

# DevOps Project

You are hired as a DevOps engineer for Analytics Pvt Ltd. This company is a product based organization which uses Docker for their containerization needs within the company. The final product received a lot of traction in the first few weeks of launch. Now with the increasing demand, the organization needs to have a platform for automating deployment, scaling, and operations of application containers across clusters of hosts. As a DevOps engineer, you need to implement a DevOps life cycle, such that all the requirements are implemented without any change in the Docker containers in the testing environment.

Up until now, this organization used to follow a monolithic architecture with just 2 developers. The product is present on

<https://github.com/hshar/website.git>

Following are the specifications of life-cycle:

1. Git workflow should be implemented. Since the company follows monolithic architecture of Development you need to take care of version control. The release should happen only on 25<sup>th</sup> of every month.
2. Code build should be triggered once the commits are made in the master Branch.
3. The code should be containerized with the help of the Docker file, The Dockerfile should be built every time if there is a push to Git-Hub. Create a custom Docker image using a Dockerfile.
4. As per the requirement in the production server, you need to use the Kubernetes cluster and the containerized code from Docker hub should be deployed with 2 replicas. Create a NodePort service and configure the same for port 30008.

5. Create a Jenkins pipeline script to accomplish the above task.
6. For configuration management of the infrastructure, you need to deploy the configuration on the servers to install necessary software and configurations.
7. Using Terraform accomplish the task of infrastructure creation in the AWS cloud provider.

## Architectural Advice

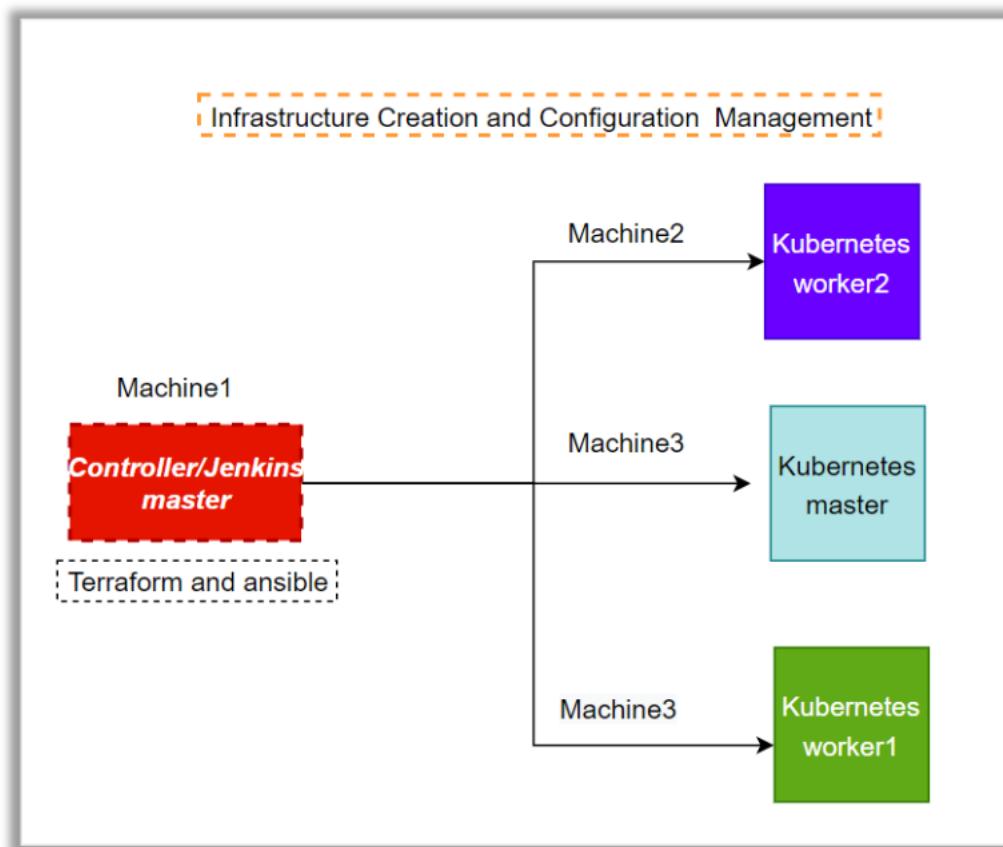
Software's to be installed on the respective machines using configuration management.

**Worker1:** Jenkins, Java.

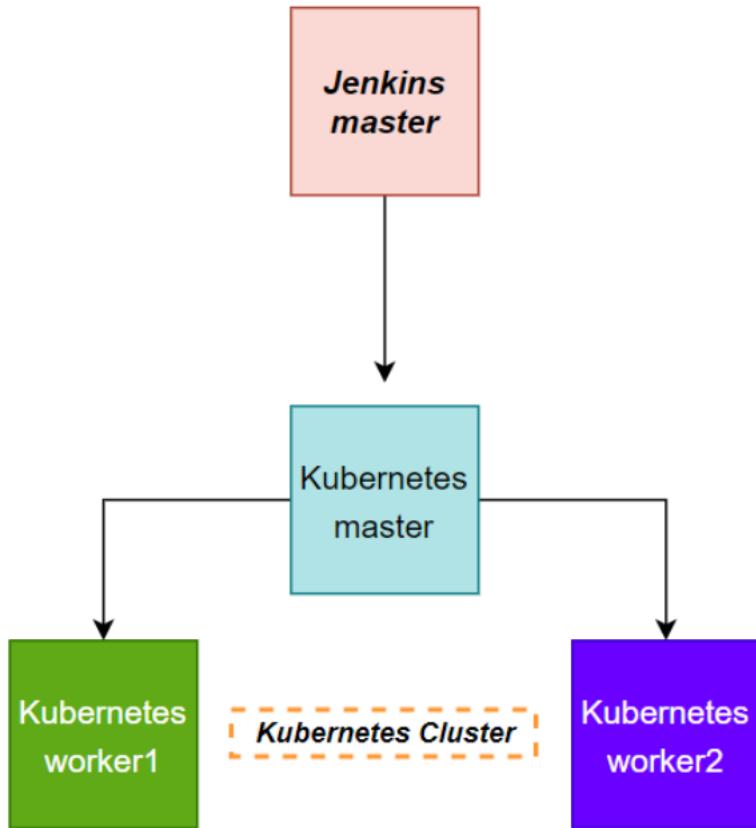
**Worker2:** Docker, Kubernetes.

**Worker3:** Java, Docker, Kubernetes

**Worker4:** Docker, Kubernetes.



### *Servers for jenkins and kubernetes configuration*



### **Task:1 Launched Instances:**

[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  
Hari-DevOps-Project

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

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

 *Search our full catalog including 1000s of application and OS images*

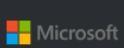
Recents

Quick Start

Amazon  
Linux  


macOS  


Ubuntu  


Windows  


Red Hat  


SUSE Li  




Browse more AMIs

Including AMIs from  
AWS, Marketplace and  
the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

ami-05e00961530ae1b55 (64-bit (x86)) / ami-072b1c33a2439c226 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

## ▼ Instance type [Info](#) | [Get advice](#)

Instance type

t2.medium

Family: t2 2 vCPU 4 GiB Memory Current generation: true

On-Demand Linux base pricing: 0.0496 USD per Hour

On-Demand Windows base pricing: 0.0676 USD per Hour

On-Demand RHEL base pricing: 0.1096 USD per Hour

On-Demand SUSE base pricing: 0.1496 USD per Hour

All generations

[Compare instance types](#)

[Additional costs apply for AMIs with pre-installed software](#)

## ▼ Key pair (login) [Info](#)

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

Key pair name - *required*

DevOps



 [Create new key pair](#)

▼ Network settings [Info](#)

[Edit](#)

Network | [Info](#)  
vpc-08bde8db1e7da9b9a | Default VPC

Subnet | [Info](#)  
No preference (Default subnet in any availability zone)

Auto-assign public IP | [Info](#)  
Enable  
Additional charges apply when outside of **free tier allowance**

Firewall (security groups) | [Info](#)  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group       Select existing security group

Common security groups [Info](#)

Select security groups ▾

default sg-04d9bdfee395aade5 X  
VPC: vpc-08bde8db1e7da9b9a

Compare security group rules

Security groups that you add or remove here will be added to or removed from all your network interfaces.

▼ Summary

Number of instances | [Info](#)  
1

Software Image (AMI)  
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)  
ami-05e00961530ae1b55

Virtual server type (instance type)  
t2.medium

Firewall (security group)  
default

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

[Cancel](#) [Launch instance](#)

Instances (1) <a href="#">Info</a>		<a href="#">C</a>	Connect	Instance state ▾	Actions ▾	<a href="#">Launch instances</a> ▾
<input type="text"/> Find Instance by attribute or tag (case-sensitive)				All states ▾	< 1 >	
	Name ▾	Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status
<input type="checkbox"/>	Hari-DevOps-Project	i-0aa471167f40b6000	Running	t2.medium	2/2 checks passed	<a href="#">View alarms</a> +

## Task: 2 Installed Terraform:

```
ubuntu@ip-172-31-0-232:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]

i-0aa471167f40b6000 (Hari-DevOps-Project)

Public IPs: 13.235.74.156 Private IPs: 172.31.0.232
```

The screenshot shows the HashiCorp Terraform website's 'Install Terraform' page for Linux. The left sidebar has a 'Linux' section selected under 'Operating Systems'. The main content area shows package manager instructions for Ubuntu/Debian, CentOS/RHEL, Fedora, Amazon Linux, and Homebrew. The Ubuntu/Debian tab is active, displaying the following command:

```
wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
```

```
wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
```

```
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
```

```
sudo apt update && sudo apt install terraform
```

```
ubuntu@ip-172-31-0-232:~$ wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
sudo apt update && sudo apt install terraform
--2024-05-12 03:47:28-- https://apt.releases.hashicorp.com/gpg
Resolving apt.releases.hashicorp.com (apt.releases.hashicorp.com)... 18.172.78.129, 18.172.78.12, 18.172.78.30, ...
Connecting to apt.releases.hashicorp.com (apt.releases.hashicorp.com)|18.172.78.129|:443... connected.
i-0aa471167f40b6000 (Hari-DevOps-Project)
Public IPs: 13.235.74.156 Private IPs: 172.31.0.232
```

```
ubuntu@ip-172-31-0-232:~$ terraform --version
Terraform v1.8.3
on linux_amd64
ubuntu@ip-172-31-0-232:~$ 
```

i-0aa471167f40b6000 (Hari-DevOps-Project)

Public IPs: 13.235.74.156 Private IPs: 172.31.0.232

```
provider "aws" {
  region = "ap-south-1"
  access_key = "AKIARPFRXAWSGACDTUNL"
  secret_key = "sxRmeNNRZdhMey2yu0J0zYY+O4ZKtChNXXXXE3Ef"
}

resource "aws_instance" "K8s-slave2" {
  ami      = "ami-05e00961530ae1b55"
  instance_type = "t2.medium"
  key_name = "DevOps"
  tags = {
    Name = "m2-slave"
  }
}

resource "aws_instance" "K8s-master" {
  ami      = "ami-05e00961530ae1b55"
  instance_type = "t2.medium"
  key_name = "DevOps"
  tags = {
    Name = "m3-master"
  }
}

resource "aws_instance" "K8s-Node2" {
  ami      = "ami-05e00961530ae1b55"
  instance_type = "t2.medium"
  key_name = "DevOps"
  tags = {
    Name = "m4-slave"
  }
}
```

```
ubuntu@ip-172-31-0-232:~$ sudo nano main.tf
ubuntu@ip-172-31-0-232:~$ terraform init
```

**Initializing the backend...**

**Initializing provider plugins...**

- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.49.0...

**i-0aa471167f40b6000 (Hari-DevOps-Project)**

Public IPs: 13.232.38.197 Private IPs: 172.31.0.232

```
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@ip-172-31-22-198:~$ █
```

**i-0bc624895711ef030 (Sagar-capstone)**

Public IPs: 3.16.143.221 Private IPs: 172.31.22.198

```
ubuntu@ip-172-31-0-232:~$ terraform plan

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

Terraform will perform the following actions:

# aws_instance.K8s-Node2 will be created
+ resource "aws_instance" "K8s-Node2" {
    + ami                               = "ami-05e00961530aelb55"
    + arn                               = (known after apply)
    + associate_public_ip_address       = (known after apply)
    + availability_zone                 = (known after apply)
```

**i-0aa471167f40b6000 (Hari-DevOps-Project)**

Public IPs: 13.232.38.197 Private IPs: 172.31.0.232

```

+ subnet_id                      = (Known after apply)
+ tags                           = {
    + "Name" = "m2-slave"
}
+ tags_all                       = {
    + "Name" = "m2-slave"
}
+ tenancy                        = (known after apply)
+ user_data                      = (known after apply)
+ user_data_base64               = (known after apply)
+ user_data_replace_on_change   = false
+ vpc_security_group_ids         = (known after apply)
}

Plan: 3 to add, 0 to change, 0 to destroy.

```

Note: You didn't use the `-out` option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.  
ubuntu@ip-172-31-0-232:~\$

i-0aa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

```

ubuntu@ip-172-31-0-232:~$ 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.K8s-Node2 will be created
+ resource "aws_instance" "K8s-Node2" {
    + ami                         = "ami-05e00961530aelb55"
    + 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)
}

i-0aa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

```

```

aws_instance.K8s-Node2: Creating...
aws_instance.K8s-slave2: Creating...
aws_instance.K8s-master: Creating...
aws_instance.K8s-slave2: Still creating... [10s elapsed]
aws_instance.K8s-Node2: Still creating... [10s elapsed]
aws_instance.K8s-master: Still creating... [10s elapsed]
aws_instance.K8s-Node2: Still creating... [20s elapsed]
aws_instance.K8s-slave2: Still creating... [20s elapsed]
aws_instance.K8s-master: Still creating... [20s elapsed]
aws_instance.K8s-slave2: Still creating... [30s elapsed]
aws_instance.K8s-Node2: Still creating... [30s elapsed]
aws_instance.K8s-master: Still creating... [30s elapsed]
aws_instance.K8s-slave2: Creation complete after 32s [id=i-049ebcf9c2c3ad094]
aws_instance.K8s-Node2: Still creating... [40s elapsed]
aws_instance.K8s-master: Still creating... [40s elapsed]
aws_instance.K8s-Node2: Creation complete after 42s [id=i-0941fbcf34abab51a]
aws_instance.K8s-master: Creation complete after 42s [id=i-0a8a43e5bf1861eac]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-0-232:~$ 

```

i-0aa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

Instances (4) <a href="#">Info</a>		<a href="#">C</a>	Connect	Instance state ▾	Actions ▾	Launch instances	⋮
<input type="text"/> Find Instance by attribute or tag (case-sensitive)				All states ▾			⋮
Instance state = running <a href="#">X</a>		<a href="#">Clear filters</a>		< 1 > ⚙			
□	Name ↗	Instance ID	Instance state	Instance type	Status check	Alarm status	⋮
□	Hari-DevOps-Project	i-0aa471167f40b6000	🕒 Running <a href="#">Q</a> <a href="#">Q</a>	t2.medium	🕒 2/2 checks passed	<a href="#">View alarms</a>	⋮
□	m2-slave	i-049ebcf9c2c3ad094	🕒 Running <a href="#">Q</a> <a href="#">Q</a>	t2.medium	🕒 2/2 checks passed	<a href="#">View alarms</a>	⋮
□	m3-master	i-0a8a43e5bf1861eac	🕒 Running <a href="#">Q</a> <a href="#">Q</a>	t2.medium	🕒 2/2 checks passed	<a href="#">View alarms</a>	⋮
□	m4-slave	i-0941fbef34abab51a	🕒 Running <a href="#">Q</a> <a href="#">Q</a>	t2.medium	🕒 2/2 checks passed	<a href="#">View alarms</a>	⋮

```
ubuntu@ip-172-31-3-13:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
```

**i-049ebcf9c2c3ad094 (m2-slave)**

Public IPs: 43.204.141.55 Private IPs: 172.31.3.13

```
ubuntu@ip-172-31-3-129:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
```

**i-0a8a43e5bf1861eac (m3-master)**

Public IPs: 65.0.122.157 Private IPs: 172.31.3.129

```
ubuntu@ip-172-31-0-49:~$ sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
```

**i-0941fbef34abab51a (m4-slave)**

Public IPs: 13.127.159.114 Private IPs: 172.31.0.49

## Installing Ansible on Ubuntu

Ubuntu builds are available [in a PPA here](#).

To configure the PPA on your system and install Ansible run these commands:

```
$ sudo apt update
$ sudo apt install software-properties-common
$ sudo add-apt-repository --yes --update ppa:ansible/ansible
$ sudo apt install ansible
```

```
$ sudo apt update
```

```
$ sudo apt install software-properties-common
```

```
$ sudo add-apt-repository --yes --update ppa:ansible/ansible
```

```
$ sudo apt install ansible
```

```
ubuntu@ip-172-31-0-232:~$ ansible --version
ansible [core 2.16.6]
  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 = 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True
ubuntu@ip-172-31-0-232:~$ []
i-0aa471167f40b6000 (Hari-DevOps-Project)
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

```
ubuntu@ip-172-31-0-232:~$ cd .ssh
ubuntu@ip-172-31-0-232:~/ssh$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_rsa
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub
```

```
i-0aa471167f40b6000 (Hari-DevOps-Project)
```

```
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

```
Your identification has been saved in /home/ubuntu/.ssh/id_rsa
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:4JAjhnhooZZVdVYtvLZEiyuIMZIvH7n2uvVj8ArMbhY ubuntu@ip-172-31-0-232
The key's randomart image is:
+---[RSA 3072]---+
| .. .... oo.. |
| oo+ . o + . |
| +=+ + . o + |
| o= + + . . = |
|   o = o S + . |
|   .oE o . . . |
|   o+++.o . |
|   .*o .+ |
|   +++oo.. |
+---[SHA256]---+
ubuntu@ip-172-31-0-232:~/ssh$ ls
authorized_keys  id_rsa  id_rsa.pub
ubuntu@ip-172-31-0-232:~/ssh$ []
```

```
i-0aa471167f40b6000 (Hari-DevOps-Project)
```

```
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

```
ubuntu@ip-172-31-0-232:~/.ssh$ ls
authorized_keys  id_rsa  id_rsa.pub
ubuntu@ip-172-31-0-232:~/.ssh$ sudo cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCoxdzIURdm/fttm0yEKi6s8MtAAjMEcOqt3fhDulTJuDZQ4UwzI2qgfbc1YXu634vh81hKBDxo+b3CcKvEQZ8a+pVvqLIZG2Vr
9tWELHL/o1KnV+OBBr9Lw//q4S1GXhr42!+o7RqHxIEX2dzUj1i2FKgDEUsq1SrZdrK5HAs9Es1MjOUVisAGwPErzUeNDpAzM6+kf991Rx7N5GpDtS3FcUK6cV+i15f8cf1MsP
4DBeqXyNQfpFN+7fYTcbxpMS62ftjh0FtYig/qNnEeYPPP1CZqmwjswdW18336Mh+cUD1TCUkbychibF860WpCAitdjZPZKG0T7e0m+G2z DevOps
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCoxdzIURdm/fttm0yEKi6s8MtAAjMEcOqt3fhDulTJuDZQ4UwzI2qgfbc1YXu634vh81hKBDxo+b3CcKvEQZ8a+pVvqLIZG2Vr
9tWELHL/o1KnV+OBBr9Lw//q4S1GXhr42!+o7RqHxIEX2dzUj1i2FKgDEUsq1SrZdrK5HAs9Es1MjOUVisAGwPErzUeNDpAzM6+kf991Rx7N5GpDtS3FcUK6cV+i15f8cf1MsP
4DBeqXyNQfpFN+7fYTcbxpMS62ftjh0FtYig/qNnEeYPPP1CZqmwjswdW18336Mh+cUD1TCUkbychibF860WpCAitdjZPZKG0T7e0m+G2z DevOps
qsaPzx6WKOaPoLWJ1eriw1JHuSV1jeSJ/WUKFQvn4l95fBn7JqM1pRRuBaRF6PmTqZ062Woz614ioBnzBt15Pf3cxnPTQKTbOoTgGFbkrDKHkmzU6SpIF38BuR8g+x9yp9MC5P
FwoPUalesAt7iteHCQjJEZ2MrY5h2m0Iq6/XePtBBM/t7qhWUWddBCJFD+Fq+ZtsPutyCbx3qUYD9jx2V1XJAEfufGjuw6rBq7qfQ6WkrFVKbf9wIc4CJfga6XIfdB1gB6CSYG
2vQn2RfpG2Fj1rUmw9U1MnvjnoFj0zNLVOEflbrfON+T7Lxen1P9imJNMUWTaFkcvMhrD0zz54t85m44xyKxYx8DE6ybwfEQV3bhzzSiANISTwhlgY1J3QIRNhf9z
wI+WlbwJeLNtkMO= ubuntu@ip-172-31-0-232:~/.ssh$ 
```

i-0aa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

```
ubuntu@ip-172-31-3-13:~$ cd .ssh
ubuntu@ip-172-31-3-13:~/.ssh$ ls
authorized_keys
ubuntu@ip-172-31-3-13:~/.ssh$ sudo nano authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCoxdzIURdm/fttm0yEKi6s8MtAAjMEcOqt3fhDulTJuDZQ4UwzI2qgfbc1YXu634vh81hKBDxo+b3CcKvEQZ8a+pVvqLIZG2Vr
9tWELHL/o1KnV+OBBr9Lw//q4S1GXhr42!+o7RqHxIEX2dzUj1i2FKgDEUsq1SrZdrK5HAs9Es1MjOUVisAGwPErzUeNDpAzM6+kf991Rx7N5GpDtS3FcUK6cV+i15f8cf1MsP
4DBeqXyNQfpFN+7fYTcbxpMS62ftjh0FtYig/qNnEeYPPP1CZqmwjswdW18336Mh+cUD1TCUkbychibF860WpCAitdjZPZKG0T7e0m+G2z DevOps
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCoxdzIURdm/fttm0yEKi6s8MtAAjMEcOqt3fhDulTJuDZQ4UwzI2qgfbc1YXu634vh81hKBDxo+b3CcKvEQZ8a+pVvqLIZG2Vr
9tWELHL/o1KnV+OBBr9Lw//q4S1GXhr42!+o7RqHxIEX2dzUj1i2FKgDEUsq1SrZdrK5HAs9Es1MjOUVisAGwPErzUeNDpAzM6+kf991Rx7N5GpDtS3FcUK6cV+i15f8cf1MsP
4DBeqXyNQfpFN+7fYTcbxpMS62ftjh0FtYig/qNnEeYPPP1CZqmwjswdW18336Mh+cUD1TCUkbychibF860WpCAitdjZPZKG0T7e0m+G2z DevOps
qsaPzx6WKOaPoLWJ1eriw1JHuSV1jeSJ/WUKFQvn4l95fBn7JqM1pRRuBaRF6PmTqZ062Woz614ioBnzBt15Pf3cxnPTQKTbOoTgGFbkrDKHkmzU6SpIF38BuR8g+x9yp9MC5P
FwoPUalesAt7iteHCQjJEZ2MrY5h2m0Iq6/XePtBBM/t7qhWUWddBCJFD+Fq+ZtsPutyCbx3qUYD9jx2V1XJAEfufGjuw6rBq7qfQ6WkrFVKbf9wIc4CJfga6XIfdB1gB6CSYG
2vQn2RfpG2Fj1rUmw9U1MnvjnoFj0zNLVOEflbrfON+T7Lxen1P9imJNMUWTaFkcvMhrD0zz54t85m44xyKxYx8DE6ybwfEQV3bhzzSiANISTwhlgY1J3QIRNhf9z
eLNtkMO= ubuntu@ip-172-31-3-13:~/.ssh$ 
```

i-049ebcf9c2c3ad094 (m2-slave)

PublicIPs: 43.204.141.55 PrivateIPs: 172.31.3.13

```
ubuntu@ip-172-31-3-129:~$ cd .ssh
ubuntu@ip-172-31-3-129:~/.ssh$ ls
authorized_keys
ubuntu@ip-172-31-3-129:~/.ssh$ sudo nano authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCoxdzIURdm/fttm0yEKi6s8MtAAjMEcOqt3fhDulTJuDZQ4UwzI2qgfbc1YXu634vh81hKBDxo+b3CcKvEQZ8a+pVvqLIZG2Vr
9tWELHL/o1KnV+OBBr9Lw//q4S1GXhr42!+o7RqHxIEX2dzUj1i2FKgDEUsq1SrZdrK5HAs9Es1MjOUVisAGwPErzUeNDpAzM6+kf991Rx7N5GpDtS3FcUK6cV+i15f8cf1MsP
4DBeqXyNQfpFN+7fYTcbxpMS62ftjh0FtYig/qNnEeYPPP1CZqmwjswdW18336Mh+cUD1TCUkbychibF860WpCAitdjZPZKG0T7e0m+G2z DevOps
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCoxdzIURdm/fttm0yEKi6s8MtAAjMEcOqt3fhDulTJuDZQ4UwzI2qgfbc1YXu634vh81hKBDxo+b3CcKvEQZ8a+pVvqLIZG2Vr
9tWELHL/o1KnV+OBBr9Lw//q4S1GXhr42!+o7RqHxIEX2dzUj1i2FKgDEUsq1SrZdrK5HAs9Es1MjOUVisAGwPErzUeNDpAzM6+kf991Rx7N5GpDtS3FcUK6cV+i15f8cf1MsP
4DBeqXyNQfpFN+7fYTcbxpMS62ftjh0FtYig/qNnEeYPPP1CZqmwjswdW18336Mh+cUD1TCUkbychibF860WpCAitdjZPZKG0T7e0m+G2z DevOps
qsaPzx6WKOaPoLWJ1eriw1JHuSV1jeSJ/WUKFQvn4l95fBn7JqM1pRRuBaRF6PmTqZ062Woz614ioBnzBt15Pf3cxnPTQKTbOoTgGFbkrDKHkmzU6SpIF38BuR8g+x9yp9MC5P
FwoPUalesAt7iteHCQjJEZ2MrY5h2m0Iq6/XePtBBM/t7qhWUWddBCJFD+Fq+ZtsPutyCbx3qUYD9jx2V1XJAEfufGjuw6rBq7qfQ6WkrFVKbf9wIc4CJfga6XIfdB1gB6CSYG
2vQn2RfpG2Fj1rUmw9U1MnvjnoFj0zNLVOEflbrfON+T7Lxen1P9imJNMUWTaFkcvMhrD0zz54t85m44xyKxYx8DE6ybwfEQV3bhzzSiANISTwhlgY1J3QIRNhf9z
eLNtkMO= ubuntu@ip-172-31-3-129:~/.ssh$ 
```

i-0a8a43e5bf1861ec (m3-master)

PublicIPs: 65.0.122.157 PrivateIPs: 172.31.3.129

```
ubuntu@ip-172-31-0-49:~$ cd .ssh
ubuntu@ip-172-31-0-49:~/.ssh$ ls
authorized_keys
ubuntu@ip-172-31-0-49:~/.ssh$ sudo nano authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCoxdzIURdm/fttm0yEKi6s8MtAAjMEcOqt3fhDulTJuDZQ4UwzI2qgfbc1YXu634vh81hKBDxo+b3CcKvEQZ8a+pVvqLIZG2Vr
9tWELHL/o1KnV+OBBr9Lw//q4S1GXhr42!+o7RqHxIEX2dzUj1i2FKgDEUsq1SrZdrK5HAs9Es1MjOUVisAGwPErzUeNDpAzM6+kf991Rx7N5GpDtS3FcUK6cV+i15f8cf1MsP
4DBeqXyNQfpFN+7fYTcbxpMS62ftjh0FtYig/qNnEeYPPP1CZqmwjswdW18336Mh+cUD1TCUkbychibF860WpCAitdjZPZKG0T7e0m+G2z DevOps
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCoxdzIURdm/fttm0yEKi6s8MtAAjMEcOqt3fhDulTJuDZQ4UwzI2qgfbc1YXu634vh81hKBDxo+b3CcKvEQZ8a+pVvqLIZG2Vr
9tWELHL/o1KnV+OBBr9Lw//q4S1GXhr42!+o7RqHxIEX2dzUj1i2FKgDEUsq1SrZdrK5HAs9Es1MjOUVisAGwPErzUeNDpAzM6+kf991Rx7N5GpDtS3FcUK6cV+i15f8cf1MsP
4DBeqXyNQfpFN+7fYTcbxpMS62ftjh0FtYig/qNnEeYPPP1CZqmwjswdW18336Mh+cUD1TCUkbychibF860WpCAitdjZPZKG0T7e0m+G2z DevOps
qsaPzx6WKOaPoLWJ1eriw1JHuSV1jeSJ/WUKFQvn4l95fBn7JqM1pRRuBaRF6PmTqZ062Woz614ioBnzBt15Pf3cxnPTQKTbOoTgGFbkrDKHkmzU6SpIF38BuR8g+x9yp9MC5P
FwoPUalesAt7iteHCQjJEZ2MrY5h2m0Iq6/XePtBBM/t7qhWUWddBCJFD+Fq+ZtsPutyCbx3qUYD9jx2V1XJAEfufGjuw6rBq7qfQ6WkrFVKbf9wIc4CJfga6XIfdB1gB6CSYG
2vQn2RfpG2Fj1rUmw9U1MnvjnoFj0zNLVOEflbrfON+T7Lxen1P9imJNMUWTaFkcvMhrD0zz54t85m44xyKxYx8DE6ybwfEQV3bhzzSiANISTwhlgY1J3QIRNhf9z
eLNtkMO= ubuntu@ip-172-31-0-49:~/.ssh$ 
```

i-0941fbcf34abab51a (m4-slave)

PublicIPs: 13.127.159.114 PrivateIPs: 172.31.0.49

```
ubuntu@ip-172-31-0-232:~/ssh$ cd
ubuntu@ip-172-31-0-232:~$ cd /etc/ansible
ubuntu@ip-172-31-0-232:/etc/ansible$ ls
ansible.cfg hosts roles
ubuntu@ip-172-31-0-232:/etc/ansible$ cd hosts
-bash: cd: hosts: Not a directory
ubuntu@ip-172-31-0-232:/etc/ansible$ cd /hosts
-bash: cd: /hosts: No such file or directory
ubuntu@ip-172-31-0-232:/etc/ansible$ sudo nano hosts
ubuntu@ip-172-31-0-232:/etc/ansible$ cat hosts
[master]
172.31.3.129
[slaves]
172.31.3.13
172.31.0.49
ubuntu@ip-172-31-0-232:/etc/ansible$ 
```

i-0aa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

```
GNU nano 0.4
[master]
172.31.3.129
[slaves]
172.31.3.13
172.31.0.49
```

^G Help ^O Write Out ^W Where Is ^I  
^X Exit ^R Read File ^\ Replace ^T

i-0aa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

```
ubuntu@ip-172-31-0-232:~$ cd /etc/ansible
ubuntu@ip-172-31-0-232:/etc/ansible$ ansible -m ping all
The authenticity of host '172.31.0.49 (172.31.0.49)' can't be established.
ED25519 key fingerprint is SHA256:mWGUwCCu9FVwIQJFxj3RzKhijFSjyDcQYHAdMezykVU.
This key is not known by any other names
The authenticity of host '172.31.3.129 (172.31.3.129)' can't be established.
ED25519 key fingerprint is SHA256:pZG68DivoC4N0iG6HhSlcR5rljHWC2qpd16iRP7fXAQ.
This key is not known by any other names
The authenticity of host '172.31.3.13 (172.31.3.13)' can't be established.
ED25519 key fingerprint is SHA256:4Iku9BQ24qrEcSaIxOug8VwU3EEAMD9C9eAkaPIsbk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
172.31.0.49 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
}
```

i-Oaa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

```
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
172.31.0.49 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
yes
172.31.3.129 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
yes
172.31.3.13 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ubuntu@ip-172-31-0-232:/etc/ansible$ 
```

i-Oaa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

```
ubuntu@ip-172-31-0-232:~$ cd /etc/ansible  
ubuntu@ip-172-31-0-232:/etc/ansible$ sudo nano script1.sh  
ubuntu@ip-172-31-0-232:/etc/ansible$ sudo nano script2.sh  
ubuntu@ip-172-31-0-232:/etc/ansible$ sudo nano script3.sh  
ubuntu@ip-172-31-0-232:/etc/ansible$ 
```

## i-0aa471167f40b6000 (Hari-DevOps-Project)

PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232

### script1.sh

```
sudo apt update  
sudo apt install openjdk-11-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 install jenkins -y  
sudo apt install docker.io -y
```

### script2.sh

```
sudo apt update  
sudo apt install docker.io -y  
sudo apt install openjdk-11-jdk -y  
sudo swapoff -a
```

```
# Create the .conf file to load the modules at bootup  
cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf  
overlay  
br_netfilter  
EOF
```

```
sudo modprobe overlay
```

```
sudo modprobe br_netfilter

# sysctl params required by setup, params persist across reboots
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-iptables = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.ipv4.ip_forward = 1
EOF

# Apply sysctl params without reboot
sudo sysctl --system

## Install CRI-O Runtime
sudo apt-get update -y
sudo apt-get install -y software-properties-common curl apt-transport-https ca-certificates gpg

sudo curl -fsSL https://pkgs.k8s.io/addons:/cri-o:/prerelease:/main/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/cri-o-apt-keyring.gpg
echo "deb [signed-by=/etc/apt/keyrings/cri-o-apt-keyring.gpg] https://pkgs.k8s.io/addons:/cri-o:/prerelease:/main/deb /" | sudo tee /etc/apt/sources.list.d/cri-o.list

sudo apt-get update -y
sudo apt-get install -y cri-o

sudo systemctl daemon-reload
sudo systemctl enable crio --now
sudo systemctl start crio.service

echo "CRI runtime installed successfully"

# Add Kubernetes APT repository and install required packages
```

```
curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.29/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg  
echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg]  
https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /' | sudo tee /etc/apt/sources.list.d/kubernetes.list
```

```
sudo apt-get update -y  
sudo apt-get install -y kubelet="1.29.0-*" kubectl="1.29.0-*" kubeadm="1.29.0-*"  
sudo apt-get update -y  
sudo apt-get install -y jq
```

```
sudo systemctl enable --now kubelet  
sudo systemctl start kubelet
```

### **script3.sh**

```
sudo apt update  
sudo apt install docker.io -y  
sudo swapoff -a  
  
# Create the .conf file to load the modules at bootup  
cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf  
overlay  
br_netfilter  
EOF
```

```
sudo modprobe overlay  
sudo modprobe br_netfilter
```

```
# sysctl params required by setup, params persist across reboots  
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf  
net.bridge.bridge-nf-call-iptables = 1  
net.bridge.bridge-nf-call-ip6tables = 1
```

```
net.ipv4.ip_forward      = 1
EOF

# Apply sysctl params without reboot
sudo sysctl --system

## Install CRIOS Runtime
sudo apt-get update -y
sudo apt-get install -y software-properties-common curl apt-transport-https ca-certificates gpg

sudo curl -fsSL https://pkgs.k8s.io/addons:/cri-o:/prerelease:/main/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/cri-o-apt-keyring.gpg
echo "deb [signed-by=/etc/apt/keyrings/cri-o-apt-keyring.gpg] https://pkgs.k8s.io/addons:/cri-o:/prerelease:/main/deb/ /" | sudo tee /etc/apt/sources.list.d/cri-o.list

sudo apt-get update -y
sudo apt-get install -y cri-o

sudo systemctl daemon-reload
sudo systemctl enable crio --now
sudo systemctl start crio.service

echo "CRI runtime installed successfully"

# Add Kubernetes APT repository and install required packages
curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.29/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg
echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.29/deb/ /' | sudo tee /etc/apt/sources.list.d/kubernetes.list

sudo apt-get update -y
sudo apt-get install -y kubelet="1.29.0-*" kubectl="1.29.0-*" kubeadm="1.29.0-*"
```

```
sudo apt-get update -y
```

```
sudo apt-get install -y jq
```

```
sudo systemctl enable --now kubelet
```

```
sudo systemctl start kubelet
```

```
ubuntu@ip-172-31-0-232:/etc/ansible$ sudo nano playbook.yaml
ubuntu@ip-172-31-0-232:/etc/ansible$
```

```
i-0aa471167f40b6000 (Hari-DevOps-Project)
```

```
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

```
--
```

```
- name: install Jenkins and Java on host
```

```
become: true
```

```
hosts: localhost
```

```
tasks:
```

```
  - name: running script to install tools on host
```

```
    script: script1.sh
```

```
- name: install docker, K8s and Java on main
```

```
become: true
```

```
hosts: master
```

```
tasks:
```

```
  - name: running script to install tools on main
```

```
    script: script2.sh
```

```
- name: install docker, K8s on nodes
```

```
become: true
```

```
hosts: slaves
```

```
tasks:
```

```
  - name: running script to install tools on node
```

```
    script: script3.sh
```

```
ubuntu@ip-172-31-0-232:/etc/ansible$ ansible-playbook playbook.yaml --syntax-check
playbook: playbook.yaml
ubuntu@ip-172-31-0-232:/etc/ansible$ []
i-Oaa471167f40b6000 (Hari-DevOps-Project)
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

```
ubuntu@ip-172-31-0-232:/etc/ansible$ ansible-playbook playbook.yaml --check
PLAY [install Jenkins and Java on host] ****
TASK [Gathering Facts] ****
ok: [localhost]
TASK [running script to install tools on host] ****
skipping: [localhost]
PLAY [install docker, K8s and Java on main] ****
i-Oaa471167f40b6000 (Hari-DevOps-Project)
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

```
PLAY [install docker, K8s and Java on main] ****
TASK [Gathering Facts] ****
ok: [172.31.3.129]
TASK [running script to install tools on main] ****
skipping: [172.31.3.129]
PLAY [install docker, K8s on nodes] ****
TASK [Gathering Facts] ****
ok: [172.31.3.13]
ok: [172.31.0.49]
TASK [running script to install tools on node] ****
skipping: [172.31.3.13]
skipping: [172.31.0.49]
i-Oaa471167f40b6000 (Hari-DevOps-Project)
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

```
PLAY RECAP ****
172.31.0.49      : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
172.31.3.129     : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
172.31.3.13      : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
localhost        : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
ubuntu@ip-172-31-0-232:/etc/ansible$ []
i-Oaa471167f40b6000 (Hari-DevOps-Project)
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

## Running Playbook:

```
ubuntu@ip-172-31-0-232:/etc/ansible$ ansible-playbook playbook.yaml
PLAY [install Jenkins and Java on host] ****
TASK [Gathering Facts] ****
ok: [localhost]
TASK [running script to install tools on host] ****
changed: [localhost]
PLAY [install docker, K8s and Java on main] ****
TASK [Gathering Facts] ****
ok: [172.31.3.129]
TASK [running script to install tools on main] ****
[...]
i-0aa471167f40b6000 (Hari-DevOps-Project)
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

The screenshot shows a web browser window with the URL `13.232.38.197:8080/login?from=%2F`. The page title is "Getting Started". The main content is titled "Unlock Jenkins". It instructs the user to copy the password from either the log or this file. A red box highlights the path `/var/lib/jenkins/secrets/initialAdminPassword`. Below this, there is a text input field labeled "Administrator password" and a blue "Continue" button.

```
ubuntu@ip-172-31-0-232:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
a0d16ae92d6941fc9a5b99bbf1b9749b
ubuntu@ip-172-31-0-232:~$ [...]
i-0aa471167f40b6000 (Hari-DevOps-Project)
PublicIPs: 13.232.38.197 PrivateIPs: 172.31.0.232
```

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

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

**Administrator password**

.....

⚠ Not secure 13.232.38.197:8080

## Getting Started

# Getting Started

Folders	Formatter		
Timestamper	Workspace Cleanup	Ant	Gradle
Pipeline	Github Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline Graph View
Git	SSH Build Agents	Matrix Authorization Strategy	PAM Authentication
LDAP	Email Extension	Mailer	Dark Theme

Folders

- OWASP Markup Formatter
- \*\* ASM API
- \*\* JSON Path API
- \*\* Structs
- \*\* Pipeline: Step API
- \*\* Token Macro
- Build Timeout
- \*\* Credentials
- \*\* Plain Credentials
- \*\* Variant
- \*\* SSH Credentials

\*\* - required dependency

Jenkins 2.458

⚠ Not secure 13.232.38.197:8080

## Getting Started

# Create First Admin User

Username

Password

Confirm password

Full name

E-mail address

⚠ Not secure 13.232.38.197:8080

## Getting Started

# Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD\_URL environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Jenkins 2.458 Not now Save and Finish

← → ⌛ ⚡ Not secure 13.232.38.197:8080

# Jenkins

Dashboard >

+ New Item Add description

Build History Manage Jenkins My Views

Build Queue +  
No builds in the queue.

Build Executor Status +  
1 Idle  
2 Idle

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job Set up a distributed build

Set up an agent Configure a cloud Learn more about distributed builds

REST API Jenkins 2.458

Dashboard > Manage Jenkins > Plugins

## Plugins

Updates Available plugins Install ↻

Available plugins

SSH Agent 367.vf9076cd4ee21 Released 1 mo 5 days ago  
This plugin allows you to provide SSH credentials to builds via a ssh-agent in Jenkins.

Installed plugins Advanced settings Download progress

## New node

### Node name

K8 node

### 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**

### 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 ?

##### Description ?

##### Username

### Jenkins Credentials Provider: Jenkins

##### Username

ubuntu

Treat username as secret ?

##### Private Key

Enter directly

##### Key

Enter New Secret Below

```
-----BEGIN RSA PRIVATE KEY-----  
MIIEowIBAAKCAQEAjjsXWSFEXZv337ZtMhCourPDLQAIzBHDqr34Q7tUybg2UOFM  
MyNqoH23ImF7ut+Lx/NYsgQ8aPm9wnCrxEgfGvqVvaiyGRtl1a/bVhCxxy/6JSp1cv
```

##### Passphrase



Dashboard > Manage Jenkins > Nodes >

Host Key Verification Strategy ?

Non verifying Verification Strategy

Advanced ▾

Availability ?

Keep this agent online as much as possible

**Node Properties**

- Disable deferred wipeout on this node ?
- Disk Space Monitoring Thresholds
- Environment variables
- Tool Locations

**Save**

**Nodes**

+ 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	3.14 GiB	! 0 B	3.14 GiB	0ms ⚡
💻	K8 node	Linux (amd64)	In sync	4.60 GiB	! 0 B	4.60 GiB	39ms ⚡
Data obtained		0.11 sec	0.11 sec	0.11 sec	0.1 sec	0.1 sec	0.1 sec

Icon: S M L

Legend

```
ubuntu@ip-172-31-3-129:~$ ls
' '$'\004' jenkins kubernetes.sh
ubuntu@ip-172-31-3-129:~$ 
i-0a8a43e5bf1861eac (m3-master)
PublicIPs: 65.0.122.157 PrivateIPs: 172.31.3.129
```

Dashboard > Manage Jenkins

Add, remove, and configure cloud instances to provision agents on-demand.

Configure the look and feel of Jenkins

### Security

- Security**  
Secure Jenkins; define who is allowed to access/use the system.
- Credentials**  
Configure credentials

- Credential Providers**  
Configure the credential providers and types
- Users**  
Create/delete/modify users that can log in to this Jenkins.

### Status Information

- System Information**  
Displays various environmental information to assist trouble-shooting.
- System Log**  
System log captures output from java.util.logging output related to Jenkins.

Dashboard > Manage Jenkins > Credentials

## Credentials

T	P	Store ↓	Domain	ID	Name
⌚	👤	System	(global)	1d28d664-9ac3-4a3c-804a-f03ea8eeb186	ubuntu

### Stores scoped to Jenkins

P	Store ↓	Domains
👤	System	(global)

Icon: S M L

Now we will click on global.

Jenkins

Search (CTRL+K) ? 🔍 1 1 Hari Govind Singh log out

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

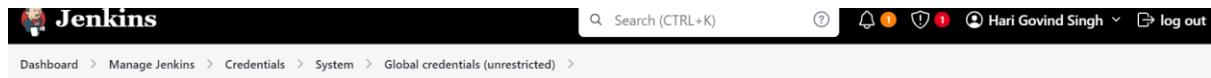
### Global credentials (unrestricted)

+ Add Credentials

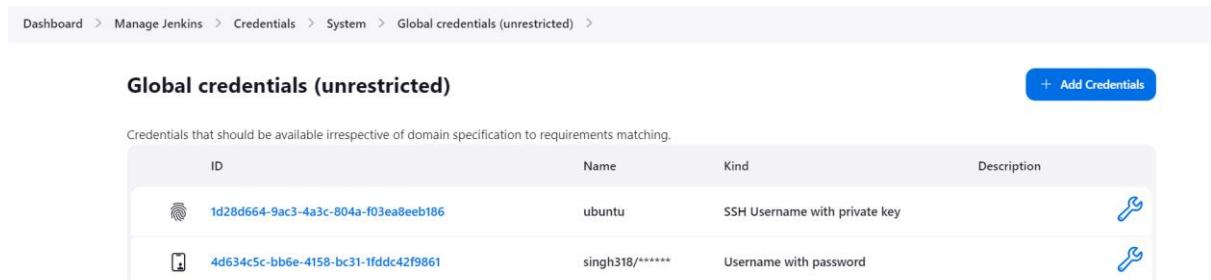
Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description
⌚ 1d28d664-9ac3-4a3c-804a-f03ea8eeb186	ubuntu	SSH Username with private key	🔧

Now we will click on Add Credentials and will enter the docker hub credentials.



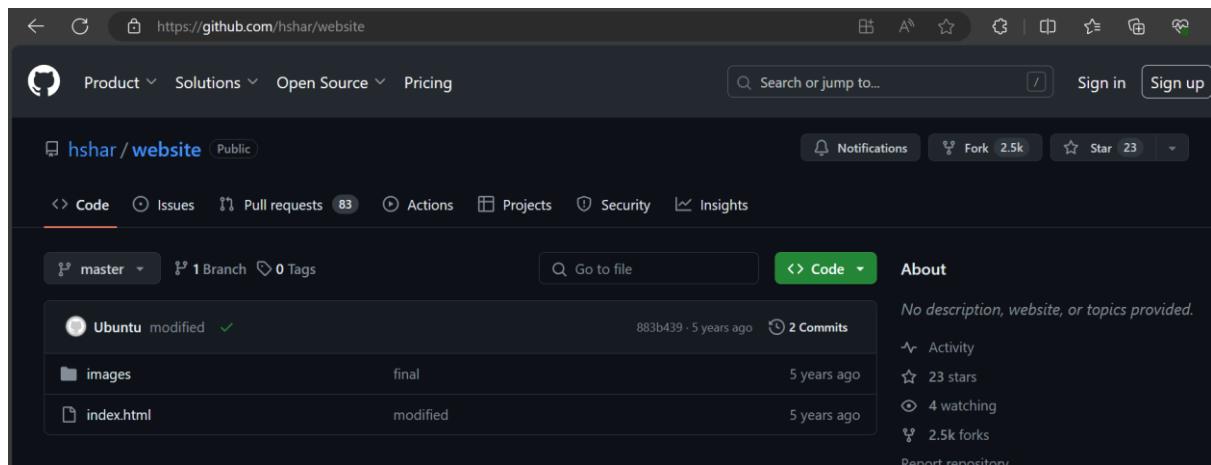
The screenshot shows the Jenkins 'Global credentials (unrestricted)' creation page. The 'Kind' dropdown is set to 'Username with password'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc.)'. The 'Username' field contains 'singh318'. The 'Password' field contains '\*\*\*\*\*'. A 'Create' button is visible at the bottom.



The screenshot shows the Jenkins 'Global credentials (unrestricted)' list page. It displays two entries:

ID	Name	Kind	Description
1d28d664-9ac3-4a3c-804a-f03ea8eeb186	ubuntu	SSH Username with private key	
4d634c5c-bb6e-4158-bc31-1fddc42f9861	singh318/*****	Username with password	

Let's fork the repository which is given in the project.



The screenshot shows a GitHub repository page for 'hshar / website'. The repository is public and has 2.5k forks. The 'Code' tab is selected, showing a commit history:

File	Commit Message	Time	Commits
Ubuntu	modified	883b439 · 5 years ago	2 Commits
images	final	5 years ago	
index.html	modified	5 years ago	

The 'About' section notes: 'No description, website, or topics provided.' It also shows activity statistics: 23 stars, 4 watching, and 2.5k forks.

<https://github.com/hshar/website/fork>

hshar / website

Code Issues Pull requests 83 Actions Projects Security Insights

## Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Required fields are marked with an asterisk (\*).

Owner \* Repository name \*

SinghH98 / projectwebsite ✓ projectwebsite is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

Copy the master branch only

Contribute back to hshar/website by adding your own branch. [Learn more.](#)

ⓘ You are creating a fork in your personal account.

Create fork

SinghH98 / projectwebsite

Code Pull requests Actions Projects Wiki Security Insights Settings

master projectwebsite / Dockerfile

SinghH98 Create Dockerfile 2b38e65 · now History

Code Blame 2 lines (2 loc) · 42 Bytes Code 55% faster with GitHub Copilot

```
1 FROM ubuntu/apache2
2 COPY . /var/www/html/
```

Jenkins

Dashboard > All > New Item

### New Item

Enter an item name

Project

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.

OK

Dashboard > Project > Configuration

## Pipeline

### Configure

General Advanced Project Options Pipeline

Definition

Script ?

```
1 * pipeline {
2     agent none
3     environment {
4         DOCKERHUB_CREDENTIALS=credentials('4d634c5c-bb6e-4158-bc31-1fddc42f9861')
5     }
6     stages {
7         stage('Hello') {
8             steps {
9                 echo 'Hello World'
10            }
11        }
12    }
13}
14}
15}
```

Hello World

Use Groovy Sandbox ?

Save Apply

Dashboard > Project > #1

### Console Output

Status Changes Console Output View as plain text Edit Build Information Delete build '#1' Timings Pipeline Overview Pipeline Console Restart from Stage

```
Started by user Hari Govind Singh
[Pipeline] Start of Pipeline
[Pipeline] withCredentials
Masking supported pattern matches of $DOCKERHUB_CREDENTIALS or $DOCKERHUB_CREDENTIALS_PSW
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Hello)
[Pipeline] echo
Hello World
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] End of Pipeline
Finished: SUCCESS
```

## Definition

Pipeline script

Script ?

```
6 ▼ stages {
7 ▼   stage('Hello') {
8 ▼     steps {
9       |   echo 'Hello World'
10      }
11    }
12 ▼   stage('git') {
13 ▼     agent{
14       |   label 'K8 node'
15     }
16 ▼     steps {
17       |   git 'https://github.com/SinghH98/projectwebsite.git'
18     }
19   }
20 }
21 }
22 }
```

Use Groovy Sandbox ?

Save

Apply

Jenkins

Search (CTRL+K) ? bell shield circle Hari Govind Singh log out

Dashboard > Project > #4

Status Changes Console Output

View plain text

Edit Build Information Delete build '#4'

Timings Git Build Data Pipeline Overview Pipeline Console

Started by user Hari Govind Singh  
[Pipeline] Start of Pipeline  
[Pipeline] withCredentials  
Masking supported pattern matches of \$DOCKERHUB\_CREDENTIALS or \$DOCKERHUB\_CREDENTIALS\_PSW  
[Pipeline] {  
[Pipeline] stage  
[Pipeline] { (Hello)  
[Pipeline] echo  
Hello World  
[Pipeline] }  
[Pipeline] // stage  
[Pipeline] stage  
[Pipeline] { (git)  
[Pipeline] node  
Running on K8 node in /home/ubuntu/jenkins/workspace/Project  
[Pipeline] {

← → G Not secure 13.232.38.197:8080/job/Project/4/console

Dashboard > Project > #4

Workspaces

← Previous Build

```
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch --tags --force --progress -- https://github.com/SinghH98/projectwebsite.git
+refs/heads/*:refs/remotes/origin/*
> git config remote.origin.url https://github.com/SinghH98/projectwebsite.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/*
Avoid second fetch
Checking out Revision 2b38e658b47dede60ce7d53eaef2652c858258e2 (refs/remotes/origin/master)
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
> git config core.sparsecheckout # timeout=10
> git checkout -f 2b38e658b47dede60ce7d53eaef2652c858258e2 # timeout=10
> git branch -a -v --no-abbrev # timeout=10
> git checkout -b master 2b38e658b47dede60ce7d53eaef2652c858258e2 # timeout=10
Commit message: "Create Dockerfile"
First time build. Skipping changelog.
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] End of Pipeline
Finished: SUCCESS
```

```
Last login: Wed May 15 15:37:51 2024 from 13.233.177.3
ubuntu@ip-172-31-3-129:~$ ls
'$'\004' jenkins kubernetes.sh
ubuntu@ip-172-31-3-129:~$ cd jenkins
ubuntu@ip-172-31-3-129:~/jenkins$ ls
remoting remoting.jar workspace
ubuntu@ip-172-31-3-129:~/jenkins$ cd workspace
ubuntu@ip-172-31-3-129:~/jenkins/workspace$ ls
Project
ubuntu@ip-172-31-3-129:~/jenkins/workspace$ 
```

i-0a8a43e5bf1861eac (m3-master)

PublicIPs: 65.0.122.157 PrivateIPs: 172.31.3.129

```
ubuntu@ip-172-31-3-129:~/jenkins/workspace$ ls
Project
ubuntu@ip-172-31-3-129:~/jenkins/workspace$ cd Project
ubuntu@ip-172-31-3-129:~/jenkins/workspace/Project$ ls
Dockerfile images index.html
ubuntu@ip-172-31-3-129:~/jenkins/workspace/Project$ 
```

i-0a8a43e5bf1861eac (m3-master)

PublicIPs: 65.0.122.157 PrivateIPs: 172.31.3.129

## Definition

Pipeline script

Script ?

```
17  
18 }  
19  
20 stage('docker') {  
21   agent{  
22     label 'K8 node'  
23   }  
24   steps {  
25     sh 'sudo apt install docker-buildx'  
26     sh 'sudo docker build /home/ubuntu/jenkins/workspace/Project -t singh318/devopsproject'  
27     sh 'sudo echo $DOCKERHUB_CREDENTIALS_PSW | sudo docker login -u $DOCKERHUB_CREDENTIALS_USR --password-stdin'  
28     sh 'sudo docker push singh318/devopsproject'  
29   }  
30 }  
31  
32 }  
33
```

## Pipeline

### Definition

Pipeline script

Script ?

```
18  
19  
20 stage('docker') {  
21   agent{  
22     label 'K8 node'  
23   }  
24   steps {  
25     sh 'sudo apt install docker-buildx'  
26     sh 'sudo docker build /home/ubuntu/jenkins/workspace/Project -t singh318/devopsproject'  
27     sh 'sudo echo $DOCKERHUB_CREDENTIALS_PSW | sudo docker login -u $DOCKERHUB_CREDENTIALS_USR --password-stdin'  
28     sh 'sudo docker push singh318/devopsproject'  
29  
30  
31  
32  
33
```

Dashboard > Project > #11

Status

</> Changes

Console Output

View as plain text

Edit Build Information

Delete build '#11'

Timings

Git Build Data

Pipeline Overview

Pipeline Console

### Console Output

```
Started by user Hari Govind Singh  
[Pipeline] Start of Pipeline  
[Pipeline] withCredentials  
Masking supported pattern matches of $DOCKERHUB_CREDENTIALS or $DOCKERHUB_CREDENTIALS_PSW  
[Pipeline] {  
[Pipeline] stage  
[Pipeline] { (Hello)  
[Pipeline] echo  
Hello World  
[Pipeline] }  
[Pipeline] // stage  
[Pipeline] stage  
[Pipeline] { (git)  
[Pipeline] node  
Running on K8 node in /home/ubuntu/jenkins/workspace/Project  
[Pipeline] {
```

Dashboard > Project > #11

```

The push refers to repository [docker.io/singh318/devopsproject]
1a102914b00d: Preparing
1ff9e12edc64: Preparing
f545706e7f85: Preparing
3b2125e3f1e0: Preparing
1ff9e12edc64: Mounted from ubuntu/apache2
3b2125e3f1e0: Mounted from ubuntu/apache2
f545706e7f85: Mounted from ubuntu/apache2
1a102914b00d: Pushed
latest: digest: sha256:3e274c5e361893a0ddd53d4d301775ad084ef2279c68d6649a42e5e8aed0ce66 size: 1158
[Pipeline]
[Pipeline] // node
[Pipeline]
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] End of Pipeline
Finished: SUCCESS

```

<https://hub.docker.com/repository/docker/singh318/devopsproject/general>

This repository does not have a description INCOMPLETE

This repository does not have a category INCOMPLETE

**Tags**

This repository contains 1 tag(s).

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

[See all](#)

**Docker commands**

To push a new tag to this repository:

```
docker push singh318/devopsproject:tagname
```

**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](#)

## Definition

Pipeline script

### Script ?

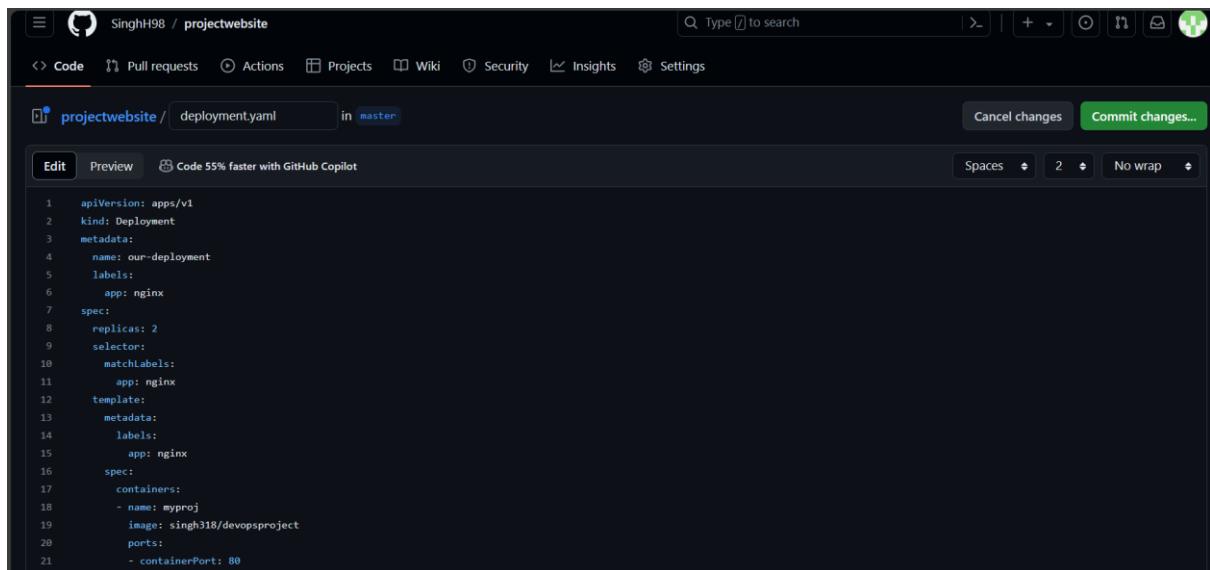
```

27      sh 'sudo echo $DOCKERHUB_CREDENTIALS_PSW | sudo docker login -u $DOCKERHUB_CREDENTIALS_USR --pa
28      sh 'sudo docker push singh318/devopsproject'
29
30  }
31  }
32  stage('k8s') {
33    agent{
34      label 'K8 node'
35    }
36    steps {
37      sh 'kubectl apply -f deployment.yaml'
38      sh 'kubectl apply -f service.yaml'
39    }
40  }
41
42

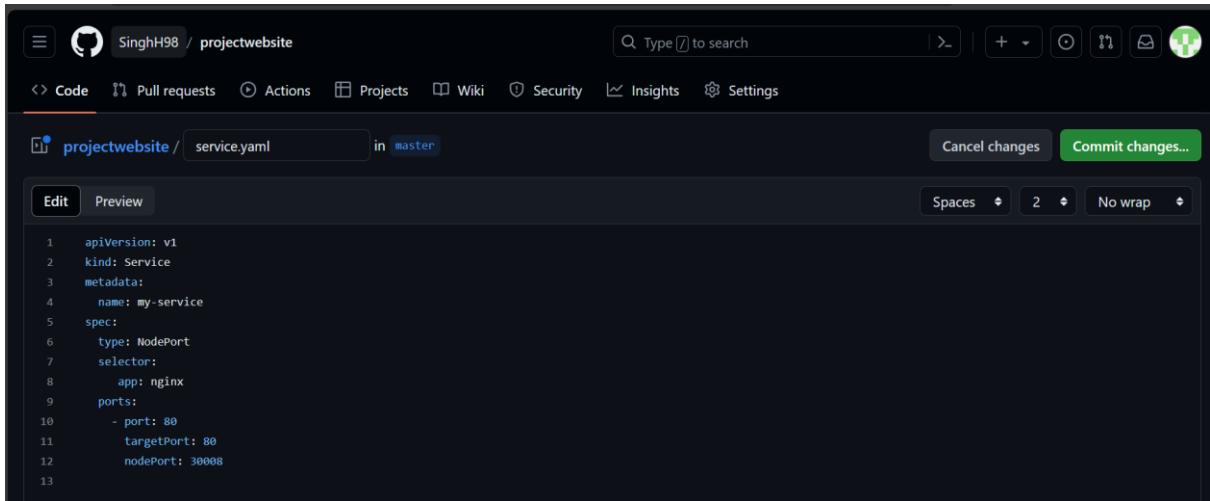
```

[Save](#)

[Apply](#)

A screenshot of a GitHub repository page for "SinghH98 / projectwebsite". The user is viewing the "deployment.yaml" file in the "master" branch. The file content is a Kubernetes Deployment configuration. The GitHub Copilot feature is shown as a suggestion above the code editor.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: our-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: myproj
          image: singh318/devopsproject
          ports:
            - containerPort: 80
```



A screenshot of a GitHub code editor interface. The repository is 'SinghH98 / projectwebsite'. The file being edited is 'service.yaml' in the 'master' branch. The code content is:

```
apiVersion: v1
kind: Service
metadata:
  name: my-service
spec:
  type: NodePort
  selector:
    app: nginx
  ports:
    - port: 80
      targetPort: 80
      nodePort: 30008
```

apiVersion: v1

kind: Service

metadata:

name: my-service

spec:

type: NodePort

selector:

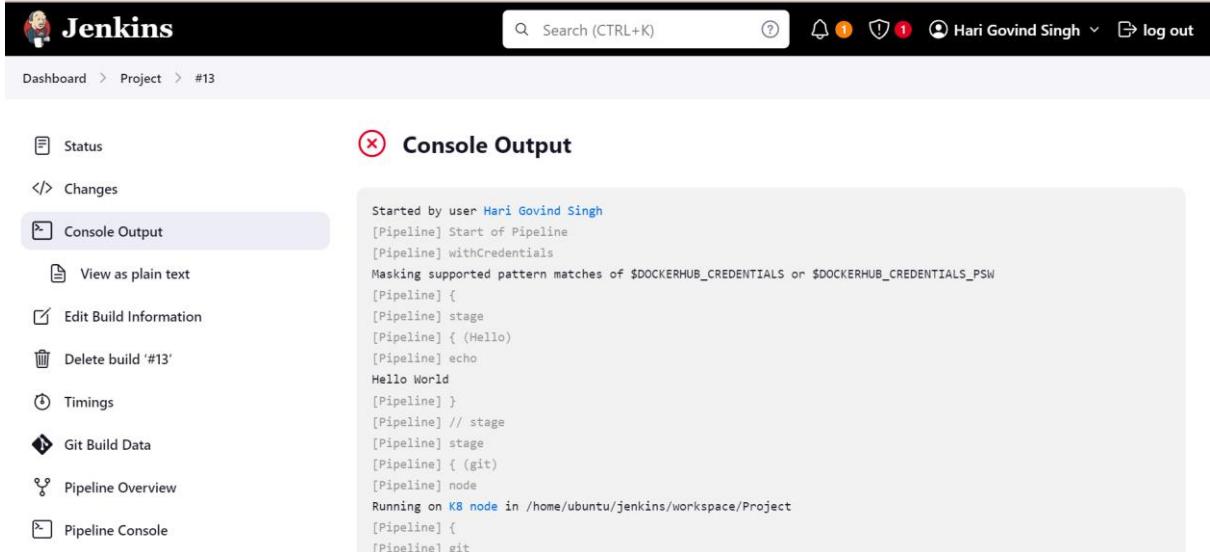
app: nginx

ports:

- port: 80

targetPort: 80

nodePort: 30008



A screenshot of a Jenkins pipeline console output. The pipeline was started by user Hari Govind Singh. The log output shows:

```
Started by user Hari Govind Singh
[Pipeline] Start of Pipeline
[Pipeline] withCredentials
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Hello)
[Pipeline] echo
Hello World
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (git)
[Pipeline] node
Running on K8 node in /home/ubuntu/jenkins/workspace/Project
[Pipeline] {
[Pipeline] git
```

Dashboard > Project > #13

```
[Pipeline] { (k8s)
[Pipeline] node
Running on K8 node in /home/ubuntu/jenkins/workspace/Project
[Pipeline] {
[Pipeline] sh
+ kubectl apply -f deployment.yaml
error: error validating "deployment.yaml": error validating data: failed to download openapi: Get "http://localhost:8080/openapi/v2?timeout=32s": dial tcp 127.0.0.1:8080: connect: connection refused; if you choose to ignore these errors, turn validation off with --validate=false
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] End of Pipeline
ERROR: script returned exit code 1
Finished: FAILURE
```

Dashboard > Project > Configuration

Throttle builds [?](#)

## Configure

### Build Triggers

 General

 Advanced Project Options

 Pipeline

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

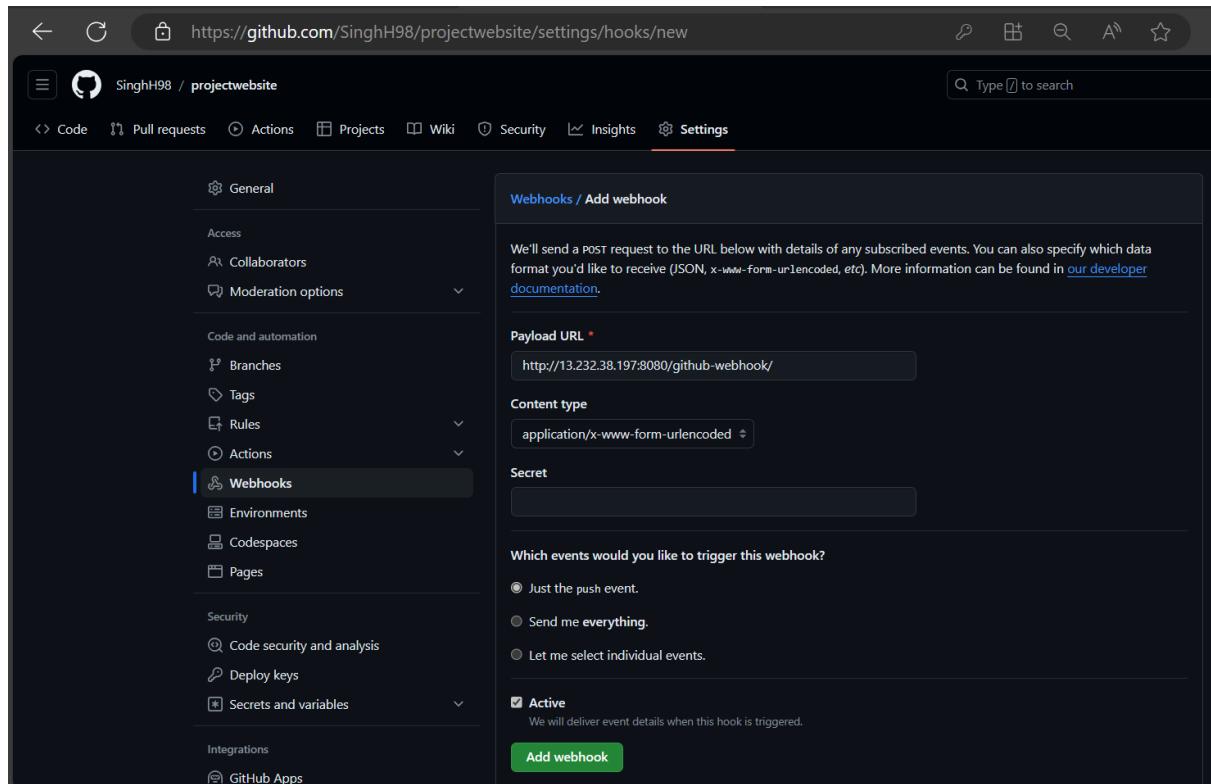
Advanced [▼](#)

← → ⌂ ⚠ Not secure 13.232.38.197:8080/job/Project/

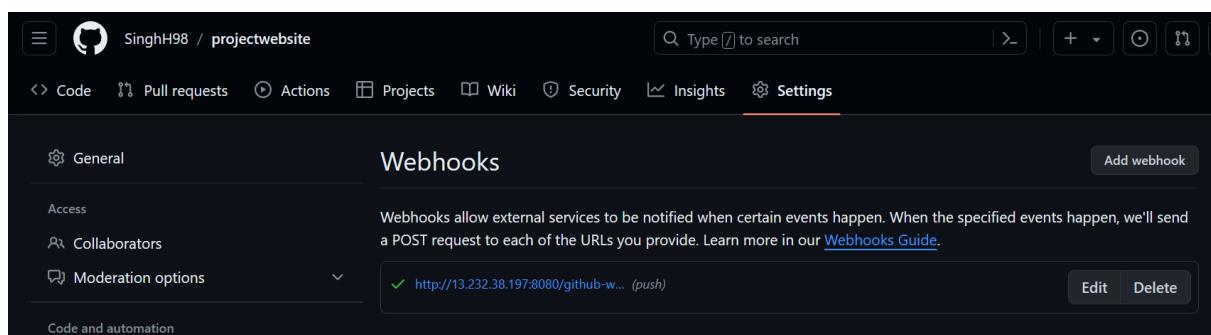


# Jenkins

Dashboard > Project >



The screenshot shows the GitHub settings interface for a repository named "projectwebsite". The left sidebar is collapsed, showing the "Webhooks" section under "Code and automation". The main content area is titled "Webhooks / Add webhook". It contains instructions about sending POST requests to a specified URL for subscribed events. The "Payload URL" field is populated with "http://13.232.38.197:8080/github-webhook/". The "Content type" dropdown is set to "application/x-www-form-urlencoded". A "Secret" input field is present but empty. Below these, there are three radio button options for event triggers: "Just the push event." (selected), "Send me everything.", and "Let me select individual events.". A checked checkbox labeled "Active" indicates the hook will deliver event details when triggered. A green "Add webhook" button is at the bottom.



The screenshot shows the GitHub settings interface for the same repository. The left sidebar is collapsed, showing the "Webhooks" section under "Code and automation". The main content area is titled "Webhooks". It provides a brief description of what webhooks do and links to the "Webhooks Guide". A single webhook entry is listed: "http://13.232.38.197:8080/github-w... (push)". To the right of the URL are two buttons: "Edit" and "Delete".

## Definition

Pipeline script

Script ?

```
28 } | sh 'sudo docker push singns18/devopsproject'
29 |
30 }
31 v stage('K8s') {
32 v   agent{
33 |     label 'K8 node'
34 }
35 v   steps {
36     sh 'kubectl delete deploy deployment.yaml'
37     sh 'kubectl apply -f deployment.yaml'
38     sh 'kubectl delete deploy service.yaml'
39     sh 'kubectl apply -f service.yaml'
40   }
41 }
42 }
43 }
44 }
```

Use Groovy Sandbox ?

Save

Apply

\*\*\*\*\* THE END \*\*\*\*\*