances](#) > [Launch an instance](#). The main section is titled 'Launch an instance' with an 'Info' link. A sub-section titled 'Name and tags' also has an 'Info' link. The 'Name' field contains 'exp10client'. An 'Add additional tags' link is visible. Below this is a section titled 'Application and OS Images (Amazon Machine Image)' with an 'Info' link. It includes a search bar and a 'Quick Start' tab selected over 'Recents'. Under 'Quick Start', there are six categories: Amazon Linux (with AWS logo), macOS (with Mac logo), Ubuntu (with ubuntu logo), Windows (with Microsoft logo), Red Hat (with Red Hat logo), and SUSE Linux (with SUSE logo). To the right of these is a 'Browse more AMIs' link with a magnifying glass icon, which includes a note about including AMIs from AWS Marketplace and the Community. On the far right, a vertical sidebar titled 'Summary' provides a quick overview of the instance configuration:

- Number of instances: 1
- Software Image (AMI): Canonical, Ubuntu, 24.04, a ami-0e86e20dae9224db8
- Virtual server type (instance type): t2.micro
- Firewall (security group): launch-wizard-5
- Storage (volumes): 1 volume(s) - 8 GiB

A callout box on the right side of the sidebar provides information about the Free tier, stating: "Free tier: In your first 750 hours of t2.micro usage in the Regions in which it's available, instances run at no charge. Includes up to 100 free tier AMIs per month, public IPv4 address usage, 1 million API calls per month, 30 GiB of EBS storage, 100 million I/Os, 1 GB of traffic to the internet."

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.

Recents **Quick Start**

Amazon Linux	macOS	Ubuntu	Windows	Red Hat	SUSE Linux Enterprise Server

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
 ami-0e86e20dae9224db8 (64-bit (x86)) / ami-096ea6a12ea24a797 (64-bit (Arm))
 Virtualization: hvm ENA enabled: true Root device type: ebs

Description
 Ubuntu Server 24.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Architecture	AMI ID	Username	Verified provider
64-bit (x86)	ami-0e86e20dae9224db8	ubuntu	

Summary

Number of instances: 1

Software Images: Canonical, Ubuntu, Red Hat, SUSE Linux Enterprise Server

Virtual server type: t2.micro

Firewall (security group): launch-wizard-1

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

Free tier
 750 hours of the Region are available for free tier AMI public IP traffic per month, up to 1 million requests per second, and 100 GB of internet bandwidth.

Cancel

Make sure to select the same key-pair login used in the exp9 machine.

The screenshot shows the AWS Launch Wizard configuration interface. On the left, there are two main sections: 'Key pair (login)' and 'Network settings'. In the 'Key pair (login)' section, a key pair named 'nagios_exp_9' is selected. In the 'Network settings' section, a VPC named 'vpc-07b6966cbfba88ee3' is chosen, and the 'Auto-assign public IP' option is enabled. A sidebar on the right provides information about the instance: it's using the Canonical AMI (ami-0e86e2c), has a t2.micro virtual service, and is associated with a free tier allowance of 750 hours. A 'Cancel' button is located at the bottom right of the sidebar.

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)

The screenshot shows the AWS Instances page with a table of running EC2 instances. The columns include Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. The instances listed are 'Master', 'node-1', 'node-2', 'nagios_host_e...', 'exp10client', all of which are currently running. The 'Master' instance is in the us-east-1b availability zone with the public IP ec2-3-82-156-160.com. The 'exp10client' instance is also running and is in the us-east-1b availability zone with the public IP ec2-54-173-58-143.co...

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
Master	i-0ab175e9c60cc3a23	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-3-82-156-160.com...
node-1	i-08ad30b7114767ca2	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-3-85-110-80.comp...
node-2	i-03c70d364fb762af5	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-54-226-209-38.co...
nagios_host_e...	i-0820376be204a7fcf	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-54-224-175-95.co...
exp10client	i-0994ca5a178801a54	Running	t2.micro	Initializing	View alarms +	us-east-1b	ec2-54-173-58-143.co...

EC2 > Instances > i-0994ca5a178801a54 > Connect to instance

Connect to instance Info

Connect to your instance i-0994ca5a178801a54 (exp10client) using any of these options

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

Instance ID
 i-0994ca5a178801a54 (exp10client)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is nagios_exp_9.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
 chmod 400 "nagios_exp_9.pem"
4. Connect to your instance using its Public DNS:
 ec2-54-173-58-143.compute-1.amazonaws.com

Command copied

ssh -i "nagios_exp_9.pem" ubuntu@ec2-54-173-58-143.compute-1.amazonaws.com

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

Host Client

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

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

PS C:\Users\Lenovo> ssh -i "C:\Users\Lenovo\Downloads\nagios_exp_9.pem" ubuntu@ec2-54-173-58-143.compute-1.amazonaws.com

The authenticity of host 'ec2-54-173-58-143.compute-1.amazonaws.com (54.173.58.143)' can't be established.
ED25519 key fingerprint is SHA256:IA3XH7f011spK084wDcZFmqRgNn0iJZ7itI2pBMmHP4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-173-58-143.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

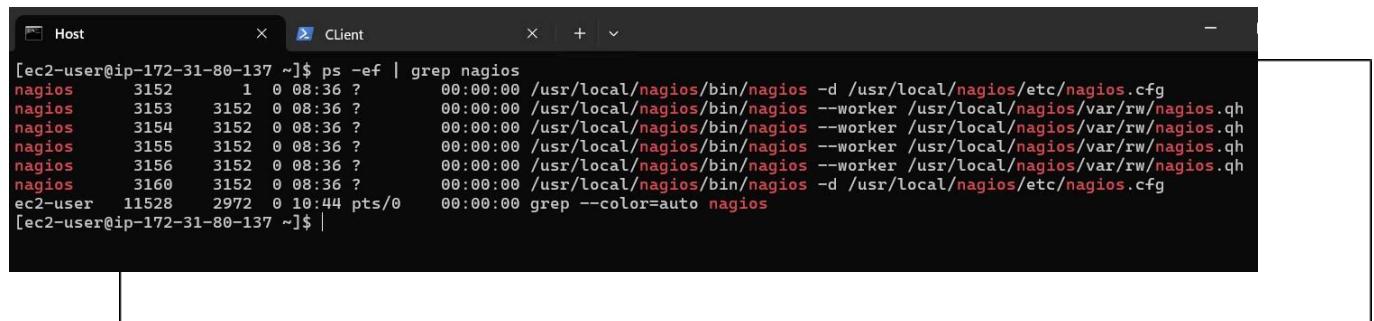
 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sat Sep 28 10:43:28 UTC 2024

System load: 0.01      Processes:          107
Usage of /:   22.8% of 6.71GB  Users logged in:    0
Memory usage: 19%        IPv4 address for enX0: 172.31.82.77
```

```
[root@ip-172-31-80-137 ec2-user]# mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts  
[root@ip-172-31-80-137 ec2-user]# ls
```

Perform the following commands



```
[ec2-user@ip-172-31-80-137 ~]$ ps -ef | grep nagios  
nagios 3152 1 0 08:36 ? 00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg  
nagios 3153 3152 0 08:36 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh  
nagios 3154 3152 0 08:36 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh  
nagios 3155 3152 0 08:36 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh  
nagios 3156 3152 0 08:36 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh  
nagios 3160 3152 0 08:36 ? 00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg  
ec2-user 11528 2972 0 10:44 pts/0 00:00:00 grep --color=auto nagios  
[ec2-user@ip-172-31-80-137 ~]$ |
```

sudo su

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

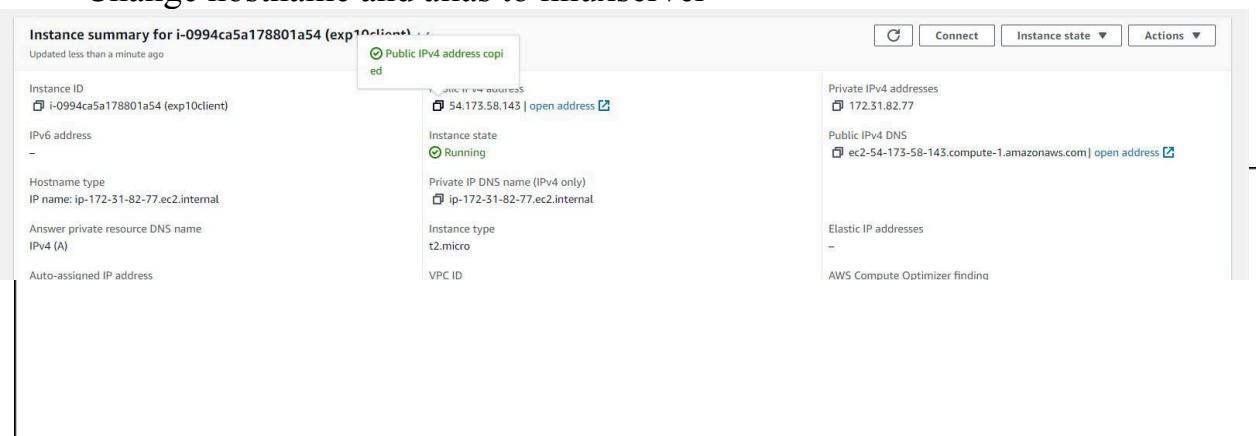
```
cp /usr/local/nagios/etc/objects/localhost.cfg  
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

```
[root@ip-172-31-80-137 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

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

```
[root@ip-172-31-80-137 ec2-user]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg|
```

Change hostname and alias to linuxserver



```

; HOST DEFINITION
#####
; Define a host for the local machine

define host {

    use          linux-server           ; Name of host template to use
                                ; This host definition will in>
                                ; in (or inherited by) the lin>

    host_name    linuxserver
    alias        linuxserver
    address     54.173.58.143
}

```

```

# Define an optional hostgroup for Linux machines

define hostgroup {

    hostgroup_name   linux-servers1      ; The name of the hostgroup
    alias            Linux Servers       ; Long name of the group
    members          localhost           ; Comma separated list of host>
}

```

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

host_name	localhost
service_description	PING

changed to

```

define service {

    use          local-service           ; Name of service template
    host_name    linuxserver
    service_description PING
    check_command check_ping!100.0,20%!500.0,60%
}

```

This is the last one

```
define service {
    use           local-service      ; Name of service template to >
    host_name     linuxserver
    service_description  HTTP
    check_command  check_http
    notifications_enabled 0
```

```
[root@ip-172-31-80-137 ec2-user]# nano /usr/local/nagios/etc/nagios.cfg
```

##Add this line below the opened nano interface where similar lines are commented.

```
GNU nano 5.8                               /usr/local/nagios/etc/nagios.cfg
# These are the object configuration files in which you define hosts,
# host groups, contacts, contact groups, services, etc.
# You can split your object definitions across several config files
# if you wish (as shown below), or keep them all in a single config file.

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

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

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

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

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

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

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

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

ctrl+o and enter for saving and ctrl+x to exit nano editor.

Verify configuration files

```
[root@ip-172-31-80-137 ec2-user]# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
Nagios Core 4.5.5
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-09-17
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 0 service dependencies
  Checked 0 host dependencies
  Checked 5 timeperiods
  Checking global event handlers...
  Checking obsessive compulsive processor commands...
  Checking misc settings...

  Total Warnings: 0
  Total Errors: 0

  Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-80-137 ec2-user]# |
```

Restart nagios
service. service
nagios restart

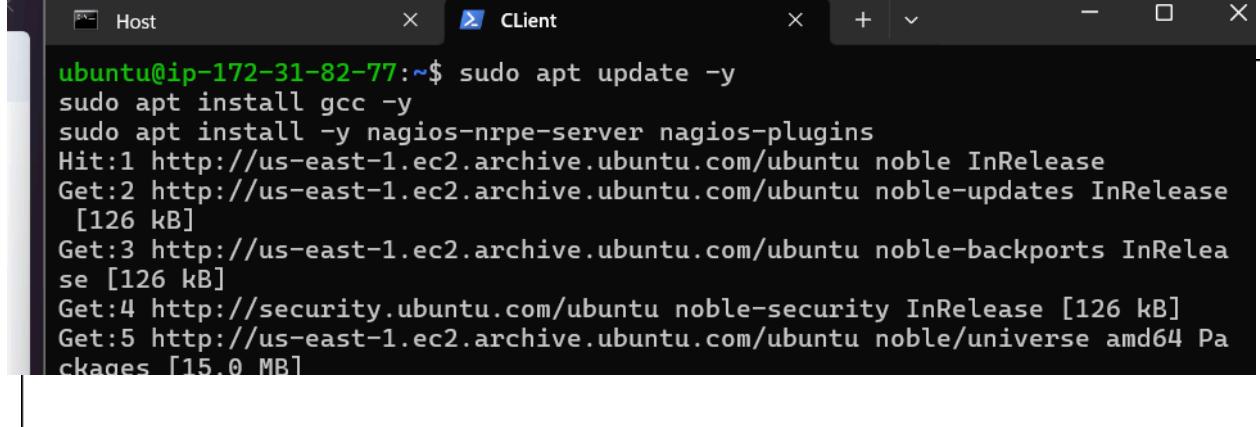
```
Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-80-137 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
[root@ip-172-31-80-137 ec2-user]# |
```

2) Go to client machine (ubuntu machine)

Perform the following commands

```
sudo apt update -y
```

```
sudo apt install gcc -y
```



```
ubuntu@ip-172-31-82-77:~$ sudo apt update -y
sudo apt install gcc -y
sudo apt install -y nagios-nrpe-server nagios-plugins
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
[126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
[126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Pa
ckages [15.0 MB]

Running kernel seems to be up-to-date.

Restarting services...

Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart serial-getty@ttyS0.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #1: sshd[990,1101]
ubuntu @ user manager service: systemd[996]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-82-77:~$ |
```

Open the nrpe.cfg file in nano editor
sudo nano /etc/nagios/nrpe.cfg

Under allowed_hosts, add the nagios host ip address (public)

```
# You can either supply a username or a UID.  
#  
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd  
nrpe_user=nagios  
  
# NRPE GROUP  
# This determines the effective group that the NRPE daemon should run as.  
# You can either supply a group name or a GID.  
#  
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd  
nrpe_group=nagios  
  
# ALLOWED HOST ADDRESSES  
# This is an optional comma-delimited list of IP address or hostnames  
# that are allowed to talk to the NRPE daemon. Network addresses with a bit  
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not curr  
# supported.  
#  
# Note: The daemon only does rudimentary checking of the client's IP  
# address. I would highly recommend adding entries in your /etc/hosts.allow  
# file to allow only the specified host to connect to the port  
# you are running this daemon on.  
#  
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd  
allowed_hosts=127.0.0.1,54.224.175.95  
  
# COMMAND ARGUMENT PROCESSING  
# This option determines whether or not the NRPE daemon will allow clients
```

again save and exit the nano editor.

3) Go to nagios dashboard and click on hosts

The screenshot shows the Nagios Core 4.5.5 dashboard. At the top right, it displays "Nagios® Core™ Version 4.5.5" and the date "September 17, 2024". A green checkmark indicates "Daemon running with PID 13935". On the left, a sidebar menu includes sections for General, Current Status (selected), Reports, and System. The Current Status section contains links for Tactical Overview, Map, Hosts, Services, and Host Groups. Below the sidebar, a "Get Started" box lists items like monitoring infrastructure, changing the look and feel, and extending Nagios with addons. A "Quick Links" box provides links to Nagios Library, Labs, Exchange, Support, and the company website. At the bottom, there are "Latest News" and "Don't Miss..." sections, along with copyright and license information.

Current Status

[Tactical Overview](#)

[Map](#)

[Hosts](#)

[Services](#)

[Host Groups](#)

4) Click on linux server

Current Network Status

Last Updated: Sat Sep 28 11:33:39 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged In as nagiosadmin

Host Status Totals				Service Status Totals			
Up	Down	Unreachable	Pending	Ok	Warning	Unknown	Critical
2	0	0	0	12	1	0	3
All Problems All Types				All Problems All Types			
0	2			4	16		

Host Status Details For All Host Groups

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	09-28-2024 11:29:10	0d 0h 8m 36s	PING OK - Packet loss = 0%, RTA = 1.18 ms
localhost	UP	09-28-2024 11:32:18	0d 3h 53m 7s	PING OK - Packet loss = 0%, RTA = 0.03 ms

Results 1 - 2 of 2 Matching Hosts

Host Information

Last Updated: Sat Sep 28 11:33:39 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged In as nagiosadmin

Host
linuxserver
(linuxserver)

Member of
No hostgroups

54.173.58.143

Host State Information

Host Status:	UP (for 0d 0h 8m 51s)
Status Information:	PING OK - Packet loss = 0%, RTA = 1.18 ms
Performance Data:	rta=1.19400ms;3000.000000;5000.000000;0.000000 pl=%;80;100;1
Current Attempt:	1/10 (HARD state)
Last Check Time:	09-28-2024 11:29:10
Check Type:	ACTIVE
Check Latency / Duration:	0.000 / 4.066 seconds
Next Scheduled Active Check:	09-28-2024 11:34:10
Last State Change:	09-28-2024 11:24:48
Last Notification:	N/A (notification 0)
Is This Host Flapping?	NO (0.00% state change)
In Scheduled Downtime?	NO
Last Update:	09-28-2024 11:33:37 (0d 0h 0m 2s ago)

Host Commands

- Locate host on map
- Disable active checks of this host
- Re-schedule the next check of this host
- Submit passive check result for this host
- Stop accepting passive checks for this host
- Stop obsessing over this host
- Disable notifications for this host
- Send custom host notification
- Schedule downtime for this host
- Schedule downtime for all services on this host
- Disable notifications for all services on this host
- Enable notifications for all services on this host
- Schedule a check of all services on this host
- Disable checks of all services on this host
- Enable checks of all services on this host
- Disable event handler for this host
- Disable flap detection for this host
- Clear flapping state for this host

Host Comments

Add a new comment

Entry Time	Author	Comment	Comment ID	Persistent	Type	Expires	Actions
This host has no comments associated with it							

5) Click on nagios services

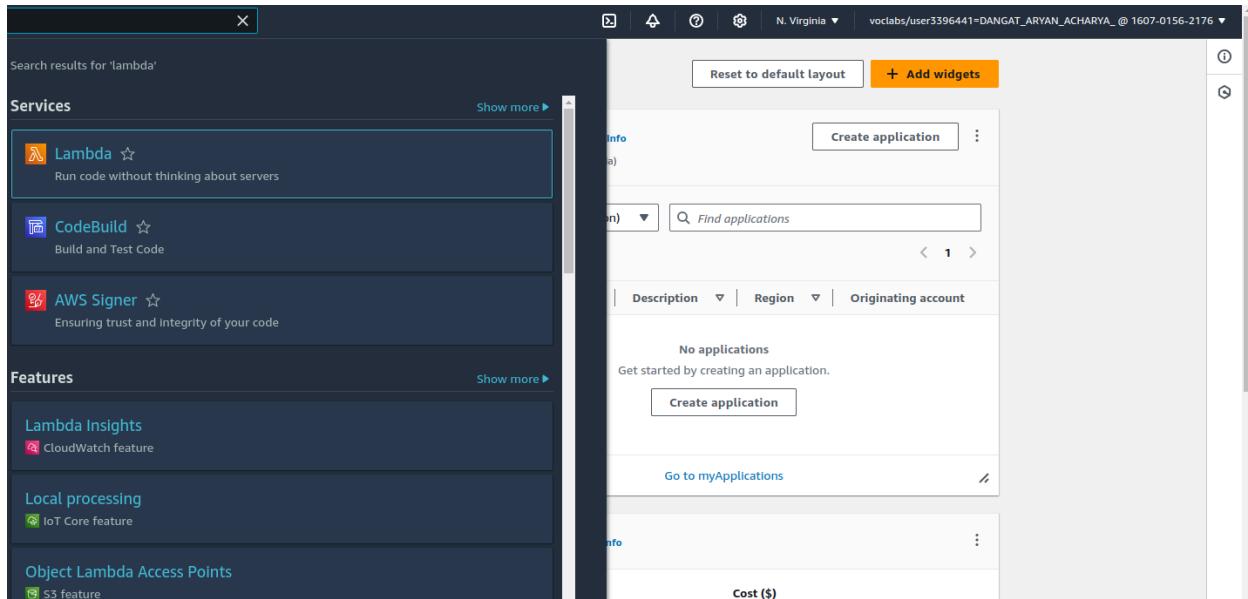
[Documentation](#)

Service Status Details For All Hosts						
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
linuxserver	Current Users	OK	09-28-2024 11:31:03	0d 0h 7m 55s	1/4	USERS OK - 2 users currently logged in
linuxserver	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
linuxserver	PING	OK	09-28-2024 11:32:18	0d 0h 6m 40s	1/4	PING OK - Packet loss = 0%, RTA = 1.03 ms
linuxserver	Root Partition	OK	09-28-2024 11:32:55	0d 0h 6m 3s	1/4	DISK OK - free space / 105 MB (75.23% used=98%)
linuxserver	SSH	CRITICAL	09-28-2024 11:33:33	0d 0h 5m 25s	1/4	SSH OK - OpenSSH_9_6p1 Ubuntu-13.4 (protocol 2.0)
linuxserver	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.
linuxserver	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
localhost	Current Users	OK	09-28-2024 11:30:17	0d 3h 52m 27s	1/4	USERS OK - 2 users currently logged in
localhost	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
localhost	PING	OK	09-28-2024 11:31:32	0d 3h 51m 12s	1/4	PING OK - Packet loss = 0%, RTA = 0.03 ms
localhost	Root Partition	OK	09-28-2024 11:32:09	0d 3h 50m 35s	1/4	DISK OK - free space / 105 MB (75.23% used=98%)
localhost	SSH	OK	09-28-2024 11:32:47	0d 3h 49m 57s	1/4	SSH OK - OpenSSH_8.7 (protocol 2.0)
localhost	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.
localhost	Total Processes	OK	09-28-2024 11:29:02	0d 3h 14m 56s	1/4	PROCS OK: 37 processes with STATE = RSZDT

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.

A screenshot of the AWS Lambda 'Get started' page. It features a large heading 'AWS Lambda' with the subtext 'lets you run code without thinking about servers.' Below this is a paragraph about Lambda's cost-efficiency and ease of use. To the right, there's a 'Get started' box containing the text 'Author a Lambda function from scratch, or choose from one of many preconfigured examples.' with a prominent 'Create a function' button. At the bottom, there's a 'How it works' section with tabs for '.NET', 'Java', 'Node.js' (which is selected), 'Python', 'Ruby', and 'Custom runtime'. A code snippet for a Node.js handler is shown:

```
1+ exports.handler = async (event) => {
2   console.log(event);
3   return 'Hello from Lambda!';
4 };
5 
```

- 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.

Create function Info

Choose one of the following options to create your function.

Author from scratch Start with a simple Hello World example.

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

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

Basic information

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

Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).

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

Architecture Info
Choose the instruction set architecture you want for your function code.
 x86_64
 arm64

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

Change default execution role

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

Tutorials

Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

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

[Learn more](#)

- Function is successfully created

pract1

Function overview Info

Successfully created the function **pract1**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

Code source Info

Code Test Monitor Configuration Aliases Versions

index.mjs

```

1 export const handler = async (event) => {
2     // TODO implement
3     const response = {
4         statusCode: 200,
5         body: JSON.stringify('Hello from Lambda!'),
6     };
7     return response;
}

```

Tutorials

Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

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

[Learn more](#)

The screenshot shows the AWS Lambda code editor interface. At the top, a green banner displays the message: "Successfully created the function PythonLambdaPractical11. You can now change its code and configuration. To invoke your function with a test event, choose *Test*." Below this, the "Code source" tab is selected, showing the file structure "PythonLambdaPractical11" with "lambda_function.py" open. The code in "lambda_function.py" is:

```

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

```

The interface includes tabs for "File", "Edit", "Find", "View", "Go", "Tools", "Window", "Test", and "Deploy". A sidebar on the left shows the environment variables. At the bottom, there are tabs for "Code properties" and "Info", along with links for "cloudShell", "Feedback", and "Cookie preferences".

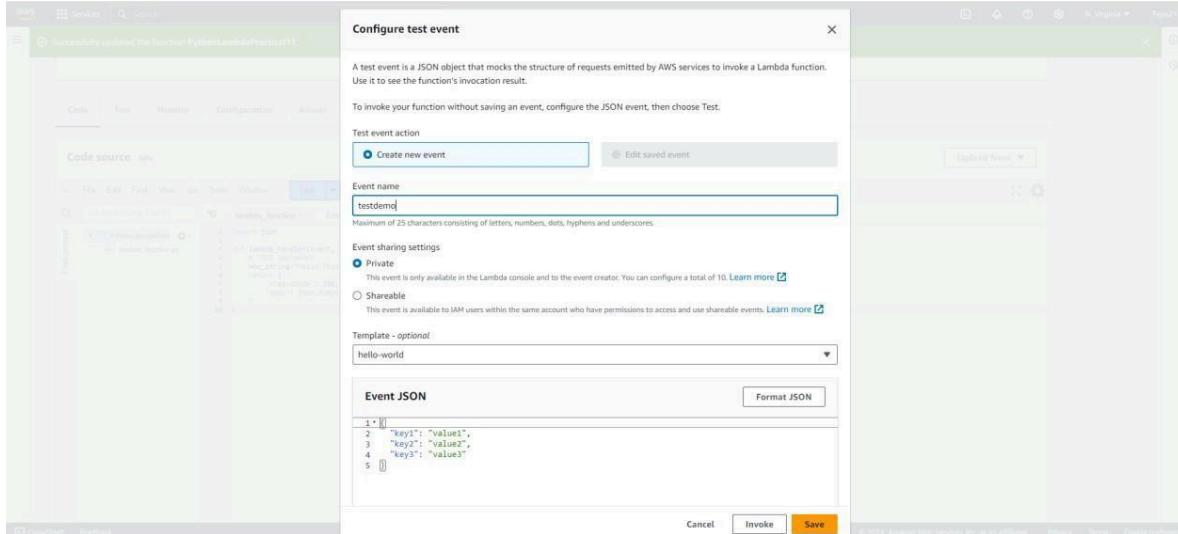
- To change the configuration, open up the Configuration tab and under General Configuration, choose Edit. Here, you can enter a description and change Memory and Timeout. I've changed the Timeout period to 1 sec since that is sufficient for now.

The screenshot shows the AWS Lambda function configuration page for "pract1". Under the "Basic settings" tab, the following configuration is visible:

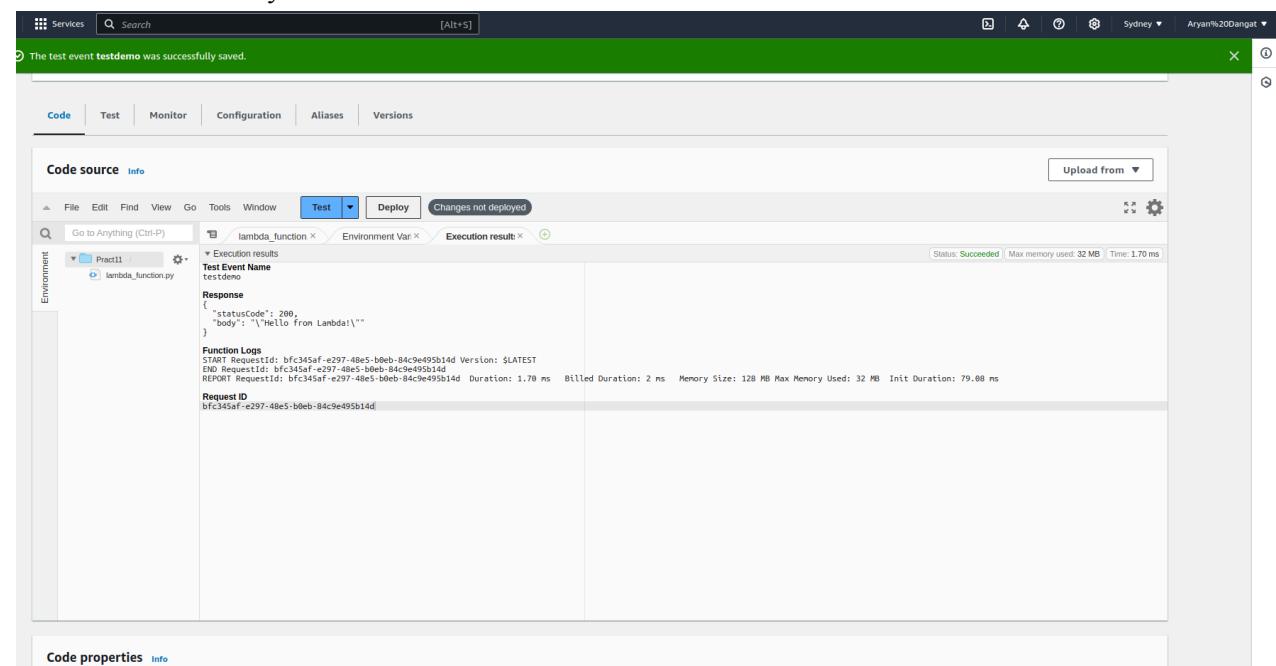
- Description - optional:** (Empty input field)
- Memory - Info:** Set to 128 MB. Description: "Your function is allocated CPU proportional to the memory configured." Options: Set memory to between 128 MB and 10240 MB.
- Ephemeral storage - Info:** Set to 512 MB. Description: "You can configure up to 10 GB of ephemeral storage (/tmp) for your function." Options: Set ephemeral storage (/tmp) to between 512 MB and 10240 MB.
- SnapStart - Info:** Set to None. Description: "Reduce startup time by having Lambda cache a snapshot of your function after the function has initialized. To evaluate whether your function code is resilient to snapshot operations, review the SnapStart compatibility considerations." Options: None, Java 11, Java 17, Java 21.
- Timeout:** Set to 1 sec. Description: "The maximum execution time for this function." Options: 0 min, 1 sec.
- Execution role:** Set to "Use an existing role". Description: "Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console." Options: Use an existing role (selected), Create a new role from AWS policy templates.
- Existing role:** Set to "service-role/pract1-role-tz60jys3". Description: "Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs." Options: service-role/pract1-role-tz60jys3.

On the right side of the configuration page, there is a "Tutorials" section titled "Create a simple web app" with a "Start tutorial" button. It also includes a "Learn how to Implement common use cases in AWS Lambda" link and a "Learn more" link.

- 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 res



Aryan Dangat D15A-12

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

The screenshot shows the AWS IAM Dashboard. In the 'Security recommendations' section, there are two items: 'Add MFA for root user' (with a note to enable multi-factor authentication for the root user) and 'Root user has no active access keys' (with a note to use access keys attached to an IAM user instead of the root user). The 'IAM resources' section displays 0 User groups, 0 Users, 8 Roles, 3 Policies, and 0 Identity providers. The 'What's new' section lists recent updates from the AWS IAM Access Analyzer. On the right side, there are sections for 'AWS Account' (Account ID: 637425278915, Sign-In URL: https://637425278915.signin.aws.amazon.com/console), 'Quick Links' (My security credentials, Policy simulator), and 'Tools' (Policy simulator).

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

The screenshot shows the 'Roles' page in the AWS IAM console. It lists eight roles: 'aryan', 'AWSCodePipelineServiceRole-ap-southeast-2-AD-CICD', 'AWSServiceRoleForAmazonSSM', 'AWSServiceRoleForSupport', 'AWSServiceRoleForTrustedAdvisor', 'CodeDeploy', 'pract1-role-tz6ojvs3', and 'Pract11-role-pvk413d'. Each role entry includes its name, trusted entities (such as AWS Services like ec2, codepipeline, ssm, support, trustedadvisor, codedeploy, lambda), and last activity date.

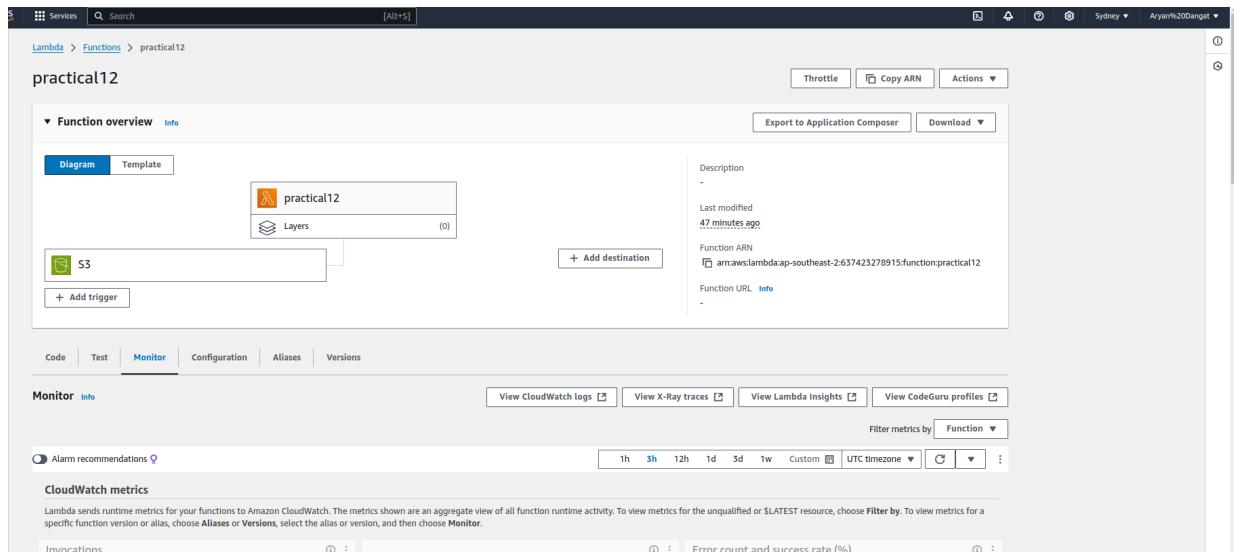
After successful attachment of policy you will see something like this you will be able to see the updated policies.

The screenshot shows the AWS IAM Roles page. The role 'Pract11-role-pvlk413d' has been created and is displayed. It has a creation date of October 18, 2024, at 06:55 (UTC+05:30), and it was last active 2 days ago. The ARN is arn:aws:iam::657423278915:role/service-role/Pract11-role-pvlk413d, and the maximum session duration is 1 hour. The 'Permissions' tab is selected, showing three managed policies attached: AmazonS3ReadOnlyAccess, AWSLambdaBasicExecutionRole-14046366-cb99-4509-b0..., and CloudWatchFullAccess. There is also a 'Permissions boundary (not set)' section.

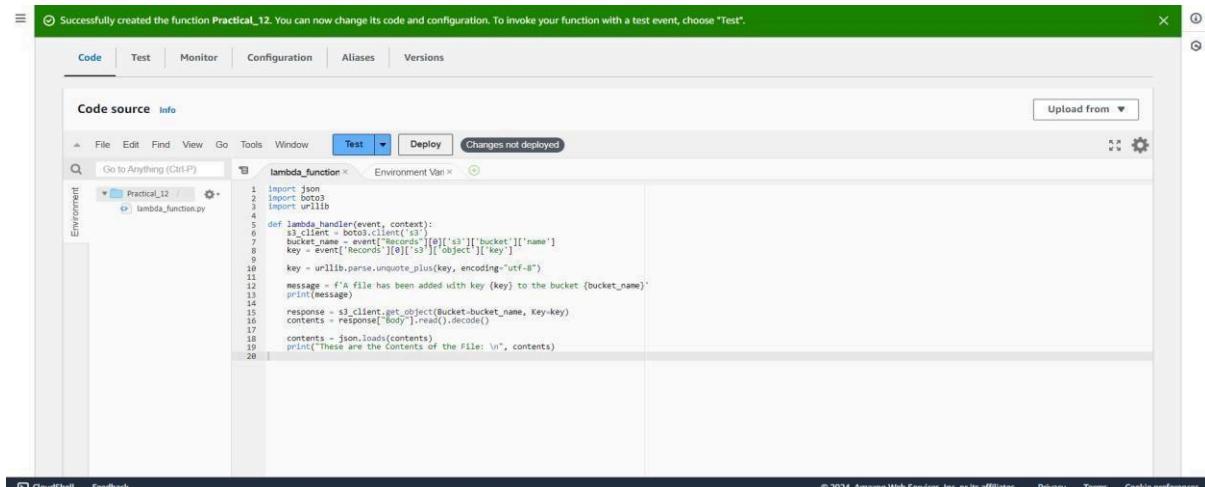
- Open up AWS Lambda and create a new Python function. Under Execution Role, choose the existing role, then select the one which was previously created and to which we just added permissions.

The screenshot shows the AWS Lambda 'Create Function' page. A new function named 'Pract12' is being created. The runtime is set to Python 3.12, and the architecture is x86_64. Under 'Permissions', the execution role is set to 'service-role/Pract11-role-pvlk413d'. In the 'Additional Configurations' section, there are fields for code signing, function URL, tags, and Amazon VPC access, all of which are currently empty. At the bottom, there are 'Cancel' and 'Create function' buttons.

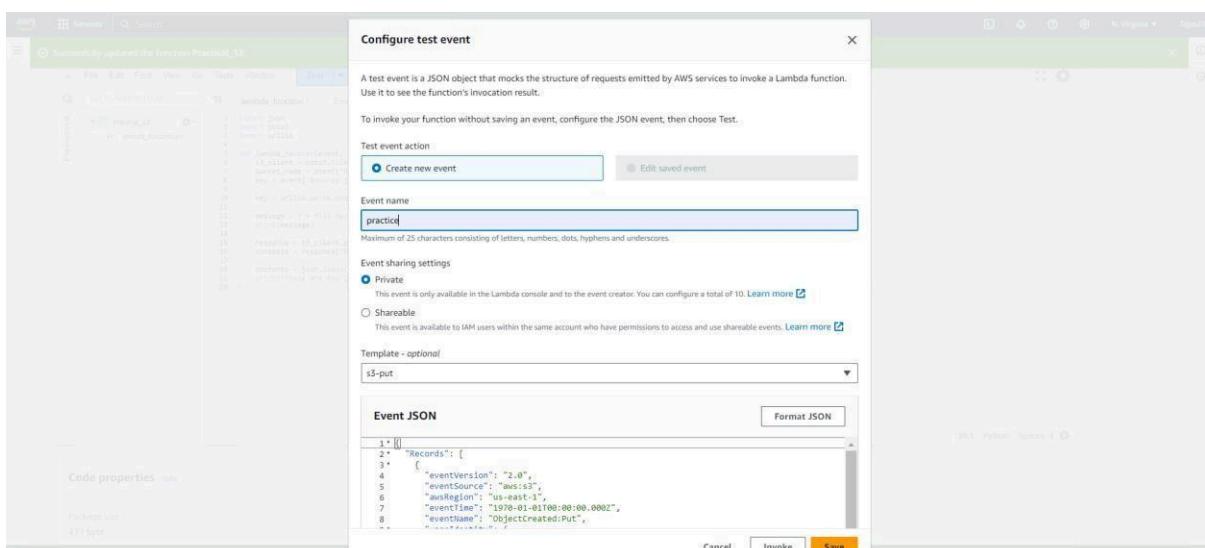
- The function is now successfully created and running



- 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.

The screenshot shows the Amazon S3 console interface. At the top, there's a header with account information and navigation links. Below it, a section titled 'Account snapshot - updated every 24 hours' provides storage usage and activity trends. The main area displays 'General purpose buckets (1)'. A table lists one bucket: 'codepipeline-ap-southeast-2-511632003555' located in 'Asia Pacific (Sydney) ap-southeast-2'. The 'Create bucket' button is prominently displayed at the top right of the list area.

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

This screenshot shows the 'Create bucket' wizard in the Amazon S3 console. It's on the 'General configuration' step. The 'Bucket name' field contains 'adidevopsxp12'. The 'Bucket type' dropdown is set to 'General purpose'. The 'Object ownership' section shows 'ACLs disabled (recommended)'. Other options like 'Copy settings from existing bucket' and 'Choose bucket' are also visible.

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

This screenshot shows the 'Event notifications' section for a specific bucket. It lists one notification rule named 's3req' which triggers on 'Put' events. The destination for this notification is a Lambda function named 'practical12'. There are buttons for 'Edit', 'Delete', and 'Create event notification'.

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

The screenshot shows the 'Create event notification' page in the Amazon S3 console. The 'Event name' field contains 's3req'. The 'Prefix - optional' field contains 'images/'. The 'Suffix - optional' field contains '.jpg'.

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

The screenshot shows the 'Destination' configuration page. The 'Destination' section is selected, showing options for 'Lambda function', 'SNS topic', and 'SQS queue'. 'Lambda function' is selected. In the 'Specify Lambda function' section, 'Choose from your Lambda functions' is selected. A dropdown menu shows 'practical12' as the chosen function.

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

The screenshot shows the Lambda function 'practical12' in the AWS Lambda console. The 'Function overview' section shows the function name 'practical12'. Under the 'Triggers' section, there is a card for 'S3' with '(0)' triggers. Other tabs include 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'.

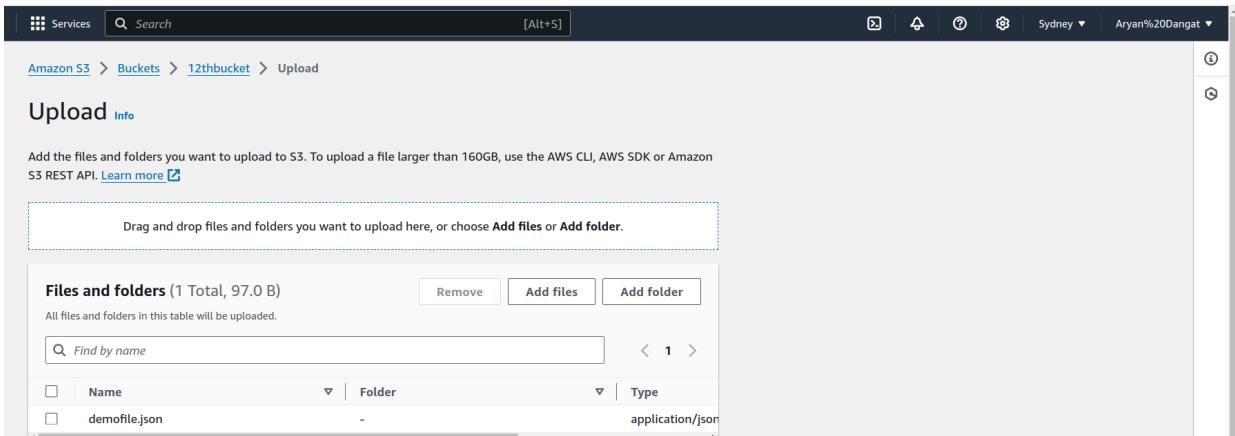
- Now, create a demofile JSON file locally.

```

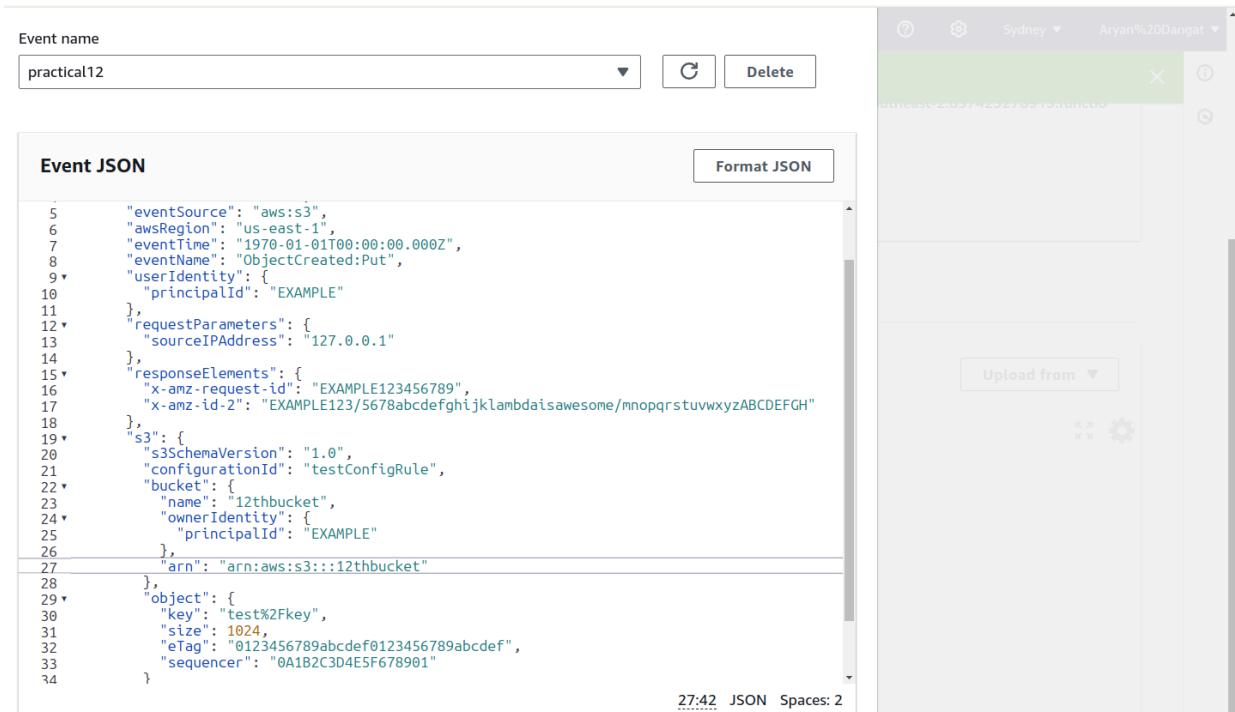
1   "firstname" : "Aryan",
2   "lastname" : "Dangat",
3   "age" : "19",
4   "gender" : "Male"
5
6

```

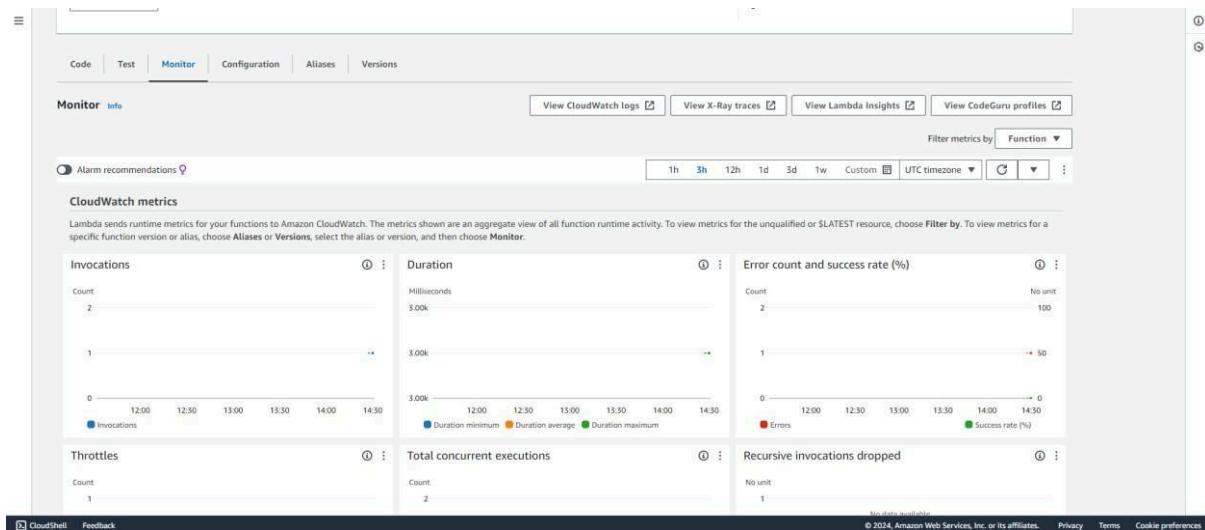
- Go back to your S3 Bucket and click on Add Files to upload a new file. Select the demofile data file from your computer and click Upload.



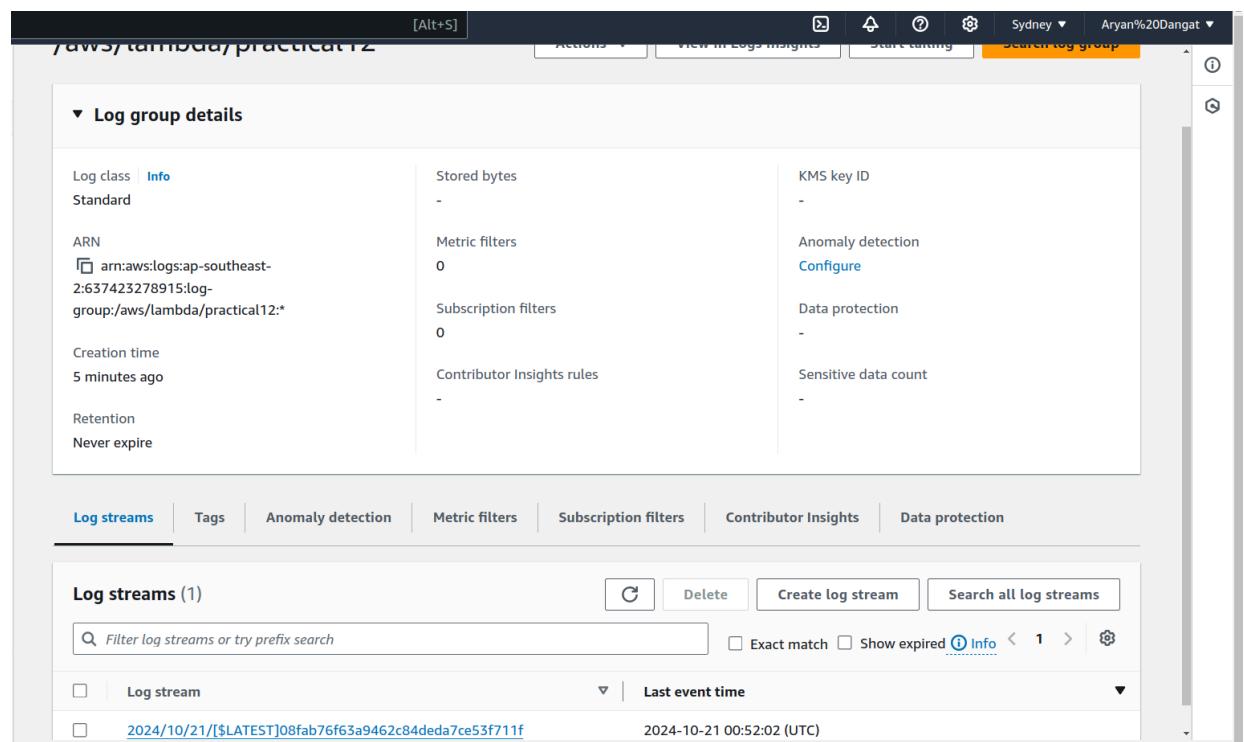
- After this make the necessary changes in the Test configuration file which we created it previously by replacing the Bucket Name and the ARN of Bucket.



- 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.

The screenshot shows the AWS CloudWatch Log Events interface. At the top, there's a navigation bar with links for CloudWatch, Log groups, /aws/lambda/practical12, and a timestamp 2024/10/21/[\$.LATEST]. On the far right of the navigation bar are buttons for Sydney (selected), Aryan%20Dangat, and a gear icon. Below the navigation bar is a header bar with tabs for Log events, Actions, Start tailing, and Create metric filter. There are also buttons for Filter events - press enter to search, Clear, and time range filters (1m, 30m, 1h, 12h, Custom, UTC timezone). To the right of these are Display and a gear icon buttons. The main content area is titled "Log events" and contains a table with two columns: "Timestamp" and "Message". The table has 10 rows of log entries. The first row shows a single entry: "No older events at this moment. [Retry](#)". The subsequent nine rows show multiple entries per timestamp, representing different requests. Each entry includes the timestamp, request ID, version, and some descriptive text like "Runtime Version: python:3.12.v36 Runtime Version ARN: arn:aws:lambda:ap-southeast-2::runtime:188...". The last row also indicates "No newer events at this moment. Auto retry paused. [Resume](#)".

Timestamp	Message
	No older events at this moment. Retry
2024-10-21T00:52:02.389Z	INIT_START Runtime Version: python:3.12.v36 Runtime Version ARN: arn:aws:lambda:ap-southeast-2::runtime:188...
2024-10-21T00:52:02.476Z	START RequestId: 8206db2f-c544-4ab7-b80a-73794df63b3f Version: \$.LATEST
2024-10-21T00:52:02.478Z	END RequestId: 8206db2f-c544-4ab7-b80a-73794df63b3f
2024-10-21T00:52:02.478Z	REPORT RequestId: 8206db2f-c544-4ab7-b80a-73794df63b3f Duration: 1.98 ms Billed Duration: 2 ms Memory Size:...
2024-10-21T00:52:24.834Z	START RequestId: abcfa8c7-8f47-4cc2-9897-5180cbce83e9 Version: \$.LATEST
2024-10-21T00:52:24.849Z	END RequestId: abcfa8c7-8f47-4cc2-9897-5180cbce83e9
2024-10-21T00:52:24.849Z	REPORT RequestId: abcfa8c7-8f47-4cc2-9897-5180cbce83e9 Duration: 1.47 ms Billed Duration: 2 ms Memory Size:...
2024-10-21T00:52:32.895Z	START RequestId: 41298836-2ce8-472c-a46f-1ccdb00dc199 Version: \$.LATEST
2024-10-21T00:52:32.897Z	END RequestId: 41298836-2ce8-472c-a46f-1ccdb00dc199
2024-10-21T00:52:32.898Z	REPORT RequestId: 41298836-2ce8-472c-a46f-1ccdb00dc199 Duration: 1.39 ms Billed Duration: 2 ms Memory Size:...
	No newer events at this moment. Auto retry paused. Resume