

DEVOPS PROJECT

SCENARIO:

YOUR ORGANIZATION IS AIMING TO AUTOMATE THE INFRASTRUCTURE CREATION AND DEPLOYMENT PROCEDURE. YOU ARE TASKED WITH SETTING UP AND CONFIGURING THE AWS ENVIRONMENT TO MEET SPECIFIC REQUIREMENTS.

QUESTION:

1. CREATE AN INSTANCE IN AWS MANUALLY WHICH WILL BE ACTING AS THE TERRAFORM MASTER MACHINE USING WHICH YOU HAVE TO CREATE A VPC AND A SUBNET IN THAT VPC, YOU WILL THEN HAVE TO CREATE 3 MORE INSTANCES IN THAT SUBNET
2. ANY CONFIGURATIONS NEEDED TO BE MADE HAS TO BE ACHIEVED USING ANSIBLE, I.E. JENKINS AND KUBERNETES INSTALLATIONS IN THE REQUIRED MACHINES.
3. ONE OF THE 3 MACHINES CREATED USING TERRAFORM WILL BE BEHAVING AS THE MASTER FOR JENKINS AND THE OTHER WILL BE THE KUBERNETES MASTER MACHINE WHICH WILL BE REQUIRED TO BE ADDED AS A KUBERNETES AGENT IN THE JENKINS DASHBOARD. THE THIRD MACHINE WILL BE FULFILLING THE ROLE OF YOUR KUBERNETES SLAVE NODE WHICH DOES NOT NEED TO BE ADDED AS A JENKINS AGENT.
4. A DOCKERFILE ALSO NEEDS TO BE CREATED TO CONTAINERIZE THE WEBSITE THAT HAS BEEN MENTIONED IN THE GITHUB LINK PROVIDED BELOW. THE SAME IS TO BE ACHIEVED BY FORKING THE REPOSITORY ON YOUR GITHUB ACCOUNT.

GITHUB URL: [HTTPS://GITHUB.COM/SAMEER-8080/WEBSITE-PRT-ORG](https://github.com/SAMEER-8080/WEBSITE-PRT-ORG)

1. A JENKINSFILE HAS TO BE CREATED THAT WILL BE INCLUDED IN THE FORKED REPOSITORY AS WELL WHICH WILL ALSO CONTAIN THE MANIFEST FILES REQUIRED. MORE INFORMATION ON THE MANIFEST FILES REQUIRED IS MENTIONED IN THE NEXT POINT. THE JENKINSFILE SHOULD BUILD A NEW IMAGE UPON CHANGES TO THE REPOSITORY TO ENSURE THAT THE LATEST WEBSITE IS DISPLAYED AT ALL TIMES.

THE MANIFEST FILE REQUIRED ARE:

- I. A DEPLOYMENT FILE THAT WILL CREATE THE REQUIRED DEPLOYMENT FROM THE IMAGE WHICH WILL BE CREATED FROM THE DOCKERFILE AND UPLOADED TO DOCKERHUB. THE DEPLOYMENT WILL HAVE A TOTAL OF 5 REPLICAS.
 - II. A SERVICE FILE WHICH WILL BE OF TYPE NODEPORT WHICH WILL EXPOSE THE ABOVE MENTIONED DEPLOYMENT ON PORT 30010.
2. THE JENKINS PIPELINE WHICH WILL BE CREATED SHOULD BE AUTOMATICALLY TRIGGERED ONCE CHANGES ARE MADE TO THE FORKED REPOSITORY.

Jul 14 4:26 PM

Start Course | Intellipaat | Launch an instance | EC2 | speed test - Google Search | +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:

YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

aws Services Search [Alt+S]

EC2 Instances Launch an instance

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 Terraform-Master Add additional tags

Application and OS Images (Amazon Machine Image) Info

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

Search our full catalog including 1000s of application and OS Images

Summary

Number of Instances Info
1

Software Image (AMI)
Canonical, Ubuntu, 24.04 LTS, ... read more
ami-04a81a99f5ec58529

Virtual server type (Instance type)
t2.micro

Firewall (security group)
default

Storage (volumes)
1 volume(s) - 8 GB

Cancel Launch instance Review commands

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

This screenshot shows the 'Launch an instance' wizard on the AWS EC2 console. It includes fields for 'Name and tags', 'Application and OS Images (Amazon Machine Image)', and a summary section with configuration details like the number of instances, AMI, instance type, and storage. The 'Launch instance' button is prominently displayed at the bottom right.

Jul 14 4:31 PM

Start Course | Intellipaat | Instances | EC2 | us-east-1 | speed test - Google Search | +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstancesinstanceId=i-043be26a1500e342b;v=3;\$case=tags=true%5C;client=false;... | All Bookmarks

YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

aws Services Search [Alt+S]

EC2 Dashboard EC2 Global View Events Console-to-Code Preview

Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations

Images AMIs AMI Catalog

Instances (1/1) Info

Find Instance by attribute or tag (case-sensitive)

Instance ID = i-043be26a1500e342b | Clear filters

All states

| Name | Instance ID | Instance state | Instanc... | Status check |
|------------------|---------------------|----------------|------------|--------------|
| Terraform-Master | i-043be26a1500e342b | Running | t2.micro | ... |

i-043be26a1500e342b (Terraform-Master)

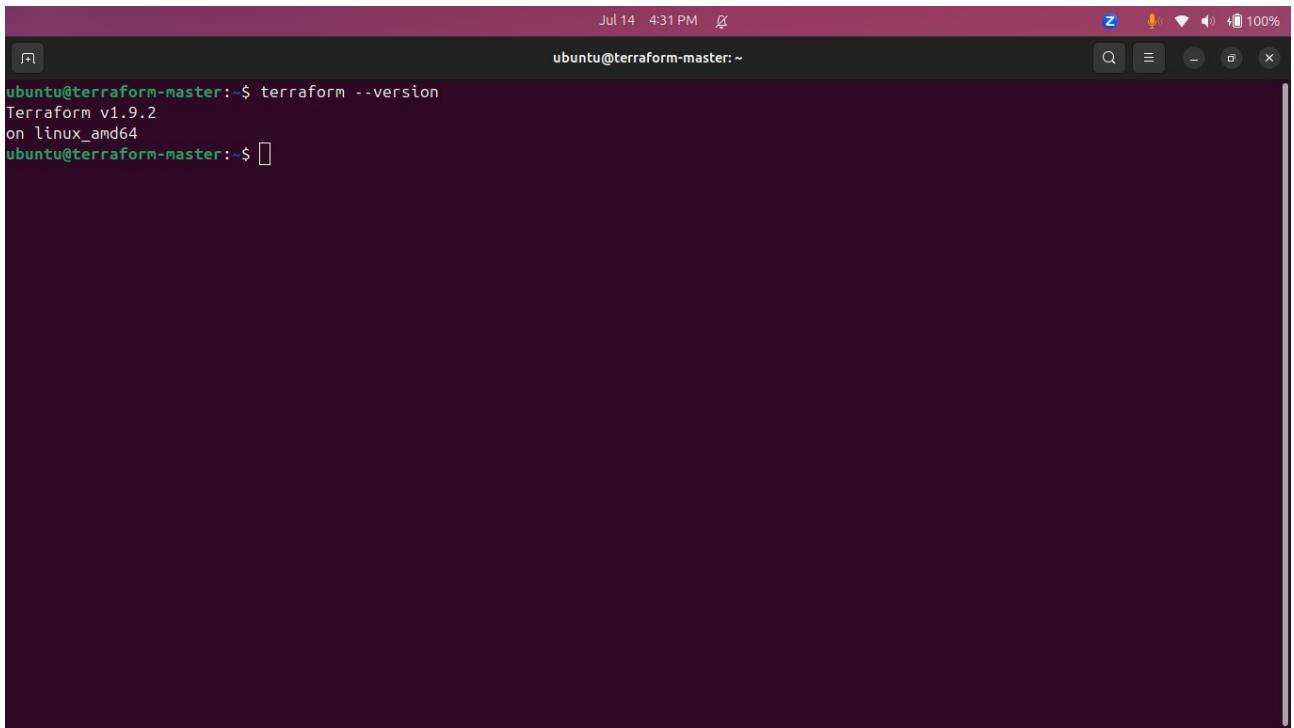
Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary Info

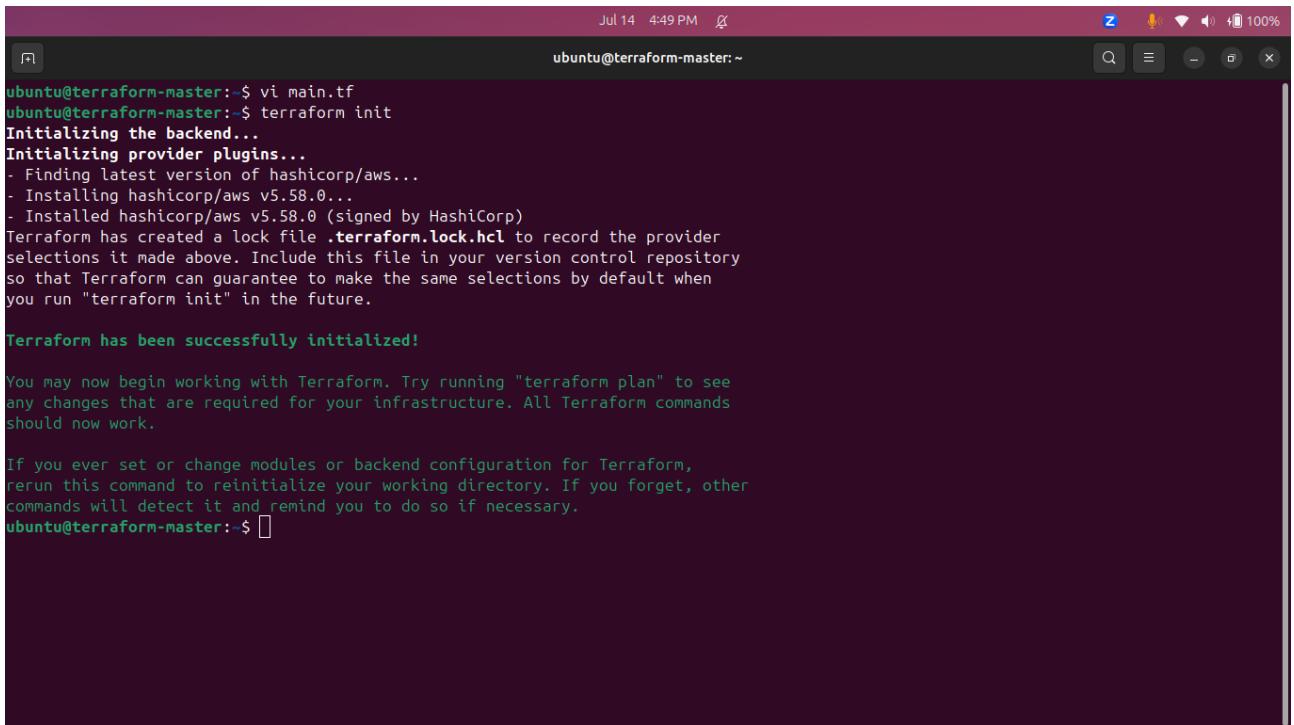
| Instance ID | Public IPv4 address | Private IPv4 addresses |
|--|----------------------------|--|
| i-043be26a1500e342b (Terraform-Master) | 54.85.1.194 open address | 172.31.80.248 |
| IPv6 address | Instance state | Public IPv4 DNS |
| - | Running | ec2-54-85-1-194.compute-1.amazonaws.com open address |

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

This screenshot shows the EC2 Instances page displaying a single instance named 'Terraform-Master'. The instance is listed in the main table with its ID, state (Running), and IP address. A detailed view of the instance is shown below, including its summary, public and private IP addresses, and DNS information.



```
Jul 14 4:31 PM ⓘ
ubuntu@terraform-master:~$ terraform --version
Terraform v1.9.2
on linux_amd64
ubuntu@terraform-master:~$ ⎵
```



```
Jul 14 4:49 PM ⓘ
ubuntu@terraform-master:~$ vi main.tf
ubuntu@terraform-master:~$ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.58.0...
- Installed hashicorp/aws v5.58.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

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

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
ubuntu@terraform-master:~$ ⎵
```

terraform script

```
provider "aws" {
  access_key = "AKIA3FLD2BXIDIVK3END"
  secret_key = "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"
  region = "us-east-1"
}
```

```

# Create a VPC
resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
  enable_dns_hostnames = true
  tags = {
    Name = "main-vpc"
  }
}

resource "aws_internet_gateway" "main-ig" {
  vpc_id = aws_vpc.main.id
  tags = {
    Name = "main-ig"
  }
}

# Setting up the route table
resource "aws_route_table" "pub-rt" {
  vpc_id = aws_vpc.main.id
  route {
    # pointing to the internet
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.main-ig.id
  }
  route {
    ipv6_cidr_block = "::/0"
    gateway_id = aws_internet_gateway.main-ig.id
  }
  tags = {
    Name = "pub-rt"
  }
}

# Create a subnet
resource "aws_subnet" "main" {
  vpc_id      = aws_vpc.main.id
  cidr_block   = "10.0.1.0/24"
  availability_zone = "us-east-1a"

  tags = {
    Name = "main-subnet"
  }
}

# Associating the subnet with the route table
resource "aws_route_table_association" "pub-sub-rt-assoc" {
  subnet_id = aws_subnet.main.id
  route_table_id = aws_route_table.pub-rt.id
}

# Create a security group
resource "aws_security_group" "instance_sg" {
  vpc_id = aws_vpc.main.id
}

```

```

ingress {
  from_port  = 0
  to_port    = 0
  protocol   = "-1"
  cidr_blocks = ["0.0.0.0/0"]
}
egress {
  from_port  = 0
  to_port    = 0
  protocol   = "-1"
  cidr_blocks = ["0.0.0.0/0"]
}

tags = {
  Name = "instance-sg"
}
}

resource "aws_instance" "jenkins-master" {
  ami          = "ami-04a81a99f5ec58529"
  instance_type = "t2.small"
  key_name     = "my-keypair"
  associate_public_ip_address = true
  subnet_id    = aws_subnet.main.id
  vpc_security_group_ids = [aws_security_group.instance_sg.id]

  tags = {
    Name = "jenkins-master"
  }
}

resource "aws_instance" "kubernetes-master" {

  key_name     = "my-keypair"
  ami          = "ami-04a81a99f5ec58529"
  instance_type = "t2.medium"
  associate_public_ip_address = true
  subnet_id    = aws_subnet.main.id
  vpc_security_group_ids = [aws_security_group.instance_sg.id]

  tags = {
    Name = "kubernetes-master"
  }
}

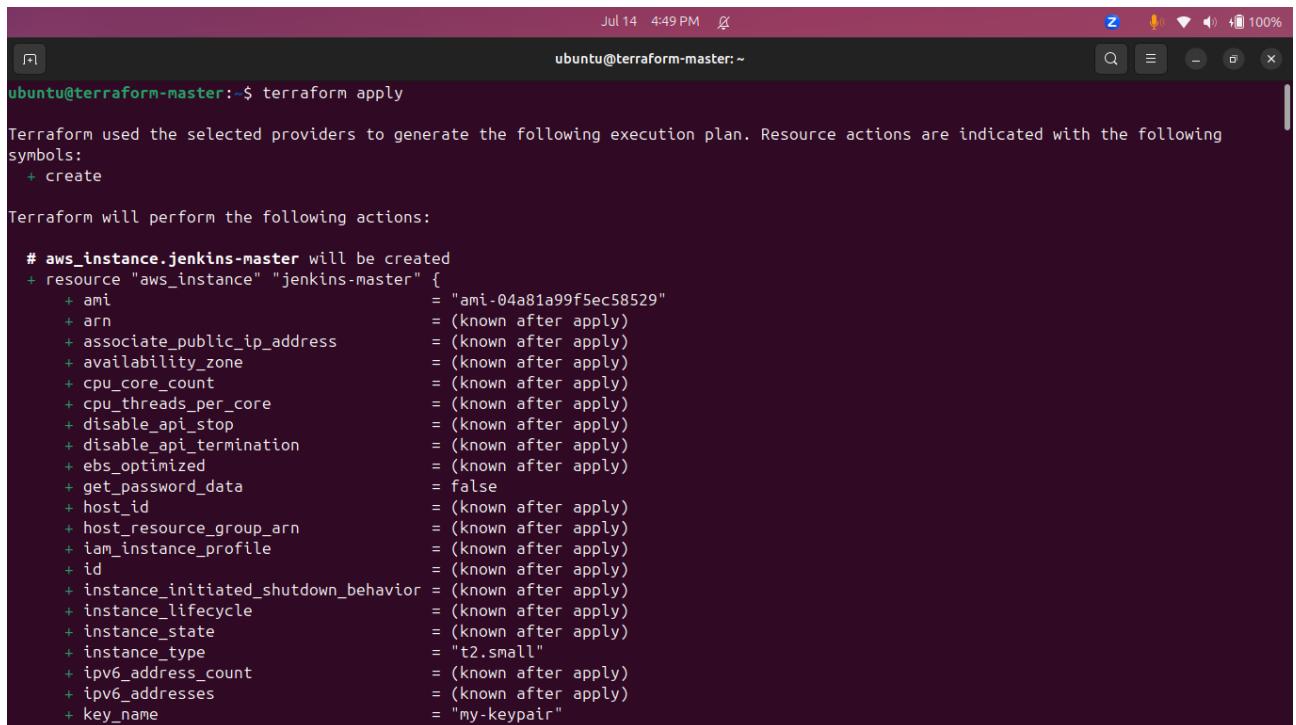
resource "aws_instance" "worker-node" {

  key_name     = "my-keypair"
  ami          = "ami-04a81a99f5ec58529"
  instance_type = "t2.micro"
}

```

```
associate_public_ip_address = true
subnet_id      = aws_subnet.main.id
vpc_security_group_ids = [aws_security_group.instance_sg.id]

tags = {
  Name = "worker-node"
}
}
```



The screenshot shows a terminal window on an Ubuntu system. The title bar indicates it's running on 'ubuntu@terraform-master:~'. The terminal displays the command 'terraform apply' followed by its execution plan. The output shows that a new AWS instance resource named 'jenkins-master' will be created. The resource block includes various configuration parameters such as AMI, security group, and instance type.

```
ubuntu@terraform-master:~$ terraform apply
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# aws_instance.jenkins-master will be created
+ resource "aws_instance" "jenkins-master" {
  + ami                               = "ami-04a81a99f5ec58529"
  + arn                               = (known after apply)
  + associate_public_ip_address       = (known after apply)
  + availability_zone                 = (known after apply)
  + cpu_core_count                   = (known after apply)
  + cpu_threads_per_core            = (known after apply)
  + disable_api_stop                = (known after apply)
  + disable_api_termination         = (known after apply)
  + ebs_optimized                   = (known after apply)
  + get_password_data               = false
  + host_id                          = (known after apply)
  + host_resource_group_arn          = (known after apply)
  + iam_instance_profile             = (known after apply)
  + id                               = (known after apply)
  + instance_initiated_shutdown_behavior = (known after apply)
  + instance.lifecycle               = (known after apply)
  + instance.state                  = (known after apply)
  + instance_type                   = "t2.small"
  + ipv6_address_count              = (known after apply)
  + ipv6_addresses                  = (known after apply)
  + key_name                        = "my-keypair"
```

Jul 14 4:53 PM

Start Course | Intellipaat Instances | EC2 | us-east-1 speed test - Google Search labs/terraform/install at +

YouTube xdhivayax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

aws Services Search iam

EC2 Dashboard EC2 Global View Events Console-to-Code Preview Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations Images AMIs AMI Catalog CloudShell Feedback

Instances (3/5) Info Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

| Name | Instance ID | Instance state | Instanc... | Status check |
|----------------------------|---------------------|----------------|------------|-------------------|
| server created by dev u... | i-0b8a7e7d148726d7a | Running | t2.micro | 2/2 checks passed |
| Jenkins-master | i-079695ec68f36f031 | Running | t2.small | Initializing |
| kubernetes-master | i-005cbd43fa9c85e8f | Running | t2.medium | Initializing |
| worker-node | i-007fb8184749b2c14 | Running | t2.micro | Initializing |
| Terraform-Master | i-043be26a1500e342b | Running | t2.micro | 2/2 checks passed |

3 instances selected

Monitoring Configure CloudWatch agent

It is taking a bit longer than usual to fetch your data

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

Jul 14 4:54 PM

Start Course | Intellipaat Instances | EC2 | us-east-1 vpcs | VPC Console speed test - Google Search labs/terraform/install at +

YouTube xdhivayax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

aws Services Search [Alt+S]

VPC dashboard EC2 Global View Filter by VPC

Virtual private cloud Your VPCs (1/2) Info Create VPC

Last updated less than a minute ago

| Name | VPC ID | State | IPv4 CIDR | IPv6 CIDR |
|----------|-----------------------|-----------|---------------|-----------|
| main-vpc | vpc-0e087138a80351039 | Available | 10.0.0.0/16 | - |
| - | vpc-0b6ed3457eb1fb30f | Available | 172.31.0.0/16 | - |

vpc-0e087138a80351039 / main-vpc

Details Resource map CIDRs Flow logs Tags Integrations

Details

| | | | | | | | |
|--------|-----------------------|-------|-----------|---------------|---------|----------------|---------|
| VPC ID | vpc-0e087138a80351039 | State | Available | DNS hostnames | Enabled | DNS resolution | Enabled |
|--------|-----------------------|-------|-----------|---------------|---------|----------------|---------|

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

```
Jul 14 5:21 PM ✘
ubuntu@jenkins-master: ~
ubuntu@jenkins-master:~/.ssh$ cd
ubuntu@jenkins-master: $ ansible --version
ansible [core 2.16.8]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Apr 10 2024, 05:33:47) [GCC 13.2.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
ubuntu@jenkins-master: $
```

```
Jul 14 5:20 PM ✘
ubuntu@jenkins-master: ~/.ssh
ubuntu@jenkins-master:~/.ssh
ubuntu@jenkins-master:~/.ssh$ ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_ed25519
Your public key has been saved in /home/ubuntu/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:9pAGZtL4gT3swjpSP+Al0u105Ssg0qIoegVHfmT5kxQ ubuntu@jenkins-master
The key's randomart image is:
+--[ED25519 256]--+
|          .E.      |
|     .*+ .       |
|    o+oXo..      |
|   o.o*.*+.     |
|+ BoB.+ S.      |
|o* X.+ o +     |
|= +.+. . .     |
|o... . .        |
|..             |
+---[SHA256]-----+
ubuntu@jenkins-master: $ cd .ssh/
ubuntu@jenkins-master:~/.ssh$ ls
authorized_keys  id_ed25519  id_ed25519.pub
ubuntu@jenkins-master:~/.ssh$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAI0VWhGfreFiluzm/2Rw9kNVUPJ04qWcNjy70LJJzNk5  ubuntu@jenkins-master
ubuntu@jenkins-master:~/.ssh$
```

```
Jul 14 5:21 PM ✘ ubuntu@kube-master: ~.ssh

ubuntu@jenkins-master: ~ x      ubuntu@kube-master: ~.ssh x
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCV2wtz9wzvP+e+LVP0QixCh+xrzmVILsdrpi+sbcnXqa3Y3A+L1mEOL5M9oMVRPDRlGfUWkKQFknuSJKT8XMmhZVvBgfUnU3R
mhS2pFubri8swyutKaeth5Ghm01J/SFzMhwspdjh6/au1i6jFOTBFa38W3uToe0IovxbBnzT2o5D1E+qx3DEdKceXD8lykkP0yF+CoMsSzjBLN54bYdmDW+7AgcwF4pRm7D2260
Hi6cFkViust3acIbipZAQc2vrEJa17LGzbabjft4qPGcs1qR4h0wGCAxSAL90IHf0izPfvxsx9siw3jYNe59fdakC8+gV/83+lwxLKFkmQFSQl my-keypair
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAIOWVhJGfreFIluzm/2Rw9kNVUPJ04qWcNjy70LJJzNk5  ubuntu@jenkins-master
:wc[]
```

```
ubuntu@jenkins-master:~
```

```
ubuntu@jenkins-master:/etc/ansible$ cat hosts
[kubemaster]
3.94.180.191 ansible_user=ubuntu ansible_ssh_private_key_file=/home/ubuntu/.ssh/id_ed25519

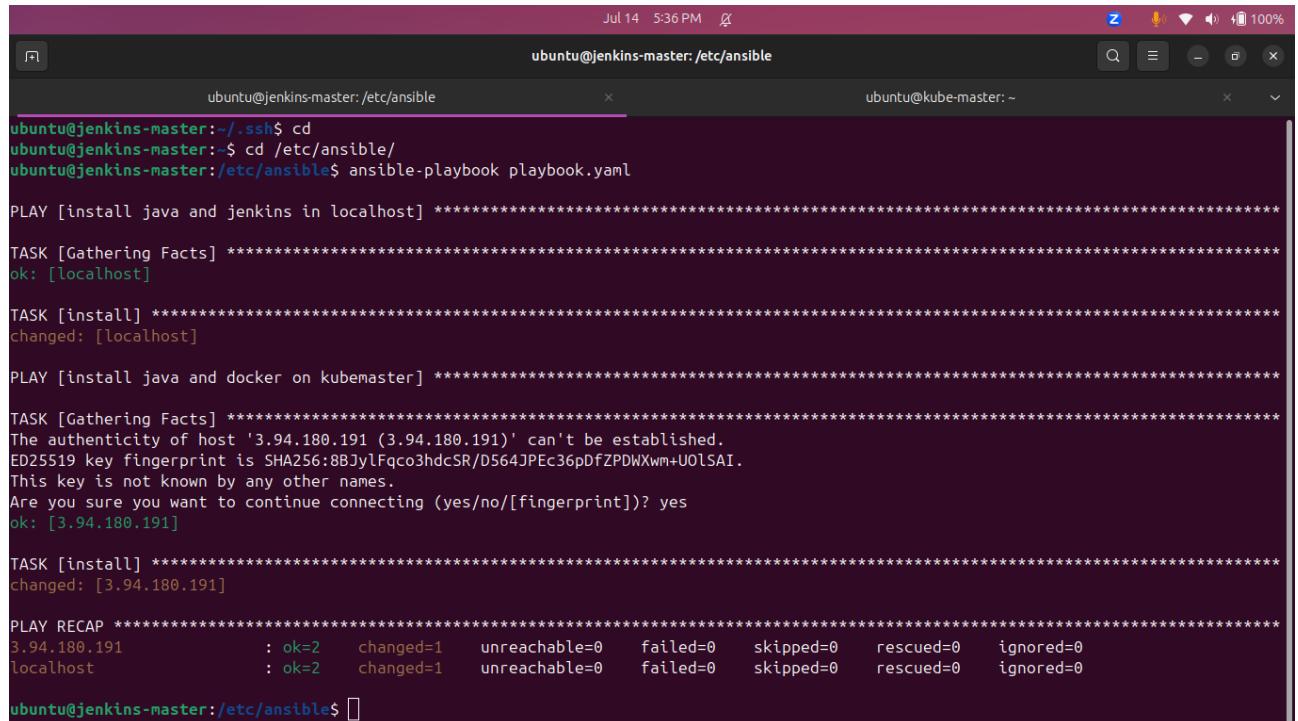
ubuntu@jenkins-master:/etc/ansible$ cat kubemaster.sh
sudo apt-get update
sudo apt-get install openjdk-17-jdk -y
sudo apt-get install docker.io -y

ubuntu@jenkins-master:/etc/ansible$ cat localhost.sh
sudo apt-get update
sudo apt-get install openjdk-17-jdk -y
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
  https://pkg.jenkins.io/debian binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins -y
ubuntu@jenkins-master:/etc/ansible$ 
```

PLAYBOOK.YAML

```
---
- name: install java and jenkins in localhost
  hosts: localhost
  become: true
  tasks:
    - name: install
      script: localhost.sh

- name: install java and docker on kubemaster
  hosts: kubemaster
  become: true
  tasks:
    - name: install
      script: kubemaster.sh
```



The screenshot shows a terminal window with two tabs. The left tab is titled 'ubuntu@jenkins-master:/etc/ansible' and the right tab is titled 'ubuntu@kube-master:~'. The terminal output is as follows:

```
ubuntu@jenkins-master:~/.ssh$ cd
ubuntu@jenkins-master:~$ cd /etc/ansible/
ubuntu@jenkins-master:/etc/ansible$ ansible-playbook playbook.yaml

PLAY [install java and jenkins in localhost] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [install] ****
changed: [localhost]

PLAY [install java and docker on kubemaster] ****
TASK [Gathering Facts] ****
The authenticity of host '3.94.180.191 (3.94.180.191)' can't be established.
ED25519 key fingerprint is SHA256:8BJylFqco3hdcSR/D564JPEc36pDFZPDWXwm+U0lSAI.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
ok: [3.94.180.191]

TASK [install] ****
changed: [3.94.180.191]

PLAY RECAP ****
3.94.180.191      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0   ignored=0
localhost          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0   ignored=0

ubuntu@jenkins-master:/etc/ansible$
```

Jul 14 5:38 PM

ubuntu@jenkins-master: ~

```
ubuntu@jenkins-master:/etc/ansible$ cd
ubuntu@jenkins-master: $ jenkins --version
2.467
ubuntu@jenkins-master: $ java --version
openjdk 17.0.11 2024-04-16
OpenJDK Runtime Environment (build 17.0.11+9-Ubuntu-1)
OpenJDK 64-Bit Server VM (build 17.0.11+9-Ubuntu-1, mixed mode, sharing)
ubuntu@jenkins-master: $ 
```

Jul 14 5:40 PM

ubuntu@jenkins-master: ~

```
ubuntu@kube-master: ~
```

ubuntu@jenkins-master: ~

```
ubuntu@kube-master: $ docker --version
Docker version 27.0.3, build 7d4bcd8
ubuntu@kube-master: $ java --version
openjdk 11.0.23 2024-04-16
OpenJDK Runtime Environment (build 11.0.23+9-post-Ubuntu-1ubuntu1)
OpenJDK 64-Bit Server VM (build 11.0.23+9-post-Ubuntu-1ubuntu1, mixed mode, sharing)
ubuntu@kube-master: $ 
```

Jul 14 5:42 PM

ubuntu@jenkins-master: ~

```
ubuntu@kube-master:~$ docker -v
cri-dockerd --version
kubeadm version -o short
kubelet --version
kubectl version --client
Docker version 27.0.3, build 7d4bcd8
cri-dockerd 0.3.4 (e88b1605)
v1.29.6
Kubernetes v1.29.6
Client Version: v1.29.6
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
ubuntu@kube-master:~$
```

Jul 14 5:43 PM

ubuntu@jenkins-master: /home/ubuntu

ubuntu@kube-master: ~

```
ubuntu@kube-master:~$ sudo kubeadm init --cri-socket unix:///var/run/cri-dockerd.sock --ignore-preflight-errors=all
I0714 12:12:38.228049    8702 version.go:256] remote version is much newer: v1.30.2; falling back to: stable-1.29
[init] Using Kubernetes version: v1.29.6
[preflight] Running pre-flight checks
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your internet connection
[preflight] You can also perform this action in beforehand using 'kubeadm config images pull'
W0714 12:12:47.644069    8702 checks.go:835] detected that the sandbox image "registry.k8s.io/pause:3.6" of the container runtime is inconsistent with that used by kubeadm. It is recommended that using "registry.k8s.io/pause:3.9" as the CRI sandbox image.
[certs] Using certificateDir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [kube-master kubernetes kubernetes.default kubernetes.default.svc kubernetes.default.svc.cluster.local] and IPs [10.96.0.1 10.0.1.245]
[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 [kube-master localhost] and IPs [10.0.1.245 127.0.0.1 ::1]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [kube-master localhost] and IPs [10.0.1.245 127.0.0.1 ::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
```

```
JUL 14 5:44 PM ⓘ
ubuntu@kube-master:~
```

root@jenkins-master: /home/ubuntu

uba@kube-master: ~

```
entials
[bootstrap-token] Configured RBAC rules to allow the csrapprover controller automatically approve CSRs from a Node Bootstrap Token
[bootstrap-token] Configured RBAC rules to allow certificate rotation for all node client certificates in the cluster
[bootstrap-token] Creating the "cluster-info" ConfigMap in the "kube-public" namespace
[kubelet-finalize] Updating "/etc/kubernetes/kubelet.conf" to point to a rotatable kubelet client certificate and key
[addons] Applied essential addon: CoreDNS
[addons] Applied essential addon: kube-proxy

Your Kubernetes control-plane has initialized successfully!

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

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

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

export KUBECONFIG=/etc/kubernetes/admin.conf

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

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

kubeadm join 10.0.1.245:6443 --token txb6ub.kgchadjf0gxmxxbv \
--discovery-token-ca-cert-hash sha256:7c79268fe96051e1384a2180982d1804d491ab3af5e7756c13829c5295ae62b7
ubuntu@kube-master: $ ]
```

```
JUL 14 5:48 PM ⓘ
ubuntu@worker-node:~
```

root@jenkins-master: /home/ubuntu

uba@kube-master: ~

uba@worker-node: ~

```
ubuntu@worker-node: $ docker -v
cri-dockerd --version
kubeadm version -o short
kubelet --version
kubectl version --client
Docker version 27.0.3, build 7d4bcd8
cri-dockerd 0.3.4 (e88b1605)
v1.29.6
Kubernetes v1.29.6
Client Version: v1.29.6
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
ubuntu@worker-node: $ ]
```

Jul 14 5:48 PM

ubuntu@worker-node:~

```
root@jenkins-master:/home/ubuntu ×      ubuntu@kube-master:~ ×      ubuntu@worker-node:~ ×
ubuntu@worker-node:~$ sudo kubeadm join 10.0.1.245:6443 --token 80zettl.kk1q75qdr9uu3njf --discovery-token-ca-cert-hash sha256:7c79268fe96051e1384a2180982d1804d491ab3af5e7756c13829c5295ae62b7 --cri-socket unix:///var/run/cri-dockerd.sock
[preflight] Running pre-flight checks
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...

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

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

ubuntu@worker-node:~$
```

Jul 14 5:49 PM

ubuntu@kube-master:~

```
root@jenkins-master:/home/ubuntu ×      ubuntu@kube-master:~ ×      ubuntu@worker-node:~ ×
ubuntu@kube-master:~$ kubectl get nodes
NAME     STATUS   ROLES      AGE    VERSION
kube-master   Ready    control-plane   6m3s   v1.29.6
worker-node   Ready    <none>    18s    v1.29.6
ubuntu@kube-master:~$
```

Jul 14 5:50 PM

Instances | EC2 | us-east-1 | kubernetes/1-intall/install | Sign in [Jenkins]

YouTube xdhivyax (Dhivya... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

EC2 Dashboard Services Search [Alt+S] Connect Instance state Actions Launch instances

Instances (1/5) Info Find Instance by attribute or tag (case-sensitive) All states

running Clear filters

| Name | Instance ID | Instance state | Instance type | Status check |
|----------------------------|---------------------|----------------|---------------|-------------------|
| server created by dev u... | i-0b8a7e7d148726d7a | Running | t2.micro | 2/2 checks passed |
| Jenkins-master | i-0becabf2d0518a115 | Running | t2.small | 2/2 checks passed |

i-0becabf2d0518a115 (jenkins-master)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary Info

| Instance ID | Public IPv4 address | Private IPv4 addresses |
|--------------------------------------|------------------------------|--|
| i-0becabf2d0518a115 (jenkins-master) | 54.165.40.156 open address | 10.0.1.72 |
| IPv6 address | Instance state | Public IPv4 DNS |
| - | Running | ec2-54-165-40-156.compute-1.amazonaws.com open address |

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

Jul 14 5:51 PM

Not secure 54.165.40.156:8080/login?from=%2F

YouTube xdhivyax (Dhivya... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

```
/var/lib/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

Continue

Jul 14 5:52 PM

Start Course | Intellipaa... Instances | EC2 | us-east ... kubernetes/1-intall/inst ... labs/jenkins/install at m ... Setup Wizard [Jenkins] + - ×

YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Getting Started

Create First Admin User

Username
admin

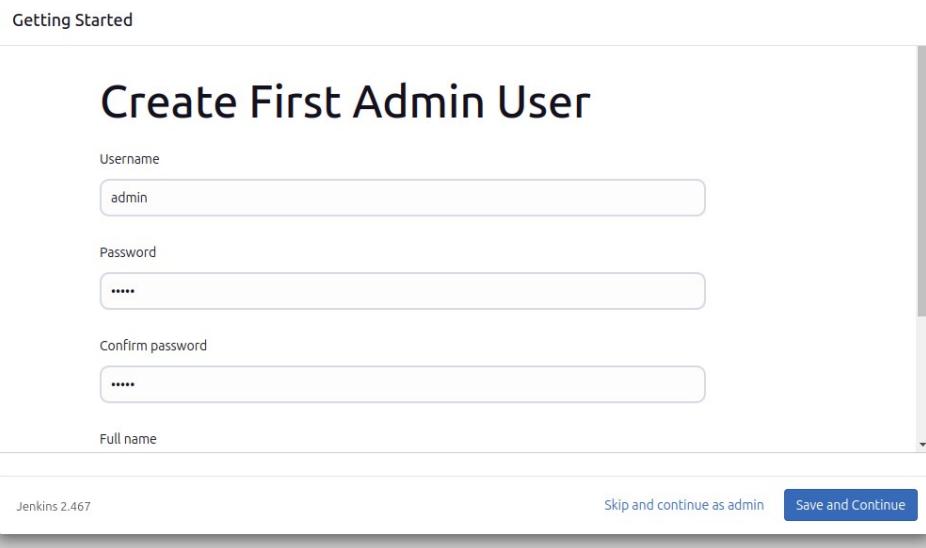
Password
.....

Confirm password
.....

Full name

Jenkins 2.467

Skip and continue as admin Save and Continue



Jul 14 5:53 PM

Start Course | Intellipaa... Instances | EC2 | us-east ... kubernetes/1-intall/inst ... labs/jenkins/install at m ... New node [Jenkins] + - ×

YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Jenkins

Dashboard > Manage Jenkins > Nodes > New node

New node

Node name
kube-master

Type
 Permanent Agent
Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

Create

REST API Jenkins 2.467

Jul 14 5:54 PM

Instances | EC2 | us-east-1 | kubernetes/1-intall/inst | labs/jenkins/install at m | Jenkins

YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

aws Services Search [Alt+S] Instance state Actions Launch instances

Instances (1/5) Info

| Name | Instance ID | Instance state | Instance type | Status check |
|--------------------------|---------------------|----------------|---------------|-------------------|
| Terraform-Master | i-043be26a1500e342b | Running | t2.micro | 2/2 checks passed |
| kubernetes-master | i-05bf35885563c145e | Running | t2.medium | 2/2 checks passed |
| worker-node | i-0ea34ca398c4d5c94 | Running | t2.micro | 2/2 checks passed |

i-05bf35885563c145e (kubernetes-master)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary

| | | |
|--|--|--|
| Instance ID i-05bf35885563c145e (kubernetes-master) | Public IPv4 address 3.94.180.191 open address | Private IPv4 addresses 10.0.1.245 |
| IPv6 address - | Instance state Running | Public IPv4 DNS ec2-3-94-180-191.compute-1.amazonaws.com open address |

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

CloudShell Feedback

Jul 14 6:06 PM

Not secure 54.165.40.156:8080/manage/computer/createItem

YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Dashboard > Manage Jenkins > Nodes >

Jenkins Credentials Provider: Jenkins

Add Credentials

Domain: Global credentials (unrestricted)

Kind: SSH Username with private key

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

ID: ssh-credentials

Description:

Save

Jul 14 6:07 PM

Start Course | Intellipaat | Instances | EC2 | us-east-1 | kubernetes/1-intall/install | labs/jenkins/install at master | Jenkins

YouTube xdhivyax (Dhivya... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Dashboard > Manage Jenkins > Nodes >

/home/ubuntu/jenkins-agent

Labels ?

Usage ?

Use this node as much as possible

Launch method ?

Launch agents via SSH

Host ?

ec2-3-94-180-191.compute-1.amazonaws.com

Credentials ?

ubuntu

+ Add ▾

Save

Jul 14 6:54 PM

Instances | EC2 | us-east-1 | labs/jenkins/install at master | Jenkins

YouTube xdhivyax (Dhivya... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Jenkins

Search (CTRL+K)

admin log out

Dashboard > Nodes >

Nodes Clouds

Build Queue No builds in the queue.

Build Executor Status 0/2

Built-In Node kube-master

Data obtained 32 sec 32 sec 32 sec 32 sec 32 sec 32 sec

Icon: S M L

+ New Node Configure Monitors

| S | Name | Architecture | Clock Difference | Free Disk Space | Free Swap Space | Free Temp Space | Response Time |
|---|---------------|---------------|------------------|-----------------|-----------------|-----------------|---------------|
| 💻 | Built-In Node | Linux (amd64) | In sync | 3.24 GiB | 0 B | 3.24 GiB | 0ms |
| 💻 | kube-master | Linux (amd64) | In sync | 1.28 GiB | 0 B | 1.28 GiB | 112ms |

Legend

Not secure 54.165.40.156:8080/view/all/newJob

YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Jenkins

Search (CTRL+K) admin log out

Dashboard > All > New Item

New Item

Enter an item name
PRT-pipeline

Select an item type

- Freestyle project**
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
- 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.

OK

Not secure 54.165.40.156:8080/pipeline/configure

YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Dashboard > PRT-pipeline > Configuration

Configure

General

Build Triggers

- Build after other projects are built ?
- Build periodically ?
- GitHub hook trigger for GITScm polling ?
- Poll SCM ?
- Quiet period ?
- Trigger builds remotely (e.g., from scripts) ?

Advanced Project Options

Advanced

Pipeline

Save Apply

Instances | EC2 | [Not secure](#) 54.165.40.156:8080/job/PRT-pipeline/configure Start Course | Intellipaat | xdhivyax/Website | Devops-Project-2/ | + YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Dashboard > PRT-pipeline > Configuration

Configure

Pipeline

Definition

General Advanced Project Options Pipeline

SCM

Git

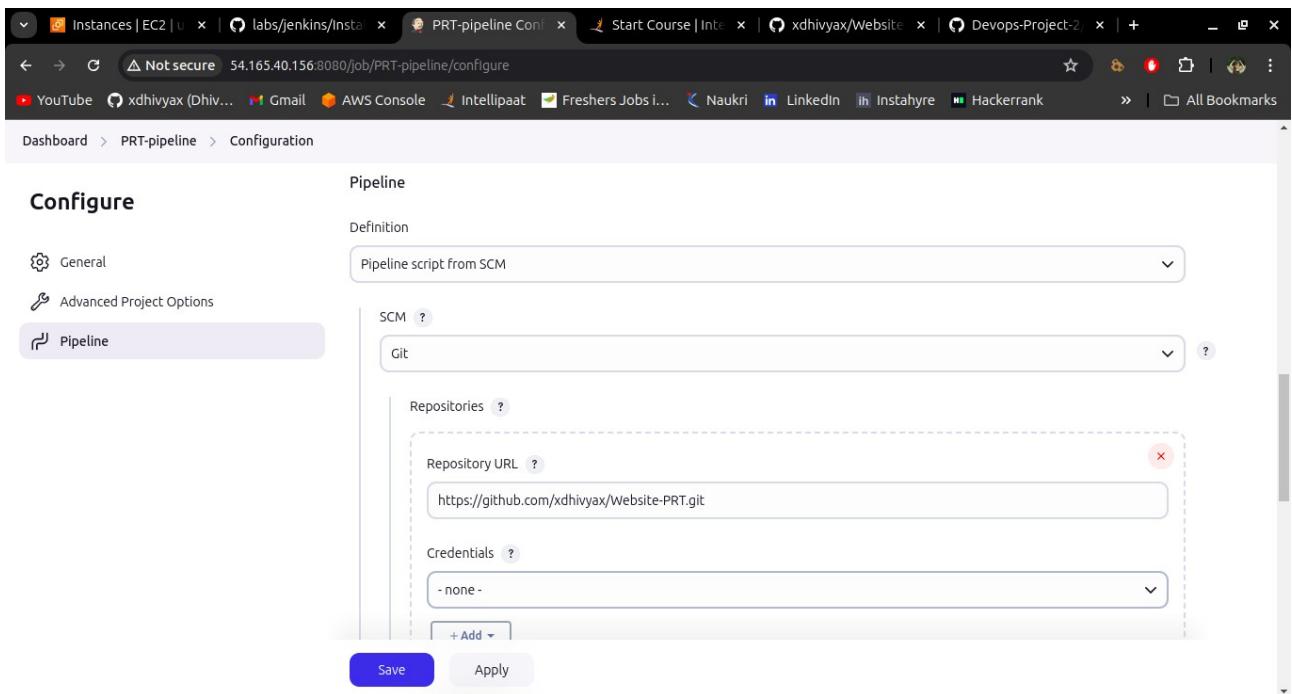
Repositories

Repository URL: https://github.com/xdhivyax/Website-PRT.git

Credentials: - none -

+ Add

Save Apply



Instances | EC2 | [Not secure](#) 54.165.40.156:8080/job/PRT-pipeline/configure Start Course | Intellipaat | xdhivyax/Website | Devops-Project-2/ | + YouTube xdhivyax (Dhiv... Gmail AWS Console Intellipaat Freshers Jobs i... Naukri LinkedIn Instahyre Hackerrank All Bookmarks

Dashboard > PRT-pipeline > Configuration

Configure

General Advanced Project Options Pipeline

Repository browser

(Auto)

Additional Behaviours

Add

Script Path

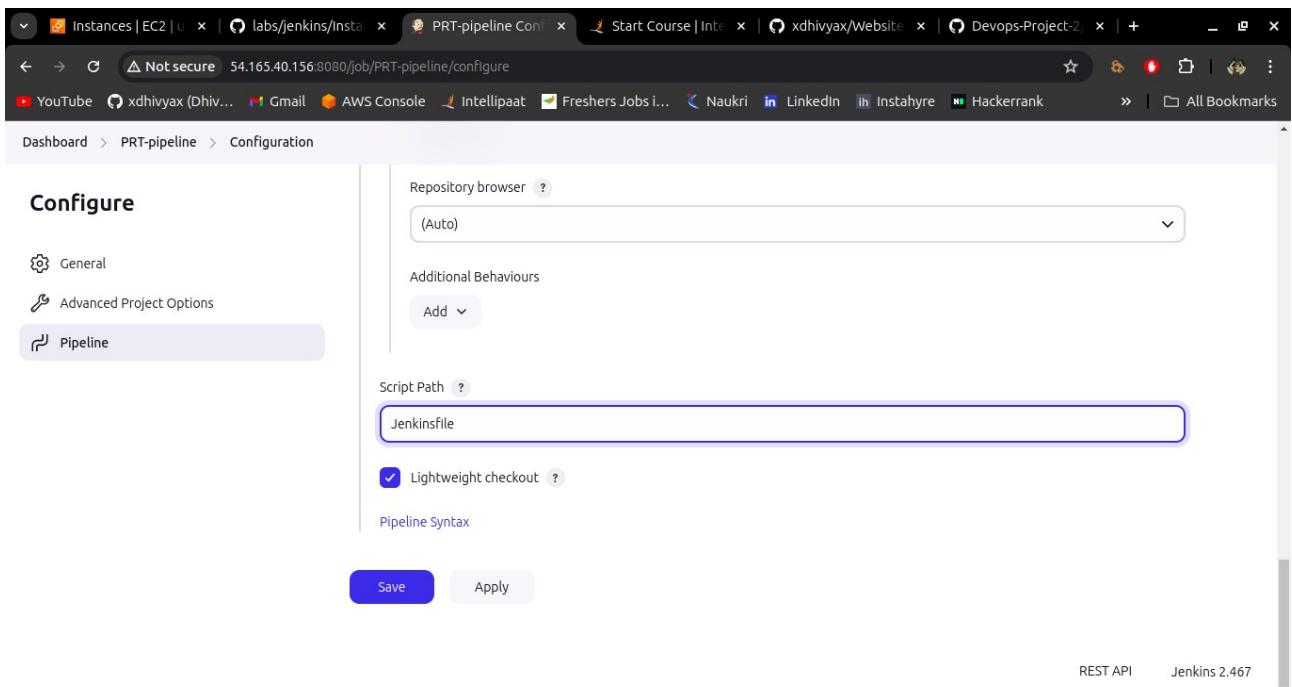
Jenkinsfile

Lightweight checkout

Pipeline Syntax

Save Apply

REST API Jenkins 2.467



DOCKERFILE

```
FROM ubuntu
RUN apt update
RUN apt install apache2 -y
ADD ./var/www/html
ENTRYPOINT apachectl -D FOREGROUND
```

JENKINSFILE

```
pipeline{
    agent any
    stages{
        stage("clone"){
            agent{
                label 'kube-master'
            }
            steps{
                git branch: 'main', url: 'https://github.com/xdhivya/Website-PRT.git'
            }
        }

        stage("docker-image") {
            agent{
                label 'kube-master'
            }
            steps{
                sh 'cd $WORSPACE'
                sh 'sudo docker build -f Dockerfile -t dhivya2409/prt .'
                withCredentials([string(credentialsId: 'DOCKER_HUB_PASSWORD', variable: 'DOCKER_HUB_PASSWORD')]) {
                    sh "sudo docker login -u dhivya2409 -p ${DOCKER_HUB_PASSWORD}"
                }
                sh 'sudo docker push dhivya2409/prt'
            }
        }

        stage("deploy"){
            agent{
                label 'kube-master'
            }
            steps{
                sh 'kubectl apply -f deploy.yaml'
                sh 'kubectl apply -f svc.yaml'
            }
        }
    }
}
```

deploy.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: deployment
  labels:
    app: project
spec:
  replicas: 5
  selector:
    matchLabels:
      app: project
  template:
    metadata:
      labels:
        app: project
    spec:
      containers:
        - name: myimage
          image: dhivya2409/prt
      ports:
        - containerPort: 80
```

svc.yaml

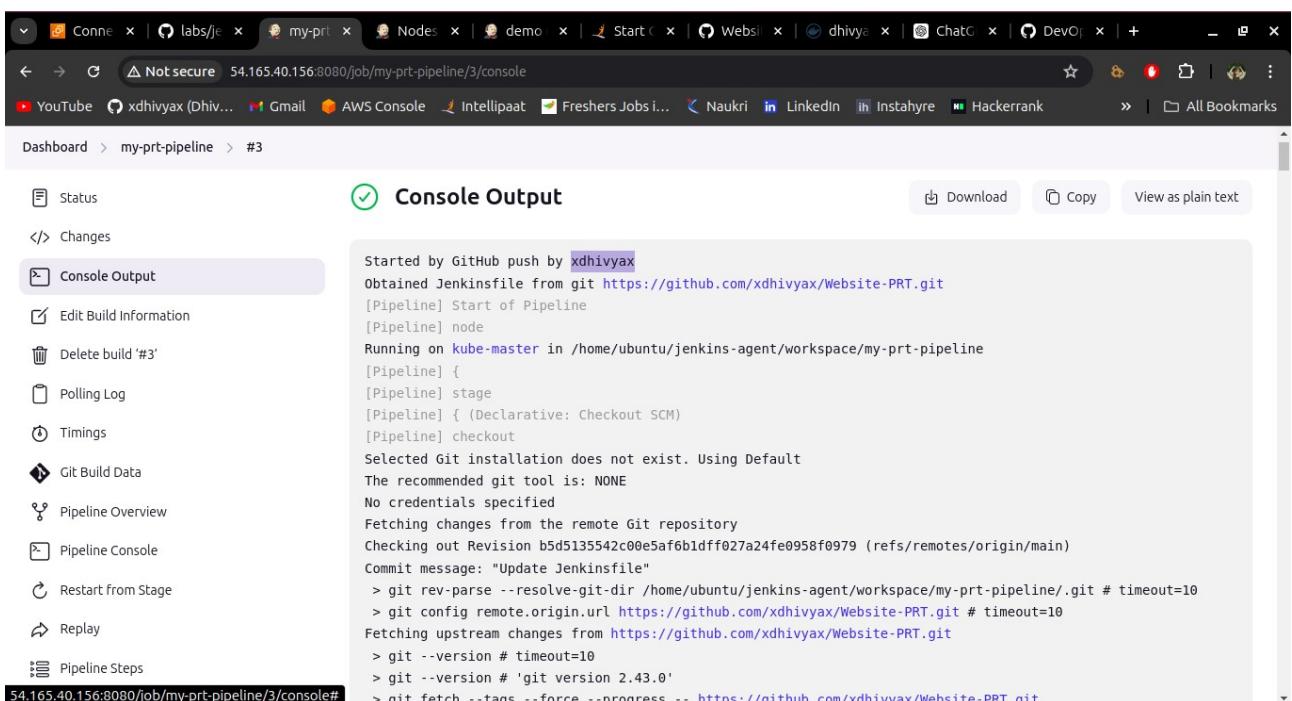
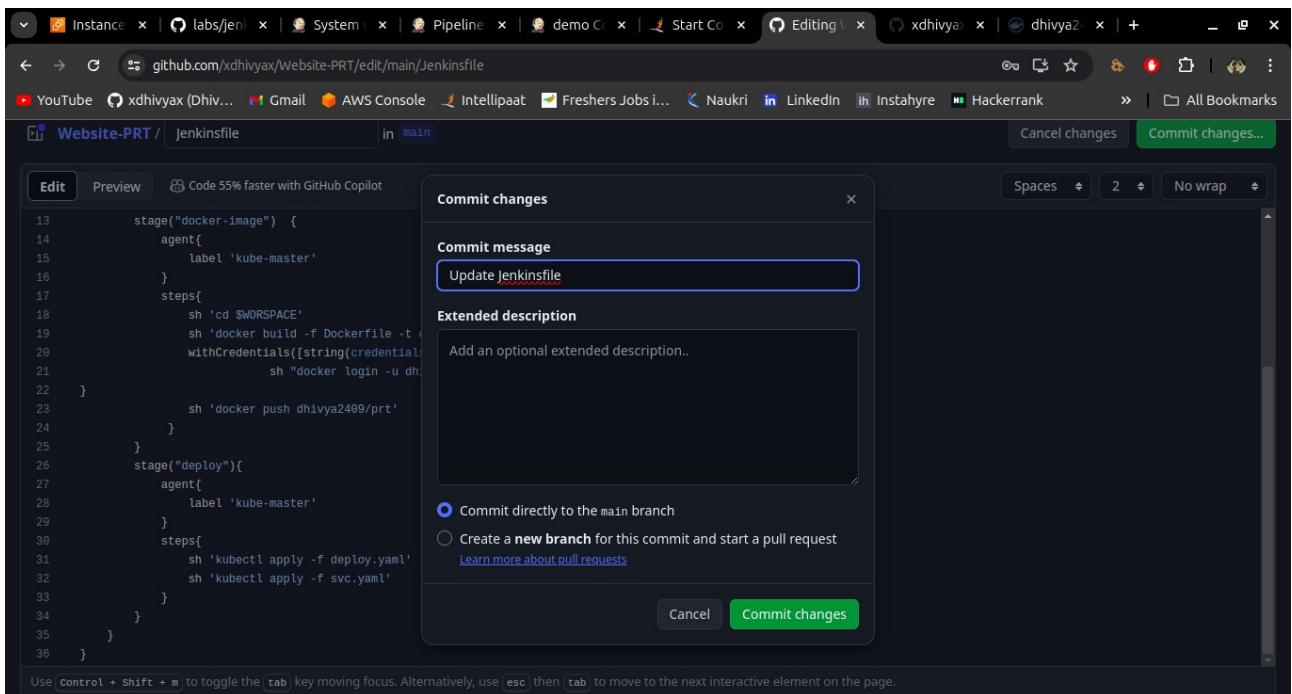
```
apiVersion: v1
kind: Service
metadata:
  name: service
spec:
  type: NodePort
  ports:
    - targetPort: 80
      port: 80
      nodePort: 30010
  selector:
    app: project
```

The screenshot shows the GitHub 'Add webhook' configuration page. The left sidebar is collapsed, and the main area displays the 'Webhooks / Add webhook' form. The 'Payload URL' field contains 'http://54.165.40.156:8080/github-webhook/'. The 'Content type' dropdown is set to 'application/json'. The 'SSL verification' section has 'Enable SSL verification' selected. The 'Secret' field is empty. The 'Which events would you like to trigger this webhook?' section has 'Just the push event.' selected. The 'Active' checkbox is checked, and a note below it states 'We will deliver event details when this hook is triggered.' A green 'Add webhook' button is at the bottom.

This screenshot shows the same GitHub 'Add webhook' page with more detailed configuration options visible. The left sidebar is expanded, showing sections like 'Code and automation', 'Security', and 'Integrations'. In the main configuration area, the 'Content type' dropdown is still set to 'application/json'. The 'SSL verification' section now includes a note: 'By default, we verify SSL certificates when delivering payloads.' The 'Which events would you like to trigger this webhook?' section includes additional options: 'Send me everything.' and 'Let me select individual events.' Below these, the 'Active' checkbox is checked, and the explanatory note is present. The 'Add webhook' button remains at the bottom.

The screenshot shows the GitHub 'Webhooks' settings page for a repository named 'Website-PRT'. The left sidebar has 'Webhooks' selected. The main area displays a single webhook entry with the URL `http://54.165.40.156:8080/github-... (push)`. A message indicates 'Last delivery was successful.' There are 'Edit' and 'Delete' buttons at the bottom of the entry.

The screenshot shows the Jenkins 'New credentials' configuration page. The 'Kind' dropdown is set to 'Secret text'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc.)'. The 'Secret' field contains a redacted password. The 'ID' field is set to 'DOCKER_HUB_PASSWORD'. The 'Description' field also contains 'DOCKER_HUB_PASSWORD'. A 'Create' button is visible at the bottom.



dhivya2409/prt ⓘ
Updated 2 minutes ago

This repository does not have a description ⓘ INCOMPLETE

This repository does not have a category ⓘ INCOMPLETE

Docker commands
To push a new tag to this repository:
docker push dhivya2409/prt:tagname

Tags
This repository contains 1 tag(s).

| Tag | OS | Type | Pulled | Pushed |
|--------|----|-------|---------------|---------------|
| latest | 🐧 | Image | 2 minutes ago | 2 minutes ago |

[See all](#)

Automated Builds
Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.
Available with Pro, Team and Business subscriptions. [Read more about automated builds](#).

[Upgrade](#)

```
ubuntu@kube-master: $ kubectl get deploy
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
deployment   5/5     5          5           2m36s
ubuntu@kube-master: $ kubectl get svc
NAME        TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)        AGE
kubernetes   ClusterIP  10.96.0.1   <none>        443/TCP       124m
service      NodePort   10.104.226.108 <none>        80:30010/TCP  2m43s
ubuntu@kube-master: $ kubectl get pods -o wide
NAME                           READY   STATUS    RESTARTS   AGE   IP           NODE   NOMINATED-NODE   READINESS   GATES
deployment-767c44f8c5-4s8rh   1/1    Running   0          4m39s   192.168.168.129   worker-node   <none>      <none>
deployment-767c44f8c5-5tcx7   1/1    Running   0          4m39s   192.168.168.131   worker-node   <none>      <none>
deployment-767c44f8c5-b2pjh   1/1    Running   0          4m39s   192.168.168.133   worker-node   <none>      <none>
deployment-767c44f8c5-jv9d6   1/1    Running   0          4m39s   192.168.168.130   worker-node   <none>      <none>
deployment-767c44f8c5-q2dx2   1/1    Running   0          4m39s   192.168.168.132   worker-node   <none>      <none>
ubuntu@kube-master: $
```

Screenshot of the AWS EC2 Instances page showing three running instances: Terraform-Master, kubernetes-master, and worker-node.

Instances (1/5) Info

| Name | Instance ID | Instance state | Instance Type | Status check |
|--------------------------|---------------------|----------------|---------------|-------------------|
| Terraform-Master | i-043be26a1500e342b | Running | t2.micro | 2/2 checks passed |
| kubernetes-master | i-05bf35885563c145e | Running | t2.medium | 2/2 checks passed |
| worker-node | i-0ea34ca398c4d5c94 | Running | t2.micro | 2/2 checks passed |

i-05bf35885563c145e (kubernetes-master)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

Instance summary

| | | |
|--|--|---|
| Instance ID I-05bf35885563c145e (kubernetes-master) | Public IPv4 address 5.94.180.191 [open address] | Private IPv4 addresses 10.0.1.245 |
| IPv6 address - | Instance state Running | Public IPv4 DNS ec2-3-94-180-191.compute-1.amazonaws.com |

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

Screenshot of a web browser showing the deployment of the Intellipaat webpage.

Not secure 3.94.180.191:30010

Successfully deployed the Intellipaat Webpage!!!



GITHUB LINK: <https://github.com/xdhivyax/Website-PRT.git>