Install Gitlab-runner and jenkins-agent

```bash
#! /bin/bash
########### Checking if the user is root

if
    [[ $(id -u) -ne 0 ]]
    then
    echo "Sorry you need to be root"
    exit 1
fi


TYPE=$(cat /etc/*rel* | grep -o ID=Ubuntu | awk -F= '{print $2}')

################# Checking OS type
if
    [[ $TYPE != Ubuntu ]]
    then
    echo "Sorry this script can only work on Ubuntu"
    exit 1
fi


echo "1. Jenkins"
echo "2. Gitlab-runner"
read -p "Please enter the number that correspond to your need: " NUM
if
[[  ${NUM} -eq 1 ]]
then
clear
echo
echo "Installing Jenkins......... please wait"

############ JENKINS NODE ################################


apt update -y
apt install tree curl vim wget ansible default-jre default-jdk  ansible-
lint maven git  unzip openssl sshpass  gnupg -y
sudo apt-get update -y
export DEBIAN_FRONTEND=noninteractive

ln -fs /usr/share/zoneinfo/America/New_York /etc/localtime
apt-get install -y tzdata
dpkg-reconfigure --frontend noninteractive tzdata
sudo git clone https://github.com/ahmetb/kubectx /usr/local/kubectx

sudo ln -s /usr/local/kubectx/kubectx /usr/local/bin/kubectx
```

```
sudo ln -s /usr/local/kubectx/kubens /usr/local/bin/kubens

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io
/release/stable.txt)/bin/linux/amd64/kubectl"

curl -LO "https://dl.k8s.io/$(curl -L -s https://dl.k8s.io/release
/stable.txt)/bin/linux/amd64/kubectl.sha256"

echo "$(cat kubectl.sha256)  kubectl" | sha256sum --check

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

chmod +x kubectl

mkdir -p ~/.local/bin

mv ./kubectl ~/.local/bin/kubectl

useradd jenkins -m -d /home/jenkins -s /bin/bash

useradd ansible -m -d /home/ansible -s /bin/bash

echo -e 'jenkins  ALL=(ALL)  NOPASSWD:  ALL' > /etc/sudoers.d/jenkins

echo -e 'ansible  ALL=(ALL)  NOPASSWD:  ALL' > /etc/sudoers.d/ansible
sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2
/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-
compose
sudo chmod +x /usr/local/bin/docker-compose
mkdir -p /usr/local/lib/docker/cli-plugins
curl -SL https://github.com/docker/compose/releases/download/v2.4.1
/docker-compose-linux-x86_64 -o /usr/local/lib/docker/cli-plugins
/docker-compose
sudo chmod +x /usr/local/lib/docker/cli-plugins/docker-compose
sudo apt-get remove docker docker-engine docker.io containerd runc -y
sudo apt-get update

 sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg

echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings
/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.
list > /dev/null
```

```
 sudo apt-get update

 sudo apt-get install docker-ce docker-ce-cli containerd.io docker-
compose-plugin -y


 sudo apt update
 sudo apt install default-jre -y
 sudo apt install default-jdk -y
   apt install -y apt-transport-https \

      ca-certificates \
      dnsutils \
      htop \
      iftop \
      iotop \
      iperf \
      iputils-ping \
      jq \
      less \
      locales \
      ltrace \
      man-db \
      manpages \
      mosh \
      mtr \
      netcat \
      nethogs \
      nfs-common \
      pass \
      psmisc \
      python3-apt \
      python3-docker \
      rkhunter \
      rsync \
      screen \
      ssl-cert \
      strace \
      tcpdump \
      time \
      traceroute \
      unhide \
      unzip \
      uuid-runtime
sudo chmod 666 /var/run/docker.sock
usermod jenkins -aG docker
usermod ansible -aG docker
usermod ubuntu -aG docker
chmod 666 /var/run/docker.sock
sudo apt-get update && sudo apt-get install -y gnupg software-
```

```
properties-common curl
curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -
sudo apt-add-repository "deb [arch=amd64] https://apt.releases.
hashicorp.com $(lsb_release -cs) main"
sudo apt-get update && sudo apt-get install terraform
usermod jenkins -aG docker
usermod ansible -aG docker


elif
    [[  ${NUM} -eq 2 ]]
then
clear
echo
echo "Installing Gitlab-runner .......... please wait"

apt update -y
apt install tree curl vim wget ansible default-jre default-jdk  ansible-
lint maven git  unzip openssl sshpass  gnupg -y
sudo apt-get update -y
export DEBIAN_FRONTEND=noninteractive


ln -fs /usr/share/zoneinfo/America/New_York /etc/localtime
apt-get install -y tzdata
dpkg-reconfigure --frontend noninteractive tzdata
sudo git clone https://github.com/ahmetb/kubectx /usr/local/kubectx

sudo ln -s /usr/local/kubectx/kubectx /usr/local/bin/kubectx

sudo ln -s /usr/local/kubectx/kubens /usr/local/bin/kubens

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io
/release/stable.txt)/bin/linux/amd64/kubectl"

curl -LO "https://dl.k8s.io/$(curl -L -s https://dl.k8s.io/release
/stable.txt)/bin/linux/amd64/kubectl.sha256"

echo "$(cat kubectl.sha256)  kubectl" | sha256sum --check

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

chmod +x kubectl

mkdir -p ~/.local/bin

mv ./kubectl ~/.local/bin/kubectl


useradd ansible -m -d /home/ansible -s /bin/bash
echo -e 'ansible  ALL=(ALL)  NOPASSWD:  ALL' > /etc/sudoers.d/ansible
sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2
```

```
/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-
compose
sudo chmod +x /usr/local/bin/docker-compose
mkdir -p /usr/local/lib/docker/cli-plugins
curl -SL https://github.com/docker/compose/releases/download/v2.4.1
/docker-compose-linux-x86_64 -o /usr/local/lib/docker/cli-plugins
/docker-compose
sudo chmod +x /usr/local/lib/docker/cli-plugins/docker-compose
sudo apt-get remove docker docker-engine docker.io containerd runc -y
sudo apt-get update

 sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg

echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings
/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.
list > /dev/null
 sudo apt-get update

 sudo apt-get install docker-ce docker-ce-cli containerd.io docker-
compose-plugin -y


 sudo apt update
 sudo apt install default-jre -y
 sudo apt install default-jdk -y
   apt install -y apt-transport-https \

      ca-certificates \
      dnsutils \
      htop \
      iftop \
      iotop \
      iperf \
      iputils-ping \
      jq \
      less \
      locales \
      ltrace \
      man-db \
      manpages \
      mosh \
      mtr \
```

```
        netcat \
        nethogs \
        nfs-common \
        pass \
        psmisc \
        python3-apt \
        python3-docker \
        rkhunter \
        rsync \
        screen \
        ssl-cert \
        strace \
        tcpdump \
        time \
        traceroute \
        unhide \
        unzip \
        uuid-runtime
sudo chmod 666 /var/run/docker.sock
usermod ansible -aG docker
usermod ubuntu -aG docker
chmod 666 /var/run/docker.sock
sudo apt-get update && sudo apt-get install -y gnupg software-
properties-common curl
curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -
sudo apt-add-repository "deb [arch=amd64] https://apt.releases.
hashicorp.com $(lsb_release -cs) main"
sudo apt-get update && sudo apt-get install terraform

usermod ansible -aG docker

curl -L "https://packages.gitlab.com/install/repositories/runner/gitlab-
runner/script.deb.sh" | sudo bash
sudo apt-get install gitlab-runner
apt-cache madison gitlab-runner
sudo apt-get install gitlab-runner
sudo gitlab-runner -version
sudo gitlab-runner status
sudo gitlab-runner start
sudo gitlab-runner stop
sudo gitlab-runner restart
echo -e 'gitlab-runner  ALL=(ALL)  NOPASSWD:  ALL' > /etc/sudoers.d
/gitlab-runner
usermod gitlab-runner -aG docker

else
echo "please select either 1 or 2"
fi
```

Follow the following documentation from Step#4