

Advanced DevOps Experiment 3

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

Step 1: Go to AWS Academia in services select EC2 and create 3 instance with instance type t2.medium and names as node1, node2 and master

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IP |
|--------|---------------------|----------------|---------------|-------------------|---------------|-------------------|-----------|
| node2 | i-0d15c704d5359f607 | Pending | t2.medium | - | View alarms + | us-east-1c | ec2-44-20 |
| master | i-0ed4c2d786c3e438f | Running | t2.medium | 2/2 checks passed | View alarms + | us-east-1c | ec2-18-20 |
| node1 | i-0ec932e19bc2d5a2f | Running | t2.medium | Initializing | View alarms + | us-east-1c | ec2-3-84 |
| node1 | i-092007b9d8e24ec14 | Terminated | t2.micro | - | View alarms + | us-east-1a | - |
| master | i-0c96d190403326eb5 | Terminated | t2.micro | - | View alarms + | us-east-1a | - |
| node2 | i-0a539b2617389125f | Terminated | t2.micro | - | View alarms + | us-east-1a | - |

Port 22 (SSH) is open to all IPv4 addresses
Port 22 (SSH) is currently open to all IPv4 addresses, indicated by **0.0.0.0/0** in the inbound rule in your security group. For increased security, consider restricting access to only the EC2 Instance Connect service IP addresses for your Region: 18.206.107.24/29. [Learn more](#).

EC2 Instance Connect Session Manager SSH client EC2 serial console

Instance ID: i-092007b9d8e24ec14 (node1)

Connection Type:

- Connect using EC2 Instance Connect
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.
- Connect using EC2 Instance Connect Endpoint
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IPv4 address: 54.162.237.253

Username: ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Step 2: Select and connect each instance and run the following commands inside the console of each instance.

```
sudo su
yum install docker -y
systemctl start docker
docker docker
--version
yum repolist
```

Amazon Linux 2023

<https://aws.amazon.com/linux/amazon-linux-2023>

```
[ec2-user@ip-172-31-33-243 ~]$ sudo su
[root@ip-172-31-33-243 ec2-user]# yum install docker -y
Last metadata expiration check: 0:10:44 ago on Wed Sep 18 13:13:43 2024.
Dependencies resolved.
```

| Package | Architecture | Version | Repository | Size |
|--------------------------|--------------|-----------------------|-------------|-------|
| Installing: | | | | |
| docker | x86_64 | 25.0.6-1.amzn2023.0.2 | amazonlinux | 44 M |
| Installing dependencies: | | | | |
| containerd | x86_64 | 1.7.20-1.amzn2023.0.1 | amazonlinux | 35 M |
| iptables-libc | x86_64 | 1.8.8-3.amzn2023.0.2 | amazonlinux | 401 K |
| iptables-nft | x86_64 | 1.8.8-3.amzn2023.0.2 | amazonlinux | 183 K |
| libcgroup | x86_64 | 3.0-1.amzn2023.0.1 | amazonlinux | 75 K |
| libnetfilter_conntrack | x86_64 | 1.0.8-2.amzn2023.0.2 | amazonlinux | 59 K |
| libnftnl | x86_64 | 1.0.1-19.amzn2023.0.2 | amazonlinux | 30 K |
| libnfntnl | x86_64 | 1.2.2-2.amzn2023.0.2 | amazonlinux | 84 K |
| pigz | x86_64 | 2.5-1.amzn2023.0.3 | amazonlinux | 83 K |
| runc | x86_64 | 1.1.13-1.amzn2023.0.1 | amazonlinux | 3.2 M |

Transaction Summary

Install 10 Packages

i-0a539b2617389125f (node2)

PublicIPs: 107.21.35.198 PrivateIPs: 172.31.33.243

aws Services Search [Alt+S] N. Virginia vocabs/user5413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9795

```
Installing : libnftnl-1.2.2-2.amzn2023.0.2.x86_64
Installing : libnftnlink-1.0.1-19.amzn2023.0.2.x86_64
Installing : libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64
Installing : iptables-libc-1.8.8-3.amzn2023.0.2.x86_64
Installing : iptables-nft-1.8.8-3.amzn2023.0.2.x86_64
Running scriptlet: iptables-nft-1.8.8-3.amzn2023.0.2.x86_64
Installing : libcgroup-3.0-1.amzn2023.0.1.x86_64
Running scriptlet: docker-25.0.6-1.amzn2023.0.2.x86_64
Installing : docker-25.0.6-1.amzn2023.0.2.x86_64
Running scriptlet: docker-25.0.6-1.amzn2023.0.2.x86_64
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.

Verifying : containerd-1.7.20-1.amzn2023.0.1.x86_64
Verifying : docker-25.0.6-1.amzn2023.0.2.x86_64
Verifying : iptables-libc-1.8.8-3.amzn2023.0.2.x86_64
Verifying : iptables-nft-1.8.8-3.amzn2023.0.2.x86_64
Verifying : libcgroup-3.0-1.amzn2023.0.1.x86_64
Verifying : libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64
Verifying : libnftnl-1.2.2-2.amzn2023.0.2.x86_64
Verifying : libnftnl-1.2.2-2.amzn2023.0.2.x86_64
Verifying : pigz-2.5-1.amzn2023.0.3.x86_64
Verifying : runc-1.1.13-1.amzn2023.0.1.x86_64

Installed:
  containerd-1.7.20-1.amzn2023.0.1.x86_64           docker-25.0.6-1.amzn2023.0.2.x86_64
  iptables-libc-1.8.8-3.amzn2023.0.2.x86_64         libcgroup-3.0-1.amzn2023.0.1.x86_64
  libnftnl-1.2.2-2.amzn2023.0.2.x86_64             libnftnl-1.2.2-2.amzn2023.0.2.x86_64
  runc-1.1.13-1.amzn2023.0.1.x86_64

Complete!
[root@ip-172-31-33-243 ec2-user]# systemctl start docker
[root@ip-172-31-33-243 ec2-user]# docker --version
Docker version 25.0.5, build 5dc9bc
[root@ip-172-31-33-243 ec2-user]# 
```

i-0a539b2617389125f (node2)

PublicIPs: 107.21.35.198 PrivateIPs: 172.31.33.243

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

Step 3: Now, go to the following link <https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-kubeadm/> and scroll down and select Red-Hat based distributions tab copy all the commands on by one in each console of instance.

 **kubernetes** Documentation Kubernetes Blog Training Partners Community Case Studies Versions English Search this site

```
aws Services Search [Alt+S] N. Virginia v vocabs/user3413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9793
[root@ip-172-31-33-243 ec2-user]# yum repolist
repo id repo name
amazonlinux Amazon Linux 2023 repository
kernel-livelpatch Amazon Linux 2023 Kernel Livepatch repository

[root@ip-172-31-33-243 ec2-user]# sudo setenforce 0
sudo: see '/sbin/SELINUX-enforcing' /SELINUX-permissive' /etc/selinux/config
[root@ip-172-31-33-243 ec2-user]# cat <>EOF | sudo tee /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=kubernetes
baseurl=https://pkgs.k8s.io/core:/stable:v1.31/rpm/
enabled=1
gpgcheck=1
gpgkey=https://pkgs.k8s.io/core:/stable:v1.31/rpm/repodata/repo-md.xml.key
exclude=kublet kubeadm kubectl cri-tools kubernetes-cni
EOF

[kubernetes]
name=Kubernetes
baseurl=https://pkgs.k8s.io/core:/stable:v1.31/rpm/
enabled=1
gpgcheck=1
gpgkey=https://pkgs.k8s.io/core:/stable:v1.31/rpm/repodata/repo-md.xml.key
exclude=kublet kubeadm kubectl cri-tools kubernetes-cni
[root@ip-172-31-33-243 ec2-user]# sudo yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes
Dependencies resolved.
54 kB/s | 9.4 kB 00:00
Package Architecture Version Repository Size
Installing:
kubeadm x86_64 1.31.1-150500.1.1 kubernetes 11 M
kubectl x86_64 1.31.1-150500.1.1 kubernetes 11 M
kubelet x86_64 1.31.1-150500.1.1 kubernetes 15 M
Installing dependencies:
comtrack-tools x86_64 1.4.6-.2.amzn2023.0.2 amazonlinux 208 k

i-0a539b2617389125f (node2)
PublicIPs: 107.21.35.198 PrivateIPs: 172.31.33.243
```

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

Step 4: Now, run the following command in the master instance -
kubeadm init

```
[root@ip-172-31-93-102 ec2-user]# kubeadm init
[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
[WARNING FileExisting-socat]: socat not found in system path
[WARNING FileExisting-tc]: tc not found in system path
[preflight] Pulling image required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your internet connection
[preflight] You can also perform this action beforehand using 'kubeadm config images pull'
W0918 14:21:55.805697    28020 checks.go:846] detected that the sandbox image "registry.k8s.io/pause:3.8" of the container runtime is inconsistent with that used by kubeadm. It is recommended to use "--registry=k8s.io/pause:3.10" as the CRI sandbox image.
[certs] Using certificateBir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [ip-172-31-93-102.ec2.internal kubernetes kubernetes.default kubernetes.default.svc kubernetes.default.svc.cluster.local] and IPs [10.96.0.1 172.31.93.102]
[certs] Generating "apiserver-kubelet-client" certificate and key
[certs] Generating "front-proxy-ca" certificate and key
[certs] Generating "front-proxy-client" certificate and key
[certs] Generating "etcd/ca" certificate and key
[certs] Generating "etcd/server" certificate and key
[certs] etcd/server serving cert is signed for DNS names [ip-172-31-93-102.ec2.internal localhost] and IPs [172.31.93.102 127.0.0.1 ::1]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [ip-172-31-93-102.ec2.internal localhost] and IPs [172.31.93.102 127.0.0.1 ::1]
[certs] Generating "etcd/healthcheck-client" certificate and key
[certs] Generating "apiserver-etcd-client" certificate and key
[certs] Generating "sa" key and public key
[kubeconfig] Using kubeconfig folder "/etc/kubernetes"
[kubeconfig] Writing "admin.conf" kubeconfig file
[kubeconfig] Writing "super-admin.conf" kubeconfig file
[kubeconfig] Writing "kubelet.conf" kubeconfig file
[kubeconfig] Writing "controller-manager.conf" kubeconfig file
[kubeconfig] Writing "scheduler.conf" kubeconfig file
[etcd] Creating static Pod manifest for local etcd in "/etc/kubernetes/manifests"
[control-plane] Using manifest folder "/etc/kubernetes/manifests"
```

i-0ed4c2d736c3e438f (master)
PublicIPs: 18.208.183.159 PrivateIPs: 172.31.93.102

Step 5: Now, run the following commands in master instance's console –

- mkdir -p \$HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf \$HOME/.kube/config
sudo chown \$(id -u):\$(id -g) \$HOME/.kube/config
- export KUBECONFIG=/etc/kubernetes/admin.conf
- kubeadm join 172.31.93.102:6443 --token 6ccgvw.o10vq5f2n5d9fa42 \
--discovery-token-ca-cert-hash
sha256:1bbcc9939e895e8de0e0ddd7ec72d881a9ef3b8f51a42f3145857e54b13c3818

```
To start using your cluster, you need to run the following as a regular user:  
mkdir -p $HOME/.kube  
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config  
sudo chown $(id -u):$(id -g) $HOME/.kube/config  
Alternatively, if you are the root user, you can run:  
export KUBECONFIG=/etc/kubernetes/admin.conf  
You should now deploy a pod network to the cluster.  
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:  
https://kubernetes.io/docs/concepts/cluster-administration/addons/  
Then you can join any number of worker nodes by running the following on each as root:  
kubeadm join 172.31.93.102:6443 --token 6ccgvw.o10vq5f2n5d9fa42 \  
--discovery-token-ca-cert-hash sha256:1bbcc9939e895e8de0e0ddd7ec72d881a9ef3b8f51a42f3145857e54b13c3818  
[root@ip-172-31-93-102 ec2-user]# mkdir -p $HOME/.kube  
[root@ip-172-31-93-102 ec2-user]# sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config  
[root@ip-172-31-93-102 ec2-user]# sudo chown $(id -u):$(id -g) $HOME/.kube/config  
[root@ip-172-31-93-102 ec2-user]# export KUBECONFIG=/etc/kubernetes/admin.conf  
[root@ip-172-31-93-102 ec2-user]# kubeadm join 172.31.93.102:6443 --token 6ccgvw.o10vq5f2n5d9fa42 \  

```

i-0ed4c2d736c3e438f (master)
PublicIPs: 18.208.183.159 PrivateIPs: 172.31.93.102

Step 6: Run this command in node1 and node2 -

```
kubeadm join 172.31.93.102:6443 --token 6ccgvw.o10vq5f2n5d9fa42 \  
--discovery-token-ca-cert-hash sha256:1bbcc9939e895e8de0e0ddd7ec72d881a9ef3b8f51a42f3145857e54b13c3818
```

```

aws Services Search [Alt+S] N. Virginia vclabs/user3413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9793 ▾
Installing : conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64 6/9
Running scriptlet: conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64 6/9
Installing : kubelet-1.31.1-150500.1.1.x86_64 7/9
Running scriptlet: kubelet-1.31.1-150500.1.1.x86_64 7/9
Installing : kubeadm-1.31.1-150500.1.1.x86_64 8/9
Installing : kubectl-1.31.1-150500.1.1.x86_64 9/9
Running scriptlet: kubectl-1.31.1-150500.1.1.x86_64 9/9
Verifying : conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64 1/9
Verifying : libnetfilter_cthelper-1.0.0-21.amzn2023.0.2.x86_64 2/9
Verifying : libnetfilter_cttimeout-1.0.0-19.amzn2023.0.2.x86_64 3/9
Verifying : libnetfilter_queue-1.0.5-2.amzn2023.0.2.x86_64 4/9
Verifying : cri-tools-1.31.1-150500.1.1.x86_64 5/9
Verifying : kubeadm-1.31.1-150500.1.1.x86_64 6/9
Verifying : kubectl-1.31.1-150500.1.1.x86_64 7/9
Verifying : kubelet-1.31.1-150500.1.1.x86_64 8/9
Verifying : kubernetes-cni-1.5.1-150500.1.1.x86_64 9/9
Installed:
conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64      cri-tools-1.31.1-150500.1.1.x86_64      kubeadm-1.31.1-150500.1.1.x86_64
kubectl-1.31.1-150500.1.1.x86_64      kubelet-1.31.1-150500.1.1.x86_64      kubernetes-cni-1.5.1-150500.1.1.x86_64
libnetfilter_cthelper-1.0.0-21.amzn2023.0.2.x86_64  libnetfilter_cttimeout-1.0.0-19.amzn2023.0.2.x86_64  libnetfilter_queue-1.0.5-2.amzn2023.0.2.x86_64
Complete!
[root@ip-172-31-95-221 ec2-user]# sudo systemctl enable --now kubelet
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /usr/lib/systemd/system/kubelet.service.
[root@ip-172-31-95-221 ec2-user]# kubeadm join 172.31.93.102:6443 --token 6ccgw.o10vg5f2n5d9fa42 \
--discovery-token-ca-cert-hash sha256:lbbcc9939e895e8de0e0dd77ec72d881a9ef3b8f51a42f3145857e54b13c3818
[preflight] Running pre-flight checks
[WARNING FileExisting-socat]: socat not found in system path
[WARNING FileExisting-tc]: tc not found in system path
error execution phase preflight: couldn't validate the identity of the API Server: failed to request the cluster-info ConfigMap: Get "https://172.31.93.102:6443/api/v1/namespaces/kube-public/configmaps/cluster-info?timeout=10s": context deadline exceeded
to see the stack trace of this error execute with --v=5 or higher
[root@ip-172-31-95-221 ec2-user]# 
```

i-0d15c704d5359f607 (node2)

PublicIPs: 44.201.192.9 PrivateIPs: 172.31.95.221

```

aws Services Search [Alt+S] N. Virginia vclabs/user3413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9793 ▾
Installing : conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64 6/9
Running scriptlet: conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64 6/9
Installing : kubelet-1.31.1-150500.1.1.x86_64 7/9
Running scriptlet: kubelet-1.31.1-150500.1.1.x86_64 7/9
Installing : kubeadm-1.31.1-150500.1.1.x86_64 8/9
Installing : kubectl-1.31.1-150500.1.1.x86_64 9/9
Running scriptlet: kubectl-1.31.1-150500.1.1.x86_64 9/9
Verifying : conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64 1/9
Verifying : libnetfilter_cthelper-1.0.0-21.amzn2023.0.2.x86_64 2/9
Verifying : libnetfilter_cttimeout-1.0.0-19.amzn2023.0.2.x86_64 3/9
Verifying : libnetfilter_queue-1.0.5-2.amzn2023.0.2.x86_64 4/9
Verifying : cri-tools-1.31.1-150500.1.1.x86_64 5/9
Verifying : kubeadm-1.31.1-150500.1.1.x86_64 6/9
Verifying : kubectl-1.31.1-150500.1.1.x86_64 7/9
Verifying : kubelet-1.31.1-150500.1.1.x86_64 8/9
Verifying : kubernetes-cni-1.5.1-150500.1.1.x86_64 9/9
Installed:
conntrack-tools-1.4.6-2.amzn2023.0.2.x86_64      cri-tools-1.31.1-150500.1.1.x86_64      kubeadm-1.31.1-150500.1.1.x86_64
kubectl-1.31.1-150500.1.1.x86_64      kubelet-1.31.1-150500.1.1.x86_64      kubernetes-cni-1.5.1-150500.1.1.x86_64
libnetfilter_cthelper-1.0.0-21.amzn2023.0.2.x86_64  libnetfilter_cttimeout-1.0.0-19.amzn2023.0.2.x86_64  libnetfilter_queue-1.0.5-2.amzn2023.0.2.x86_64
Complete!
root@ip-172-31-94-95 ec2-user]# sudo systemctl enable --now kubelet
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /usr/lib/systemd/system/kubelet.service.
root@ip-172-31-94-95 ec2-user]# kubeadm join 172.31.93.102:6443 --token 6ccgw.o10vg5f2n5d9fa42 \
--discovery-token-ca-cert-hash sha256:lbbcc9939e895e8de0e0dd77ec72d881a9ef3b8f51a42f3145857e54b13c3818
[preflight] Running pre-flight checks
[WARNING FileExisting-socat]: socat not found in system path
[WARNING FileExisting-tc]: tc not found in system path
error execution phase preflight: couldn't validate the identity of the API Server: failed to request the cluster-info ConfigMap: Get "https://172.31.93.102:6443/api/v1/namespaces/kube-public/configmaps/cluster-info?timeout=10s": context deadline exceeded
to see the stack trace of this error execute with --v=5 or higher
[root@ip-172-31-94-95 ec2-user]# 
```

i-0ec932e19bc2d5a2f (node1)

PublicIPs: 3.84.157.220 PrivateIPs: 172.31.94.95

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

Step 7: Run the following command in master instance console -

kubectl get nodes

Using cluster from kubectl context: workshop.k8s.local

Validating cluster workshop.k8s.local

INSTANCE GROUPS

| NAME | ROLE | MACHINETYPE | MIN | MAX | SUBNETS |
|-------------------|--------|-------------|-----|-----|------------|
| master-us-west-2a | Master | t3.medium | 1 | 1 | us-west-2a |
| nodes-us-west-2a | Node | t3.medium | 1 | 1 | us-west-2a |

NODE STATUS

| NAME | ROLE | READY |
|---|--------|-------|
| ip-172-20-40-55.us-west-2.compute.internal | master | True |
| ip-172-20-58-174.us-west-2.compute.internal | node | True |

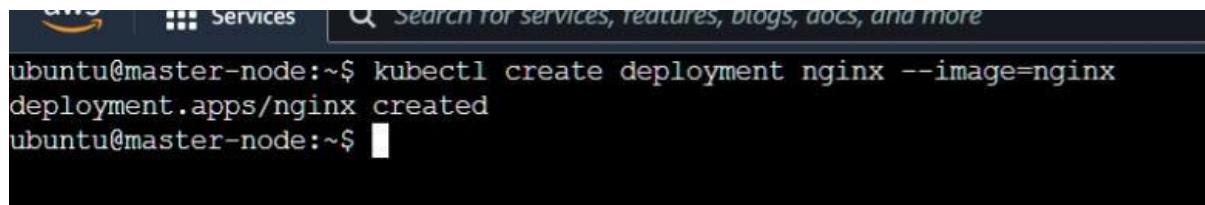
Your cluster workshop.k8s.local is ready

ADVANCE DEVOPS EXP 4

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

Step 1: As the cluster is up and running, we can deploy our nginx server on this cluster. Apply this deployment file using this command to create a deployment.

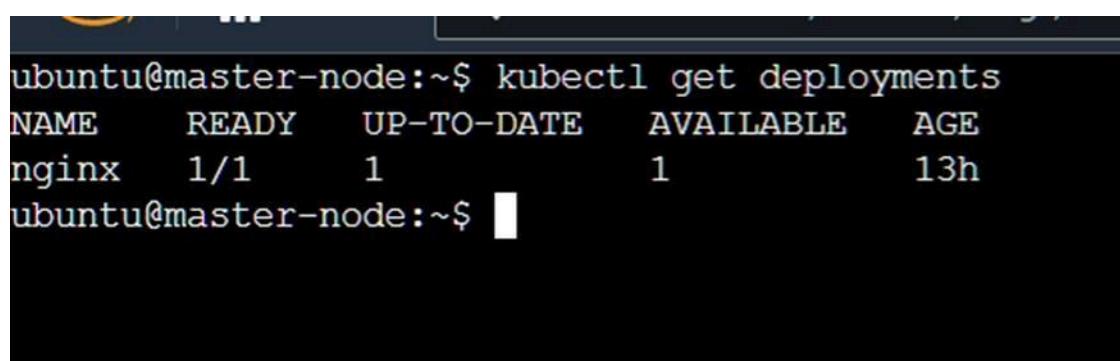
```
$kubectl create deployment nginx --image=nginx
```



```
aws Services Search for services, features, blogs, docs, and more
ubuntu@master-node:~$ kubectl create deployment nginx --image=nginx
deployment.apps/nginx created
ubuntu@master-node:~$
```

Step 2: Verify the deployment using the command:

```
$kubectl get deployments
```

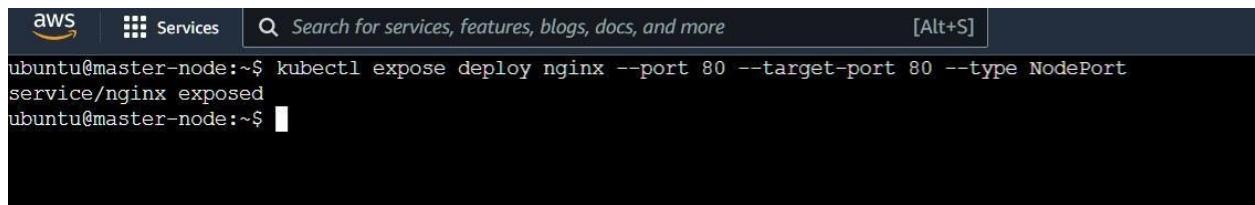


```
ubuntu@master-node:~$ kubectl get deployments
NAME      READY    UP-TO-DATE   AVAILABLE   AGE
nginx     1/1      1           1           13h
ubuntu@master-node:~$
```

Step 3: Next, run the following command to create a service named nginx that will expose the app publicly. It will do so through a NodePort, a scheme that will make the pod accessible through an arbitrary port opened on each node of the cluster

with this service-type, Kubernetes will assign this service on ports on the **30000+** range.

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

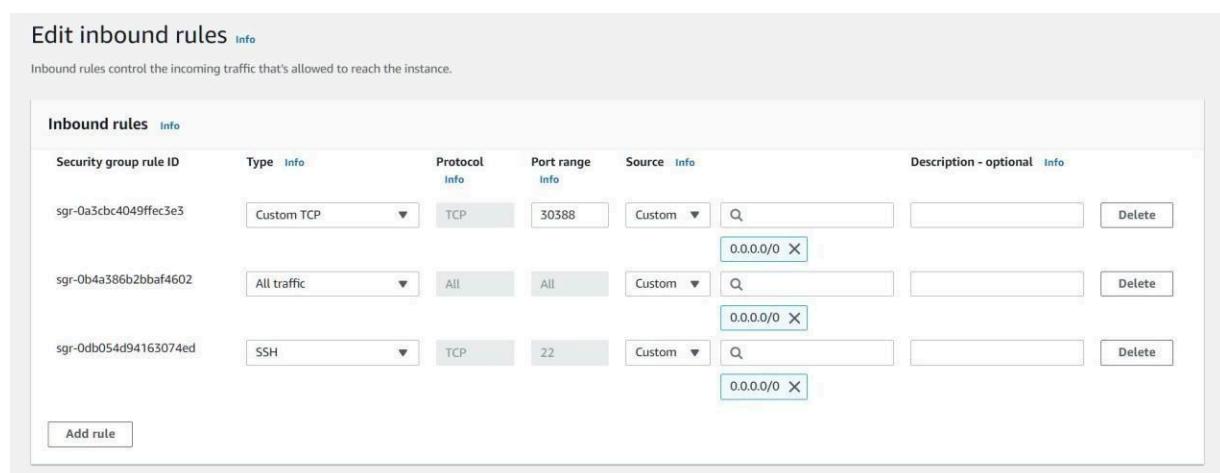


```
aws | Services | Search for services, features, blogs, docs, and more [Alt+S]
ubuntu@master-node:~$ kubectl expose deploy nginx --port 80 --target-port 80 --type NodePort
service/nginx exposed
ubuntu@master-node:~$
```

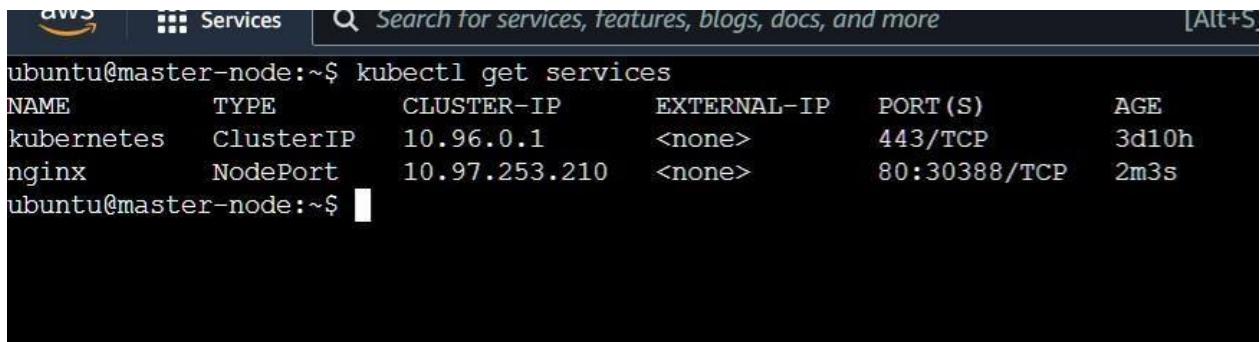
Step 4: Run this command to see a summary of the service and the ports exposed.

```
$kubectl get services
```

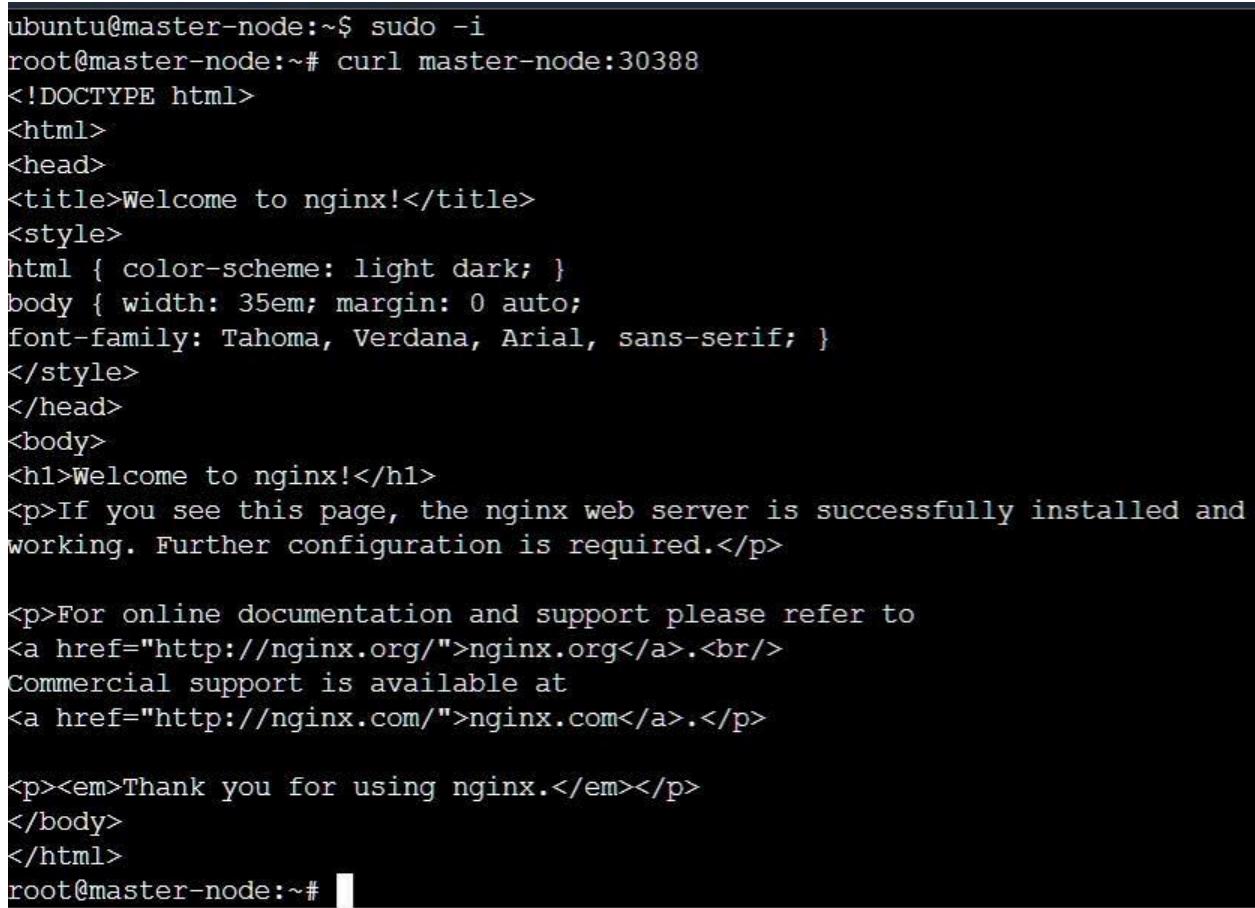
Step 5: Add the port which is displayed i.e. 30388 (in our case) in the inbound rules of the security group.



Step 6: Now you can verify that the Nginx page is reachable on all nodes using the curl command.



```
ubuntu@master-node:~$ kubectl get services
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes  ClusterIP  10.96.0.1      <none>        443/TCP      3d10h
nginx     NodePort   10.97.253.210  <none>        80:30388/TCP  2m3s
ubuntu@master-node:~$ █
```



```
ubuntu@master-node:~$ sudo -i
root@master-node:~# curl master-node:30388
<!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>

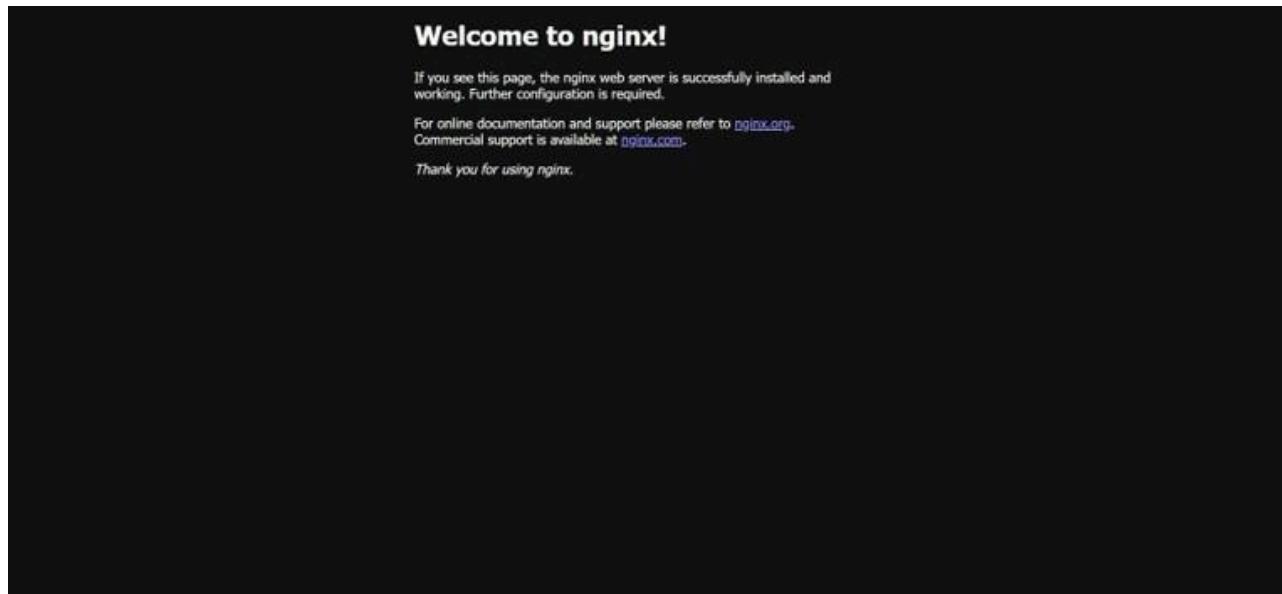
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
root@master-node:~# █
```

As you can see, the “**WELCOME TO NGINX!**” page can be reached.

Step 7: To test that everything is working, visit `http://worker_1_ip:nginx_port` or `http://worker_2_ip:nginx_port` through a browser on your local machine. You will see Nginx’s familiar welcome page.

<http://52.90.129.234:30388>



ADVANCE DEVOPS EXP 6

Vedang Wajge

D15A/66

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

Docker Installation:

```
Windows Command Prompt
Microsoft Windows [Version 10.0.22621.2715]
(c) Microsoft Corporation. All rights reserved.

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

Before using terraform commands:

```
C:\Terraform scripts\Dockers>docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
react-img      latest    5f0b23d1bdea  2 weeks ago   320MB
<none>        <none>   3bd8656788a8  2 weeks ago   320MB
```

```
C:\Terraform scripts\Dockers>docker container list
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
C:\Terraform scripts\Dockers>
```

Code:

```
docker.tf  ×

  docker.tf > resource "docker_container" "nginx"
1   terraform {
2       required_providers {
3           docker = {
4               source = "kreuzwerker/docker"
5               version = "~> 3.0.1"
6           }
7       }
8   }
9   provider "docker" {
10      host = "npipe:///./pipe//docker_engine"
11  }
12  resource "docker_image" "nginx" {
13      name = "nginx:latest"
14      keep_locally = false
15  }
16  resource "docker_container" "nginx" {
17      image = docker_image.nginx.image_id
18      name = "tutorial"
19      ports {
20          internal = 80
21          external = 8000
22      }
23 }
```

Terraform Commands:

- PS C:\Terraform scripts\ Docker> **terraform init**
Initializing the backend...
Initializing provider plugins...
 - Finding kreuzwerker/docker versions matching "~> 3.0.1"...
 - Installing kreuzwerker/docker v3.0.2...
 - Installed kreuzwerker/docker v3.0.2 (self-signed, key ID **BD080C4571C6104C**)Partner and community providers are signed by their developers.
If you'd like to know more about provider signing, you can read about it here:
<https://www.terraform.io/docs/cli/plugins/signing.html>
Terraform has created a lock file **.terraform.lock.hcl** to record the provider selections it made above. Include this file in your version control repository so that Terraform can guarantee to make the same selections by default when you run "terraform init" in the future.

Terraform has been successfully initialized!

```
● PS C:\Terraform scripts\ Docker> terraform plan
```

```
Terraform used the selected providers to generate the following execution plan. Resources: + create
```

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

```
# docker_container.nginx will be created
+ resource "docker_container" "nginx" {
    + attach = false
    + bridge = (known after apply)
    + command = (known after apply)
    + container_logs = (known after apply)
    + container_read_refresh_timeout_milliseconds = 15000
    + entrypoint = (known after apply)
    + env = (known after apply)
    + exit_code = (known after apply)
    + hostname = (known after apply)
    + id = (known after apply)
```

```
● PS C:\Terraform scripts\ Docker> terraform apply
```

```
Terraform used the selected providers to generate the following execution plan.
```

```
+ create
```

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

```
# docker_container.nginx will be created
+ resource "docker_container" "nginx" {
    + attach = false
    + bridge = (known after apply)
    + command = (known after apply)
    + container_logs = (known after apply)
    + container_read_refresh_timeout_milliseconds = 15000
```

```
Enter a value: yes
```

```
docker_image.nginx: Creating...
docker_image.nginx: Still creating... [10s elapsed]
docker_image.nginx: Still creating... [20s elapsed]
docker_image.nginx: Still creating... [30s elapsed]
docker_image.nginx: Creation complete after 38s [id=sha256:5ef79149e0ec84a7a9f9284c3f91aa3c20608f8391f5445eabe92ef07dbda0]
docker_container.nginx: Creating...
docker_container.nginx: Creation complete after 1s [id=19c3b26e694e3b26a5daa18288d68c790f0168d547f94171a49e3491bd173ae9]
```

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

```
● PS C:\Terraform scripts\ Docker>
```

After using Terraform commands:

```
C:\Terraform\scripts\Docker>docker images
REPOSITORY      TAG      IMAGE ID      CREATED       SIZE
nginx           latest   5ef79149e0ec  12 days ago  188MB
react-img       latest   5f0b23d1bdea  2 weeks ago  320MB
<none>          <none>   3bd8656788a8  2 weeks ago  320MB
```

```
C:\Terraform scripts\ Docker> docker container list
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
19c3b26e694e 5ef79149e0ec "/docker-entrypoint..." 2 minutes ago Up About a minute 0.0.0.0:8000->80/tcp tutorial

C:\Terraform scripts\ Docker>
```

To Delete the Containers created:

```
PS C:\Terraform scripts\docker> terraform destroy
docker_image.nginx: Refreshing state... [id=sha256:5ef79149e0ec84a7a9f9284c3f91aa3c20608f8391f5445eabe92ef07dbda03]
docker_container.nginx: Refreshing state... [id=19c3b26e694e3b26a5daa18288d68c790f0168d547f94171a49e3491bd173ae9]

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

Terraform will perform the following actions:

Adv. DevOps Assignment 1

Using S3 Bucket Video Streaming

- a) Create an S3 bucket

Sign in to AWS and select create bucket, enter appropriate name and choose region and for Block Public Access for this bucket, keep default settings. Choose Create Bucket.

- b) Upload a video to the S3 Bucket

To upload a file to the bucket open the created bucket and on objects tab, choose upload, then select files and folders and Add Files. Select Open and locate file then choose Upload.

- c) Create a CloudFront origin access identity

To create a CloudFront OAI, sign in to the AWS Management Console, under the Security section, choose Origin access. Under the identities tab, choose Create origin access identity. Choose create.

- d) Create a cloudfront distribution

To use CloudFront to serve and distribute the video in your S3 Bucket, you must create a CloudFront distribution.

- e) Access the S3 video through the CloudFront distribution with the custom domain name.

We can access this video through that custom URL created and SSL certificate for safety and security is also assigned.

Q.2

BMW and Hotstar on AWS Services



On 15 Nov, 2023 when India was playing against New Zealand at the World Cup semifinal match in Wankhede Stadium in Mumbai more than 5 core people were watching the match on Hotstar apps on global level. AWS serves as the primary hosting provider for Disney. It relies on AWS as the backbone of their hosting, and provide details about the infrastructure, including the use of AWS instances during high-impact events. It offers a scalable and reliable cloud computing platform, and allows Disney to manage their infrastructure based on demand.

One of the best known names in the automotive industry, BMW Group has been providing premium driving experiences of millions of customers since 1916. Always at the forefront of innovation, it has now developed a groundbreaking In-Car Cloud Assistance solution in collaboration with AWS. Part of the company's infrastructure optimization program, the ICCA is empowering BMW Group DevOps with generative AI to boost performance and efficiency. With access to large language models (LLM) and with data encrypted in transit and at rest - BMW Group can securely deliver high-quality connected mobility solutions.

Kubernetes and how industries use it.

Kubernetes, commonly referred to K8's, it is like a manager for your computer programs. Imagine you have a bunch of different programs running on your computer. They need to work together, and sometimes you want to run more copies of a program when things get busy. This can be hard to do manually, like trying to juggle a lot of balls at once. Adidas is a globally renowned sportswear and athletic footwear company headquartered in Germany. Founded in 1949, it offers a wide range of sports-related products, collaborates with subsidiary brands like Reebok, focuses on innovation in athletic footwear technology, and maintains a strong global presence. Adidas is committed to sustainability and social responsibility.

Adidas stated "we started from the developer point of view," and looked for ways to shorten the time it took to get a project up and running and into the adidas infrastructure, says Senior Director of Platform Engineering Fernando Cornago. They found the solution with containerization, agile development, continuous delivery, and a cloud native platform that includes Kubernetes and Prometheus.

Q.4] What are Nagios and its use in E-Services.

→ Nagios is an Open Source IT system monitoring tool. It was designed to run on the Linux OS and can monitor device running Linux, Windows and Unix OSes. Nagios software runs periodic checks on critical parameters of application, network server resources. Nagios can monitor memory use, disk use and microprocessor load, as well as the number of currently running processes. Bitnetix with Nagios in an IT consulting organization which is into networking, datacenter, monitoring and voice over IP, through their offerings, they make small business look big. Their solution help you managing customer relationships in a better by increasing more engagement and improving their satisfaction. They say they are in business of communication, hence, communicating right message to their customers at right time is very important for them.

Creating a REST API with serverless framework
Creating Rest Api with serverless framework is an efficient way to deploy serverless applications that can scale automatically without managing server.

- i) Serverless framework: A powerful tool that deployment of services and serverless applications across various app cloud providers like Aws.
- ii) Serverless Architecture: This design model allows developers to build applications without worrying about underlying infrastructure, enabling focus on code.
- iii) Rest Api: Representational state transfer is architecture style for designing network applications.

Steps for creating Rest APIs for serverless framework:

- 1) Install serverless frameworks
- 2) Creating a Node.js serverless project.
- 3) Project structure define.
- 4) Create a Rest Api Resource
- 5) Deploy the Service.
- 6) Testing the API's.
- 7) Storing data in Dynamic DB's
- 8) Adding more functionalities : 'list all candidates', etc.
- 9) Aws IAM Permissions.
- 10) Monitoring and Maintenance.

Q.2]

Case study for sonarqube.

Creating own profile in sonarqube for testing project quality. Use sonarqube to analyze your Github code. Install sonarlint in your IntelliJ IDE and analyze java code. Analyze python project with sonarqube.



Sonarqube is an open source platform used for continuous inspection of code quality.

1) Profile updation/creation in sonarqube:

Quality profiles in sonarqube are essential configurations that define rules applied during code analysis. Each project has a quality profile for every supported language with default being 'sonarqube'.

2) Using sonarcloud to analyze Github code.

Sonarqube / sonar cloud is cloud based counterpart of sonarqube that integrates directly with Github, BitBucket, Azure and Gitlab repos.

To get started with sonarcloud via

Github sign up via sonarcloud setup, complete result can be viewed in both sonar-cloud & Github including security import issue.

3) Sonarlint in Java IDE:

Sonarlint is an IDE that performs on the fly code analysis as you write code. It helps developers detect bugs, security vulnerabilities and code smells directly in the development environment such as IntelliJ Idea or Eclipse.

4) Analyzing Python Projects with SonarQube:

SonarQube supports Python test coverage reporting, but it requires third party tool like coverage.py to generate the coverage report. To enable coverage adjust your build process so that coverage tool runs before sonar scanner and ensures report.

5) Analyzing Node.js projects with SonarQube.

For node.js project sonarqube can analyze Javascript and Typescript code. Similar to the python setup you can configure sonarqube to analyze node.js projects by installing the appropriate.

At a large organization, your centralized operations team may get repetitive infrastructure requests. You can use Terraform to build a 'self-serve' infrastructure.

Terraform's self-serve infrastructure provides a powerful use case in large organizations.

- i) Self serve infra : By using Terraform modules, you can create reusable and standardized infrastructure config. Module creation in Terraform, main.tf, variables.tf, and outputs.tf.

Also after module creation its standardization is equally important.

ii) Enabling self-service for Product Teams :

Create a self service or version control access

and provide pre-configured Terraform workflows. Onboard and train product teams, and the most important RBAC (Role-based Access Control) for preventing unauthorized access.

3] Automate Infrastructure Request via Ticketing systems:

Integrate Terraform Cloud or Terraform Enterprise with the ticketing system, automate approval workflows and monitor & log requests.

4] Workspaces setup for Environment Segregation.

To manage different environments, Terraform workspaces were set up. This ensured that teams could deploy the same infrastructure across different environments without overlap.

ADVANCE DEVOPS - 9

NAME - Vedang V. Wajge

D15A

66

Create an EC2 instance at AWS

The screenshot shows the AWS EC2 Instances page. The left sidebar includes links for EC2 Dashboard, EC2 Global View, Events, Console-to-Code, Instances (selected), Instances Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots). The main content area displays a table titled 'Instances (4) Info' with columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. The instances listed are:

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|-------------|---------------------|----------------|---------------|-------------------|-----------------------------|-------------------|
| Master | i-0420c7d89658aa5e2 | Terminated | t2.medium | - | View alarms | us-east-1c |
| Worker | i-0032d9bfa820603a4 | Terminated | t2.medium | - | View alarms | us-east-1c |
| nagios-host | i-09e8ea019f24f4be2 | Running | t2.micro | 2/2 checks passed | View alarms | us-east-1c |
| nagios-host | i-0ab4023c9d183ce0e | Terminated | t2.micro | - | View alarms | us-east-1c |

Below the table, a modal window titled 'Select an instance' is open, showing the same list of instances.

After this install following commands:-

```
sudo apt install apache2 libapache2-mod-php php php-gd libgd-dev
gcc make autoconf libssl-dev wget unzip bc gawk dc build-essential
snmp libnet-snmp-perl gettext -y
```

The terminal window displays the following output:

```
System information as of Sat Sep 28 12:51:58 UTC 2024
System load: 0.16      Processes:          106
Usage of /: 10.6% of 14.46GB  Users logged in:    0
Memory usage: 21%        IPv4 address for enX0: 172.31.89.161
Swap usage: 0%          Instance type: t2.medium
                        Status check:   -
                        Alarm status:  -
                        Availability zone: us-east-1c

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

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

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

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

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

ubuntu@ip-172-31-89-161:~$ sudo apt update
```

```

ubuntu@ip-172-31-89-161:~$ sudo apt install apache2 php
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-utils libapache2-mod-php8.3 libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap
  libaprutil1t64 liblua5.4-0 php-common php8.3 php8.3-cli php8.3-common php8.3-opcache php8.3-readline ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser php-pear
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapache2-mod-php8.3 libapr1t64 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libaprutil1t64 liblua5.4-0 php php-common php8.3 php8.3-cli php8.3-common php8.3-opcache
  php8.3-readline ssl-cert
0 upgraded, 18 newly installed, 0 to remove and 143 not upgraded.
Need to get 6998 kB of archives.
After this operation, 30.5 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libapr1t64 amd64 1.7.2-3.1ubuntu0.1 [108 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1t64 amd64 1.6.3-1.1ubuntu7 [91.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.3-1.1ubuntu7 [11.2 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-ldap amd64 1.6.3-1.1ubuntu7 [9116 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblua5.4-0 amd64 5.4.6-3build2 [166 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-bin amd64 2.4.58-1ubuntu8.4 [1329 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-data all 2.4.58-1ubuntu8.4 [163 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-utils amd64 2.4.58-1ubuntu8.4 [97.1 kB]

```

```

ubuntu@ip-172-31-89-161:~$ sudo apt install gcc build-essential
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86_64-linux-gnu bzip2 cpp cpp-13 cpp-13-x86_64-linux-gnu cpp-x86_64-linux-gnu
  dpkg dpkg-dev fakeroot fontconfig-config fonts-dejavu-core fonts-dejavu-mono g++ g++-13 g++-13-x86_64-linux-gnu
  g++-x86_64-linux-gnu gcc-13 gcc-13-base gcc-13-x86_64-linux-gnu gcc-x86_64-linux-gnu libalgorithm-diff-perl
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libbaom3 libasan8 libatomic1 libbinutils libbz2-1.0 libc-bin
  libc-dev-bin libc-devtools libc6 libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0 libde265-0 libdeflate0
  libdpkg-perl libfakeroot libfile-fcntllock-perl libfontconfig1 libgcc-13-dev libgd3 libgomp1 libgprofng0
  libheif-plugin-aomdec libheif-plugin-aomenc libheif-plugin-libde265 libheif1 libhwasan0 libisl23 libitm1 libjbig0
  libjpeg-turbo8 libjpeg8 liblerc4 liblsan0 libmpc3 libquadmath0 libsframe1 libsharpyuv0 libstdc++-13-dev libtiff6
  libtsan2 libubsan1 libwebp7 libxpm4 linux-libc-dev linux-tools-common locales lto-disabled-list make manpages-dev
  rpcsvc-proto
Suggested packages:
  binutils-doc gprofng-gui bzip2-doc cpp-doc gcc-13-locales cpp-13-doc debsig-verify debian-keyring g++-multilib
  g++-13-multilib gcc-13-doc gcc-multilib autoconf automake libtool flex bison gdb gcc-doc gcc-13-multilib
  gdb-x86_64-linux-gnu glibc-doc libnss-nis libnss-nisplus bzr libgd-tools libheif-plugin-x265
  libheif-plugin-ffmpegdec libheif-plugin-jpegdec libheif-plugin-jpegenc libheif-plugin-j2kdec libheif-plugin-j2kenc
  libheif-plugin-ravle libheif-plugin-svtenc libstdc++-13-doc make-doc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86_64-linux-gnu build-essential bzip2 cpp cpp-13 cpp-13-x86_64-linux-gnu
  cpp-x86_64-linux-gnu dpkg-dev fakeroot fontconfig-config fonts-dejavu-core fonts-dejavu-mono g++ g++-13
  g++-13-x86_64-linux-gnu g++-x86_64-linux-gnu gcc gcc-13 gcc-13-base gcc-13-x86_64-linux-gnu gcc-x86_64-linux-gnu
  libalgorithm-diff-perl libalgorithm-merge-perl libbaom3 libasan8 libatomic1 libbinutils
  libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0 libde265-0 libdeflate0 libdpkg-perl
  libfakeroot libfile-fcntllock-perl libfontconfig1 libgcc-13-dev libgd3 libgomp1 libgprofng0 libheif-plugin-aomdec
  libheif-plugin-aomenc libheif-plugin-libde265 libheif1 libhwasan0 libisl23 libitm1 libjbig0 libjpeg-turbo8 libjpeg8
  liblerc4 liblsan0 libmpc3 libquadmath0 libsframe1 libsharpyuv0 libstdc++-13-dev libtiff6 libtsan2 libubsan1 libwebp7

```

```

ubuntu@ip-172-31-89-161:~$ sudo apt install libgd-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
libgd-dev is already the newest version (2.3.3-9ubuntu5).
0 upgraded, 0 newly installed, 0 to remove and 119 not upgraded.
ubuntu@ip-172-31-89-161:~$

```

Use this command to setup a new user called as nagios and adding it to nagcmd user group
 sudo useradd nagios
 sudo groupadd nagcmd

```

sudo usermod -aG nagcmd nagios
sudo usermod -aG nagcmd www-data

```

Now download and install nagios core

```
cd  
/tmp  
wget  
https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.  
6.tar.gz  
tar -zxvf nagios-*.tar.gz  
cd nagios-4.4.6
```

Now compile and install nagios

```
sudo ./configure --with-command-group=nagcmd  
sudo make all  
sudo make install  
sudo make install-init  
sudo make install-commandmode  
sudo make install-config  
sudo make install-webconf
```

```
***Config files installed ***  
bash  
  
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.  
mon Not Running  
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/apache2/sites-available/nagios.conf  
if [ 1 -eq 1 ]; then \  
    ln -s /etc/apache2/sites-available/nagios.conf /etc/apache2/sites-enabled/nagios.conf; \  
nification Error  
ln: failed to create symbolic link '/etc/apache2/sites-enabled/nagios.conf': File exists
```

Now install Nagios plugins.

```
cd /tmp  
wget  
https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz tar  
-zxvf nagios-plugins-*.tar.gz  
cd nagios-plugins-2.3.3  
sudo ./configure --with-nagios-user=nagios --with-nagios-group=nagios  
sudo make  
sudo make install
```

```

GNU nano 7.2                               /usr/local/nagios/etc/objects/contacts.cfg *

# 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             pranavtitambe04@gmail.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
}

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

# We only have one contact in this simple configuration file, so there is
# no need to create more than one contact group.

define contactgroup {

```

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

Here by executing this command you need to set the password for 'nagiosadmin'

```
sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

```

ubuntu@ip-172-31-89-161:/tmp/nagios-plugins-2.3.3$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:                                     sudo systemctl restart apache2
Adding password for user nagiosadmin
ubuntu@ip-172-31-89-161:/tmp/nagios-plugins-2.3.3$ sudo a2enmod rewrite
sudo a2enmod cgi                                         Ensure your firewall allows HTTP traffic (port 80):
sudo systemctl restart apache2                           bash
Module rewrite already enabled
Module cgi already enabled
ubuntu@ip-172-31-89-161:/tmp/nagios-plugins-2.3.3$ |
```

Set proper permissions for the configuration:

```
sudo a2enmod rewrite
sudo a2enmod cgi
sudo systemctl restart apache2
```

```

ubuntu@ip-172-31-89-161:~$ sudo a2enmod rewrite
sudo a2enmod cgi
sudo systemctl restart apache2
Module rewrite already enabled
Module cgi already enabled
ubuntu@ip-172-31-89-161:~$ |
```

Ensure your firewall allows HTTP traffic (port 80):

```
sudo ufw allow Apache
sudo ufw reload
```

```
ubuntu@ip-172-31-89-161:~$ sudo ufw allow Apache
sudo ufw reload
Skipping adding existing rule
Skipping adding existing rule (v6)
Firewall not enabled (skipping reload)
ubuntu@ip-172-31-89-161:~$ |
```

Starting the nagios services

```
sudo systemctl enable nagios
sudo systemctl start nagios
sudo systemctl status nagios
```

```
ubuntu@ip-172-31-89-161:~$ sudo systemctl enable nagios
sudo systemctl start nagios
ubuntu@ip-172-31-89-161:~$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-09-28 13:59:55 UTC; 19min ago
     Docs: https://www.nagios.org/documentation
     Main PID: 103799 (nagios)
        Tasks: 6 (limit: 1130)
       Memory: 2.3M (peak: 4.4M)
          CPU: 341ms
        CGroup: /system.slice/nagios.service
                └─103799 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
                  ├─103800 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                  ├─103801 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                  ├─103802 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                  ├─103803 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                  └─103805 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 28 14:04:17 ip-172-31-89-161 nagios[103799]: SERVICE ALERT: localhost;Swap Usage;CRITICAL;SOFT;1;SWAP CRITICAL - 0%
Sep 28 14:05:17 ip-172-31-89-161 nagios[103799]: SERVICE ALERT: localhost;Swap Usage;CRITICAL;SOFT;2;SWAP CRITICAL - 0%
Sep 28 14:06:17 ip-172-31-89-161 nagios[103799]: SERVICE ALERT: localhost;Swap Usage;CRITICAL;SOFT;3;SWAP CRITICAL - 0%
Sep 28 14:07:17 ip-172-31-89-161 nagios[103799]: SERVICE NOTIFICATION: nagiosadmin;localhost;Swap Usage;CRITICAL;notify>
Sep 28 14:07:17 ip-172-31-89-161 nagios[103799]: SERVICE ALERT: localhost;Swap Usage;CRITICAL;HARD;4;SWAP CRITICAL - 0%
Sep 28 14:07:17 ip-172-31-89-161 nagios[103799]: wproc: NOTIFY job 4 from worker Core Worker 103800 is a non-check help>
Sep 28 14:07:17 ip-172-31-89-161 nagios[103799]: wproc: host=localhost; service=Swap Usage; contact=nagiosadmin
Sep 28 14:07:17 ip-172-31-89-161 nagios[103799]: wproc: early_timeout=0; exited_ok=1; wait_status=32512; error_code=0;
Sep 28 14:07:17 ip-172-31-89-161 nagios[103799]: wproc: stderr line 01: /bin/sh: 1: /bin/mail: not found
Sep 28 14:07:17 ip-172-31-89-161 nagios[103799]: wproc: stderr line 02: /usr/bin/printf: write error: Broken pipe
Lines 1-26/26 (END)
```

Here it will prompt to enter the username and password provide that we set earlier.

Go to your AWS EC2 dashboard and copy the public ip address. The url should look like this

'[http://<public-ip-address>/nagios'](http://<public-ip-address>/nagios)

Not secure | 3.83.157.235/nagios/

YouTube WhatsApp Google Download Custom... BIP39 - Mnemonic... Text structures - Pre... awsacademy.com/vt... GitHub Other favorites

Nagios® Core™

✓ Daemon running with PID 103799

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

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

General

- Home
- Documentation

Current Status

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

Quick Search:

Reports

- Availability
- Timeline (Legacy)
- Alerts
 - History
 - Summary
 - Histogram (Legacy)
- Notifications
- Event Log

System

- Comments
- Downtime
- Process Info
- Performance Info
- Scheduling Queue
- Configuration

Get Started

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

Quick Links

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

Latest News

Don't Miss...

Copyright © 2018 Nagios Core Development Team and Community Contributors. Copyright © 2018 Nagios Core, Inc. All Rights Reserved. See the LICENSE file for more information.

Page Tour

ADVANCE DEVOPS - 10

NAME - Vedang V. Wajge

D15A

66

Check if the nagios service is running by executing following command

```
sudo systemctl status nagios
```

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

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

Now, create a new EC2 instance on AWS

| Instances (2) Info | | Last updated less than a minute ago | C | Connect | Instance state ▾ | Actions ▾ | Launch instances | ▼ |
|------------------------------------|--------------|--|--|-------------------------|--|---------------------------|----------------------------------|----------------------|
| | | | | | All states ▾ | | | |
| <input type="checkbox"/> | Name ↗ | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | ▶ |
| <input type="checkbox"/> | nagios-host | i-09e8ea019f24f4be2 | Running Q Q | t2.micro | 2/2 checks passed View alarms + | us-east-1c | Edit | Logs |
| <input type="checkbox"/> | linux-client | i-0ad38836f030e3784 | Running Q Q | t2.micro | Initializing View alarms + | us-east-1c | Edit | Logs |

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

On the server, run this command

```
ps -ef | grep nagios
```

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

Become a root user and create 2 folders

```
sudo su
```

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

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

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

Copy localhost.cfg file to the mentioned location

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

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

```
nano
```

```
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/localhost.cfg
GNU nano 7.2
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/localhost.cfg
#####
#
# HOST DEFINITION
#
#####
# Define a host for the local machine

define host {

    use          linux-server      ; Name of host template
                ; This host definition
                ; in (or inherits from) a
                ; global host definition

    host_name    linuxserver
    alias        linuxserver
    address     52.207.253.18
}

#####
#
# HOST GROUP DEFINITION

^G Help      ^O Write Out   ^W Where Is   ^K Cut       ^T Exchange
^X Exit      ^R Read File   ^\ Replace    ^U Paste    ^J Jump
```

Note - Here replace hostname with linuxserver

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

Add the following line to the nagios.cfg file

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

```
# Definitions for monitoring a router/switch  
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg  
  
# Definitions for monitoring a network printer  
#cfg_file=/usr/local/nagios/etc/objects/printer.cfg  
  
# You can also tell Nagios to process all config files (with a .cfg  
# extension) in a particular directory by using the cfg_dir  
# directive as shown below:  
  
#cfg_dir=/usr/local/nagios/etc/servers  
#cfg_dir=/usr/local/nagios/etc/printers  
#cfg_dir=/usr/local/nagios/etc/switches  
#cfg_dir=/usr/local/nagios/etc/routers  
  
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/
```

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

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

```
License: GPL

Website: https://www.nagios.org
Reading configuration data...
    Read main config file okay...
    Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
    Checked 16 services.
    Checked 2 hosts.
    Checked 2 host groups.
    Checked 0 service groups.
    Checked 1 contacts.
    Checked 1 contact groups.
    Checked 24 commands.
    Checked 5 time periods.
    Checked 0 host escalations.
    Checked 0 service escalations.

Checking for circular paths...
    Checked 2 hosts
    Checked 0 service dependencies
    Checked 0 host dependencies
    Checked 5 timeperiods

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

Total Warnings: 0
Total Errors:   0

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

Now restart the service by using this command

```
service nagios restart
```

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

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

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

```
sudo apt update -y
sudo apt install gcc -y
sudo apt install -y nagios-nrpe-server nagios-plugins
```

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

```

sudo nano /etc/nagios/nrpe.cfg
# Supported.
#
# Note: The daemon only does rudimentary checking of the
# address. I would highly recommend adding entries to the
# file to allow only the specified host to connect if
# you are running this daemon on.
#
# NOTE: This option is ignored if NRPE is running
#       as a separate process.
allowed_hosts=127.0.0.1,54.167.169.0

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

```

Now restart nrpe server by using this command

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

Now, check nagios dashboard, you should see linuxserver up and running, if not check security groups of the EC2 instances.

The screenshot shows the Nagios 4.6.6 dashboard. On the left is a navigation sidebar with links for General, Current Status, Reports, and System. The main content area has three primary sections: "Current Network Status" (with a message about being logged in as nagiosadmin), "Host Status Totals" (2 Up, 0 Down, 0 Unreachable, 0 Pending), and "Service Status Totals" (12 Ok, 0 Warning, 0 Unknown, 4 Critical, 0 Pending). Below these are two tables: "Host Status Details For All Host Groups" and "Service Status Details For All Service Groups". The host table shows two hosts: "linuxserver" and "localhost", both marked as UP. The service table shows various services across different groups, with some marked as OK and others as PENDING.

| Host Status Details For All Host Groups | | | | |
|---|--------|---------------------|---------------|--|
| Host | Status | Last Check | Duration | Status Information |
| linuxserver | UP | 09-28-2024 18:45:20 | 0d 0h 2m 21s | PING OK - Packet loss = 68%, RTA = 0.63 ms |
| localhost | UP | 09-28-2024 18:44:05 | 0d 4h 47m 45s | PING OK - Packet loss = 0%, RTA = 0.04 ms |

| Service Status Details For All Service Groups | | | | |
|---|------|---------------------|---------------|---|
| Service | Type | Last Check | Duration | Status Information |
| http://localhost | HTTP | 09-28-2024 18:44:05 | 0d 4h 47m 45s | OK - Response code = 200, RTA = 0.04 ms |
| ssh://localhost | SSH | 09-28-2024 18:44:05 | 0d 4h 47m 45s | PENDING - No response from host |
| ping://localhost | PING | 09-28-2024 18:44:05 | 0d 4h 47m 45s | PENDING - No response from host |

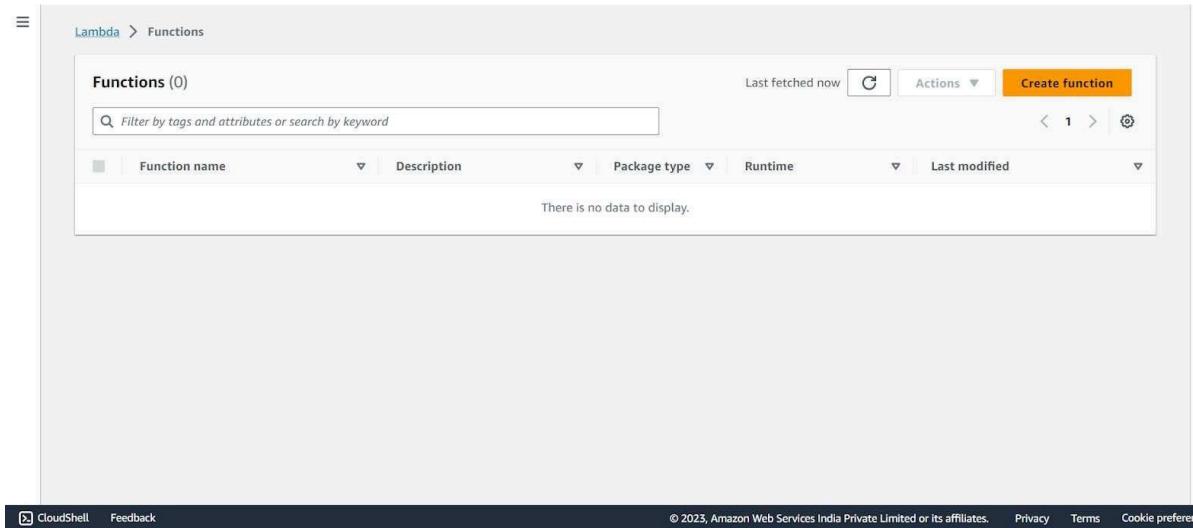
Experiment No 11

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

Steps to create an AWS Lambda function

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

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



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

Lambda > Functions > Create function

Create function Info

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

Author from scratch
Start with a simple Hello World example.

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

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

Basic information

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

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
`Node.js 18.x`

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

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

Lambda > Functions > Create function Info

Create function Info

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

Author from scratch
Start with a simple Hello World example.

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

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

Basic information

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

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
`Python 3.11`

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

arm64

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

<https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/create/applications> © 2023, Amazon Web Services India Private Limited or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Lambda > Functions > Create function

Create function Info

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

Author from scratch
Start with a simple Hello World example.

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

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

Basic information

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

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
`Python 3.11`

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

arm64

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

▶ Change default execution role

▶ Advanced settings

[Cancel](#) [Create function](#)

Click on the Create button.

3. This process will take a while to finish and after that, you'll get a message that your function was successfully created.

The screenshot shows the AWS Lambda Functions console. A success message at the top states: "Successfully created the function myPythonLambdaFunction. You can now change its code and configuration. To invoke your function with a test event, choose 'Test'." Below this, the "myPythonLambdaFunction" function card is displayed. It includes sections for "Function overview" (with a thumbnail icon, layers count, and buttons for "Add trigger" and "Add destination"), "Description" (empty), "Last modified" (15 seconds ago), "Function ARN" (arn:aws:lambda:ap-south-1:447953971928:function:myPythonLambdaFunction), and "Function URL" (info). At the bottom, tabs for "Code", "Test", "Monitor", "Configuration", "Aliases", and "Versions" are visible, with "Code" being the active tab.

This screenshot is identical to the one above, but it also shows the code editor pane on the right. The code editor displays the Python lambda function code:

```
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     }
```

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

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

The screenshot shows the AWS Lambda function configuration interface. At the top, a green banner indicates "Successfully created the function myPythonLambdaFunction. You can now change its code and configuration. To invoke your function with a test event, choose 'Test'." On the left, a sidebar lists various configuration tabs: General configuration, Triggers, Permissions, Destinations, Function URL, Environment variables, Tags, VPC, Monitoring and operations tools, Concurrency, Asynchronous invocation, Code signing, Database proxies, File systems, and State machines. The "General configuration" tab is selected. The main panel displays the "General configuration" section with the following details:

| Description | Memory | Ephemeral storage |
|-------------|-----------|-------------------|
| - | 128 MB | 512 MB |
| Timeout | SnapStart | |
| 0 min 3 sec | Info | None |

At the bottom right of the main panel is an "Edit" button.

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

Edit basic settings

Basic settings [Info](#)

Description - optional
[Empty input field]

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

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

SnapStart [Info](#)
Reduce startup time by having Lambda cache a snapshot of your function after the function has initialized. To evaluate whether your function code is resilient to snapshot operations, review the [SnapStart compatibility considerations](#).
None
Supported runtimes: Java 11, Java 17.

Timeout
0 min 1 sec

Execution role

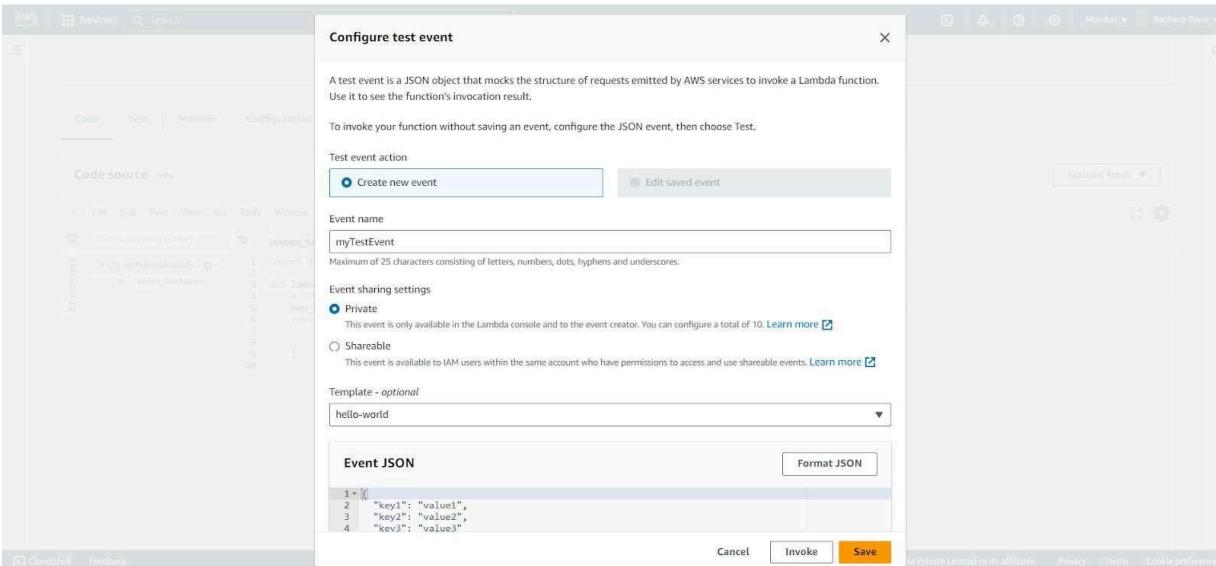
CloudShell Feedback

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

The screenshot shows the AWS Lambda function editor interface. At the top, there are tabs for Code, Test, Monitor, Configuration, Aliases, and Versions. The Code tab is selected. Below the tabs is a toolbar with File, Edit, Find, View, Go, Tools, Window, Test, Deploy, and a dropdown for Changes not deployed. To the right of the toolbar is an Upload from button. The main area contains a code editor titled 'lambda_function' with the following Python code:

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

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



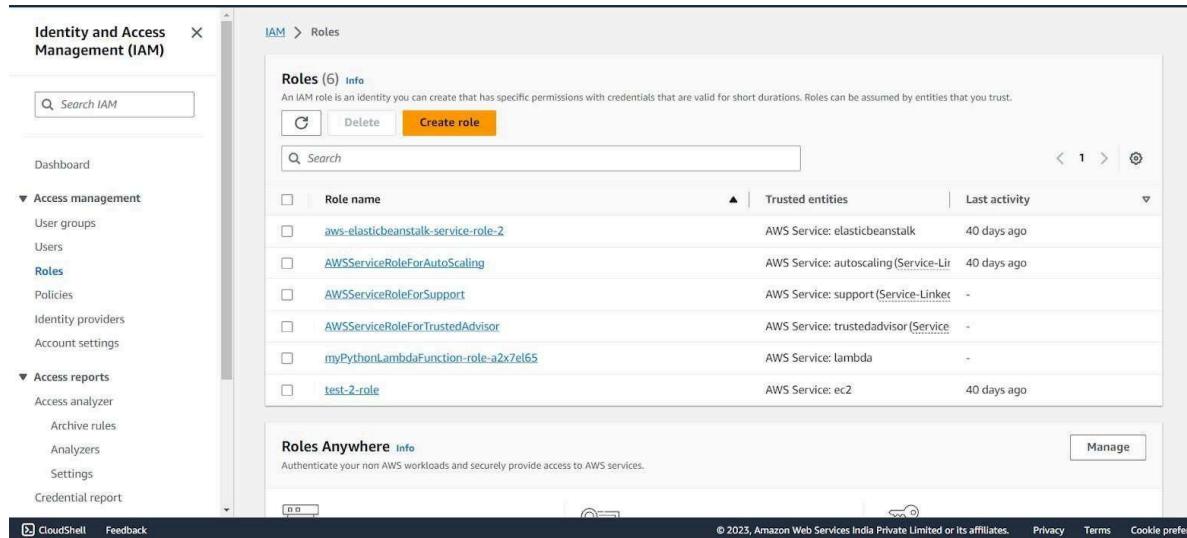
7. Now click on Test and you should be able to see the results.

The screenshot shows the AWS Lambda Test interface. At the top, a green banner displays the message "The test event myTestEvent was successfully saved." Below this, the main window has a toolbar with "File", "Edit", "Find", "View", "Go", "Tools", "Window", "Test" (which is highlighted in blue), "Deploy", and "Changes not deployed". On the left, there's a sidebar titled "Environment" with sections for "lambda_function" and "myTestEvent". The main content area shows "Execution results" with a response object containing "statusCode": 200 and "body": "\'Hello From Lambda\'". It also displays "Function Logs" and a "Request ID". At the bottom, there are links for "CloudShell", "Feedback", and copyright information from 2023.

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

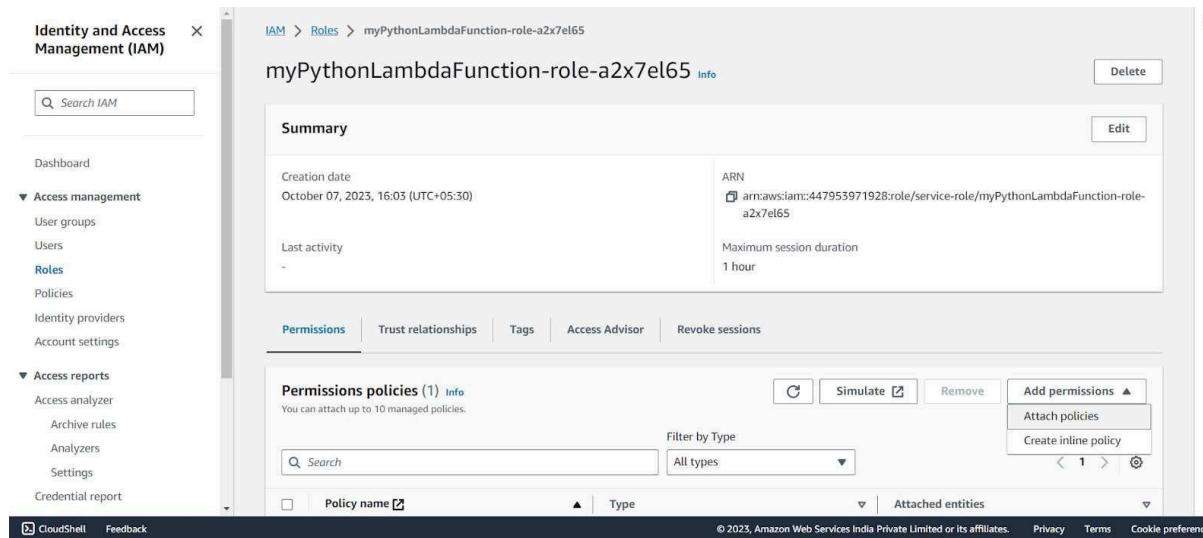
Experiment No 12

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



The screenshot shows the AWS IAM Roles page. On the left, there's a navigation sidebar with options like Dashboard, Access management, and Access reports. The main area is titled 'Roles (6) Info' and contains a table of roles. The table columns are 'Role name', 'Trusted entities', and 'Last activity'. The roles listed are: 'aws-elasticbeanstalk-service-role-2' (AWS Service: elasticbeanstalk, 40 days ago), 'AWSRoleForAutoScaling' (AWS Service: autoscaling (Service-Linker), 40 days ago), 'AWSRoleForSupport' (AWS Service: support (Service-Linker), -), 'AWSRoleForTrustedAdvisor' (AWS Service: trustedadvisor (Service-Linker), -), 'myPythonLambdaFunction-role-a2x7el65' (AWS Service: lambda, -), and 'test-2-role' (AWS Service: ec2, 40 days ago). There's also a section for 'Roles Anywhere' with a 'Manage' button.

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



The screenshot shows the 'myPythonLambdaFunction-role-a2x7el65' role details page. The left sidebar has the same navigation as the previous screenshot. The main area has a 'Summary' section with details like Creation date (October 07, 2023, 16:05 (UTC+05:30)), ARN (arn:aws:iam::447953971928:role/service-role/myPythonLambdaFunction-role-a2x7el65), Last activity (-), and Maximum session duration (1 hour). Below this is the 'Permissions' tab, which shows 'Permissions policies (1) Info'. A modal window is open over this tab, titled 'Add permissions ▲', with options to 'Attach policies' or 'Create inline policy'. The 'Attached entities' section below the modal is currently empty.

S3-ReadOnly

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

Attach policy to myPythonLambdaFunction-role-a2x7el65

▶ Current permissions policies (1)

Other permissions policies (882)

Filter by Type

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

Cancel Add permissions

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

Attach policy to myPythonLambdaFunction-role-a2x7el65

▶ Current permissions policies (2)

Other permissions policies (881)

Filter by Type

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

Cancel Add permissions

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

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

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

Access reports

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

Policy was successfully attached to role.

Last activity: Maximum session duration: 1 hour

Permissions Trust relationships Tags Access Advisor Revoke sessions

Permissions policies (3) Info You can attach up to 10 managed policies.

Filter by Type

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

Permissions boundary (not set)

CloudShell Feedback

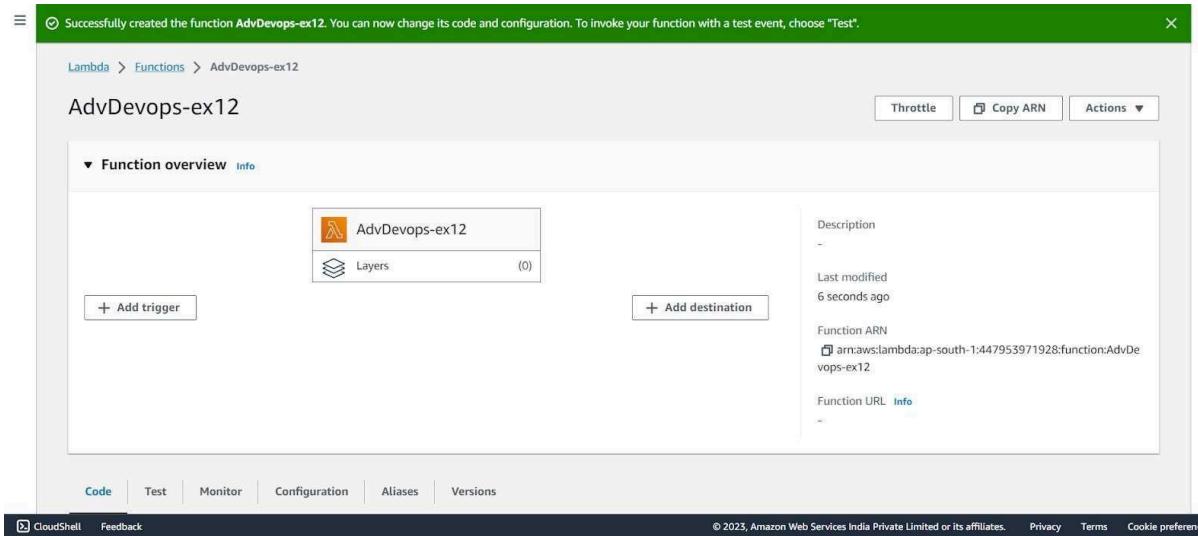
Step 3: Open up AWS Lambda and create a new Python function.

The screenshot shows the 'Create function' wizard in the AWS Lambda console. It includes sections for 'Basic information' (Function name: AdvDevops-ex12, Runtime: Python 3.11), 'Architecture' (x86_64 selected), and 'Permissions' (Use an existing role selected). At the bottom, there are links for CloudShell, Feedback, and a note about CloudWatch Logs permissions.

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 'Create function' wizard with the 'Change default execution role' section expanded. It shows the 'Use an existing role' option selected. The 'Existing role' dropdown contains 'service-role/myPythonLambdaFunction-role-a2x7el65'. At the bottom, there are links for CloudShell, Feedback, and a note about CloudWatch Logs permissions.

Step 4: The function is up and running.



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

```
1 import json
2 import boto3
3 import urllib
4
5 def lambda_handler(event, context):
6
7     s3_client = boto3.client('s3')
8     bucket_name = event['Records'][0]['s3']['bucket']['name']
9     key = event['Records'][0]['s3']['object']['key']
10    key_urllib_parse_unquote_plus = (key, encoding='utf-8')
11    message = "An file has been added with key " + key + " to the bucket" + bucket_name
12    print(message)
13    response = s3_client.get_object(Bucket=bucket_name, Key=key)
14    contents = response['Body'].read().decode()
15    contents = json.loads(contents)
16
17    print("These are the Contents of the File: \n", contents)
18
19
```

CloudShell Feedback

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Step 6: Click on Test and choose the 'S3 Put' Template.

Screenshot of the AWS Lambda console showing a successful function creation and configuration.

The top navigation bar includes the AWS logo, Services, a search bar, and a keyboard shortcut [Alt+S]. A green banner at the top right says "Successfully created the function **AdvDevops-ex12**. You can now change its code and configuration. To invoke your function, choose Test." Below the banner, the tabs are Code, Test, Monitor, Configuration, Aliases, and Versions. The Code tab is selected.

The main area shows the "Code source" section with the "Info" tab selected. The code editor displays the following Python code:

```
1 import json
2 import boto3
3 import urllib
4
5 def lambda_handler(event, context):
```

The toolbar above the code editor includes File, Edit, Find, View, Go, Tools, Window, Test, Deploy, and a status message "Changes not deployed".

A modal window titled "Configure test event" is open. It contains the following information:

- A descriptive text: "A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result."
- Text: "To invoke your function without saving an event, configure the JSON event, then choose Test."
- "Test event action" section:
 - Create new event
 - Edit saved event
- "Event name" field: "test".

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.
- "Event sharing settings":
 - Private

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

This event is available to IAM users within the same account who have permissions to access and use shareable events. [Learn more](#)
- "Template - optional" dropdown: "s3-put".
- "Event JSON" input field:

```
{<empty>}
```

Format JSON
- Action buttons: Cancel, Invoke, Save.

And Save it.

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

The screenshot shows the 'Buckets' section of the Amazon S3 console. It lists three buckets:

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

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

The screenshot shows the 'Create bucket' page. In the 'General configuration' section, the 'Bucket name' field is set to 'AdvDevopsexp12' and the 'AWS Region' dropdown is set to 'Asia Pacific (Mumbai) ap-south-1'. The 'Object Ownership' section is also visible.

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

The screenshot shows the 'Event notifications' section of the bucket properties. It displays a table with columns 'Name', 'Event types', 'Filters', 'Destination type', and 'Destination'. A message indicates 'No event notifications' and provides a link to 'Create event notification'. Other sections like 'Amazon EventBridge' and 'Transfer acceleration' are also present.

Click on Create Event Notification.

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

General configuration

Event name
S3putrequest

Event name can contain up to 255 characters.

Prefix - optional
Limit the notifications to objects with key starting with specified characters.
images/

Suffix - optional
Limit the notifications to objects with key ending with specified characters.
.jpg

Event types

Specify at least one event for which you want to receive notifications. For each group, you can choose an event type for all events, or you can choose one or more individual events.

Object creation

All object create events
s3:ObjectCreated:*

Put
s3:ObjectCreated:Put

Post
s3:ObjectCreated:Post

CloudShell Feedback © 2023, Amazon Web Services India Private Limited

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

Destination

Before Amazon S3 can publish messages to a destination, you must grant the Amazon S3 principal the necessary permissions to call the relevant API to publish messages to an SNS topic, an SQS queue, or a Lambda function. [Learn more](#)

Destination
Choose a destination to publish the event. [Learn more](#)

Lambda function
Run a Lambda function script based on S3 events.

SNS topic
Fanout messages to systems for parallel processing or directly to people.

SQS queue
Send notifications to an SQS queue to be read by a server.

Specify Lambda function

Choose from your Lambda functions

Enter Lambda function ARN

Lambda function
AdvDevops-ex12

Cancel Save changes

CloudShell Feedback © 2023, Amazon Web Services India Private Limited

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

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

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

Step 13: Go back to your S3 Bucket and click on Add Files to upload a new file.

Step 14: Select the dummy data file from your computer and click Upload.

The screenshot shows the AWS S3 'Upload' interface. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, a search bar containing 'Search', and a keyboard shortcut '[Alt+S]'. Below the navigation is a breadcrumb trail: 'Amazon S3 > Buckets > advopssexp12 > Upload'. The main title is 'Upload Info'. A note below says: 'Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more'.

A large dashed box area is labeled 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.' Below this is a table titled 'Files and folders (1 Total, 89.0 B)'. It contains one item: 'dummy.json' (application/json, 89.0 B). There are 'Remove', 'Add files', and 'Add folder' buttons above the table. A search bar 'Find by name' and a pagination indicator '< 1 >' are also present.

The 'Destination' section shows 'Destination' set to 's3://advopssexp12'. At the bottom, there are 'CloudShell' and 'Feedback' links, and a copyright notice: '© 2023, Amazon Web Services India Private Limited or its affiliates'.

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

The screenshot shows the AWS Lambda 'Event JSON' editor. The JSON code is as follows:

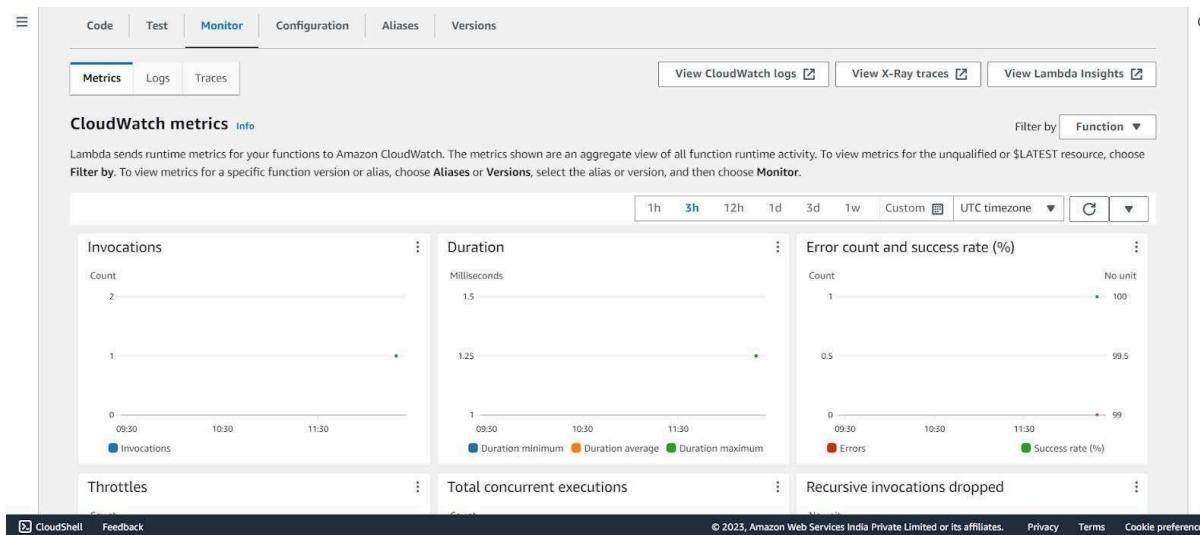
```

10     "principalId": "EXAMPLE"
11   },
12   "requestParameters": {
13     "sourceIPAddress": "127.0.0.1"
14   },
15   "responseElements": {
16     "x-amz-request-id": "EXAMPLE123456789",
17     "x-amz-id-2": "EXAMPLE123/5678abcdefghijklambdaisawesome/mnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
18   },
19   "s3": {
20     "s3SchemaVersion": "1.0",
21     "configurationId": "testConfigRule",
22     "bucket": {
23       "name": "advopssexp12",
24       "ownerIdentity": {
25         "principalId": "EXAMPLE"
26       },
27       "arn": "arn:aws:s3:::advopssexp12"
28     },
29     "object": {
30       "key": "test%2Fkey",
31       "size": 1024,
32       "eTag": "0123456789abcdef0123456789abcdef",
33       "sequencer": "0A1B2C3D4E5F678901"
34     }
35   }
36 }
37 ]
38 }

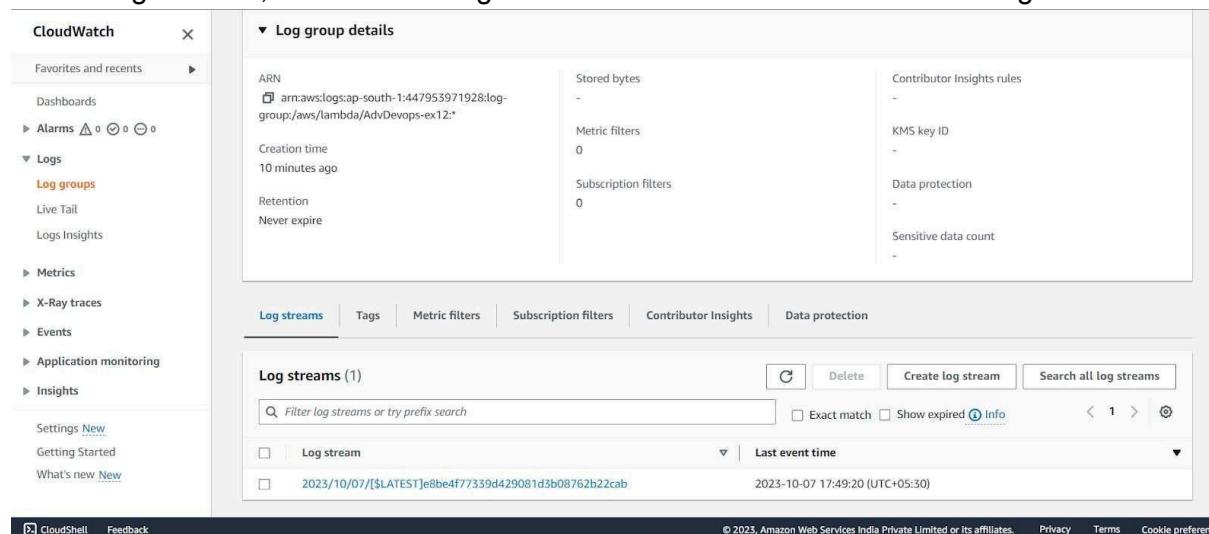
```

On the right side of the editor, there is a 'Format JSON' button.

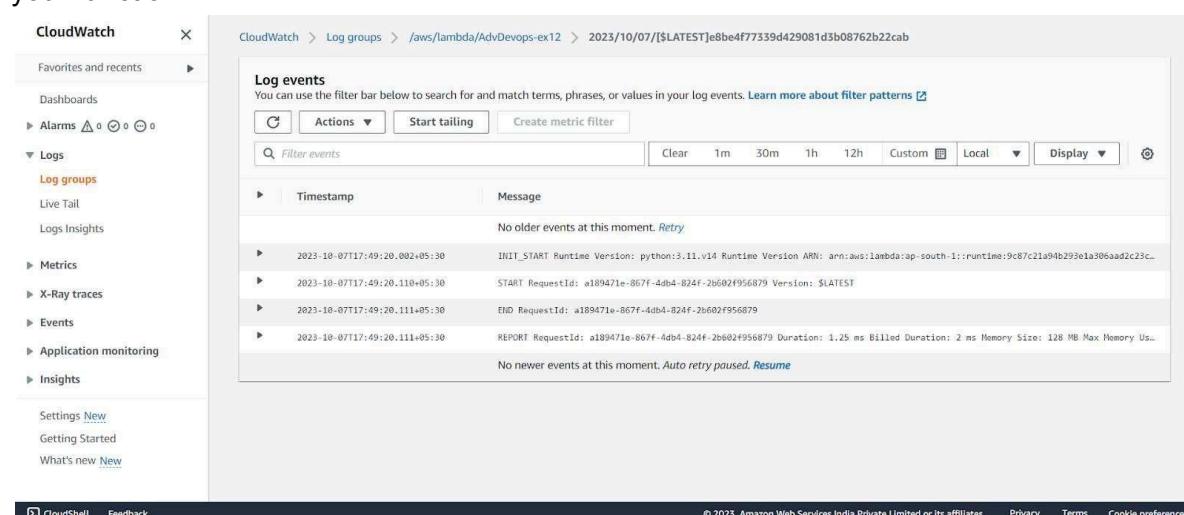
Step 16: Go back to your Lambda function , Refresh it and check the Monitor tab.



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



Step 17: Click on this log Stream that was created to view what was logged by your function.



Conclusion: Thus, we have created a Lambda function which logs “An Image has been added” once you add an object to a specific bucket in S3.

ADVANCE DEVOPS EXP-1

Vedang Wajge

D15A/66

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

Launch an instance Info

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

Name and tags Info

Name
Vedang's Server

▼ 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

Quick Start

      Including AMIs from AWS, Marketplace and the Community

▼ Instance type Info | Get advice

Instance type
t3.micro Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true
On-Demand RHEL base pricing: 0.0596 USD per Hour
On-Demand SUSE base pricing: 0.0108 USD per Hour
On-Demand Linux base pricing: 0.0108 USD per Hour
On-Demand Windows base pricing: 0.02 USD per Hour

All generations
Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

▼ Network settings [Info](#)

[Edit](#)[Network](#) | [Info](#)

vpc-0775017352e40f883

[Subnet](#) | [Info](#)

No preference (Default subnet in any availability zone)

[Auto-assign public IP](#) | [Info](#)

Enable

[Additional charges apply when outside of free tier allowance](#)[Firewall \(security groups\)](#) | [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

 Create security group Select existing security group

We'll create a new security group called '**launch-wizard-1**' with the following rules:

Allow SSH traffic from
Helps you connect to your instance

Anywhere
0.0.0.0/0

- Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server
- Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. [X](#)

Success
Successfully initiated launch of instance (i-0adcab46a9d7c2ae2)

Instances (1) [Info](#)

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS | Pub |
|-----------------|---------------------|----------------|---------------|--------------|-------------------------------|-------------------|--------------------------|------|
| Vedang's Server | i-0adcab46a9d7c2ae2 | Running | t3.micro | Initializing | View alarms + | eu-north-1b | ec2-13-61-2-113.eu-no... | 13.0 |

```
* Support:      https://ubuntu.com/pro

System information as of Tue Aug 20 07:27:04 UTC 2024

System load:  0.17          Temperature:      -273.1 C
Usage of /:   22.7% of 6.71GB  Processes:        112
Memory usage: 24%           Users logged in:   0
Swap usage:   0%            IPv4 address for ens5: 172.31.37.143

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

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

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

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

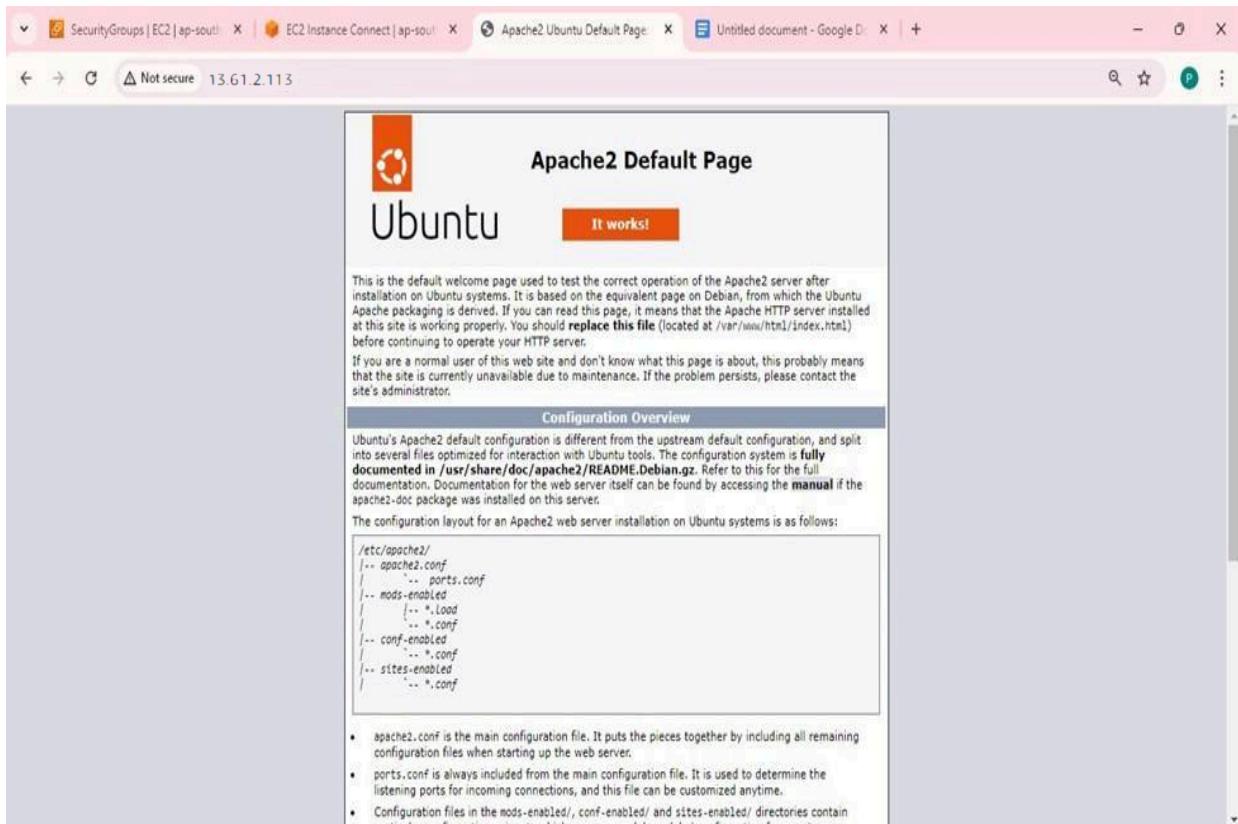
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-37-143:~$
```

```
i-0adcab46a9d7c2ae2 (Vedang's Server)
PublicIPs: 13.61.2.113  PrivateIPs: 172.31.37.143
```

```
root@ip-172-31-37-143:/home/ubuntu# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Tue 2024-08-20 07:28:04 UTC; 2min 10s ago
     Docs: https://httpd.apache.org/docs/2.4/
      Main PID: 2454 (apache2)
        Tasks: 55 (limit: 1078)
       Memory: 5.5M (peak: 5.9M)
          CPU: 45ms
        CGroup: /system.slice/apache2.service
                  ├─2454 /usr/sbin/apache2 -k start
                  ├─2456 /usr/sbin/apache2 -k start
                  └─2458 /usr/sbin/apache2 -k start
```

```
root@ip-172-31-37-143:/home/ubuntu# cd /var/www/html
root@ip-172-31-37-143:/var/www/html# /var/www/html
bash: /var/www/html: Is a directory
root@ip-172-31-37-143:/var/www/html#
```



ADVANCE DEVOPS EXP-2

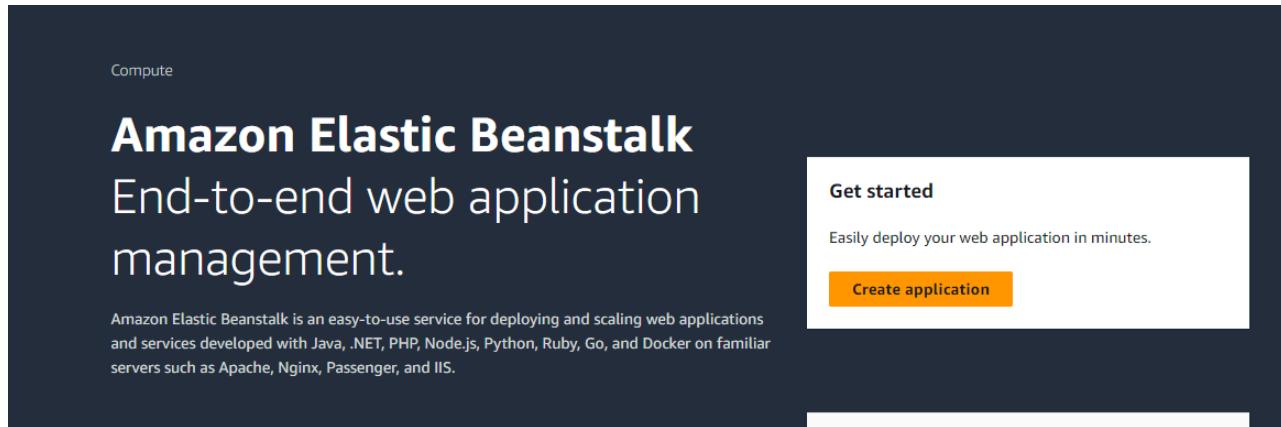
Vedang Wajge

D15A/66

Aim: To build your application using AWS Codebuild and deploy on S3 using AWS CodePipeline deploy sample application on EC2 instance using AWS codedeploy.

Code and Output :

Using elastic beanstalk:



The screenshot shows the "Environment tier" section of the Amazon Elastic Beanstalk application creation wizard. It has two options: "Web server environment" (selected) and "Worker environment". Both options have a "Learn more" link. Below this, the "Application information" section is shown, where the "Application name" is set to "Vedangbean". A note says "Maximum length of 100 characters." At the bottom, there is a section for "Application tags (optional)".

Platform Info

Platform type

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

Platform

PHP

Platform branch

PHP 8.3 running on 64bit Amazon Linux 2023

Platform version

4.3.2 (Recommended)

Environment properties

| Key | Value |
|--|-------|
| No environment properties There are no environment properties defined | |

Cancel Previous **Submit**

Vedangbean-env [Info](#)

Environment overview

| | |
|-----------|------------------|
| Health | Environment ID |
| ⊖ Unknown | e-vbejiswjb |
| Domain | Application name |
| - | Vedangbean |

Platform

| | |
|--|----------------|
| Platform | Change version |
| PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2 | |
| Running version | Platform state |
| - | Supported |

[Upload and deploy](#)

Developer Tools > CodePipeline > Pipelines

Introducing the new V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model. [Learn more](#)

Pipelines [Info](#)

[Create pipeline](#)

| Name | Latest execution status | Latest source revisions | Latest execution started | Most recent executions |
|--|-------------------------|-------------------------|--------------------------|------------------------|
| No results There are no results to display. | | | | |

a

Choose pipeline settings Info

Step 1 of 5

Pipeline settings

Pipeline name
Enter the pipeline name. You cannot edit the pipeline name after it is created.

No more than 100 characters

Pipeline type

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

Superseded
A more recent execution can overtake an older one. This is the default.

Queued (Pipeline type V2 required)
Executions are processed one by one in the order that they are queued.

Parallel (Pipeline type V2 required)
Executions don't wait for other runs to complete before starting or finishing.

eu-north-1.console.aws.amazon.com/codesuite/settings/connections/create?origi... G

WS Services Stockholm ▼ VedangWaje

Developer Tools > Connections > Create connection

Create a connection Info

Create GitHub App connection Info

Connection name

▶ Tags - optional

Source

Source provider
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

GitHub (Version 2) ▾

New GitHub version 2 (app-based) action

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

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

or [Connect to GitHub](#)

 **Ready to connect**
Your GitHub connection is ready for use.

Repository name
Choose a repository in your GitHub account.

X

You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

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

X

Output artifact format
Choose the output artifact format.

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

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

Add build stage Info

Step 3 of 5

1. [source stage](#)

2. [build stage](#)

3. [deploy stage](#)

4. [Your stage](#)

Build - optional

Build provider

Skip build stage X

Your pipeline will not include a build stage. Are you sure you want to skip this stage?

[Cancel](#) [Skip](#) [Skip build stage](#) [Next](#)

Info Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage.

Deploy

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

AWS Elastic Beanstalk ▾

Region

Europe (Stockholm) ▾

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

▾
No more than 100 characters

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

Vedangbean

Environment name
Choose an environment that you have already created in the AWS Elastic Beanstalk console. Or create an environment in the AWS Elastic Beanstalk console and then return to this task.

Vedangbean-env

Configure automatic rollback on stage failure

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

Step 4: Add deploy stage

Deploy action provider

Deploy action provider
AWS Elastic Beanstalk

ApplicationName
Vedangbean

EnvironmentName
Vedangbean-env

Configure automatic rollback on stage failure
Disabled

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

Success
Congratulations! The pipeline vedangpipeline has been created.

Create a notification rule for this pipeline >

Developer Tools > CodePipeline > Pipelines > vedangpipeline

vedangpipeline

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded
Pipeline execution ID: [449dc2d4-3f27-40f5-9285-c48e69e63ed6](#)

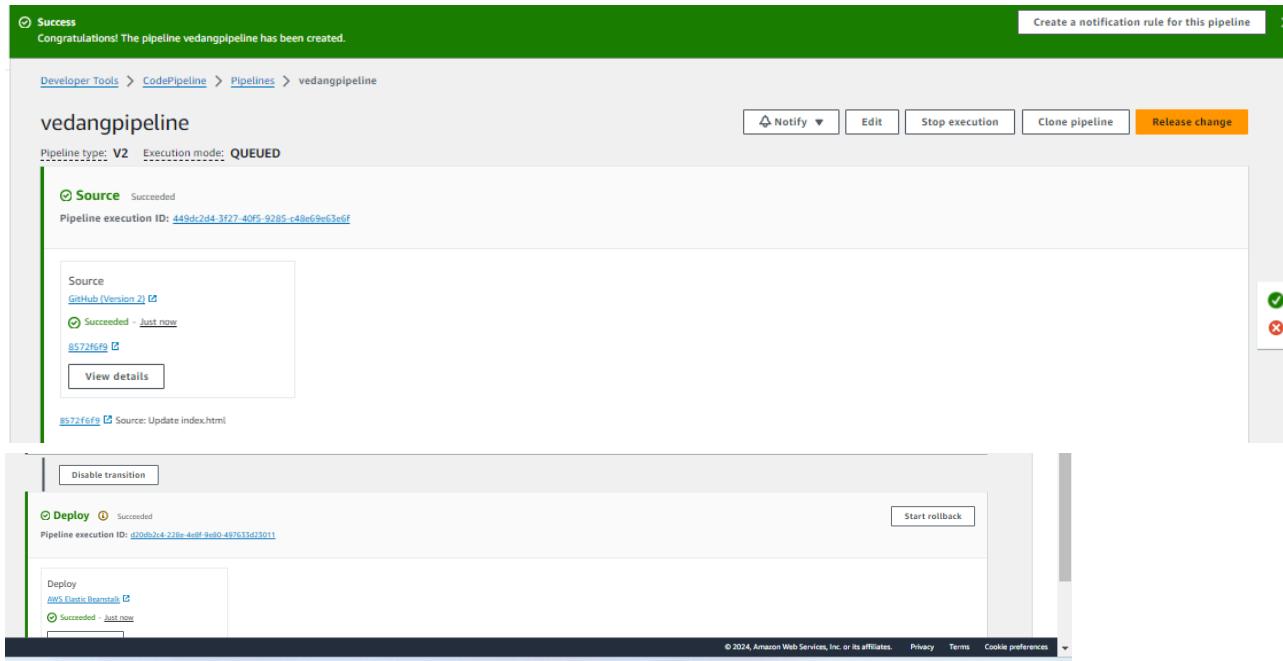
Source
GitHub (Version 2) [View details](#)
Succeeded - Just now
[8572f6fb](#) [View details](#)

Deploy Succeeded
Pipeline execution ID: [d2ndbu2c4-270e-4e0f-9e80-497c3d32011](#)

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

Disable transition Start rollback

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



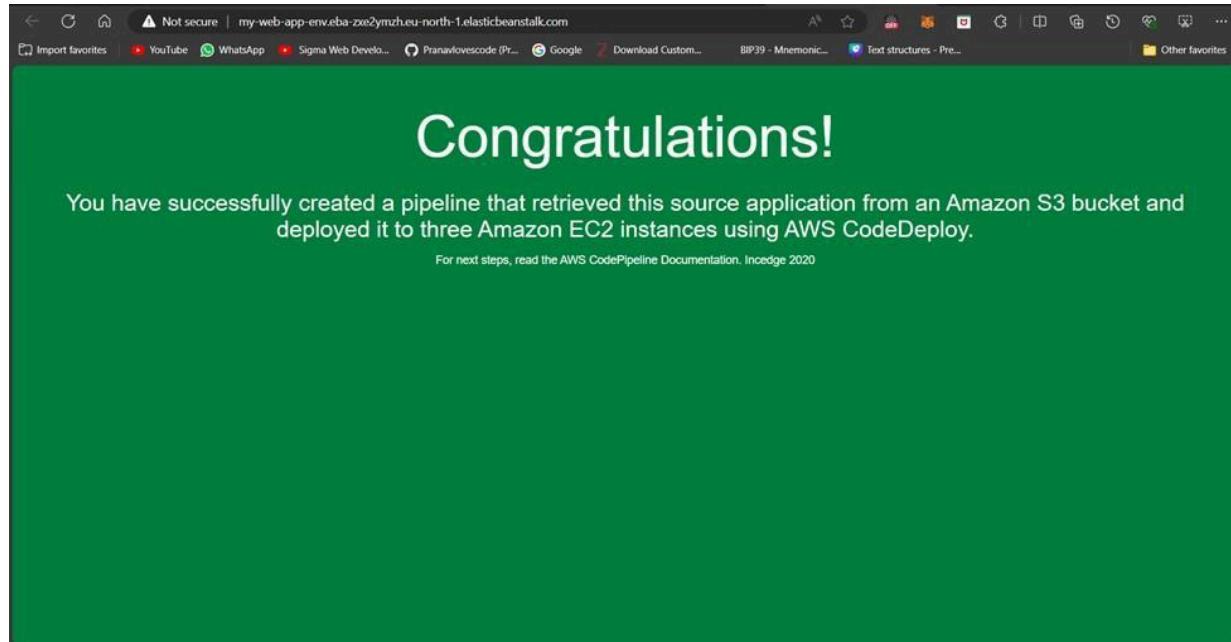
Not secure | my-web-app-env.eba-zxe2ymzh.eu-north-1.elasticbeanstalk.com

Import favorites YouTube WhatsApp Sigma Web Devlo... Pranavlovescode (Pr... Google Download Custom... BIP39 - Mnemonic... Text structures - Pre... Other favorites

Congratulations!

You have successfully created a pipeline that retrieved this source application from an Amazon S3 bucket and deployed it to three Amazon EC2 instances using AWS CodeDeploy.

For next steps, read the AWS CodePipeline Documentation. Incedege 2020



Using S3 Bucket:

Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region
Europe (Stockholm) eu-north-1

Bucket type Info

General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

Directory - New
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name Info

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#) Info

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix

Upload Info

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

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

| Files and folders (1 Total, 315.0 B) | | |
|--|--------|-----------|
| <small>All files and folders in this table will be uploaded.</small> | | |
| <input type="text" value="Find by name"/> <small>< 1 ></small> | | |
| <input type="checkbox"/> Name | Folder | Type |
| <input type="checkbox"/> index.html | - | text/html |

Edit Block public access (bucket settings) Info

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Block public access to buckets and objects granted through new access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

Block public access to buckets and objects granted through any access control lists (ACLs)

S3 will ignore all ACLs that grant public access to buckets and objects.

Block public access to buckets and objects granted through new public bucket or access point policies

S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

Block public and cross-account access to buckets and objects through any public bucket or access point policies

Object Ownership

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

ACLs disabled (recommended)

All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled

Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

⚠ We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a bucket policy instead of ACLs to share data with users outside of your account simplifies permissions management and auditing.

⚠ Enabling ACLs turns off the bucket owner enforced setting for Object Ownership

Once the bucket owner enforced setting is turned off, access control lists (ACLs) and their associated permissions are restored. Access to objects that you do not own will be based on ACLs and not the bucket policy.

I acknowledge that ACLs will be restored.

Upload succeeded
View details below.

Summary

| Destination | Succeeded | Failed |
|-------------------|---------------------------|-------------------|
| s3://vedangbucket | 1 file, 315.0 B (100.00%) | 0 files, 0 B (0%) |

Files and folders | Configuration

Files and folders (1 Total, 315.0 B)

| Name | Folder | Type | Size | Status | Error |
|------------|--------|-----------|---------|-----------|-------|
| index.html | - | text/html | 315.0 B | Succeeded | - |

Static website hosting

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

Static website hosting

Disable

Enable

Hosting type

Host a static website
Use the bucket endpoint as the web address. [Learn more](#)

Redirect requests for an object
Redirect requests to another bucket or domain. [Learn more](#)

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

Index document

Specify the home or default page of the website.

Using EC2:

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like EC2 Global View, Events, Console-to-Code, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots). The main content area displays a table titled 'Instances (1/1) Info'. It shows one instance: 'dynamic-server' (Instance ID: i-0ecbd8d07a55bd2e3, Status: Running, Type: t2.micro). Below the table, detailed information for the instance is shown, including its Public IPv4 address (34.201.70.101), Private IP DNS name (ip-172-31-85-104.ec2.internal), and Instance type (t2.micro). The URL at the bottom is https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1.

This screenshot shows the 'Connect' dialog box for the instance 'dynamic-server'. It asks for a 'Username' (ubuntu) and shows the 'Public IP address' (34.201.70.101). A note says the default username is ubuntu. A warning message states: 'You have insufficient IAM permissions to connect to an instance using EC2 Instance Connect. To connect to an instance via EC2 Instance Connect, you must have an attached IAM policy that grants the following permissions: ec2:InstanceConnect:SendSSHPublicKey, ec2:DescribeInstances'. It also notes that access can be restricted by specific EC2 instances or resource tags. At the bottom are 'Cancel' and 'Connect' buttons.

```
ubuntu@ip-172-31-85-104:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-85-104:~$ mkdir pranav
ubuntu@ip-172-31-85-104:~$ cd pranav
ubuntu@ip-172-31-85-104:~/pranav$ git clone https://github.com/Pranavlovescode/Dynamic-website-hosting-sample.git
Cloning into 'Dynamic-website-hosting-sample'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 6 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), 11.16 KiB | 5.58 MiB/s, done.
```

```

ubuntu@ip-172-31-85-104:~/pranav$ ls
Dynamic-website-hosting-sample
ubuntu@ip-172-31-85-104:~/pranav$ cd Dynamic-website-hosting-sample/
ubuntu@ip-172-31-85-104:~/pranav/Dynamic-website-hosting-sample$ ls
index.js  package-lock.json  package.json
ubuntu@ip-172-31-85-104:~/pranav/Dynamic-website-hosting-sample$ npm i

added 93 packages, and audited 94 packages in 3s

16 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New patch version of npm available! 10.8.1 -> 10.8.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.2
npm notice To update run: npm install -g npm@10.8.2
npm notice

```

```

ubuntu@ip-172-31-85-104:~/pranav/Dynamic-website-hosting-sample$ npm start

> hosting-dynamic-website@1.0.0 start
> nodemon index.js

[nodemon] 3.1.4
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): ***!
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node index.js`
Server is running on port 3000

```

| Security group rule... | IP version | Type | Protocol | Port range | Source |
|------------------------|------------|------------|----------|------------|-----------|
| sgr-09762f34ff97dc77a | IPv4 | Custom TCP | TCP | 3000 | 0.0.0.0/0 |
| sgr-05780e80302575... | IPv4 | SSH | TCP | 22 | 0.0.0.0/0 |
| sgr-0f28e3996f5f4c2d0 | IPv4 | HTTP | TCP | 80 | 0.0.0.0/0 |
| sgr-0ba94a6c403d52a8 | IPv4 | HTTPS | TCP | 443 | 0.0.0.0/0 |

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

- AMIs
- AMI Catalog

Elastic Block Store

- Volumes
- Snapshots
- Lifecycle Manager

Network & Security

- Security Groups
- Elastic IPs
- Placement Groups
- Key Pairs
- Network Interfaces

Load Balancing

- Load Balancers
- Target Groups

| | | | | | | | |
|---------------------|--------------------|---------------------|----------------------|--------------------|--|----------------------|-----------------------|
| Security group name | sg-launch-wizard-1 | Security group ID | sg-09444ecdb8b403eb6 | Description | launch-wizard-1 created 2024-07-23T09:30:42.912Z | VPC ID | vpc-0dd4c1c56f9eb78a7 |
| Owner | 433618061107 | Inbound rules count | 4 | Permission entries | 1 | Outbound rules count | 1 |

Inbound rules | Outbound rules | Tags

Inbound rules (4)

| Security group rule... | IP version | Type | Protocol | Port range |
|------------------------|------------|------------|----------|------------|
| sgr-033454d2717167... | IPv4 | HTTP | TCP | 80 |
| sgr-0810859d39a92a... | IPv4 | HTTPS | TCP | 443 |
| sgr-08756637bd2e26fe7 | IPv4 | SSH | TCP | 22 |
| sgr-05bbf31ac11f942fe | IPv4 | Custom TCP | TCP | 3000 |

Hosting:

Not secure | 34.201.70.101:3000

Import favorites YouTube WhatsApp Sigma Web Devolo... Pranavlovescode (Pr... Google Download Custom... BIP39 - Mnemonic... Text structures - Pre... Other favorites

Hey this is Dynamic Website.

Not secure | 34.201.70.101:3000/about

Import favorites YouTube WhatsApp Sigma Web Devolo... Pranavlovescode (Pr... Google Download Custom... BIP39 - Mnemonic... Text structures - Pre... Other favorites

Hey this is about page.

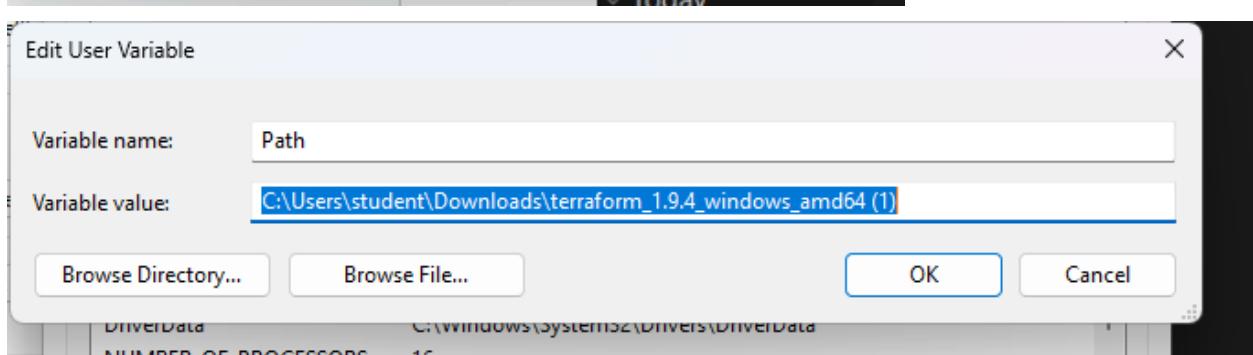
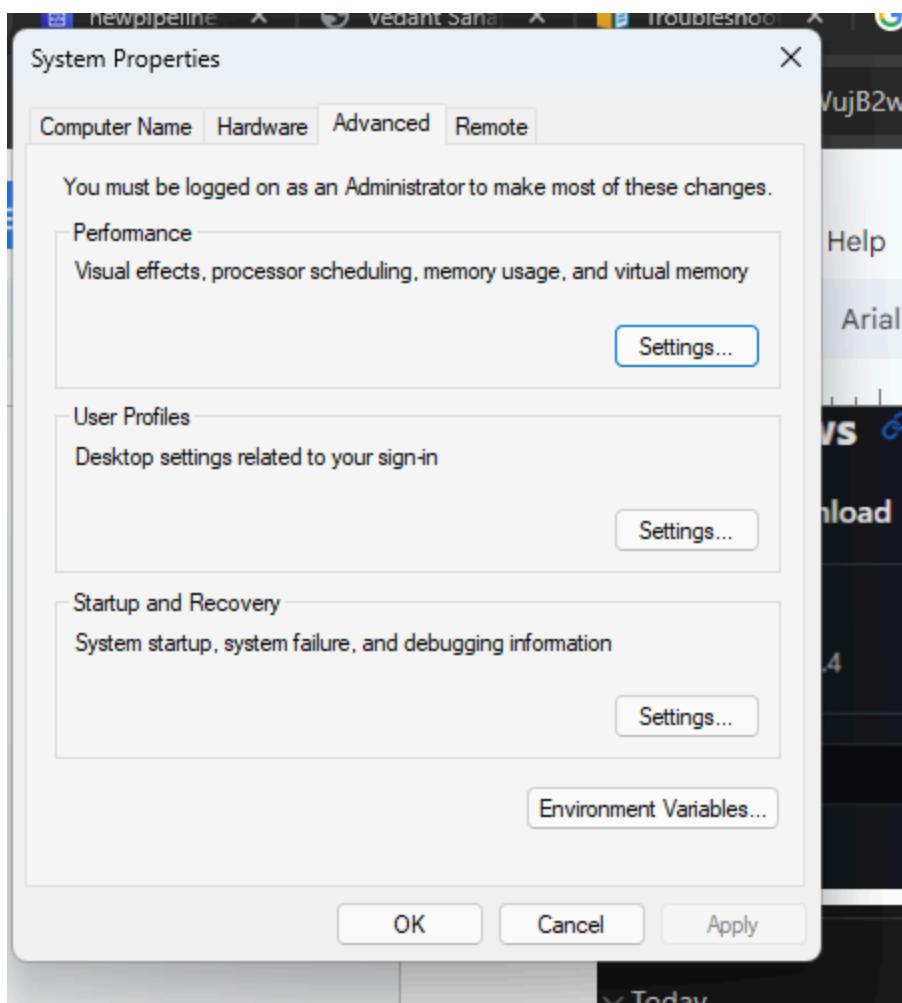
ADVANCE DEVOPS EXP-5

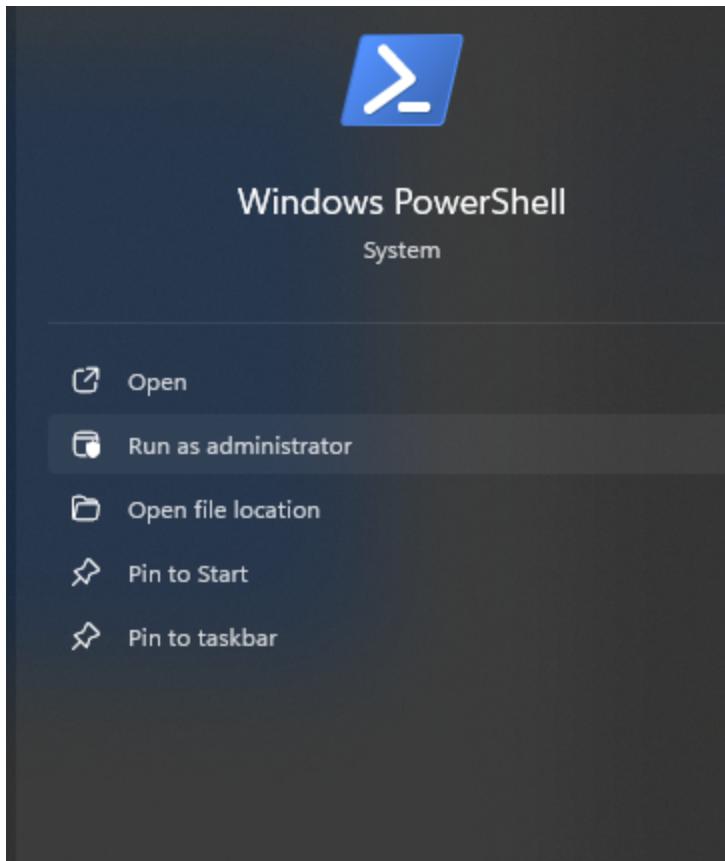
Vedang Wajge

D15A/66

The screenshot shows a software download interface. At the top, there are two download buttons for 'Version: 1.9.4' with 'Download' labels. Below this, under the 'Windows' section, there are two more download buttons: one for '386' (Version: 1.9.4) and one for 'AMD64' (Version: 1.9.4). Under the 'Linux' section, there is a table listing files:

| Name | Date modified | Type | Size |
|-----------|------------------|---------------|-----------|
| ▼ Today | | | |
| LICENSE | 13-08-2024 14:00 | Text Document | 5 KB |
| terraform | 13-08-2024 14:00 | Application | 88,918 KB |





```
PS C:\Users\student> 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
```

Experiment No 7

AIM: Installing SonarQube from the Docker Image

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

Sonarqube:latest

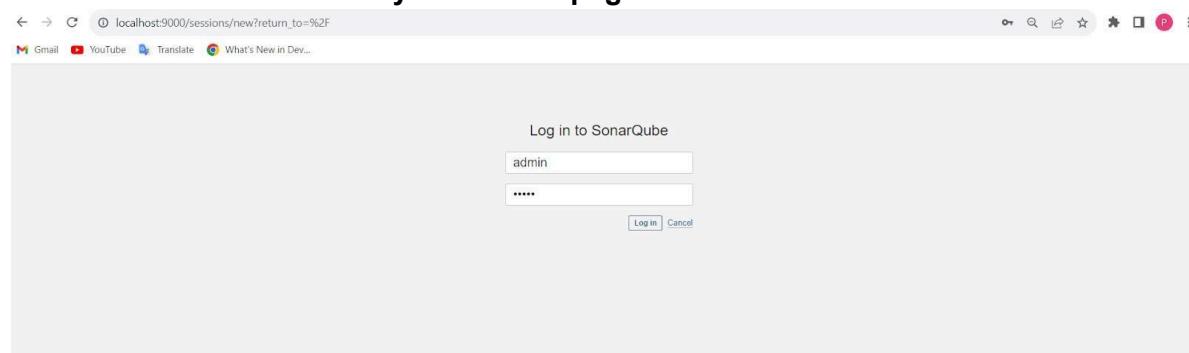
```
PS D:\Desktop\DockerFile> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
Unable to find image 'sonarqube:latest' locally
latest: Pulling from library/sonarqube
44uba2882f8eb: Pull complete
2cabec57fa36: Pull complete
c20481384b6a: Pull complete
bf7b17ee74f8: Pull complete
38617faac714: Pull complete
b795b715553d: Pull complete
c5244f6c9231: Pull complete
Digest: sha256:1fffd122cfb37ce982289dc7f5d38bb782ba05af7b5a50f7cb077ae25e60b5b9a
Status: Downloaded newer image for sonarqube:latest
1442c4e613b25aaedec05c060f020a00002b1c6dbaa27e8c5c0dad4ed8fc1f76
```



go to the SonarQube page by typing:

<http://localhost:9000/> on your browser.

Installation is successful if you see this page



Update to new password

The screenshot shows the SonarQube interface at localhost:9000/projects/create. The top navigation bar includes links for Gmail, YouTube, Translate, and What's New in Dev... The main heading is "How do you want to create your project?". Below it, a note says "Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform." A note below that says "First, you need to set up a DevOps platform configuration." There are six buttons for importing from different platforms: "Import from Azure DevOps" (Setup), "Import from Bitbucket Cloud" (Setup), "Import from Bitbucket Server" (Setup), "Import from GitHub" (Setup), "Import from GitLab" (Setup), and a "Create project manually" button.

Create project manually: Here project name is “AdDevops”

The screenshot shows the "Create a local project" step of the SonarQube project creation wizard. It is the first of two steps. The form fields are: "Project display name" (sonarqube-exp7), "Project key" (sonarqub-key-exp7), and "Main branch name" (main). Below the form is a note: "The name of your project's default branch [Learn More](#)". At the bottom are "Cancel" and "Next" buttons. A yellow warning box at the bottom left says: "⚠️ Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine." The footer includes "Community Edition v18.6 (92116) ACTIVE", "GPL v3", "Community", "Documentation", "Plugins", and "Web API".

The screenshot shows the "Set up project for Clean as You Code" step of the SonarQube project creation wizard. It is the second of two steps. The note says: "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](#)". The "Choose the baseline for new code for this project" section contains three radio button options: "Use the global setting" (selected), "Previous version" (Any code that has changed since the previous version is considered new code. Recommended for projects following regular versions or releases.), "Number of days" (Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will become part of the overall code. Recommended for projects following continuous delivery.), and "Reference branch" (Choose a branch as the baseline for the new code. Recommended for projects using feature branches.). At the bottom are "Back" and "Create project" buttons.

open Jenkins

Go to Dashboard ->Manage Jenkins -> Plugin Manager and search for SonarQube Scanner under Available plugins for Jenkins and install without restart.

Plugins

The screenshot shows the Jenkins Plugin Manager interface. A search bar at the top contains the text "sonarqube". Below the search bar, there are two tabs: "Install" and "Name ↴". The "Install" tab is selected. A list of available plugins is displayed, with one plugin checked for installation:

| Install | Name ↴ | Released |
|-------------------------------------|---|---------------|
| <input checked="" type="checkbox"/> | SonarQube Scanner 2.15 External Site/Tool Integrations Build Reports | 10 mo ago |
| <input type="checkbox"/> | Sonar Gerrit 384.vdb_755265c28d External Site/Tool Integrations | 20 days ago |
| <input type="checkbox"/> | SonarQube Generic Coverage 1.0 TODO | 4 yr 1 mo ago |

At the bottom of the screen, there are two buttons: "Install without restart" and "Download now and install after restart". A status message indicates "Update information obtained: 1 day 11 hr ago" and a "Check now" button.

Download progress

The screenshot shows the Jenkins download progress page. It lists several steps and their statuses:

| Step | Status |
|---------------------------|--|
| Preparation | <ul style="list-style-type: none">• Checking internet connectivity• Checking update center connectivity• Success |
| SSH server | Success |
| Deploy to container | Success |
| Loading plugin extensions | Success |
| SonarQube Scanner | Success |
| Loading plugin extensions | Success |

Below the table, there are two links:

- [Go back to the top page](#) (you can start using the installed plugins right away)
- Restart Jenkins when installation is complete and no jobs are running

Under Jenkins ,

Dashboard -> Manage Jenkins -> Configure System ,
Look for SonarQube Servers and enter the details. Enter the Server Authentication Token if needed.

The screenshot shows the Jenkins Global Tool Configuration page for SonarQube servers. It includes fields for Name (SonarQube), Server URL (http://localhost:9000), and Server authentication token (none). Buttons for Save and Apply are at the bottom.

Search SonarQube Scanner under Dashboard -> Manage Jenkins -> Global Tool Configuration.

Choose the latest configuration and choose Install Automatically.

The screenshot shows the Jenkins Global Tool Configuration page for SonarQube Scanner installations. It includes a section for SonarQube Scanner with a checked 'Install automatically' checkbox and a dropdown for 'Install from Maven Central' set to 'SonarQube Scanner 6.2.0.4584'. Buttons for Save and Apply are at the bottom.

create a New Item in Jenkins, choose a freestyle project.

New Item

Enter an item name
adv DevOps exp-7

Select an item type

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

OK

Choose this GitHub repository in Source Code Management.

https://github.com/shazforiot/MSBuild_firstproject.git

Under Build ->Execute SonarQube Scanner, enter these Analysis properties. Mention the SonarQube Project Key, Login, Password, and Host URL.

sonar.projectKey=SonarQueue-key
 sonar.login=admin
 sonar.password=admin
 sonar.hosturl=<http://localhost:9000/>

Configure

Description
adv DevOps exp - 7

Plain text Preview

Discard old builds

GitHub project

Project url
https://github.com/shazforiot/MSBuild_firstproject.git/

This project is parameterized

Throttle builds

Execute concurrent builds if necessary

Advanced

Source Code Management

None

Git

Repositories

Save Apply

Go to <http://localhost:9000/> and enter your previously created username.
Go to Permissions and grant the Admin user Execute Permissions.

| Group | Administer System | Administer | Execute Analysis | Create |
|----------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| sonar-administrators | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| sonar-users | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Anyone DEPRECATED | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Administrator admin | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

4 of 4 shown

Warning: Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

Build and Run:

localhost:8080/job/adv%20DevOps%20exp-7/

Jenkins

Dashboard > adv DevOps exp-7 >

Status: ✓ adv DevOps exp-7

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

Permalinks

- Last build (#6), 1 hr 59 min ago
- Last stable build (#6), 1 hr 59 min ago
- Last successful build (#6), 1 hr 59 min ago
- Last failed build (#5), 2 hr 6 min ago
- Last unsuccessful build (#5), 2 hr 6 min ago
- Last completed build (#6), 1 hr 59 min ago

Build History trend

Filter... /

| # | Date |
|----|------------------------|
| #6 | Sep 26, 2024, 12:08 AM |
| #5 | Sep 26, 2024, 12:00 AM |
| #4 | Sep 25, 2024, 11:37 PM |
| #3 | Sep 25, 2024, 11:37 PM |
| #2 | Sep 25, 2024, 11:36 PM |

localhost:8080/job/adv%20DevOps%20exp-7/6/

Jenkins

Dashboard > adv DevOps exp-7 > #6

Status: ✓ #6 (Sep 26, 2024, 12:08:16 AM)

</> Changes
Console Output
Edit Build Information
Delete build '#6'
Timings
Git Build Data

Started by user Prathamesh parmeshwar palve

This run spent:

- 15 ms waiting;
- 27 sec build duration;
- 27 sec total from scheduled to completion.

git Revision: f2bc042c04c6e72427c380bcae6d6fee7b49adf
Repository: https://github.com/shazforiot/MSBuild_firstproject.git

</> No changes.

Console Output:

Jenkins

Dashboard > adv DevOps exp-7 > #6 > Console Output

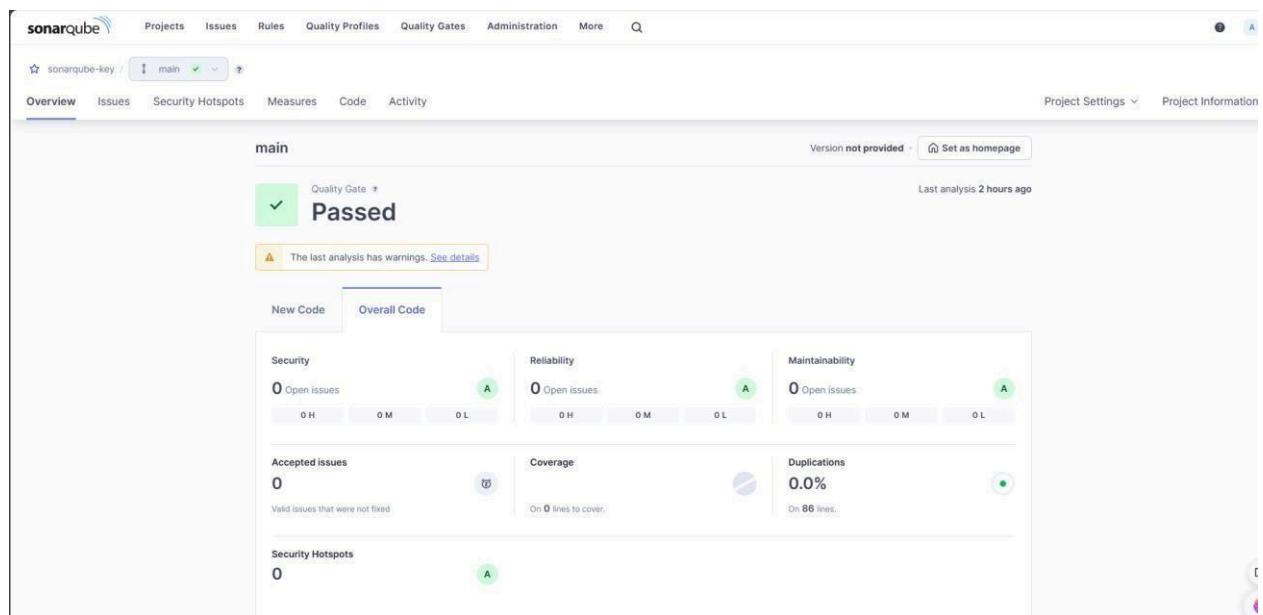
Console Output

Started by user Prathamesh Parmeshwar Patel
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\jenkins\workspace\SonarQube
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\jenkins\workspace\SonarQube\.git # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/shafzoriot/MSBuild_FirstProject.git # timeout=10
Fetching upstream changes from https://github.com/shafzoriot/MSBuild_FirstProject.git
> git.exe --version # timeout=10
> git -v version 2.46.2.windows.1
> git.exe config remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Checking out Revision f2bc042c04c6e72427c3800caeadd6fee7b49adf (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2bc042c04c6e72427c3800caeadd6fee7b49adf # timeout=10
Commit message: "updated"
> git.exe rev-list --no-walk f2bc042c04c6e72427c3800caeadd6fee7b49adf # timeout=10
Injecting SonarQube environment variables using the configuration: SonarQube
[SonarQube] \$ C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\SonarQube\bin\sonar-scanner.bat -Dsonar.host.url=http://localhost:9000 -Dsonar.projectKey=sonarqube -Dsonar.projectName=sonarqube-key -Dsonar.host.url=http://localhost:9000 -Dsonar.login=admin -Dsonar.password=SP@live@0 -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\jenkins\workspace\SonarQube
00:08:18.421 WARN Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
00:08:18.669 INFO Scanner configuration file: C:\ProgramData\Jenkins\jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\SonarQube\bin\..\conf\sonar-scanner.properties
00:08:18.672 INFO Project root configuration file: NONE
00:08:18.689 INFO SonarScanner CLI 6.2.0.4584
00:08:18.691 INFO Java 22.0.2 Oracle Corporation (64-bit)
00:08:18.692 INFO Windows 11 10.0. and64
00:08:18.722 INFO User cache: C:\Windows\system32\config\systemprofile\.sonar\cache
00:08:19.348 INFO JRE provisioning: os.windows, arch amd64
00:08:22.294 INFO Communicating with SonarQube Server 10.6.0.92116
00:08:22.802 INFO Starting SonarScanner Engine...
00:08:22.802 INFO Java 17.0.11 Eclipse Adoptium (64-bit)
.....

Console Output

00:08:42.039 INFO Using git CLI to retrieve untracked files
00:08:42.103 INFO Analyzing language associated files and files included via "sonar.text.inclusions" that are tracked by git
00:08:42.148 INFO 14 source files to be analyzed
00:08:42.322 INFO 14/14 source files have been analyzed
00:08:42.323 INFO Sensor TextAndSecretsSensor [text] [done] | time=910ms
00:08:42.329 INFO ----- Run sensors on project
00:08:42.417 INFO Sensor C# [csharp]
00:08:42.418 WARN Your project contains C# files which cannot be analyzed with the scanner you are using. To analyze C# or VB.NET, you must use the SonarScanner for .NET S.x or higher, see <https://redirect.sonarsource.com/doc/install-configure-scanner-msbuild.html>
00:08:42.418 INFO Sensor C# [csharp] [done] | time=0ms
00:08:42.419 INFO Sensor Analysis Warnings Import [csharp]
00:08:42.419 INFO Sensor Analysis Warnings Import [csharp] [done] | time=0ms
00:08:42.419 INFO Sensor C# File Caching Sensor [csharp]
00:08:42.419 WARN Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir' property.
00:08:42.419 INFO Sensor C# File Caching Sensor [csharp] [done] | time=1ms
00:08:42.420 INFO Sensor Zero Coverage Sensor
00:08:42.427 INFO Sensor Zero Coverage Sensor [done] | time=8ms
00:08:42.429 INFO SCM Publisher SCM provider for this project is: git
00:08:42.430 INFO SCM Publisher 4 source files to be analyzed
00:08:42.455 INFO SCM Publisher 4/4 source files have been analyzed (done) | time=425ms
00:08:42.857 INFO CP Executor Calculating CPD for 0 files
00:08:42.858 INFO CP Executor CPD calculation finished (done) | time=0ms
00:08:42.863 INFO SCM revision ID 'f2bc042c04c6e72427c3800caeadd6fee7b49adf'
00:08:43.105 INFO Analysis report generated in 99ms, dir size=199.9 kB
00:08:43.142 INFO Analysis report compressed in 24ms, zip size=22.3 kB
00:08:43.681 INFO Analysis report uploaded in 53ms
00:08:43.683 INFO ANALYSIS SUCCESSFUL, you can find the results at: <http://localhost:9000/dashboard?id=sonarqube>
00:08:43.684 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
00:08:43.684 INFO More about the report processing at <http://localhost:9000/api/ce/task?id=5ba2c6da-d3cb-4ec9-9888-308ad22f8670>
00:08:43.700 INFO Analysis total time: 17.889 s
00:08:43.703 INFO SonarScanner Engine completed successfully
00:08:43.778 INFO EXECUTION SUCCESS
00:08:43.782 INFO Total time: 25.150s
Finished: SUCCESS

Project on sonarqube:



Conclusion: Thus, we have successfully installed SonarQube from Docker image.

Experiment 8

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

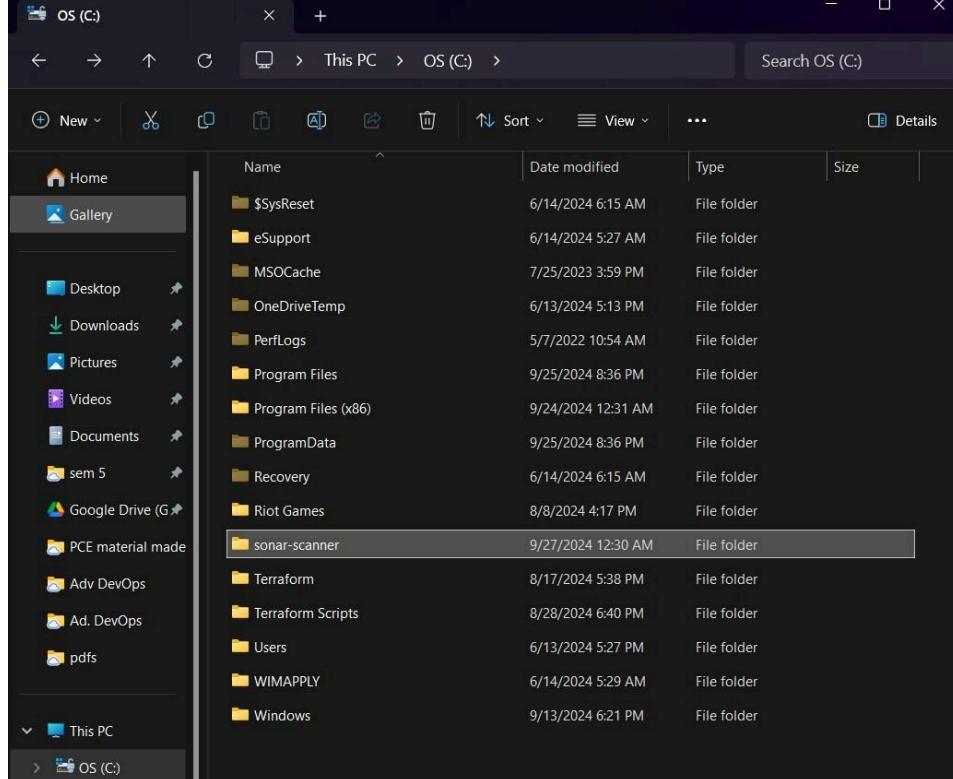
Step 1 : Visit the following link to download the SonarScanner CLI -

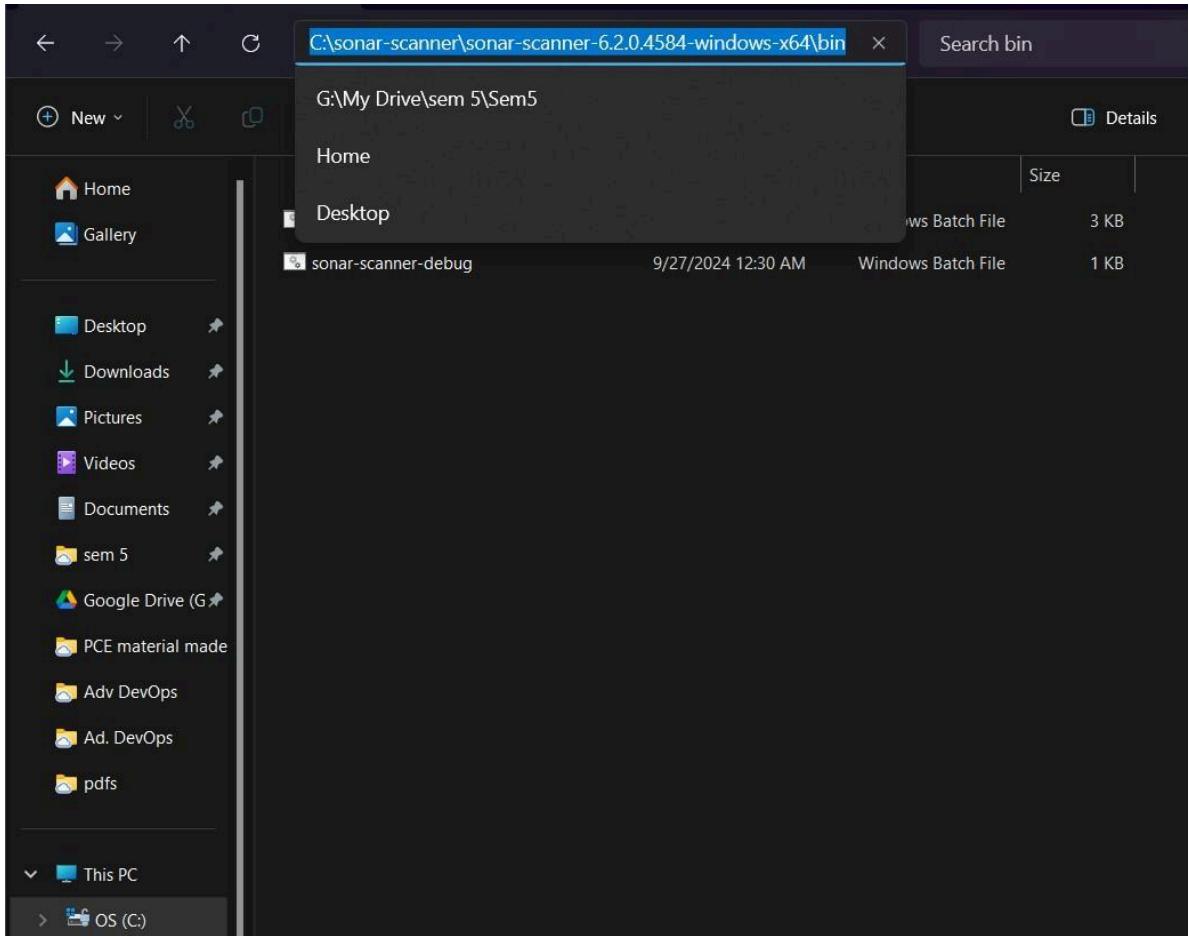
<https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscanner/> and then click on Windows x-64 to download the zip file.

The screenshot shows the SonarScanner CLI documentation page. The left sidebar contains navigation links for SonarQube, Server installation and setup, Analyzing source code, Scanners, SonarScanner for NPM, Analysis parameters, Languages, Test coverage, and Importing external issues. The main content area features a title 'SonarScanner CLI' with tabs for 'SonarScanner' and 'Issue Tracker'. Below this is a section titled '6.2' with a date '2024-09-17'. It includes a note about support for PKCS12 truststore generated with OpenSSL, download links for various platforms, and release notes. A callout box points to a note: 'The SonarScanners run on code that is checked out. See [Verifying the code checkout step of your build](#).'. To the right, a sidebar titled 'On this page' lists various documentation topics.

https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-6.2.0.4584-windows-x64.zip?ql=1+w3exrs*.qcl_au*MTUyMDY5NDYzMS4xNzI3MjczNDMS*_qa*NTAzMjA1MzcLjE3MjcyNzI3MzQzOS4xLjEuMTcyNzI3OTAxNs4yM...

Step 2: Extract the content in C drive and name the folder sonar-scanner





Step 3: Open Command Prompt and run as administrator and run the following commands –

```
cd C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin dir
```

```
sonar-scanner.bat
```

```
C:\Administrator: Command Prompt
C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>dir
 Volume in drive C is OS
 Volume Serial Number is E83B-22B8

 Directory of C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin

25-09-2024 21:18 <DIR> .
25-09-2024 21:18 <DIR> ..
25-09-2024 21:18 805 sonar-scanner-debug.bat
25-09-2024 21:18 2,553 sonar-scanner.bat
2 File(s) 3,358 bytes
2 Dir(s) 8,509,411,328 bytes free

C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>sonar-scanner.bat
22:44:22.348 INFO Scanner configuration file: C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin..\conf\sonar-scanner.properties
22:44:22.353 INFO Project root configuration file: NONE
22:44:22.369 INFO SonarScanner CLI 6.2.0.4584
22:44:22.370 INFO Java 17.0.12 Eclipse Adoptium (64-bit)
22:44:22.371 INFO Windows 11 10.0 amd64
22:44:22.389 INFO User cache: C:\Users\User\.sonar\cache
22:44:22.827 INFO JRE provisioning: os[windows], arch[amd64]
22:44:23.921 INFO EXECUTION FAILURE
22:44:23.923 INFO Total time: 1.577s
22:44:23.923 ERROR Error during SonarScanner CLI execution
java.lang.IllegalStateException: Error status returned by url [https://api.sonarcloud.io/analysis/jres?os=windows&arch=amd64]: 401
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callUrl(ServerConnection.java:182)
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callApi(ServerConnection.java:145)
        at org.sonarsource.scanner.lib.internal.http.ServerConnection.callRestApi(ServerConnection.java:123)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreMetadata(JavaRunnerFactory.java:159)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreFromServer(JavaRunnerFactory.java:138)
        at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.createRunner(JavaRunnerFactory.java:85)
        at org.sonarsource.scanner.lib.internal.ScannerEngineLauncherFactory.createLauncher(ScannerEngineLauncherFactory.java:53)
        at org.sonarsource.scanner.lib.ScannerEngineBootstrapper.bootstrap(ScannerEngineBootstrapper.java:118)
        at org.sonarsource.scanner.cli.Main.analyze(Main.java:75)
        at org.sonarsource.scanner.cli.Main.main(Main.java:63)
22:44:23.925 ERROR
22:44:23.926 ERROR Re-run SonarScanner CLI using the -X switch to enable full debug logging.

C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin>
```

Step 4: Open Jenkins and create a pipeline and name the pipeline SonarQube Pipeline and then click on okay.

Step 5: In the configuration, under the Pipeline Section write the following Pipeline Script -

```

node {
    stage('Cloning the GitHub Repo') {
        git 'https://github.com/shazforiot/GOL.git'
    }
    stage('SonarQube analysis') {
        withSonarQubeEnv('sonarqube') {
            sh "C:/sonar-scanner/sonar-scanner-6.2.0.4584-windows-x64/bin/sonar-
scanner.bat" +
                "-Dsonar.login=admin" +
                "-Dsonar.password=5Palve@08" +
                "-Dsonar.projectKey=sonarqube" +
                "-Dsonar.exclusions=vendor/**,resources/**,/**/*.java" +
                "-Dsonar.host.url=http://192.168.1.40:9000/"
        }
    }
}

```

Then click on the save button.

Dashboard > SonarQube Pipeline > Configuration

Configure

General

Advanced Project Options

Pipeline

Definition

Pipeline script

```

1+ node {
2+   stage('Cloning the GitHub Repo') {
3+     git 'https://github.com/shazfariot/GOL.git'
4+   }
5+   stage('SonarQube analysis') {
6+     withSonarQubeEnv('sonarqube') {
7+       sh "C:/sonar-scanner/sonar-scanner-6.2.0.4584-windows-x64/bin/sonar-scanner.bat" +
8+         "-DSonar.login=admin" +
9+         "-DSonar.password=5Palue@08" +
10+        "-DSonar.projectKey=sonarqube" +
11+        "-DSonar.exclusions=vendor/**,resources/**/*.*.java" +
12+        "-DSonar.host.url=http://192.168.1.40:9000"
13+     }
14+   }
15+ }
16+

```

Use Groovy Sandbox

Pipeline Syntax

Save **Apply**

REST API Jenkins 2.462.2

Jenkins

Dashboard > SonarQube Pipeline >

SonarQube Pipeline

Status

</> Changes

▷ Build Now

⚙ Configure

Delete Pipeline

SonarQube

Stages

✍ Rename

?

Pipeline Syntax

Build History trend

Filter... /

- Last build (#8), 23 min ago
- Last stable build (#8), 23 min ago
- Last successful build (#8), 23 min ago
- Last failed build (#6), 1 hr 1 min ago
- Last unsuccessful build (#7), 31 min ago
- Last completed build (#8), 23 min ago

Sep 27, 2024, 2:00 AM #8

Sep 27, 2024, 1:52 AM #7

Jenkins

Dashboard > SonarQube Pipeline > #8

Console Output

Status

</> Changes

Console Output

Edit Build Information

Delete build '#8'

Timings

Git Build Data

Pipeline Overview

Pipeline Console

Replay

Pipeline Steps

Workspaces

Previous Build

Download Copy View as plain text

Skipping 4,246 KB. Full Log

```

02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 789. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 512. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 248. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 886. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 249. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 662. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 615. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 664. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 913. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 810. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-
web/tools/jmeter/docs/api/org/apache/jmeter/visualizers/PropertyControlGui.html for block at line 668. Keep only the first 100 references.
02:06:47.449 WARN Too many duplication references on file gameoflife-

```

Step 7: Now, visit <http://192.168.1.40:9000/dashboard?id=sonarqube> to see the result.

The screenshot shows the SonarQube dashboard for the 'main' project. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, More, and a search bar. The main content area displays a summary of the analysis: 683k Lines of Code, Version not provided, and a 'Passed' status with a green checkmark icon. A message encourages users to clean their code base. Below this, tabs for 'New Code' and 'Overall Code' are shown, along with sections for New issues (0), Accepted issues (0), Coverage (On 0 New Lines to cover), Duplications (On 0 Nine Lines), and Security Hotspots (0). The last analysis was 25 minutes ago.

This screenshot shows the SonarQube dashboard for the 'main' project, but the analysis has failed. The status is displayed as 'Failed' with a red X icon. The overall summary shows 683k Lines of Code, Version not provided, and a 'Failed' status. The 'Overall Code' tab is selected. The dashboard highlights several critical issues: 0 Open issues under Security, 68k Open issues under Reliability, and 164k Open issues under Maintainability. The 'Accepted issues' section shows 0 valid issues. Coverage is listed as 50.6% (On 759k lines). The last analysis was 26 minutes ago.