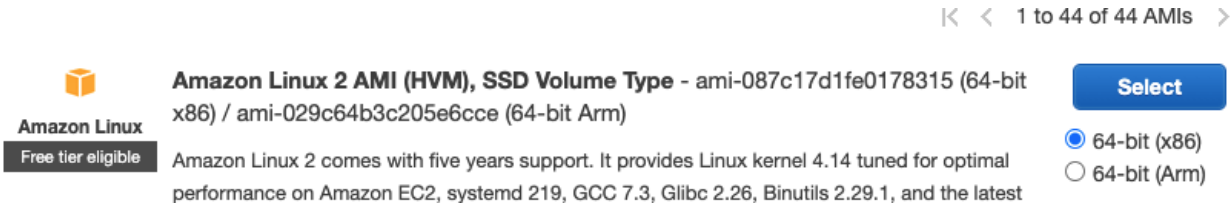
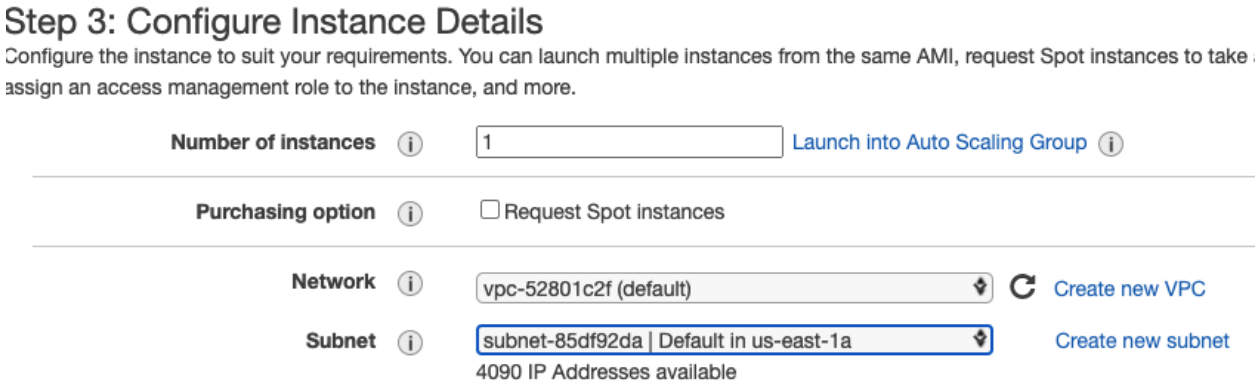
Deployment 6

By following the task we created 3 EC2 instances 1-Master and 2-Agent EC2s

1. First EC2 is Master with ports ssh anywhere-22, Custom TCP 8080 and 500



Subnet: Default in us-east-1a



Add Tags:
Name Master

Finally, select keypair.pem and launch an instance

In the terminal ssh into that EC2 Master

- ssh -i keypair.pem ec2-user@PublicIPv4
- nano 8.sh
- Copy and past
  User data script

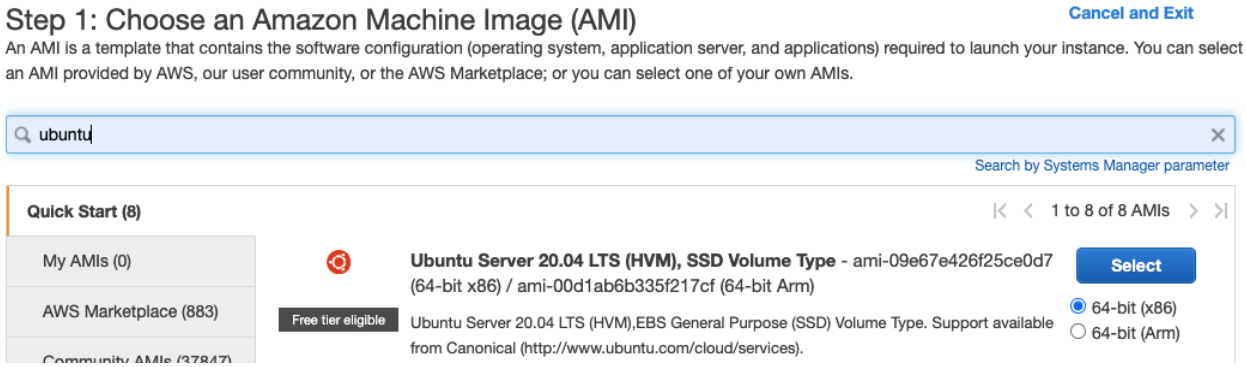
sudo amazon-linux-extras install java-openjdk11
sudo amazon-linux-extras install epel
sudo wget -O /etc/yum.repos.d/jenkins.repo \

https://pkg.jenkins.io/redhat-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
sudo yum upgrade
sudo yum install epel-release java-11-openjdk-devel
sudo yum install jenkins
sudo systemctl start jenkins

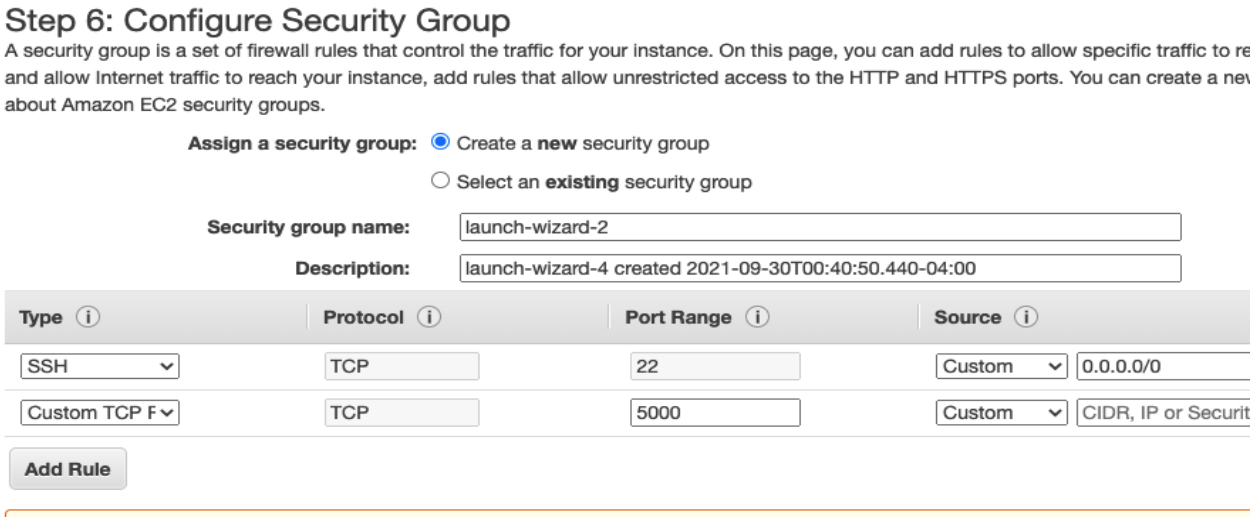
- save and exit
- bash 8.sh (should start installation process all dependencies)

2. Second EC2 is Agent1 will use Ubuntu, AMI and port ssh -22, Custom TCP 5000

Selected: ubuntu



Selected: t2.micro
Configure Instance: by default
Add tag: Name Agent1
Security group: ports ssh - 22 and Custom TCP - 5000



Finally, select keypair.pem and launch an instance

In the terminal ssh into that EC2 Agent1

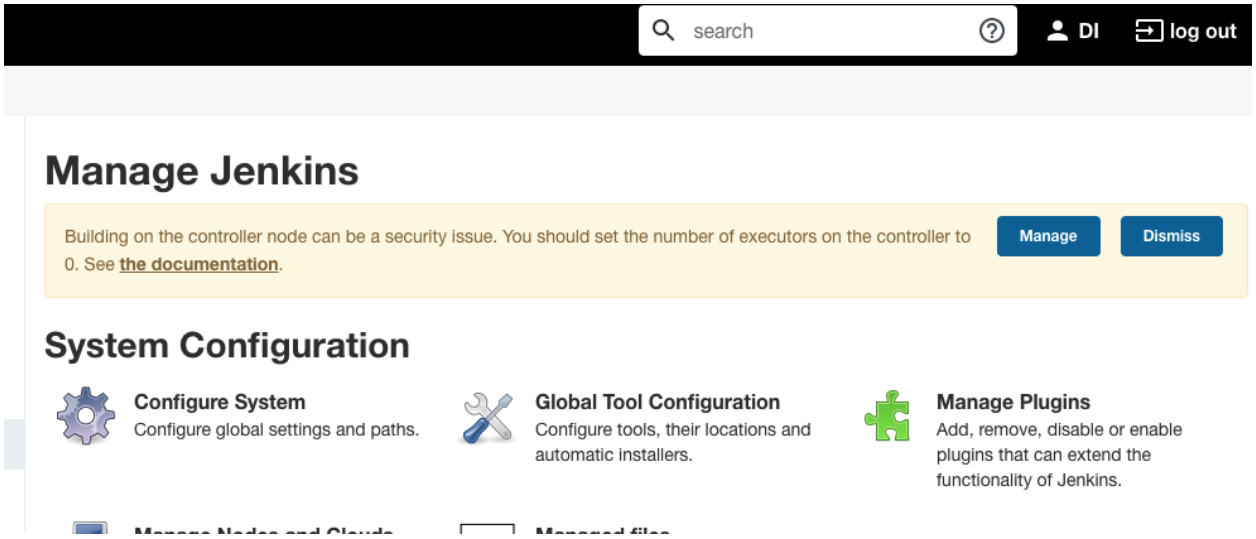
- ls: 8.sh
- nano keypair.pem
- ls: 8.sh keypair.pem
- ssh -i keypair.pem ec2-user@Agent1 Public IPv4
- chmod 4000 keypair.pem
- ls: 8.sh keypair.pem
- ssh -i keypair.pem ec2-user@18.233.98.167 (Permission denied (publickey)
- cat keypair.pem
- copy keypair.pem (open with TextEdit) and past
- ssh -i keypair.pem ubuntu@Agent1 Public IPv4
- nano a.sh
- Copy and past
  User data script

```
#!/bin/bash
sudo apt-get update && sudo apt-get upgrade -y
sudo apt-get install -y\
default-jre \
git \
nodejs -y \
npm -y
sudo apt install default-jre git nodejs npm
```

- Save and exit
- bash a.sh (should start installation process all dependencies)
- node --version (Command 'node' not found)
- sudo apt install default-jre git nodejs npm
- Exit
- Then check
- maven --version (command not found)
- java --version (already installed)
- Open new tab and go to the Jenkins using Master Public IPv4 and port 8080:
  Ipv4:8080
- Copy the path and paste to terminal usind cat command:
  sudo cat /var/lib/jenkins/secrets/initialAdminPassword

93cbd89e096b4d008952bbfe0099f18c

- Log in to the Jenkins
- Install suggested plugins
- Create an account
- We need to download plugins go to Manage Jenkins
- Manage Plugins



- Install: NodeJs, Amazon EC2, Maven Integration
- Select: Download now and install after restart

3. Third EC2 is Agen2 will use Ubuntu, AMI and port ssh -22
- Select ubuntu
- t2 micro
- Add tag: name Agent2
- Configure SG: SSH port 22
- Select keypair and launch it.
- SSH third EC2 go to terminal
- ssh -i keypair.pem ubuntu@Agent2 Public IPv4
- nano a.sh
- Copy and paste

```
!/bin/bash
sudo apt-get update && sudo apt-get upgrade -y
sudo apt-get install -y\
default-jre \
git \
nodejs -y  \
npm -y  \
maven  \
libgtk2.0-0  \
libgtk-3-0  \
libgbm-dev  \
libnotify-dev  \
libgconf-2-4  \
```

libnss3 \
libxss1 \
libasound2 \
libxtst6 \
xauth \
xvfb

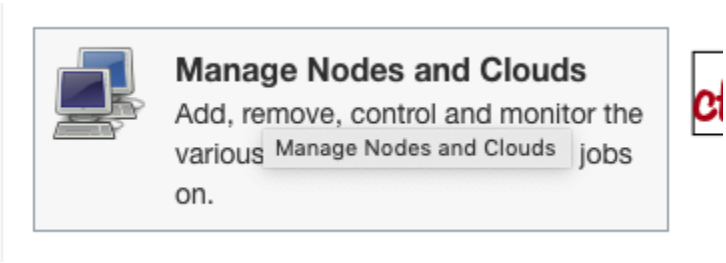
- Save and exit
- bash a.sh
- nano b.sh
- Copy and paste

sudo apt install default-jre git nodejs npm maven libgtk2.0-0 libgtk-3-0 libgbm-dev libnotify-dev libgconf-2-4 libnss3 libxss1 libasound2 libxtst6 xauth xvfb

- Save and exit
- bash b.sh (should start installation process all dependencies)
- sudo apt install libnotify-dev
- Exit
- sudo yum install git
- Exit
- sudo apt install xvfb

Go to Jenkins

- Dashboard
- Manage Jenkins
- Manage Nodes and Clouds



- New Node
- Node name: anything my Ubuntu1
- Select: Permanent Agent and OK

Name

Ubuntu1

Description

Deployment6 - ubuntu1

Number of executors

2

Remote root directory

/home/ubuntu/jenkins/app

Labels

agent-ubuntu1

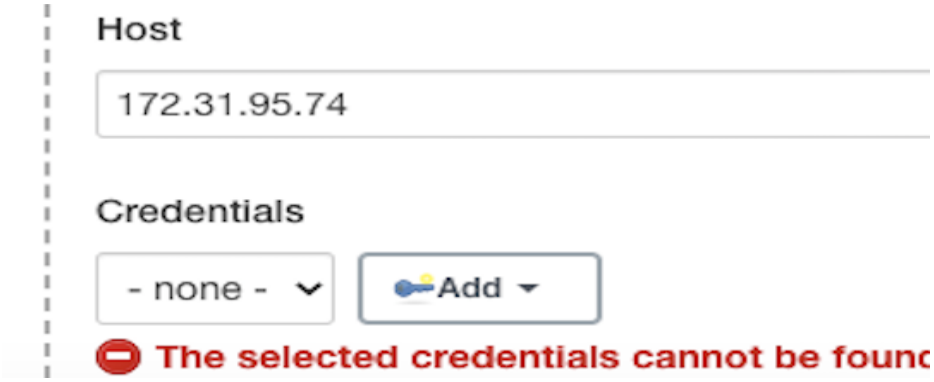
Usage

Use this node as much as possible

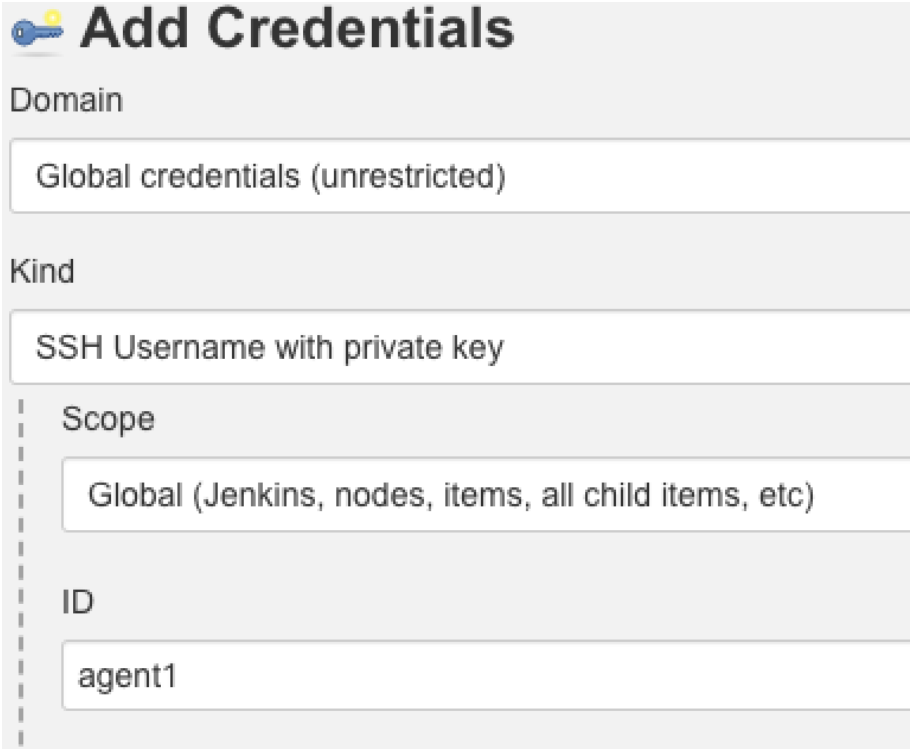
Launch method

Launch agents via SSH

The host should be the private IPv4 of the Agent1

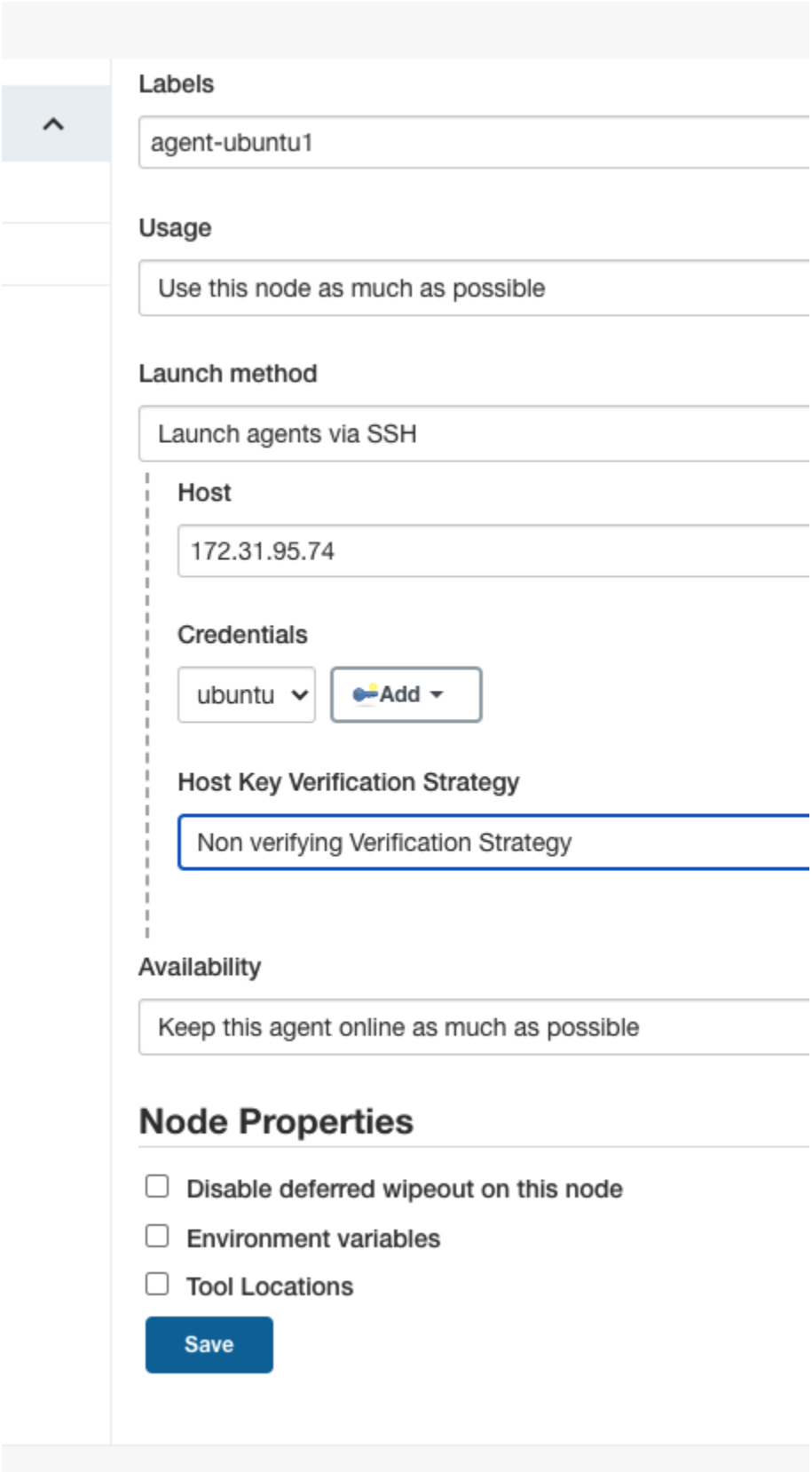


Add Credentials

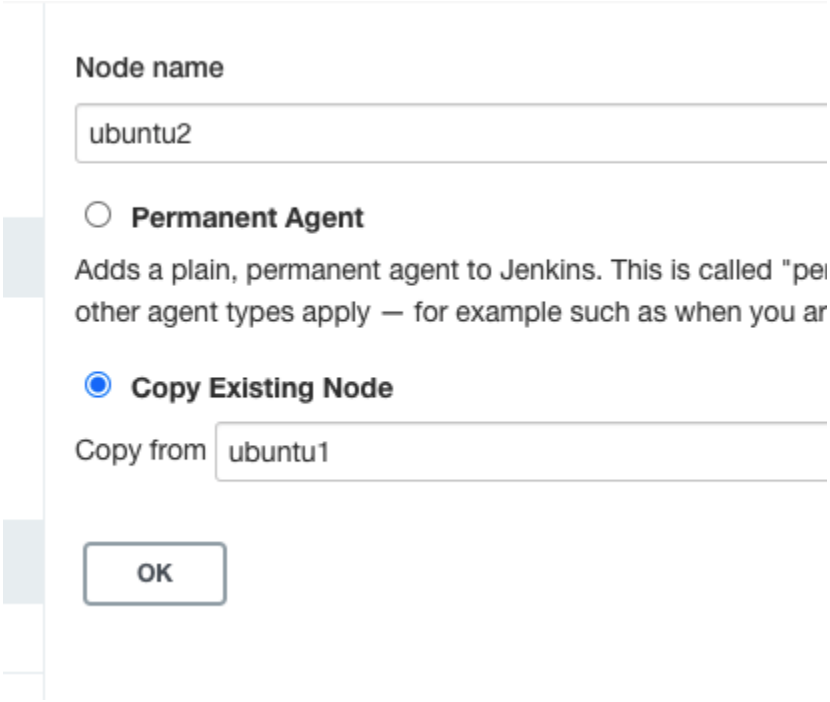


- User name: ubuntu
- Select: Enter directly
- Key: copy keypair.pem (open with TextEdit) and past
- Add

- Credentials -none-: select GitHub acc name



- Save
- Create the second node
- New Node
- Node name: anything my Ubuntu2
- Select: Permanent Agent and OK



-
- Host: Host should be the private IPv4 of the Agent2
- Select ssh credentials
  Username: Ubuntu
  Key used to log in

Name

ubuntu2

Description

Number of executors

2

Remote root directory

/home/ubuntu/jenkins/app

Labels

agent-ubuntu2

Usage

Use this node as much as possible

Launch method

Launch agents via SSH

Host

172.31.86.182

The Host must be specified

Credentials

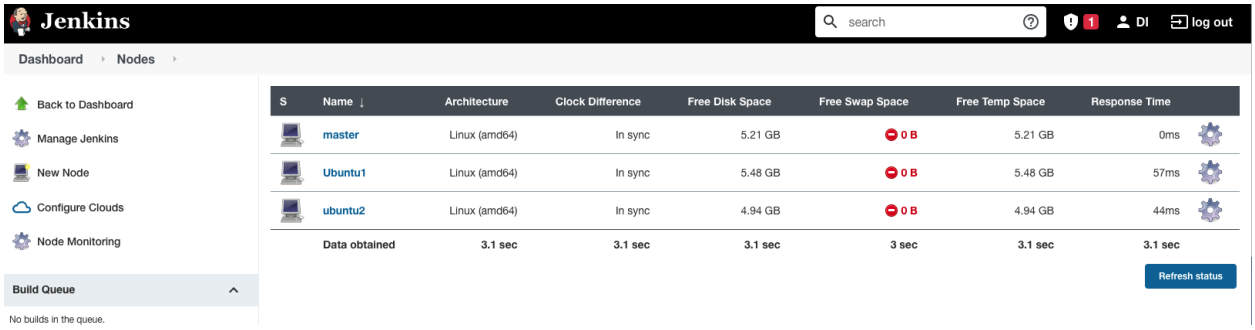
ubuntu Add

Host Key Verification Strategy

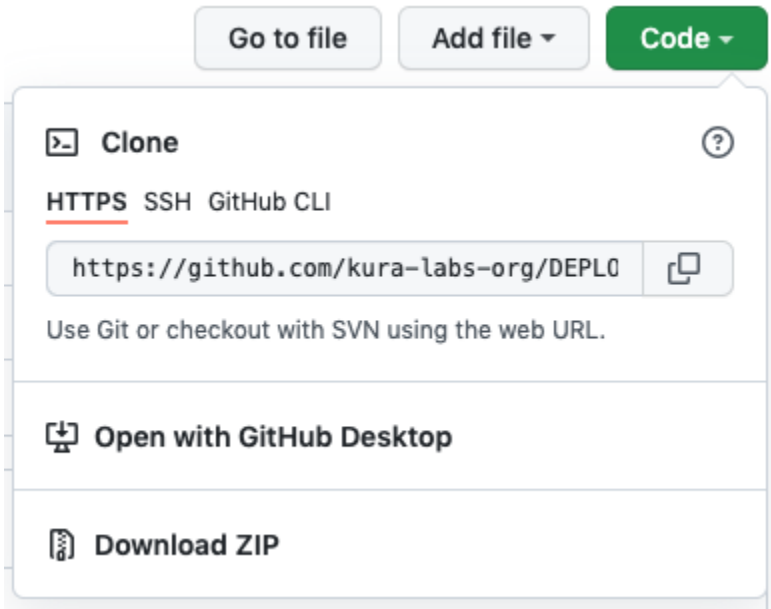
Save

- Save it

- Got to the master
-



Download the repository from Kura GitHub https://github.com/kura-labs-org/DEPLOY6_FE - Code - Download ZIP



Go to the Jenkinsfile and edit it, enter your label name for your first agent1

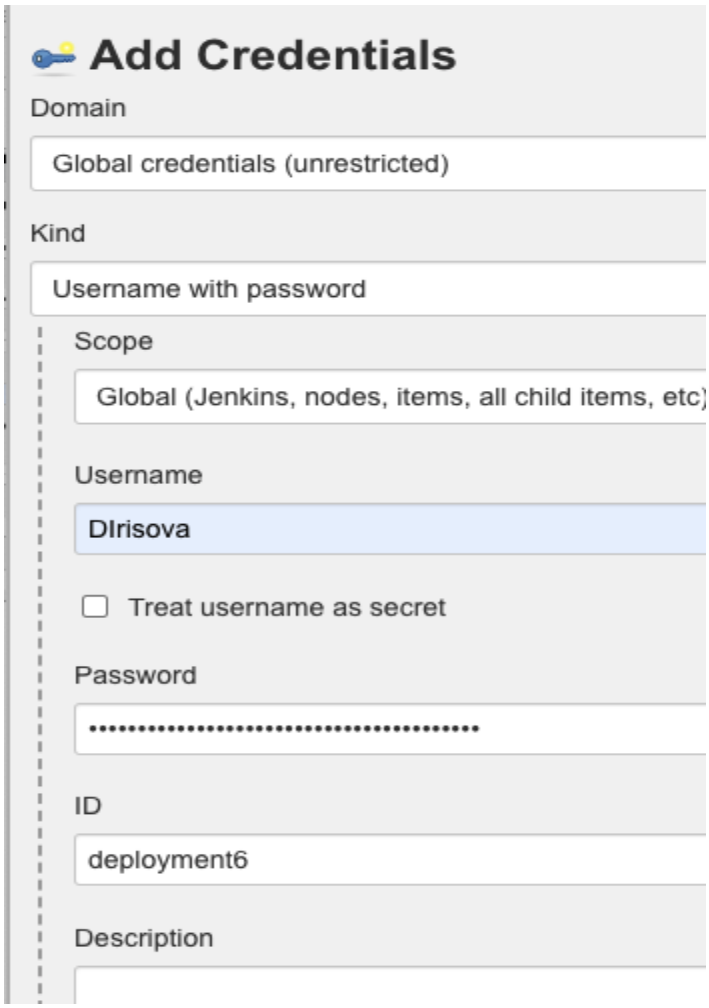
```
pipeline {
  agent {
      label 'agent-ubuntu1'
  }
  stages {
    stage ('Build') {
      steps {
      sh 'rm -rf ./kura_test_repo/cypress2'
      sh '''
        npm install
        npm run build
        sudo npm install -g serve
        serve -s build &
        '''
      }
    }
    stage ('Second') {
```

```
    stage ('Second') {
      agent {
        label 'agent-ubuntu2'
      }
      steps {
      sh '''
        npm install cypress
        npm install mocha
        npx cypress run --spec ./cypress/integration/test.spec.js
        '''
      }
      post {
        always {
          junit 'results/cypress-report.xml'
        }

      }
    }
  }
}
```

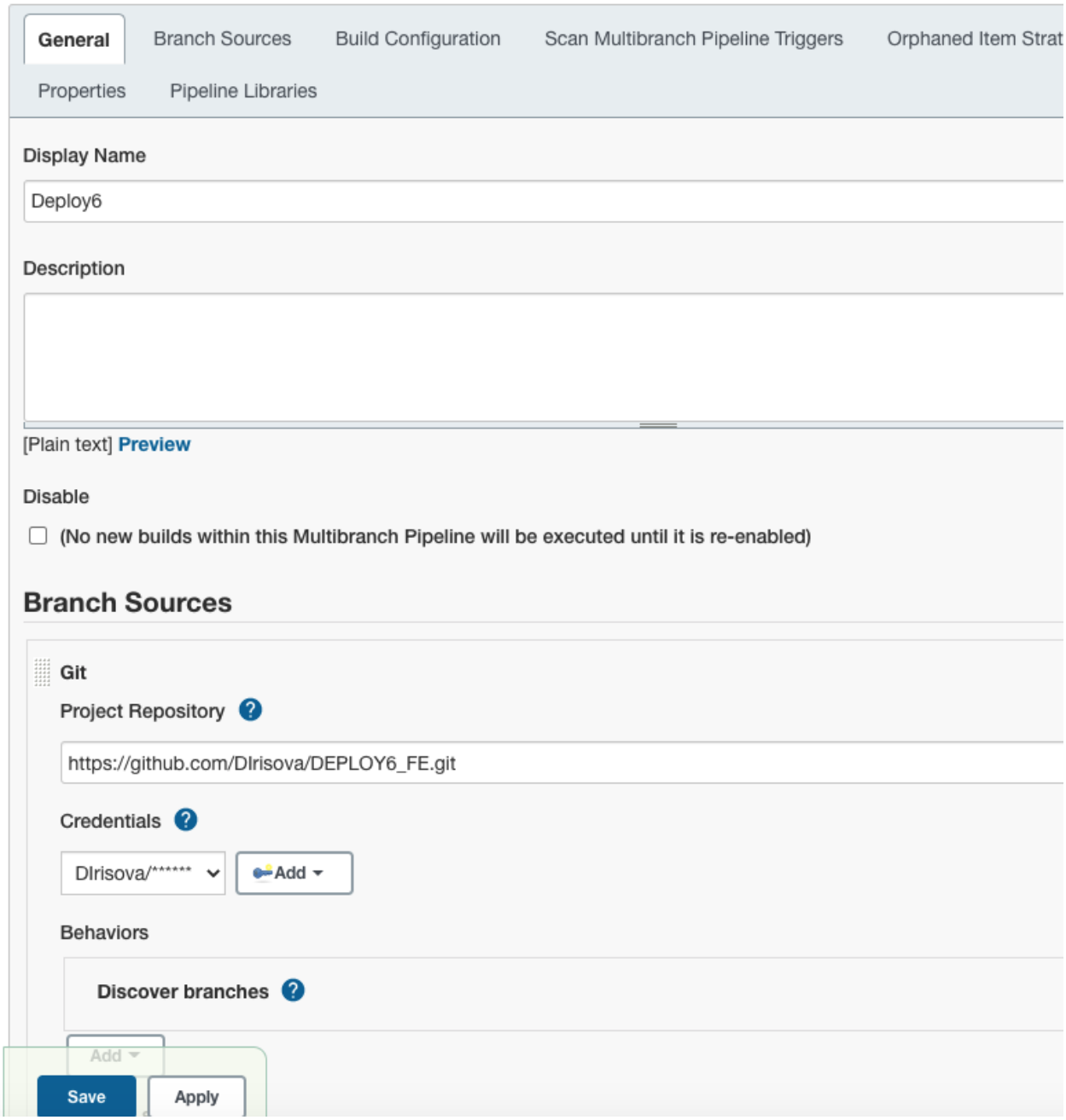
Go to the cypress/integration/test.spec.js and edit line 3, change the IP address to the private IP agent1

```
describe('Heading', () => {
    it('has the right title', () => {
        cy.visit('http://172.31.95.74:5000/example-1')

        cy.get('h1')
            .invoke('text')
            .should("equal", "My Awesome Web Application")
        Cypress.Screenshot.defaults({
        capture: 'runner',
        })
        cy.screenshot();
    });

});
```

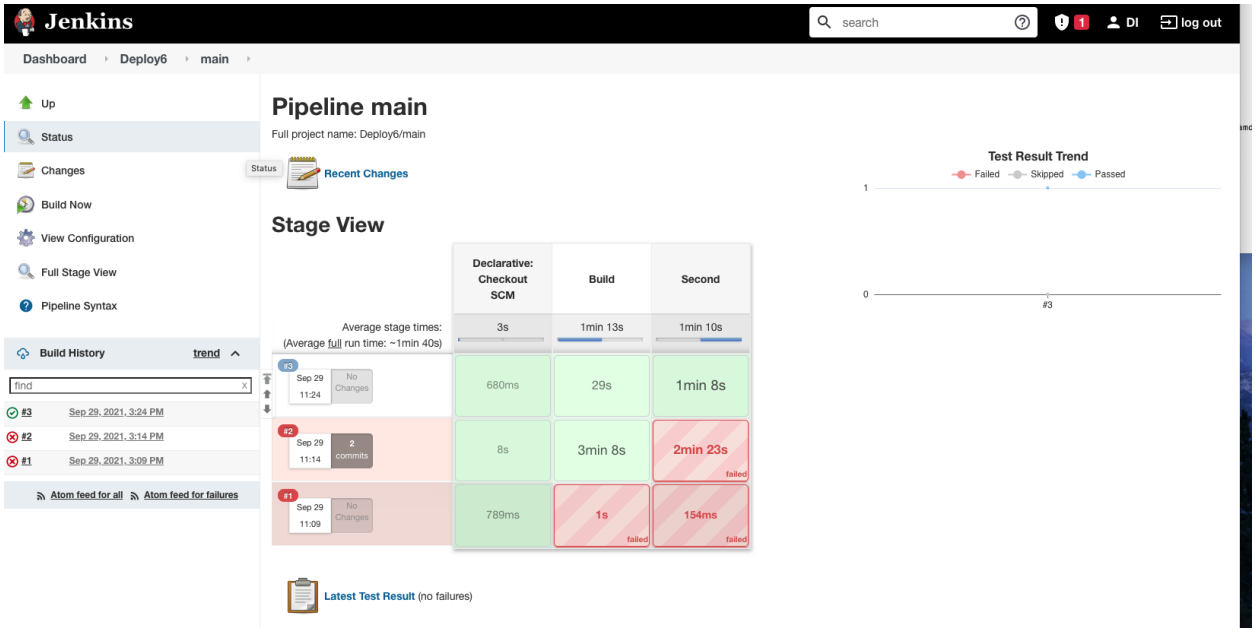
Go to the Jenkins to run the build
Create new item
Select Multibranch pipeline project -Deploy6

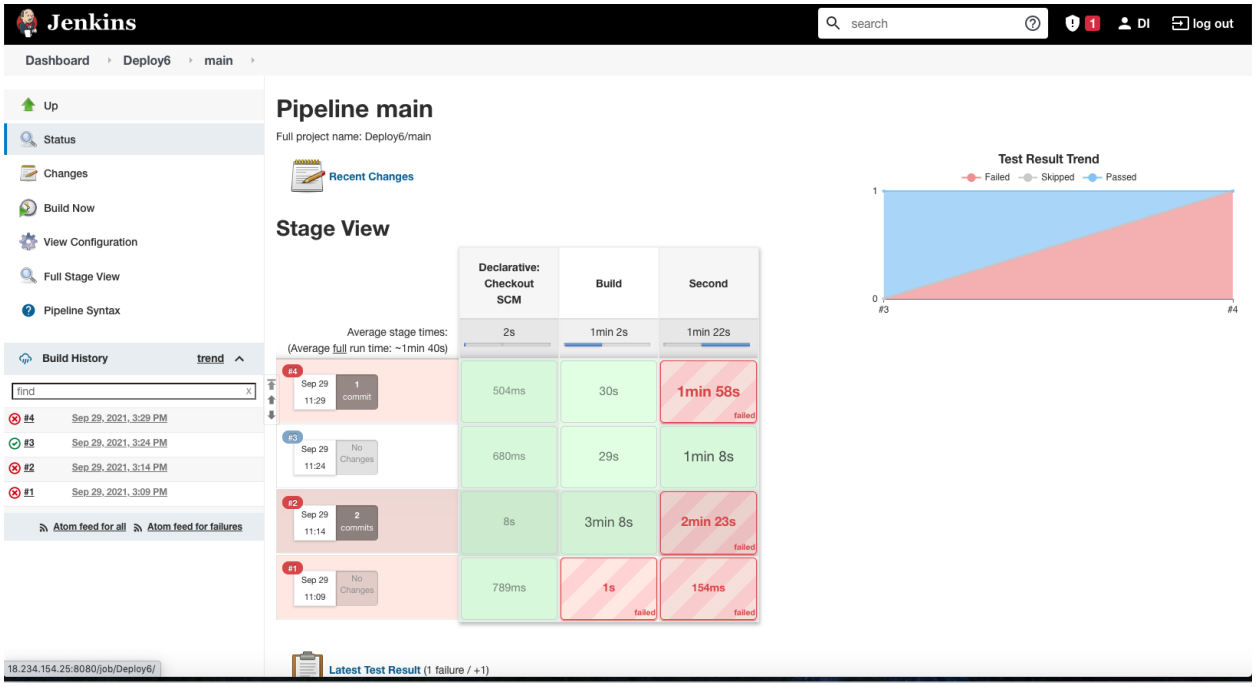


The username should be your GitHub username
For passwords, you can create new or use already existed your GitHub Personal Access Token
To create a new password go to the GitHub/Settings/Developer settings/Personal access tokens
Add
Select project repository
Select the credentials



Save
Finally, build and test





Next, we need to find the video that cypress created
Go to the terminal:

- find /home -name *.mp4

/home/ubuntu/jenkins/app/workspace/Deploy6_main/cypress/videos/test.spec.js.mp4

- cd /home/ubuntu/jenkins/app/workspace/Deploy6_main/cypress
- ls
- mv videos screenshots ~
- cd ~
- ls
- ssh-keygen
- ls
- cat video1.pub
- eval `ssh-agent -s`
- ssh-add video1.pub
- ssh-add video1
- git clone git@github.com:DIrisova/DEPLOY6_FE.git
- mv screenshots videos DEPLOY6_FE
- cd DEPLOY6_FE
- git status
- git add .
- git commit -m "add"
- git push origin main