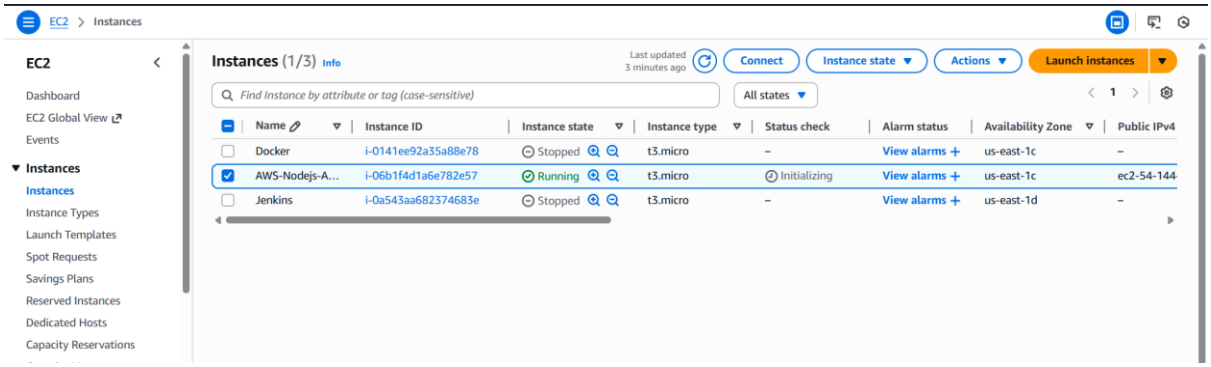
AWS-Nodejs-Application

STEP 1 — Launch EC2 for Application



Install Git

sudo apt install git -y

git –version

```
ubuntu@ip-172-31-26-4:~$ git --version
git version 2.43.0
ubuntu@ip-172-31-26-4:~$
```

Install npm

Sudo curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -

Install Node.js

sudo apt install nodejs -y

node -v

```
ubuntu@ip-172-31-26-4:~$ node -v
v18.19.1
ubuntu@ip-172-31-26-4:~$
```

sudo npm install -g pm2

```
ubuntu@ip-172-31-81-149:~$ sudo npm install -g pm2

added 133 packages in 8s

13 packages are looking for funding
  run `npm fund` for details
npm notice
npm notice New major version of npm available! 10.8.2 -> 11.10.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.10.0
npm notice To update run: npm install -g npm@11.10.0
npm notice
ubuntu@ip-172-31-81-149:~$
```

sudo apt install git

Verify :

node -v

npm -v

pm2 -v

STEP 2 — Install Jenkins Server

Install Java

sudo apt update

sudo apt install fontconfig openjdk-21-jre

java -version

Install Jenkins

sudo wget -O /etc/apt/keyrings/jenkins-keyring.asc \

https://pkg.jenkins.io/debian-stable/jenkins.io-2026.key

echo "deb [signed-by=/etc/apt/keyrings/jenkins-keyring.asc]" \

https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

/etc/apt/sources.list.d/jenkins.list > /dev/null

sudo apt update

sudo apt install Jenkins

sudo systemctl start jenkins

sudo systemctl enable Jenkins

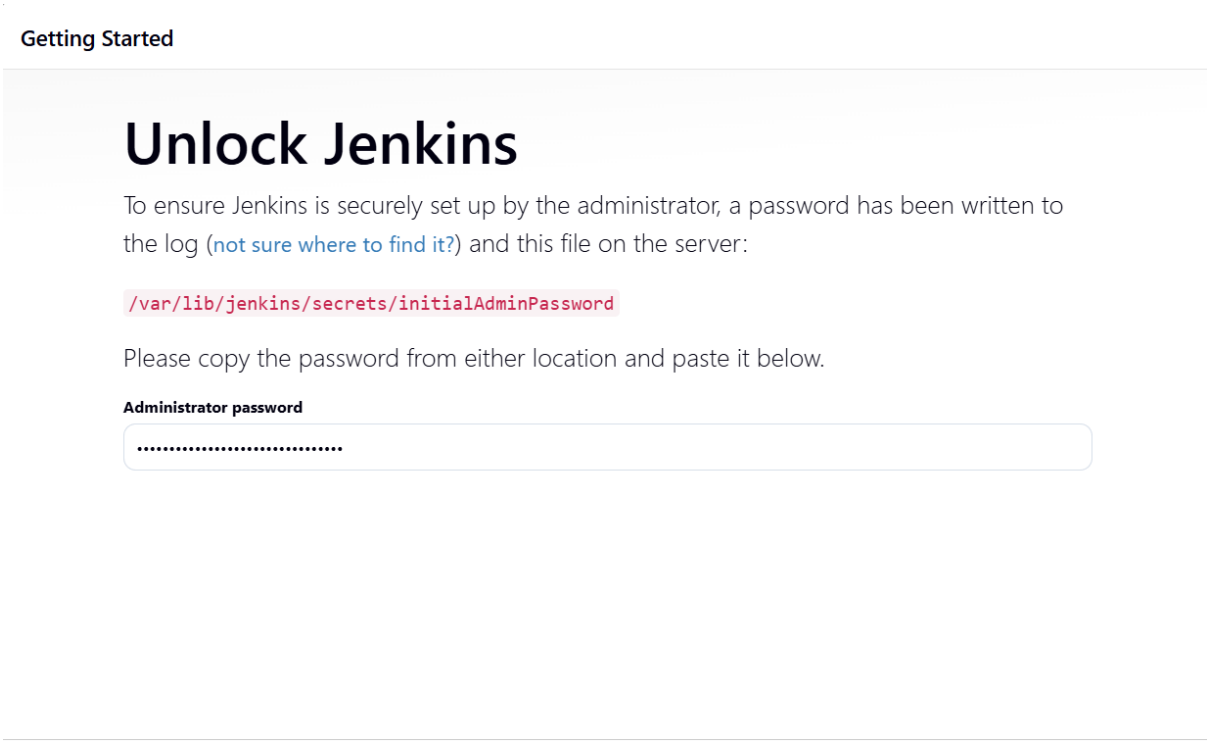
sudo systemctl status jenkins

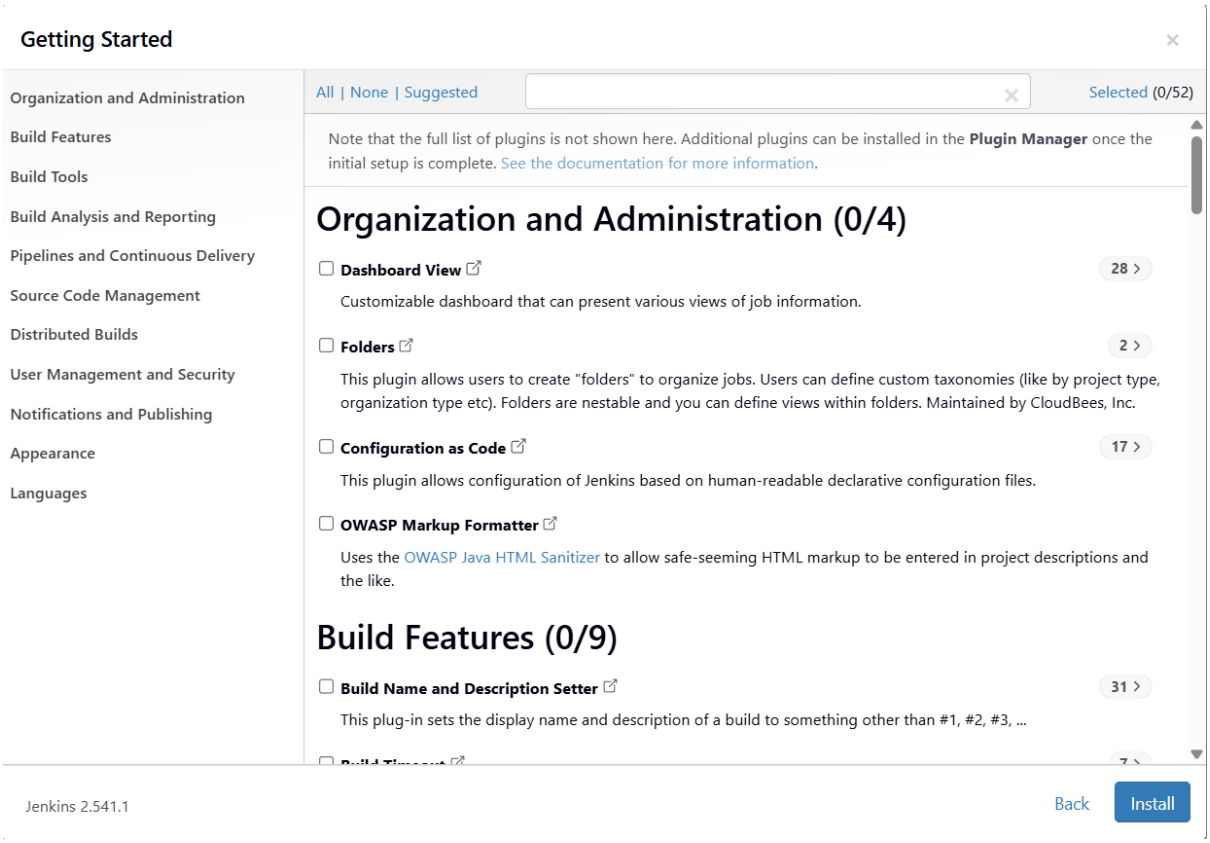
```
ubuntu@ip-172-31-26-4:~$ sudo systemctl start jenkins
sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
ubuntu@ip-172-31-26-4:~$ sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
ubuntu@ip-172-31-26-4:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
     Active: active (running) since Wed 2026-02-18 11:26:52 UTC; 11min ago
   Main PID: 3981 (java)
      Tasks: 37 (limit: 1008)
     Memory: 299.4M (peak: 306.2M)
        CPU: 20.792s
     CGroup: /system.slice/jenkins.service
             └─3981 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache
```

Open in browser:

Public ip - 3.86.138.82:8080

Unlock using: sudo cat /var/lib/jenkins/secrets/initialAdminPassword





Getting Started

Create First Admin User

Username

admin

Password

•••••

Confirm password

•••••

Full name

Arati Yadav

Jenkins 2.541.1

Skip and continue as admin Save and Continue

Getting Started

Instance Configuration

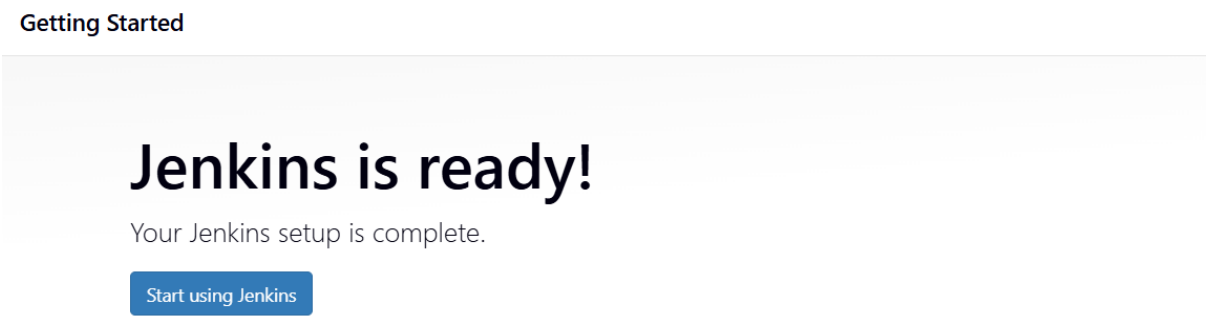
Jenkins URL: http://54.144.80.115:8080/

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 2.541.1

Not now Save and Finish

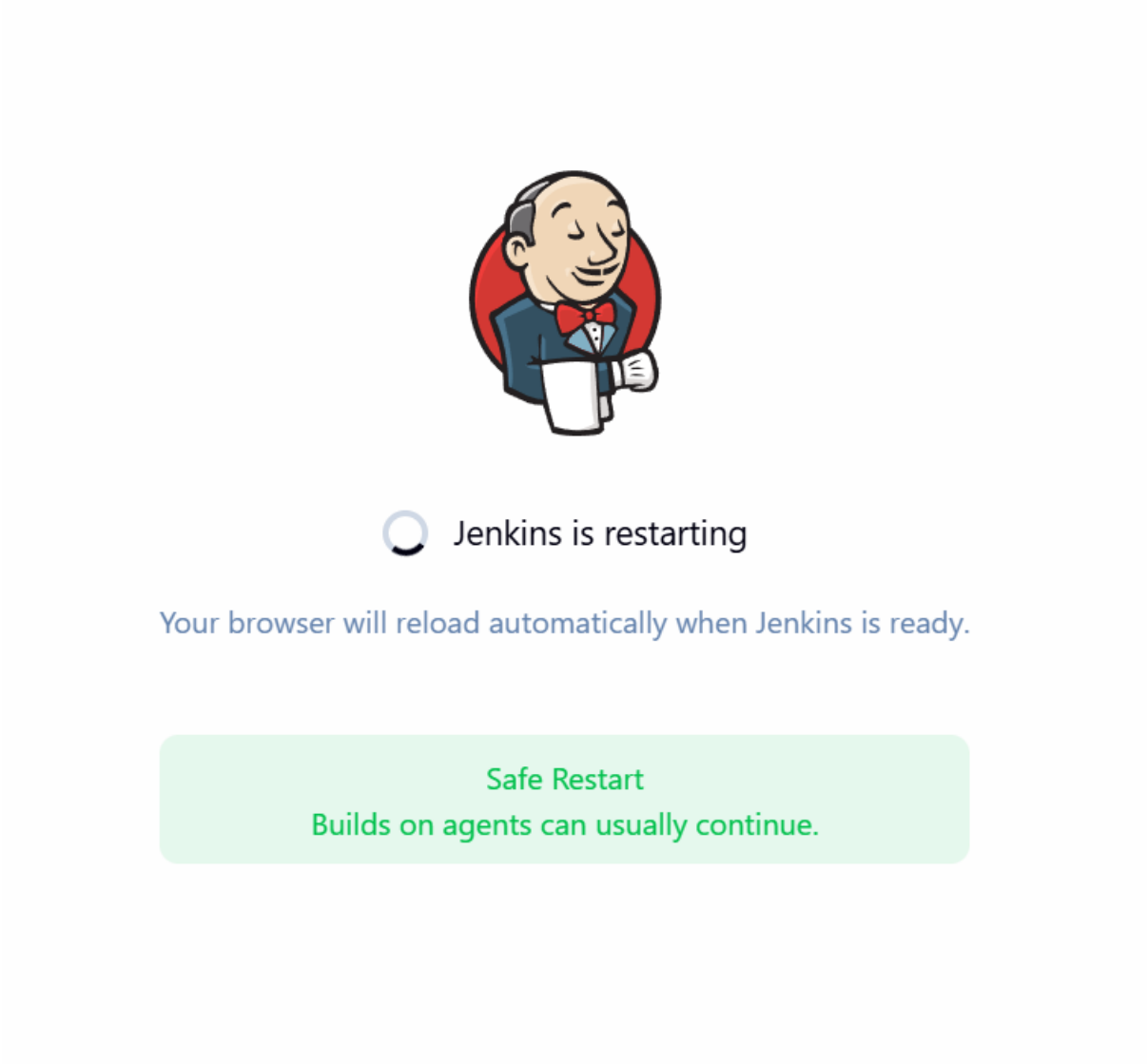


STEP 3 — Install Required Jenkins Plugins

Inside Jenkins → Manage Plugins:

Install:

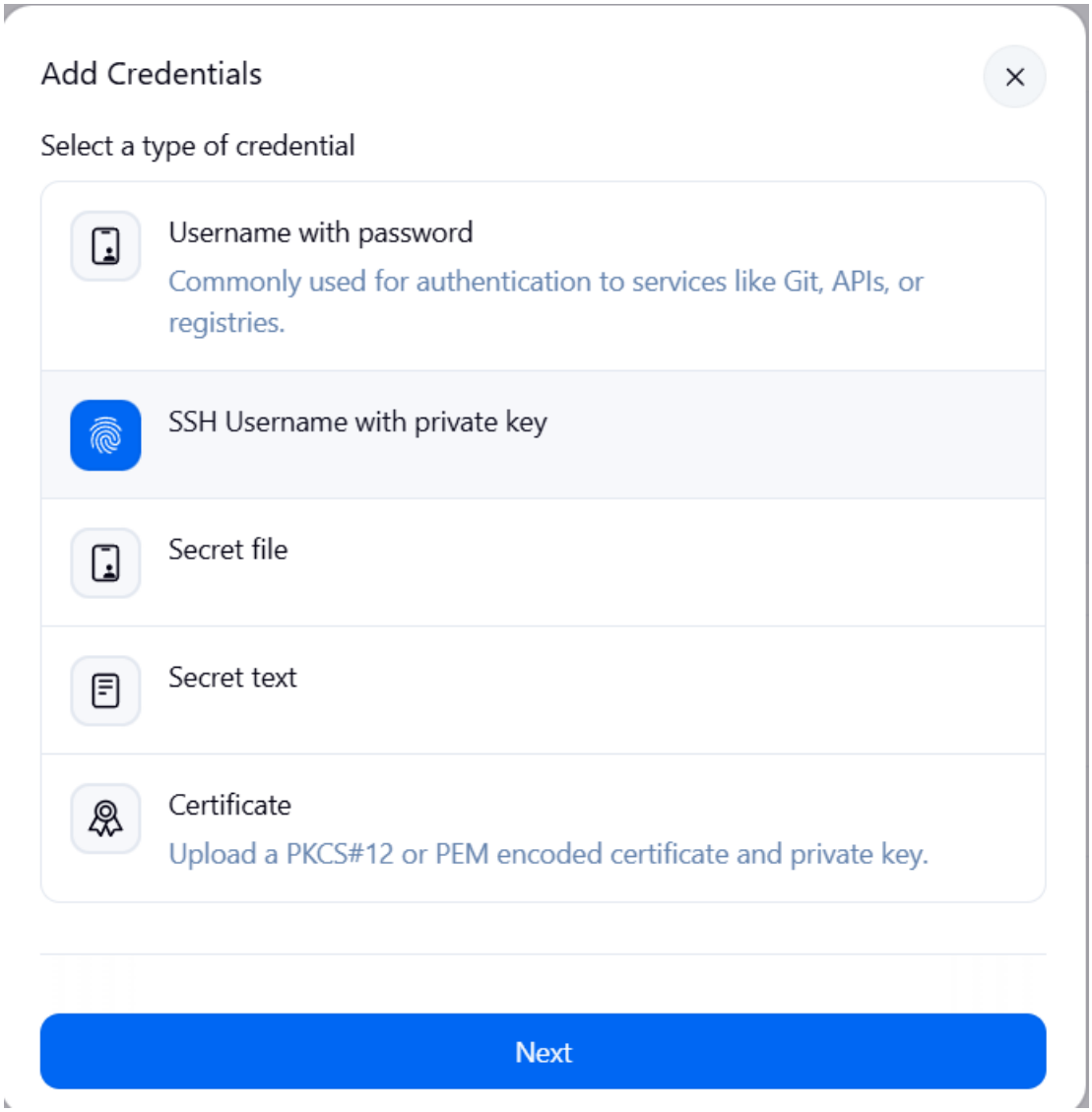
- Git Plugin
- Pipeline Plugin
- SSH Agent Plugin
- NodeJS Plugin (optional)

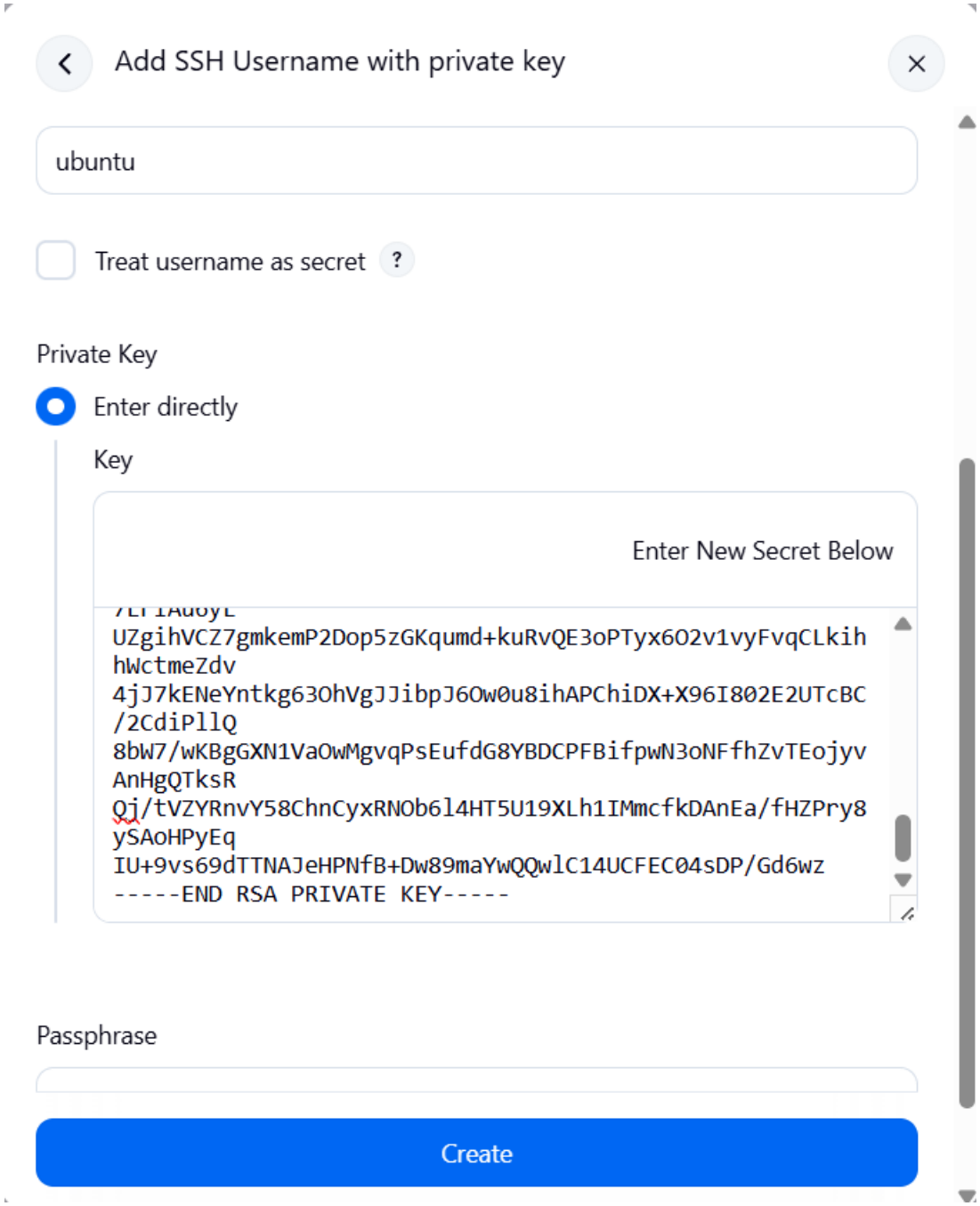


STEP 4 — Setup SSH Access from Jenkins to App EC2

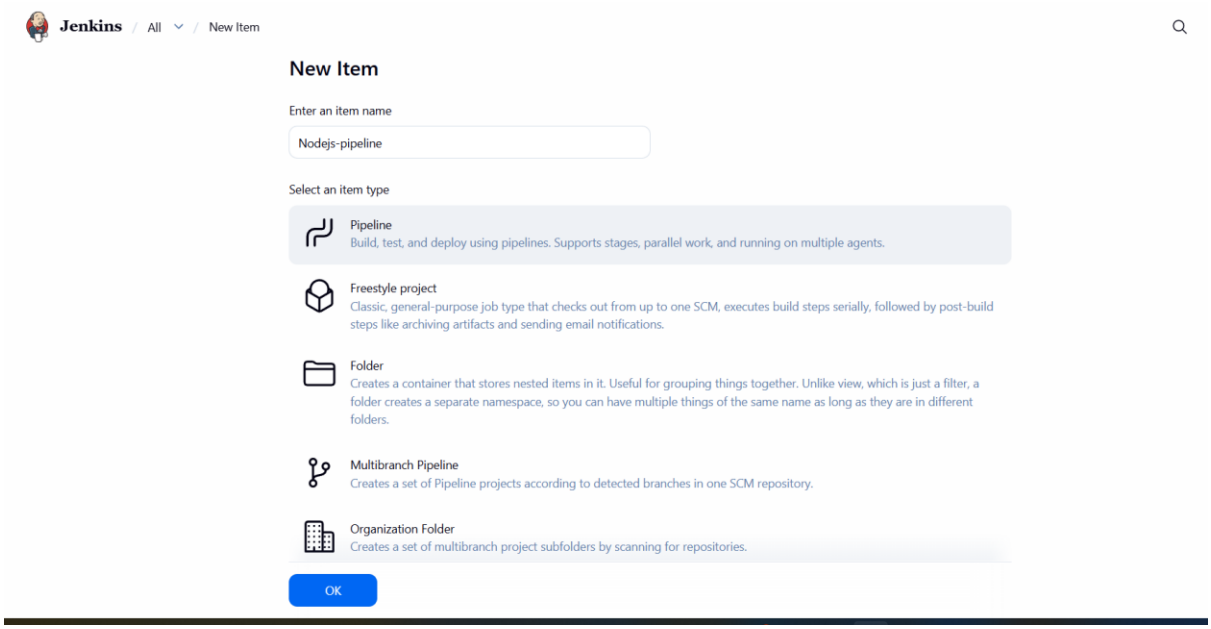
In Jenkins:

1. Go to **Manage Jenkins → Credentials**
2. Add Credentials:
    - Kind: SSH Username with private key
    - Username: ubuntu
    - Private Key
    - ID: ec2-key


Add Credentials
Select a type of credential
Username with password
Commonly used for authentication to services like Git, APIs, or registries.
SSH Username with private key
Secret file
Secret text
Certificate
Upload a PKCS#12 or PEM encoded certificate and private key.
Next



STEP 5 — Create Jenkins Pipeline job




```
pipeline {
    agent any

    environment {
        EC2_HOST = "3.86.138.82"
        EC2_USER = "ubuntu"
        APP_DIR = "/home/ubuntu/nodeapp"
    }

    stages {

        stage('Clone Code') {
            steps {
                git branch: 'main',
                    url: 'https://github.com/yourusername/your-repo.git'
            }
        }

        stage('Install Dependencies') {
```

```
    steps {
        sh 'npm install'
    }
}

stage('Build') {
    steps {
        sh 'echo "Build Step (if needed)"'
    }
}

stage('Deploy to EC2') {
    steps {
        sshagent(['ec2-key']) {
            sh """
            ssh -o StrictHostKeyChecking=no $EC2_USER@$EC2_HOST '
                mkdir -p $APP_DIR
            '

            scp -r * $EC2_USER@$EC2_HOST:$APP_DIR

            ssh $EC2_USER@$EC2_HOST '
                cd $APP_DIR
                npm install
                pm2 delete nodeapp || true
                pm2 start app.js --name nodeapp
                pm2 save
            '
            """
        }
```

```
        }
      }
    }
}
```

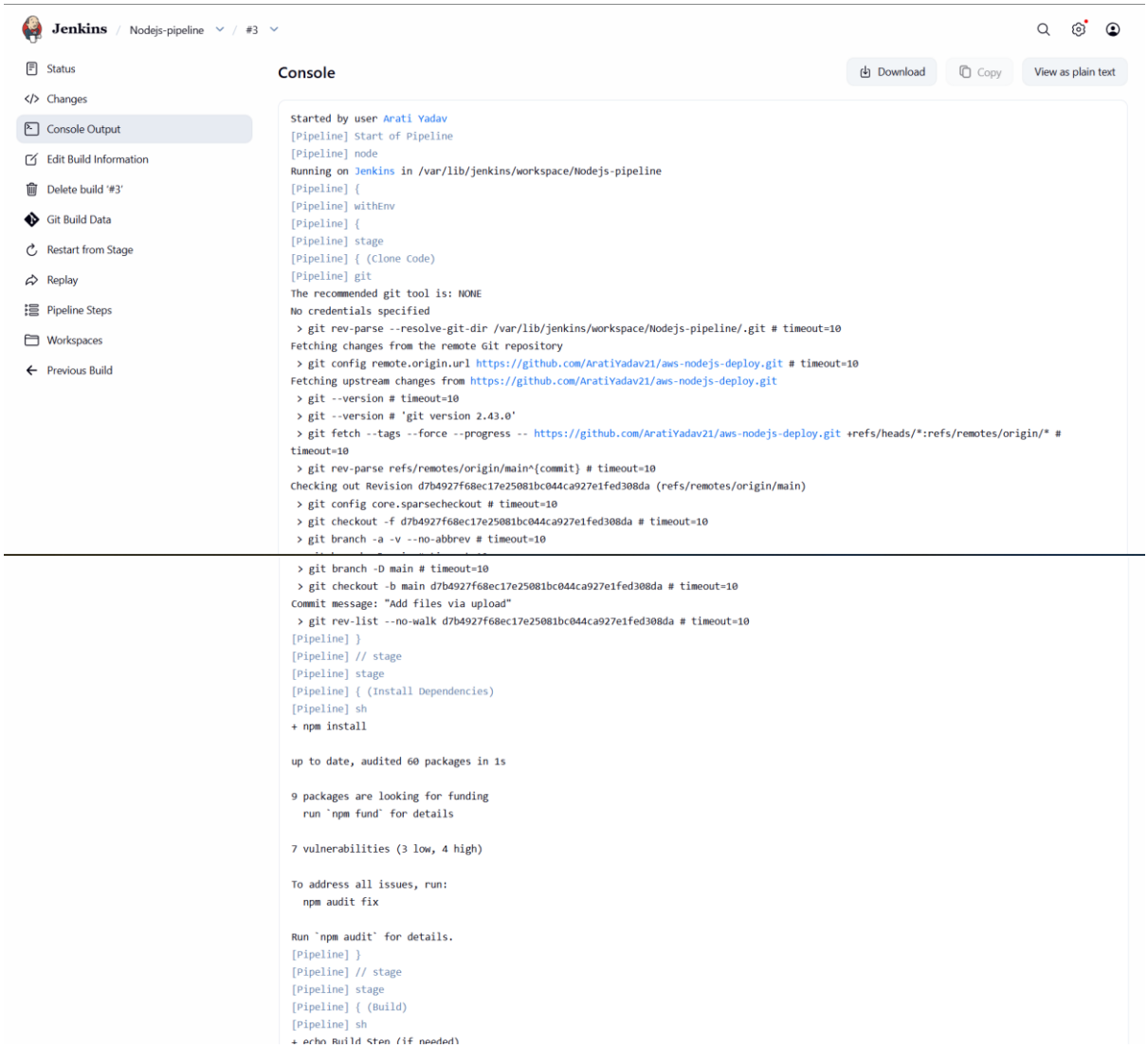
Step 6: Configure GitHub Webhook (Auto Deployment)

In GitHub:

1. Repo → Settings → Webhooks
2. Add webhook:
3. http:// 3.86.138.82:8080/github-webhook/
4. Choose: Push events

Output:

Job Successfully Build.



```
Build Step (if needed)
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy to EC2)
[Pipeline] sshagent
[ssh-agent] Using credentials ubuntu
$ ssh-agent
SSH_AUTH_SOCK=/tmp/ssh-sklqLCkWli3v/agent.6554
SSH_AGENT_PID=6557
Running ssh-add (command line suppressed)
Identity added: /var/lib/jenkins/workspace/Nodejs-pipeline@tmp/private_key_16323703221816698274.key (/var/lib/jenkins/workspace/Nodejs-
pipeline@tmp/private_key_16323703221816698274.key)
[ssh-agent] Started.
[Pipeline] {
[Pipeline] sh
+ ssh -o StrictHostKeyChecking=no ubuntu@3.86.138.82
                        mkdir -p /home/ubuntu/nodeapp

Warning: Permanently added '3.86.138.82' (ED25519) to the list of known hosts.
+ scp -r README.md node_modules package-lock.json package.json server.js ubuntu@3.86.138.82:/home/ubuntu/nodeapp
+ ssh ubuntu@3.86.138.82
                        cd /home/ubuntu/nodeapp
                        npm install
                        pm2 delete nodeapp || true
                        pm2 start app.js --name nodeapp
                        pm2 save
```



```
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.10.0
npm notice To update run: npm install -g npm@11.10.0
npm notice
[PM2][ERROR] Process or Namespace nodeapp not found
[PM2][ERROR] Script not found: /home/ubuntu/nodeapp/app.js
[PM2] Saving current process list...
[PM2][WARN] PM2 is not managing any process, skipping save...
[PM2][WARN] To force saving use: pm2 save --force
[Pipeline] }
$ ssh-agent -k
unset SSH_AUTH_SOCK;
unset SSH_AGENT_PID;
echo Agent pid 6557 killed;
[ssh-agent] Stopped.
[Pipeline] // sshagent
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```