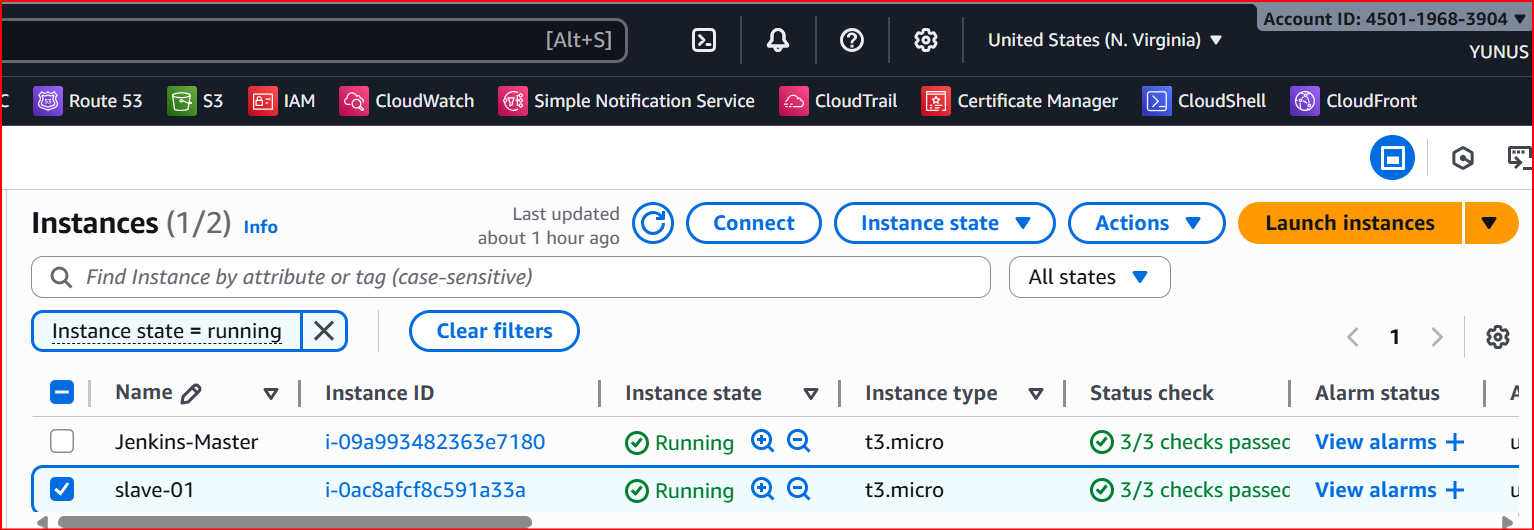
**Jenkins-Task-2**

**1) Configure 2 slave machines in Jenkins master.  
2) Configure webhooks to Jenkins job.  
3) Configure poll scm and build periodical options in Jenkins job.  
4) Take backup of Jenkins server by using bash script.  
5) Take backup of Jenkins using rethin backup plugin.  
6) Setup a new Jenkins server and dump the backup taken in task4**

**1.Setup two Instances Master and Slave**

****

**In Slave (Connect SSH)**

**Download the java**

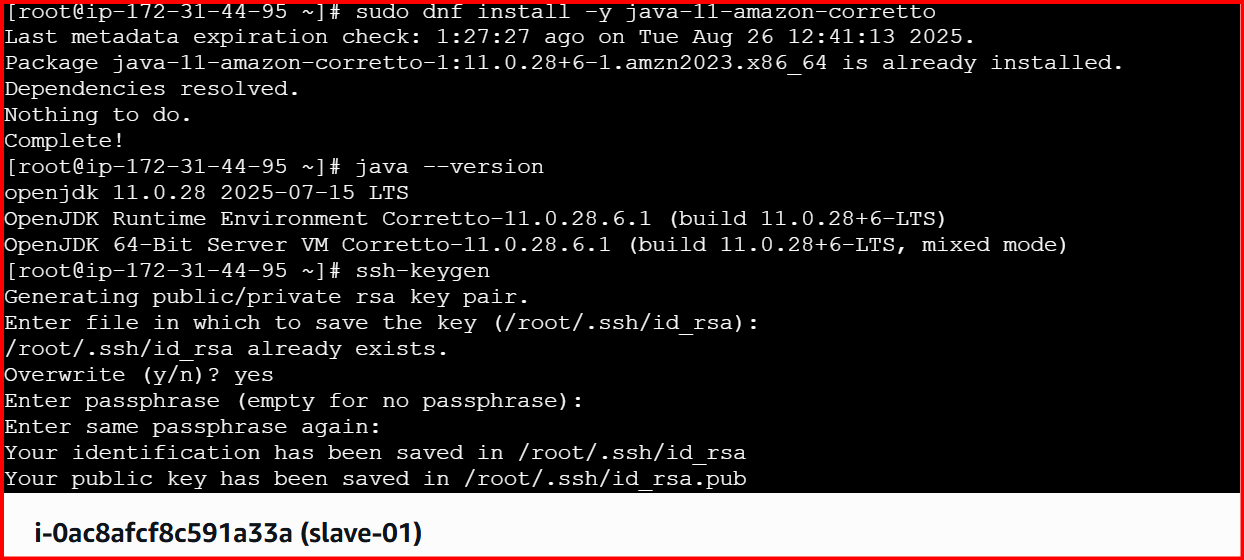
**Java –version**

**SSh-keygen**

**cd.ssh/**

**ls, cat id\_rsa.pub > authorized\_keys**

**chmod 700 authorized\_keys**

****

**Master: cd /var/lib/Jenkins**

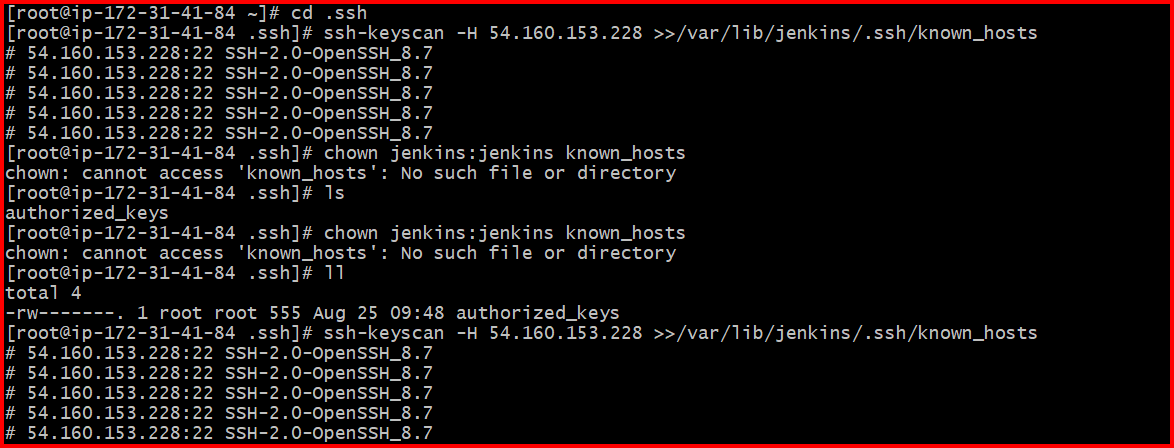
**Mkdir .ssh**

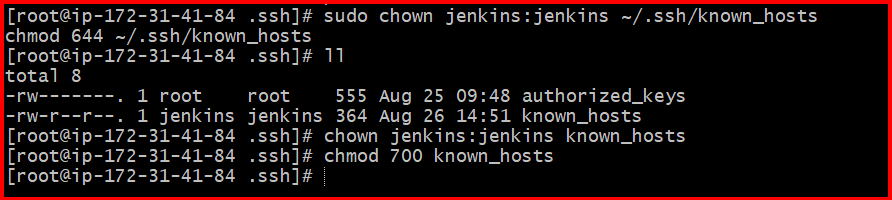
**Cd .ssh**

**ssh-keyscan -H 54.160.153.228 >>/var/lib/jenkins/.ssh/known\_hosts (Slave IP)**

**chown jenkins:jenkins known\_hosts**

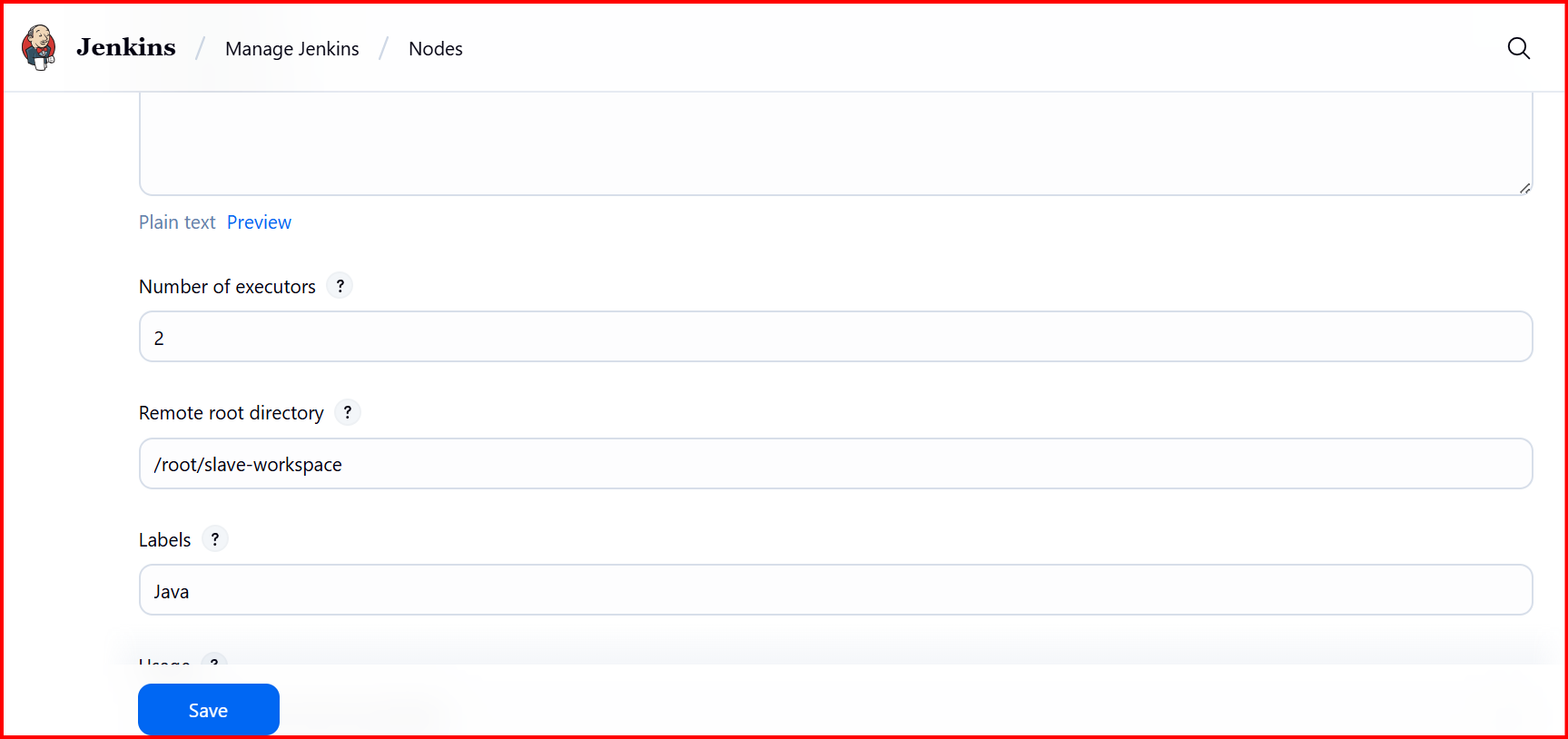
**chmod 700 known\_hosts**

****

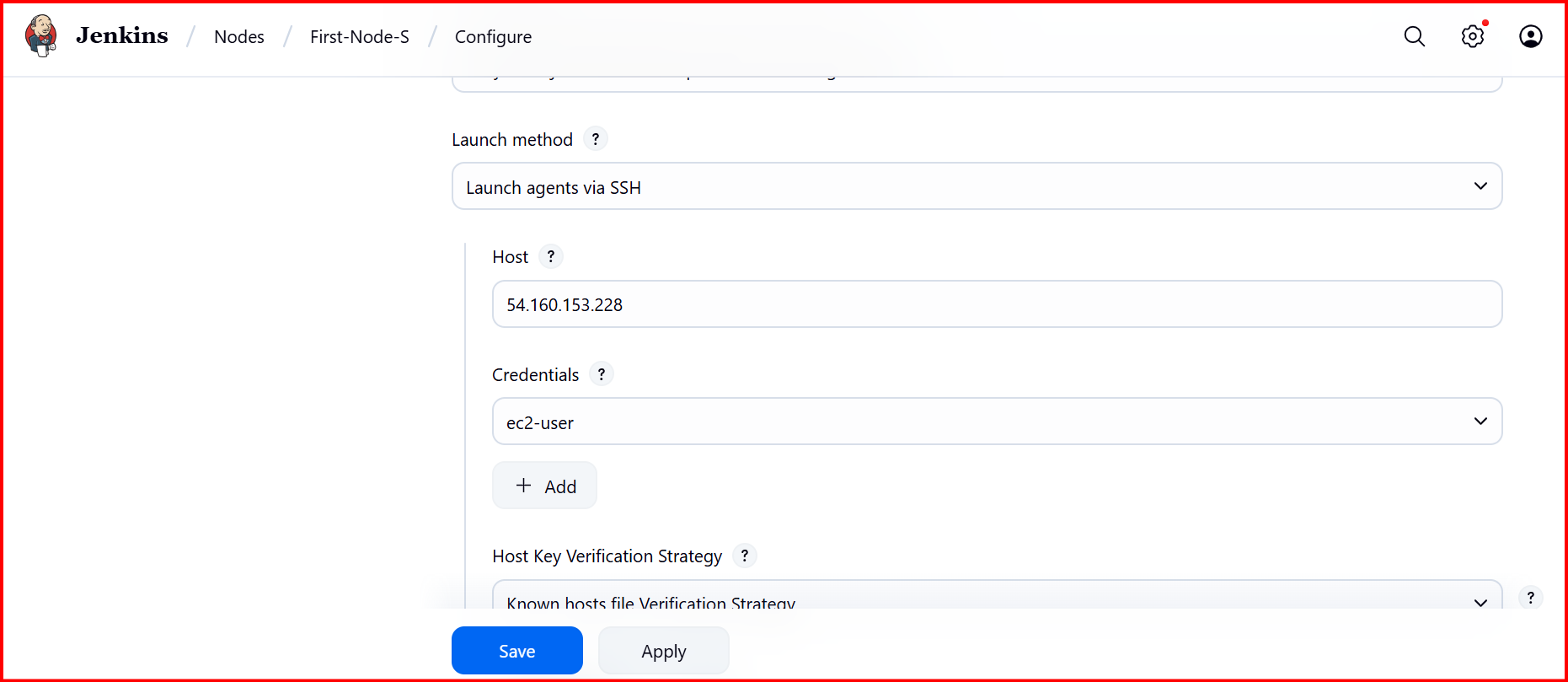
****

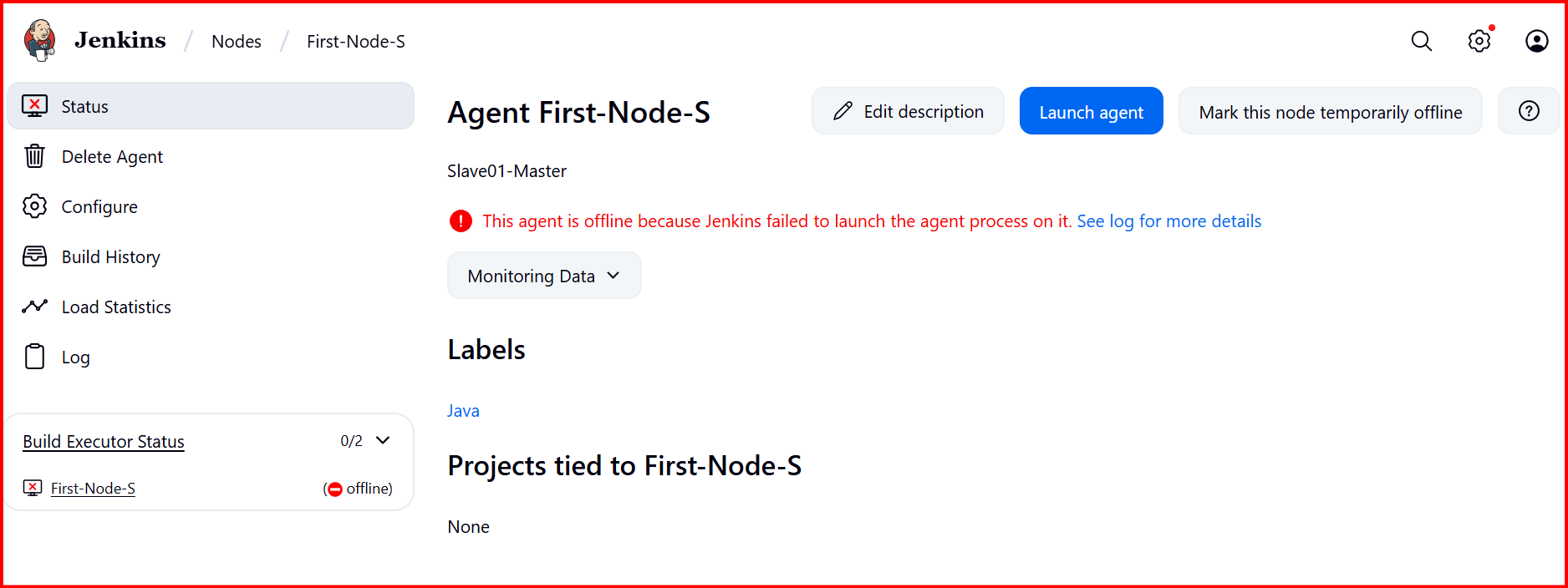
**Go to Jenkins**

**Management Jenkins 🡪 Nodes🡪 Create Node**

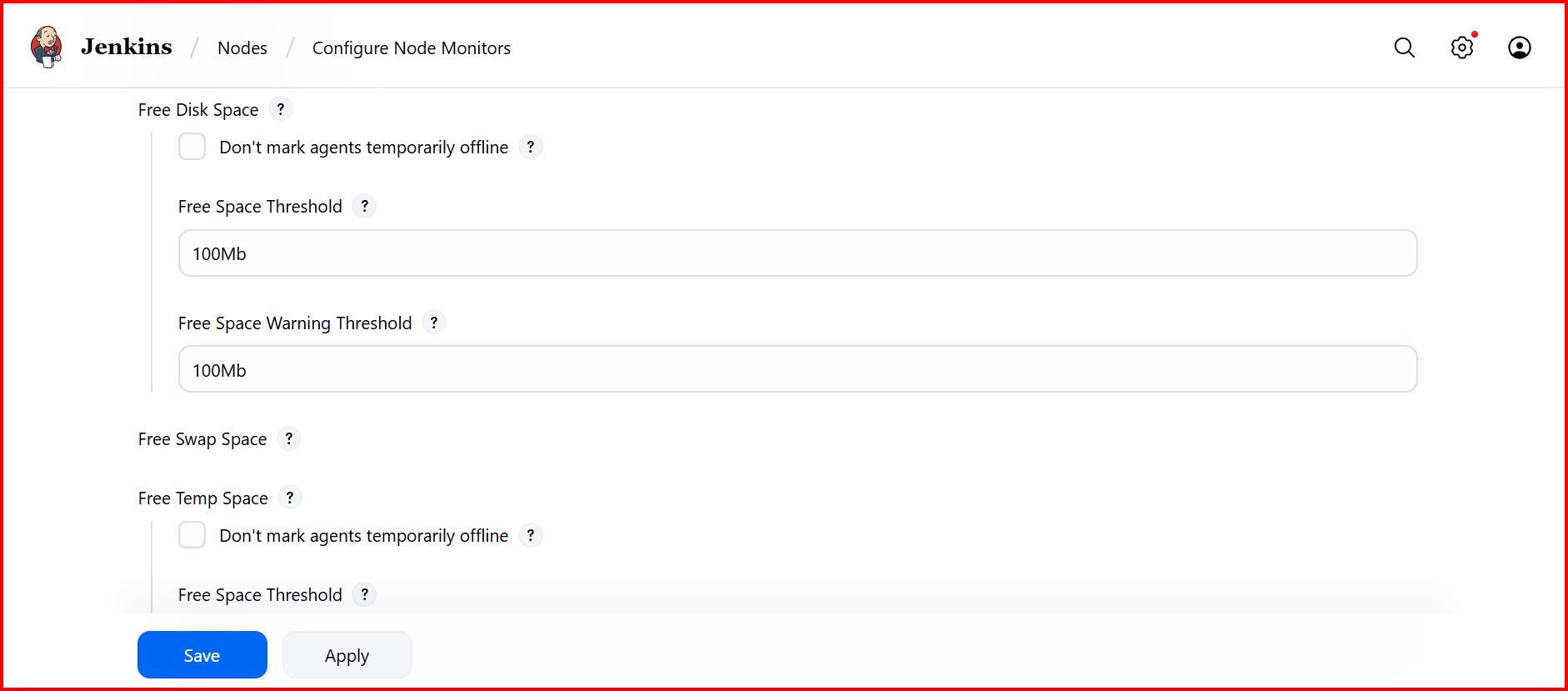
****

**Here Slave IP (Host) and add credentials by using Pem key**

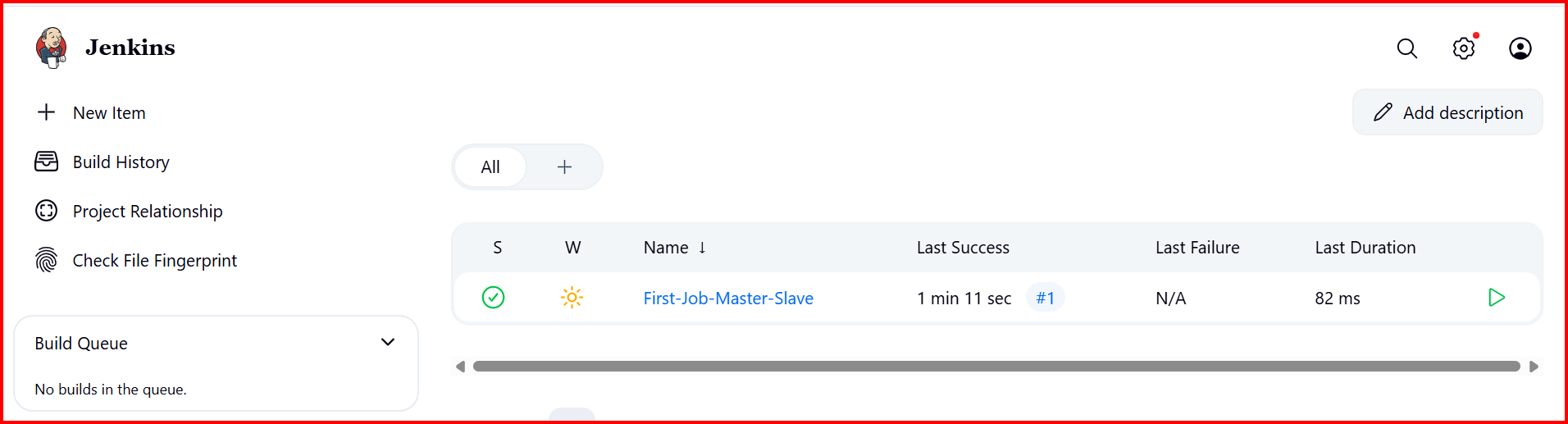
****

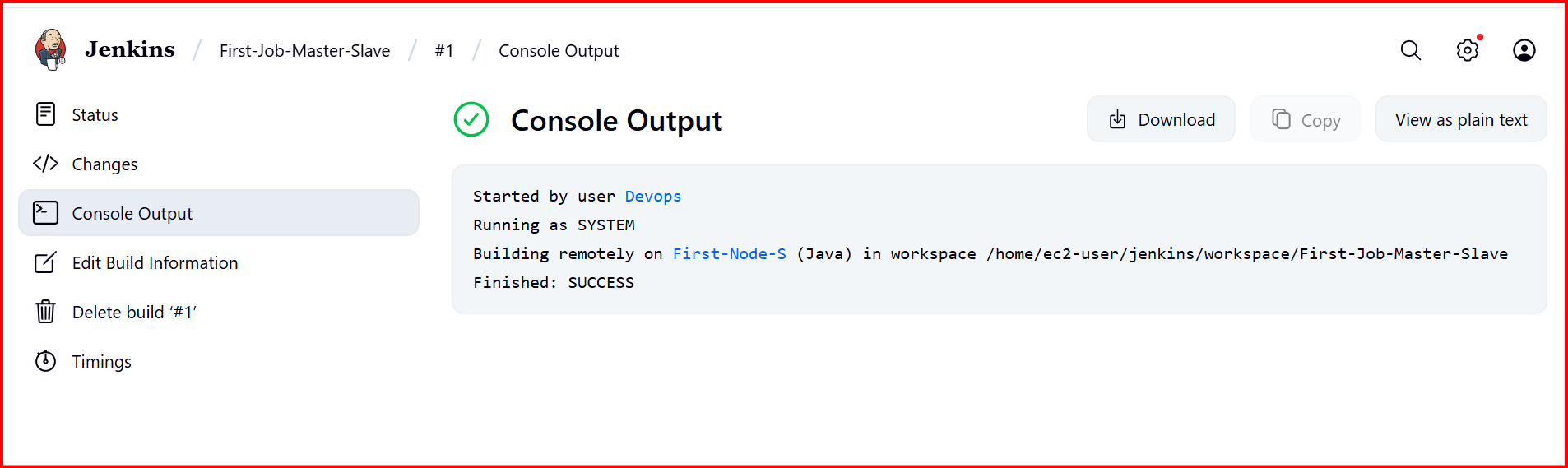
****

**Error (make all100Mb)**

****

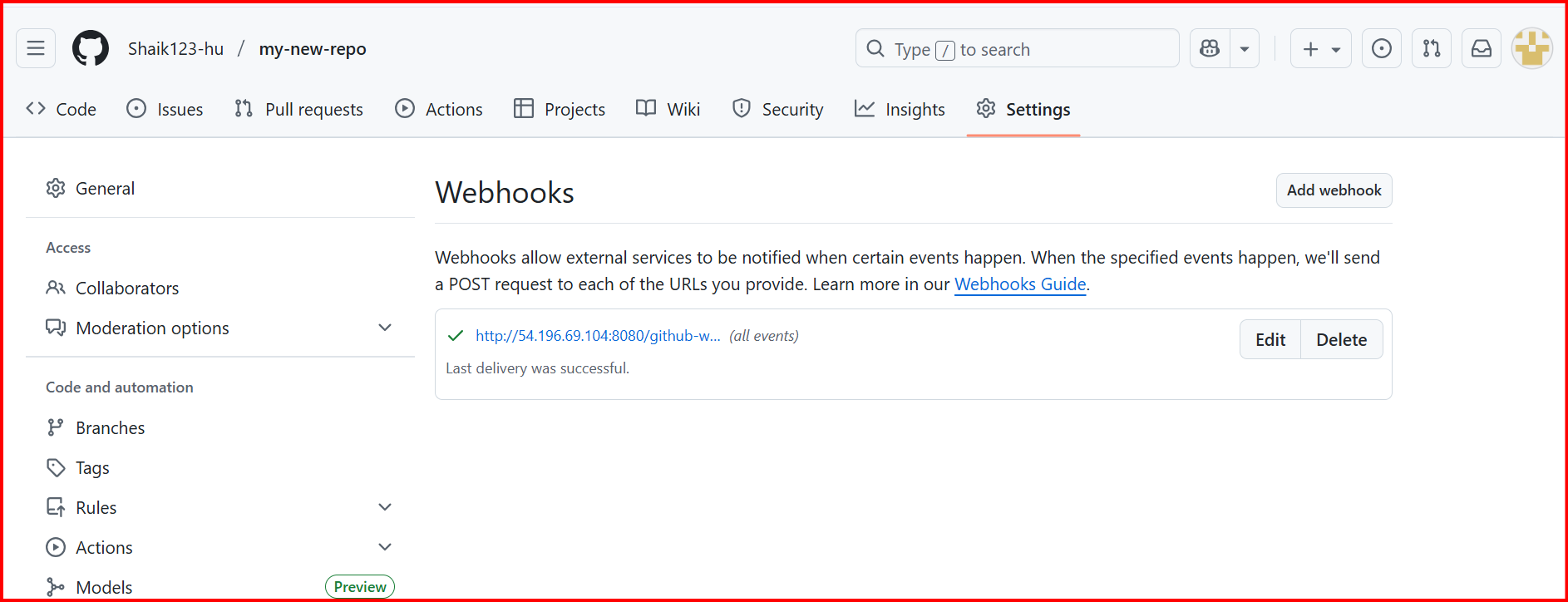
**Create a job and call Java in labels**

****

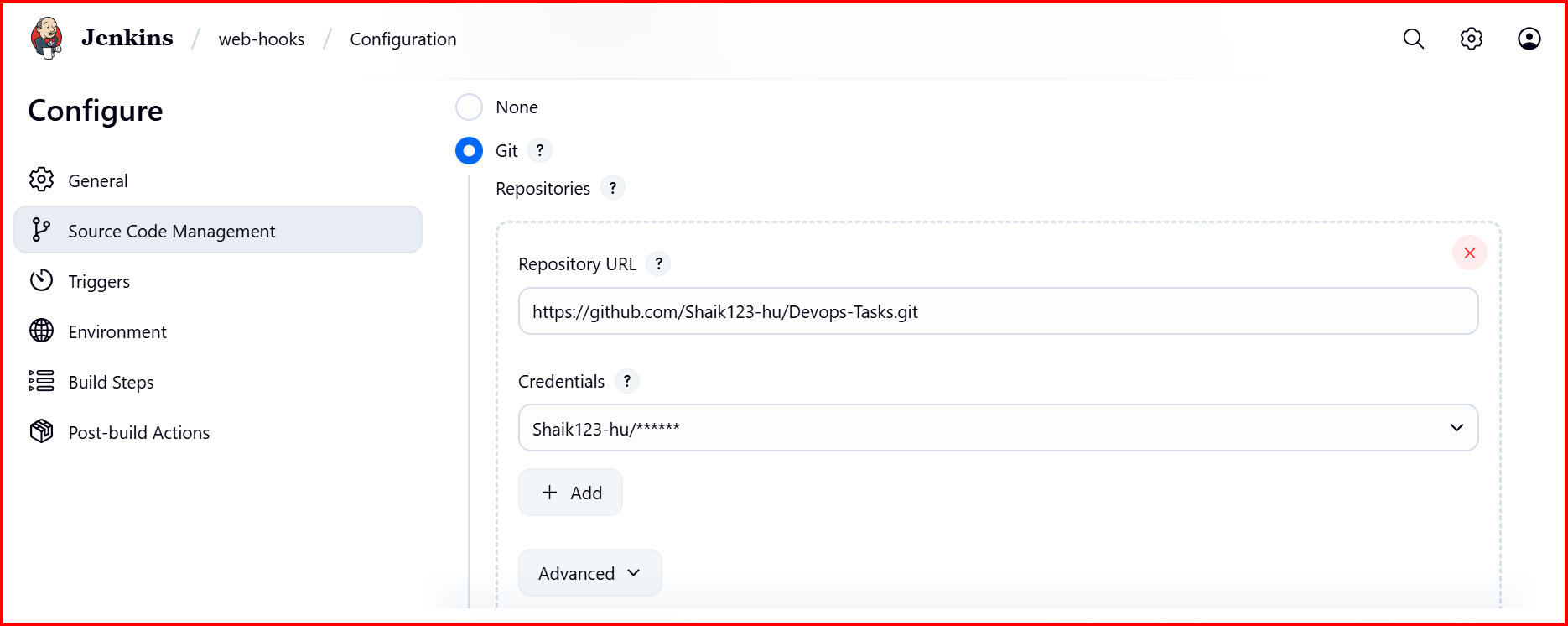
****

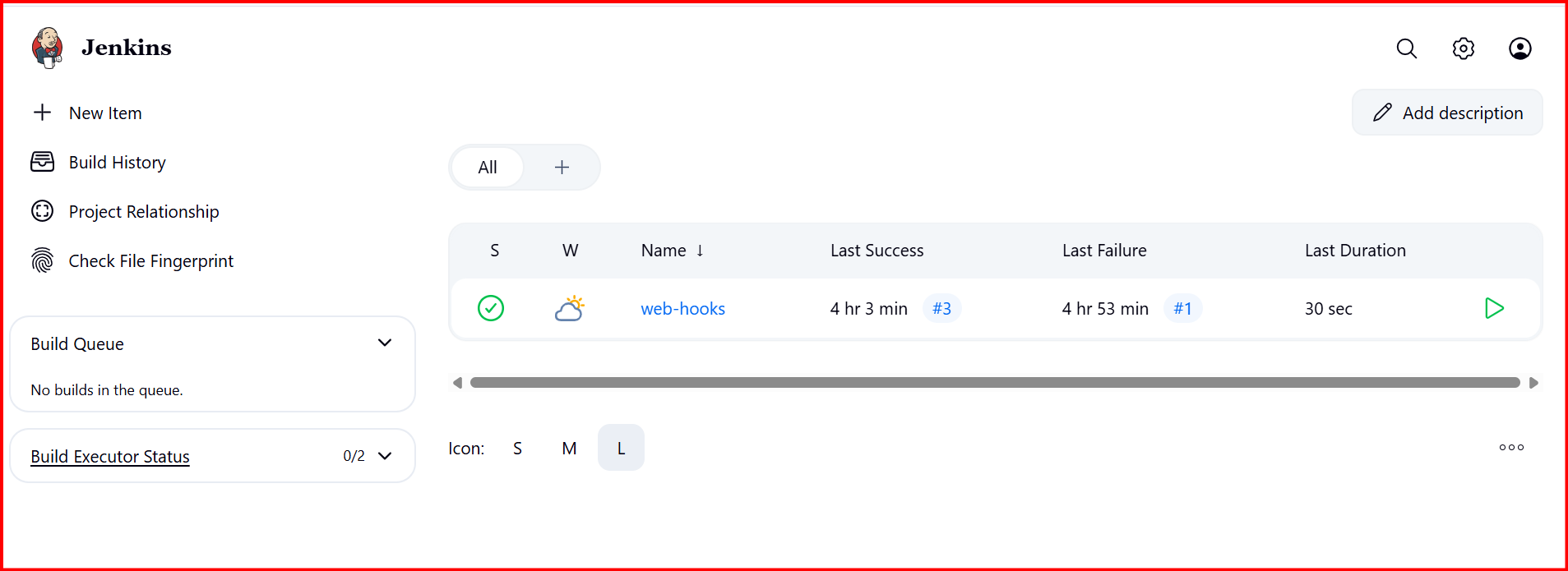
**2.Configure webhooks to Jenkins job.**

First, we need to configure the webhooks in git

****

**Create a job and configure the git url**

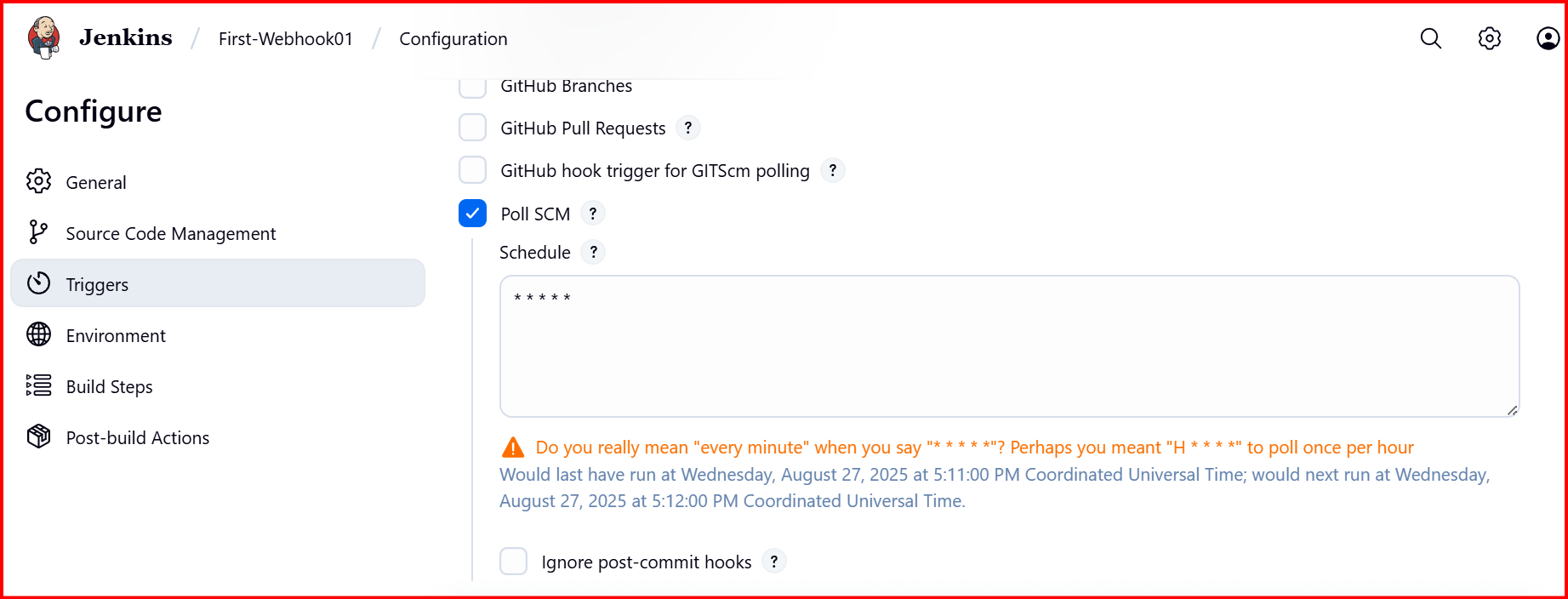
****

****

**3.Configure poll scm and build periodical options in Jenkins job**

Poll scm will check if anything changes in git repo then only execute

According to cronjobs







**4.Take backup of Jenkins server by using bash script**

**Create a backup script**

#!/bin/bash

# Define Jenkins home and backup directories

JENKINS\_HOME="/var/lib/jenkins"

BACKUP\_DIR="jenkins-backups"

BACKUP\_FILE="$BACKUP\_DIR/backup1.tar.gz"

# Create the backup directory if it doesn't exist

mkdir -p "$BACKUP\_DIR"

# Create the compressed tarball of the Jenkins home directory

echo "Creating backup of $JENKINS\_HOME..."

sudo tar -czf "$BACKUP\_FILE" -C "$JENKINS\_HOME" .

# Verify the backup file was created

if [ -f "$BACKUP\_FILE" ]; then

echo "Backup successful! File saved to $BACKUP\_FILE"

else

echo "Backup failed!"

exit 1

fi

# Set permissions for the backup file

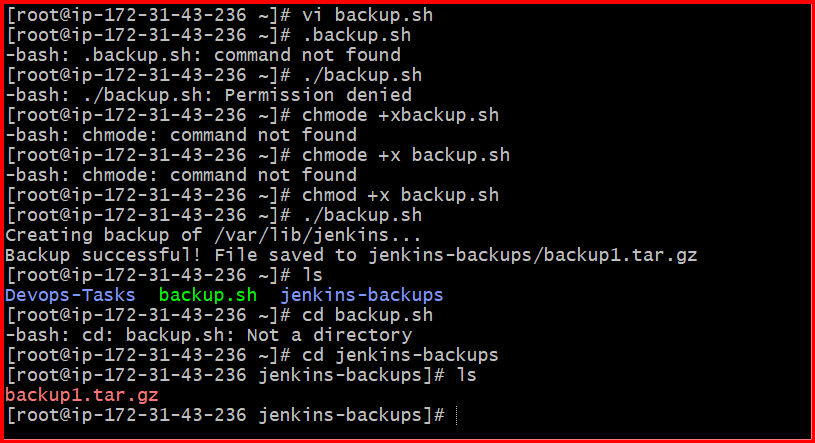
sudo chown -R jenkins:jenkins "$BACKUP\_DIR"

C

Create Vi backup.sh

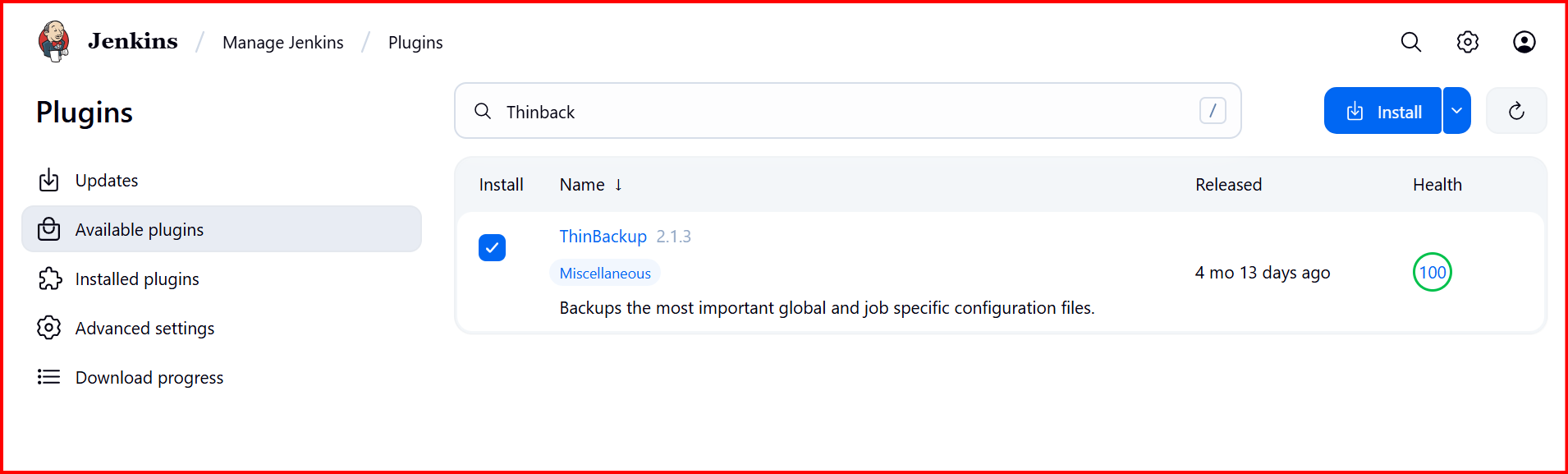
Give permissions chmod +x backup.sh

Execute: ./backup.sh

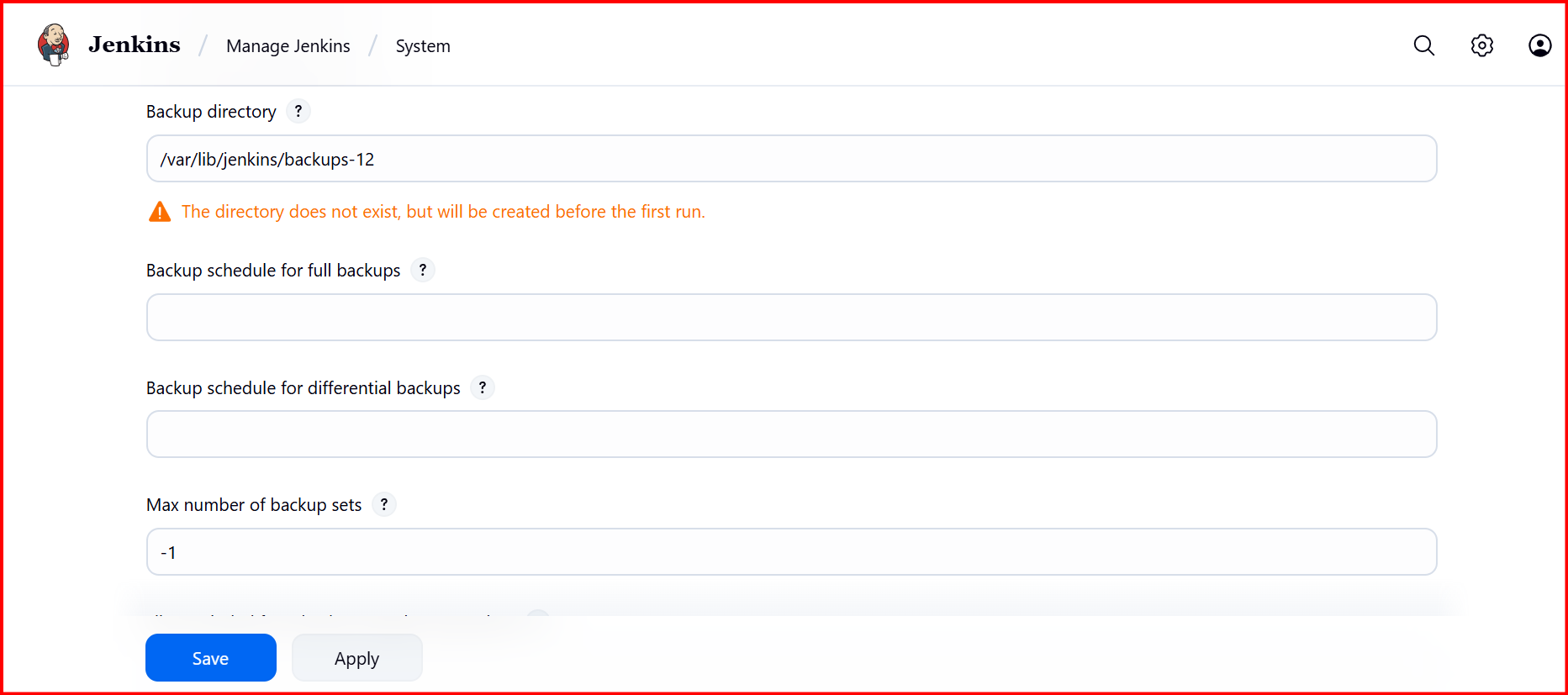


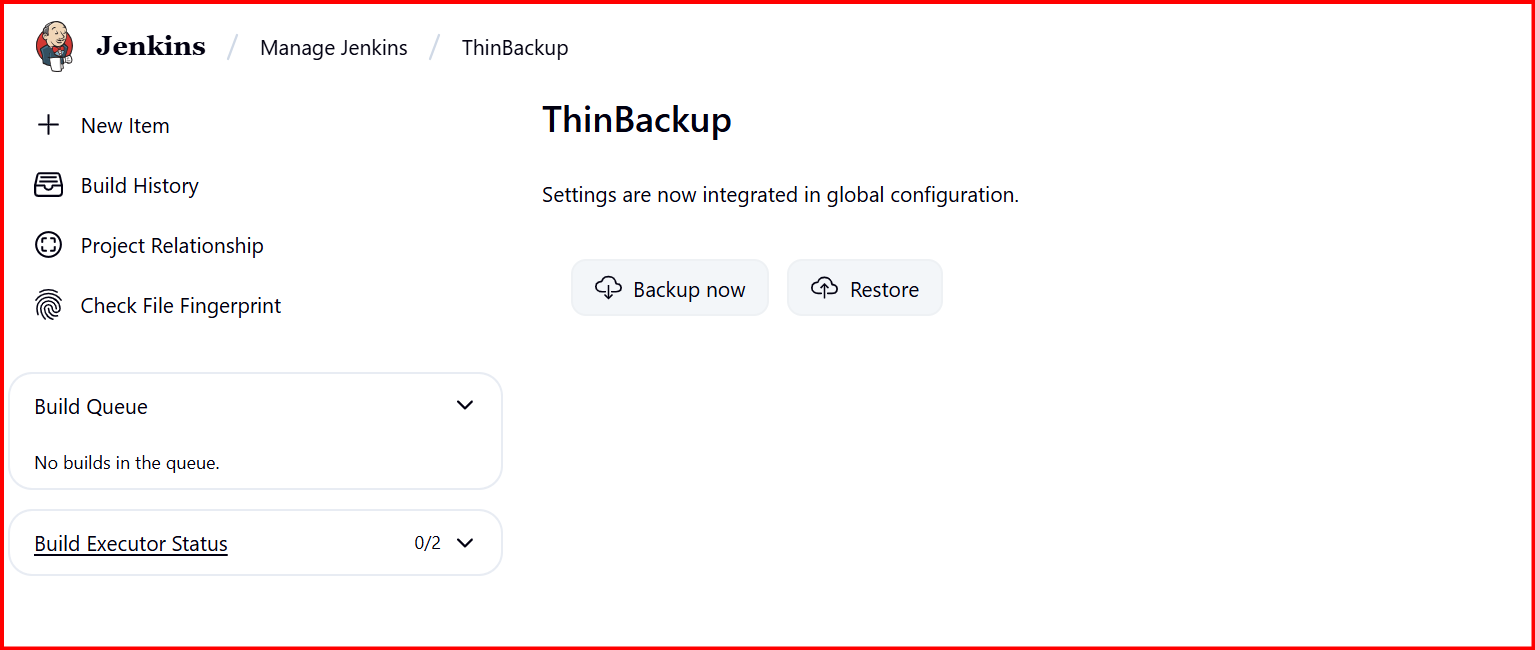
5. **Take backup of Jenkins using rethin backup plugin**

**Install the ThinBackup Plugin**

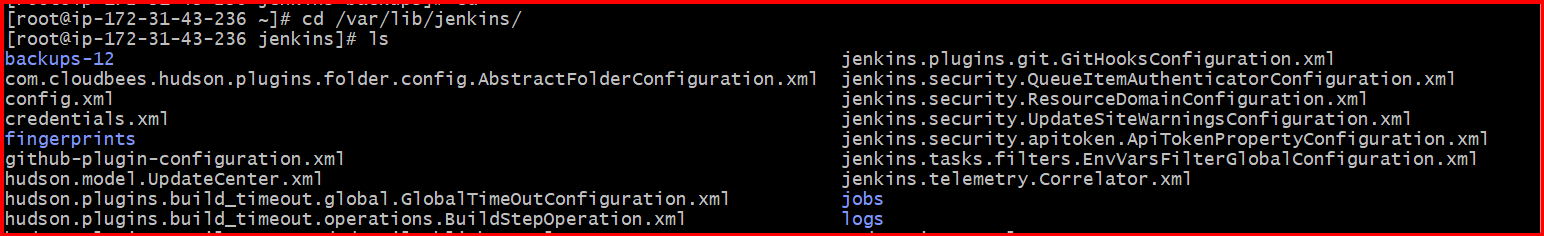


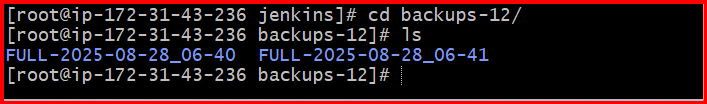
In Manage Jenkins configure the Thinbackup directory



****

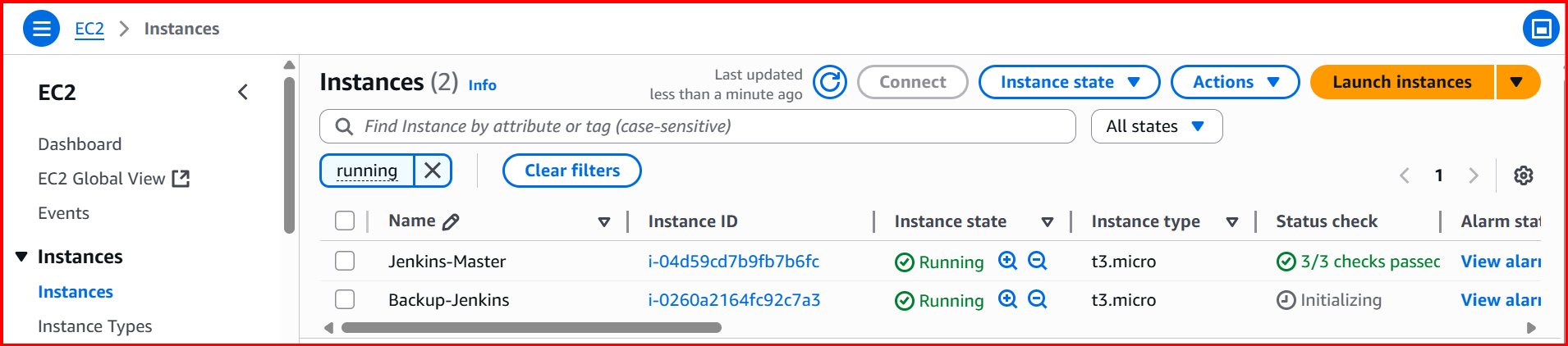
created folder will be configured in (/var/Jenkins/backups-12)

****

****

**6.Setup a new Jenkins server and dump the backup taken in task4**

**Create Ec2 for backup**



SSH and Install Jenkins

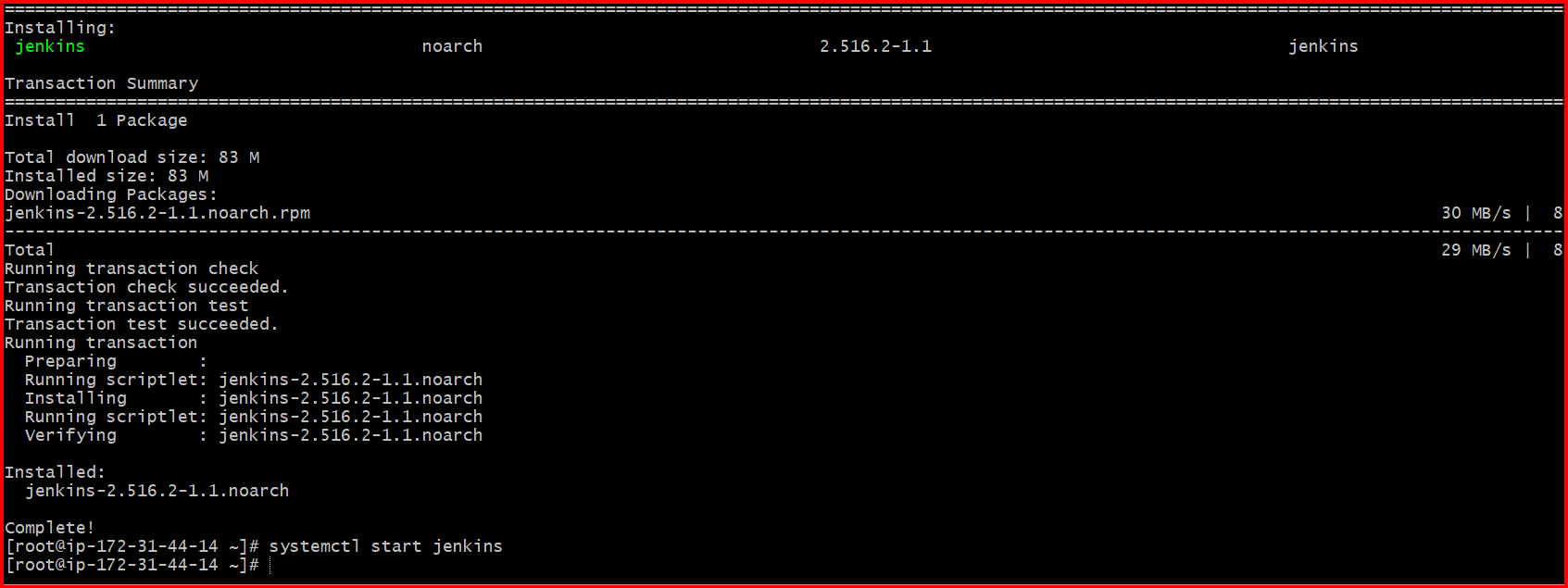
Install Jenkins and start Jenkins

Sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo

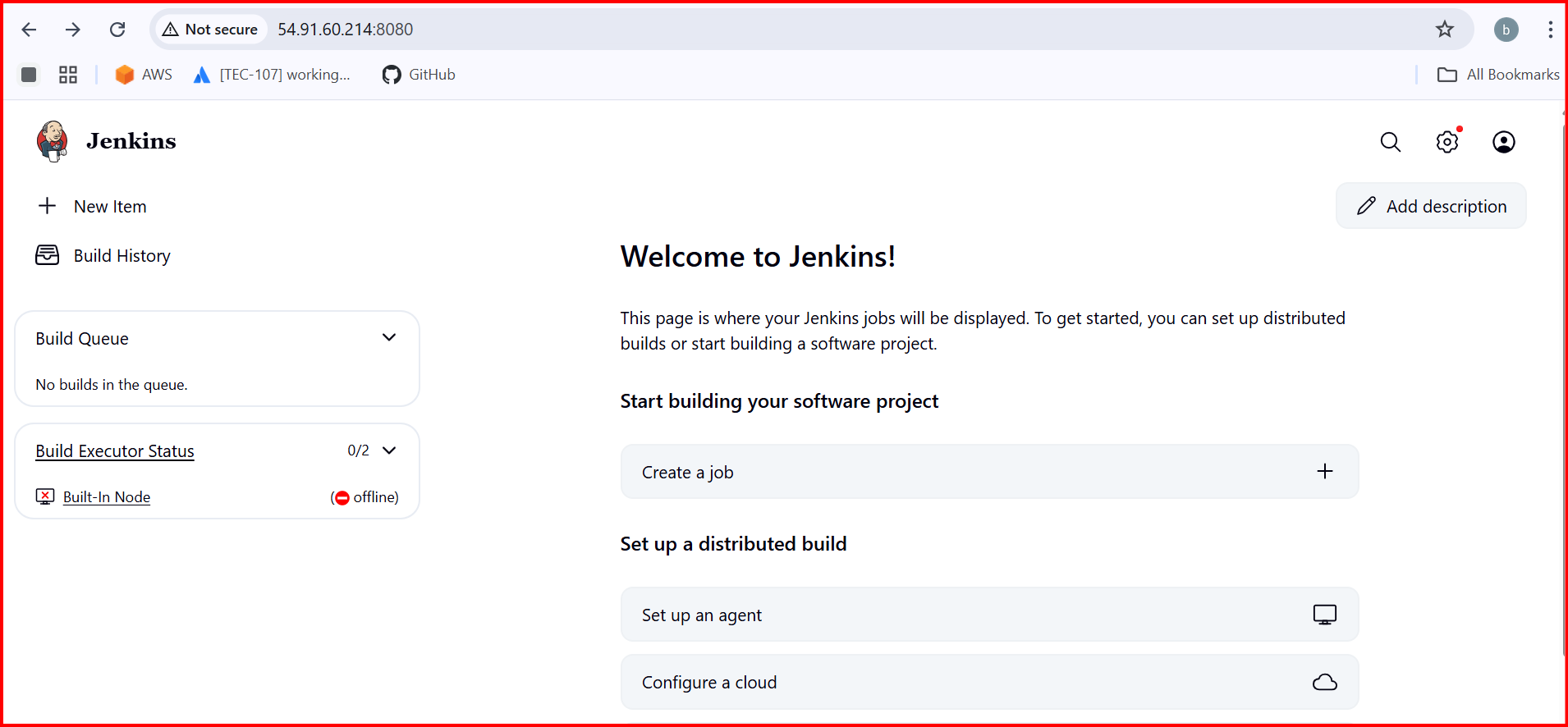
sudo rpm --import <https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key>

sudo yum clean all

sudo yum install jenkins -y



I don’t have any jobs in backup



Create Vi hello.pem

Chmod 600 test.pem

Scp -I hello.pem backup.tar ec2-user@backupserver ip:temp/

This will copy the files from main server to backup server



In Backup server

Cd /tmp

Cp backup.tar /var/lib/

Cd /var/lib

ls

tar xvf backup.tar

then restart the Jenkins

<http://54.91.60.214:8080/restart>