# Install Jenkins

Install bằng docker

docker run -d --name jenkins -p 8080:8080 -p 50000:50000 -v /home/vietdung/:/var/jenkins jenkins/jenkins:lts



Docker ps

Docker logs [jenkins]

Để lấy pass



Cài ngrok để tạo public URL

* Tạo repo Github
* Tạo project trên Jenkins
* Thêm link Jenkins vào Github webhook
* Đẩy code lên Github

## Install trực tiếp

2. Java installation

yum install -y java-1.8.0-openjdk-devel.x86\_64

alternatives --config java

java -version

3. Jenkins install

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

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

yum install -y jenkins

4. Start Jenkins

systemctl start jenkins

5. Setting password (lấy pass authen)

$ less /var/lib/jenkins/secrets/initialAdminPassword

Less tương tự vi, nhưng less chỉ xem đk, ko sửa dk

## Deploy test jenkins build server

1. Web server installation

#!/bin/bash

yum update -y

yum install httpd -y

yum install git -y

amazon-linux-extras install epel

yum install epel-release

rpm -Uvh http://rpms.famillecollet.com/enterprise/remi-release-7.rpm

yum install -y php70 php70-php php70-php-fpm php70-php-pecl-memcached php70-php-mysqlnd php70-php-xml

ln -s /usr/bin/php70 /usr/bin/php

service httpd start

chkconfig httpd on

Tạo 1 user cho việc push code

2. Configure Apache

useradd www-user

cd /var/www/html/

mkdir tinhocthatladongian

chown -Rf www-user:apache tinhocthatladongian

vi /etc/httpd/conf/httpd.conf

Listen 81

NameVirtualHost \*:81

<VirtualHost \*:81>

DocumentRoot /var/www/html/tinhocthatladongian/

<Directory "/var/www/html/tinhocthatladongian">

Order deny,allow

Allow from all

AllowOverride All

Require all granted

</Directory>

</VirtualHost>

Ssh: user abc

Web-server

Jenkins

User/pass

Thằng jenkins sẽ ssh đến web-server để pull code

Vì vậy nó sẽ cần user để ssh đến web-server

* User sẽ cần quyền đọc thực thi trên folder của web-server

Vậy tạo 1 user có quyền cho folder web server pull code

3. Creating ssh key

$ sudo su - www-user

$ ssh-keygen -t rsa

$ cd /home/www-user/.ssh/

$ mv id\_rsa.pub authorized\_keys

4. Copy private key to Jenkin server

cd /home

mkdir jenkins

vi web-key.pem

chown -Rf jenkins:root

chmod 400 web-key.pem

Key này là key của user www-user

Mà thằng jenkins access đến web-server qua user www-user

* Jenkins phải có private key của user www-user
* Web server chỉ cần public key copy vào file authen ở user ww-user cho việc xác thực

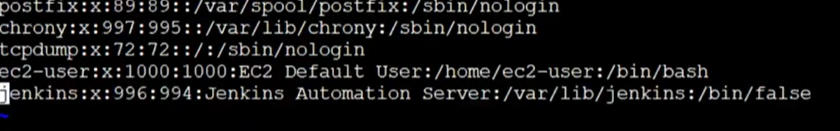
Thế cái file private key copy vào đâu ở jenkins

* Copy vào đâu cũng dk , nếu jenkins hđ user root thì copy vào /root/.ssh
* Nếu là user jenkins thì copy vào /home/jenkins/.ssh

Chmod 600 cho authen.key

Su sang user jenkins, nếu ko login được

Vi /etc/pass



Đổi cái false =bash

Test thử thằng jenkins ssh được ko

Ssh user jenkins

ssh www-user@[Web Server Address] -i /home/jenkins/web-key.pem

/home/jenkins/web-key.pem : là file chứa key

Nếu ko muốn thì đẩy vào /home/jenkins/.ssh

Tại sao phải su sang user jenkins

Vì để config vs webhook cho thằng jenkins tự chạy script thì n sẽ dùng user jenkins để chạy

## Webhook

Ý nghĩ webhook: khi code đẩy lên git, thì thằng webhook sẽ thông báo cho jenkins là có sự thay đổi ở repo, m chạy đi

Lên github -> setting -> webhook -> add webhook -> copy URL jenkins vào

## Cấu hình Jenkins cho repo nào

Đảm bảo trên jenkins có git

# Tích hợp github – Jenkins

Đẩy code lên github

Cấu hình Jenkins tự động nhận biết sự kiện đó và kéo code về



Mỗi lần github có thay đổi, thì webhook sẽ call đến github bằng cái URL của Jenkins (service của Jenkins)

Webhook nhận biết thay đổi của github thì n sẽ call đến jenkins

Jenkins thực hiện trên repository nào của github



Bất kỳ thay đổi nào của nhánh master trên repository này thay đổi thì jenkin n sẽ tự động build

Còn nhánh khác có thay đổi thì Jenkins sẽ ko care

# Jenkins + ansible

Jenkins – ansible trên cùng 1 con

Cài plugin ansible

Install git trên con jenkins – ansible

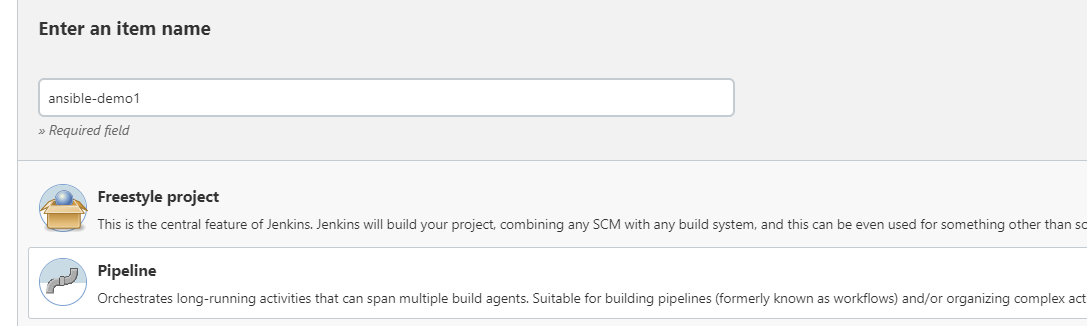
Manager Jenkins -> [Global Tool Configuration](http://192.168.226.128:8080/configureTools/)



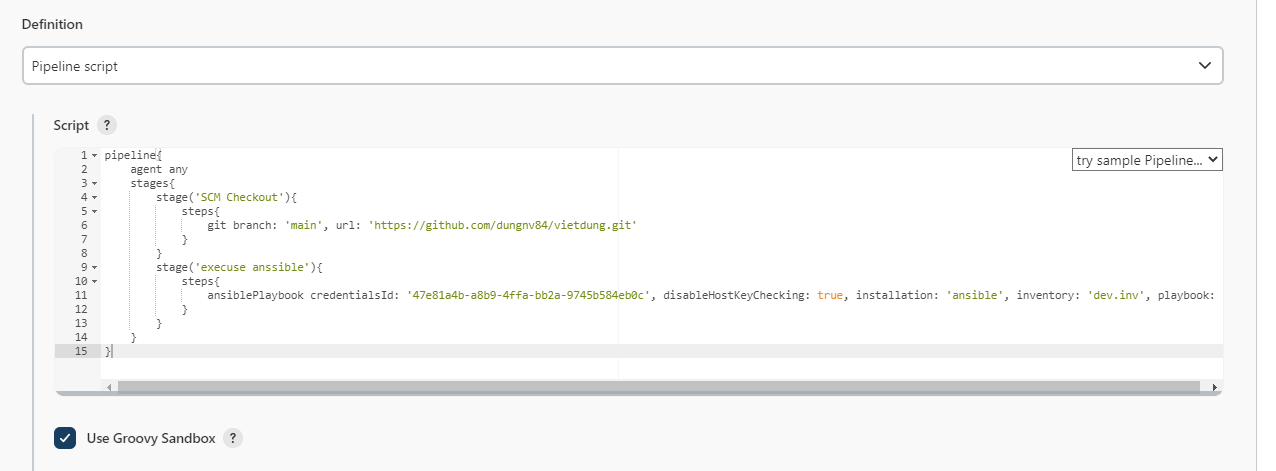


Tên là bất kỳ, tool này để tý dùng

Tạo mới 1 pipeline, dùng pipeline để có thể refer đến git, ansible, …



Add script



Step là được clone từ pipeline Syntax

# Step

pipeline{

    agent any

    stages{

        stage('SCM Checkout'){

            steps{

                git branch: 'main', url: 'https://github.com/dungnv84/vietdung.git'

            }

        }

        stage('execuse anssible'){

            steps{

                ansiblePlaybook credentialsId: '47e81a4b-a8b9-4ffa-bb2a-9745b584eb0c', disableHostKeyChecking: true, installation: 'ansible', inventory: 'dev.inv', playbook: 'apache.yml'

            }

        }

    }

}

Step1: Clone git repo, trong git repo có

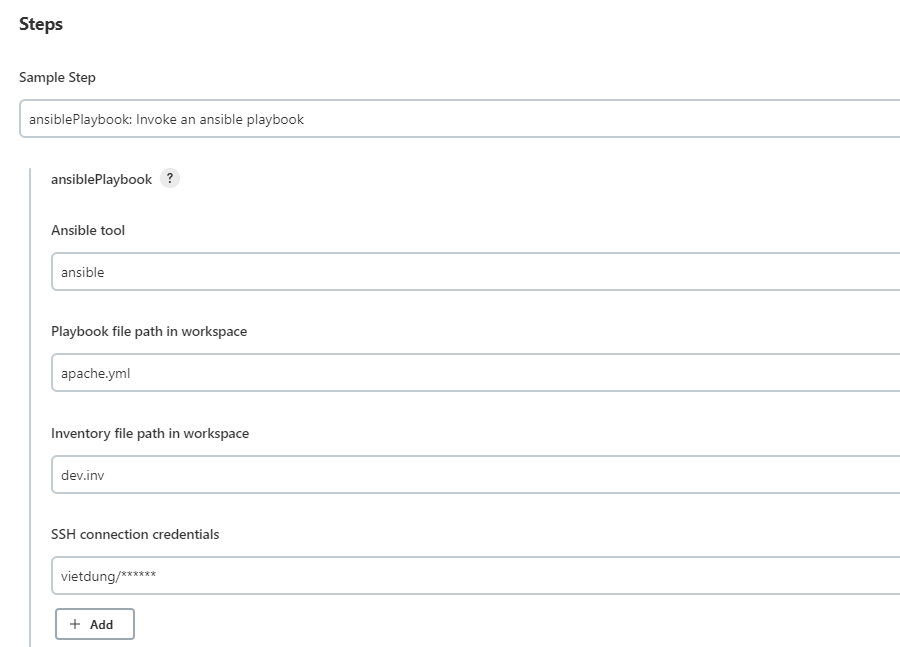
* file playbook để install web server và đẩy source code từ repo về server web
* inventory – IP server
* file HTML để chạy



Gen ra rồi copy vào step 1: coi như là đã pull được git repo về,

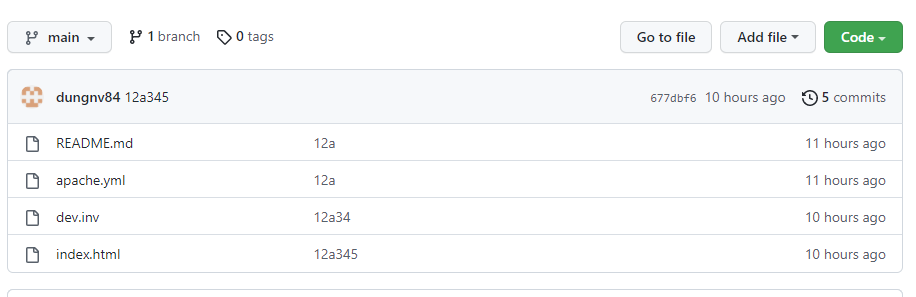
Gọi là task check out

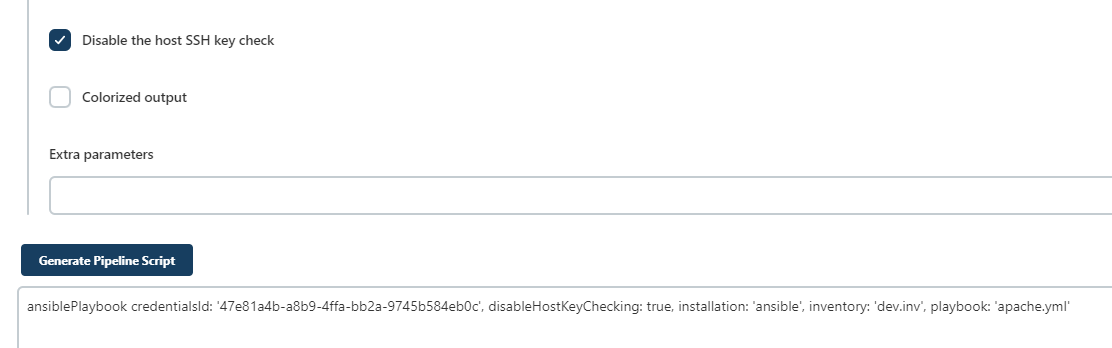
Step2: chạy các file ansible



Asible tool: tức là dùng ansible trên chính con jenkins đang chạy, tên ansible là tên vừa được tạo ra ở **global tool**

Còn các file playbook, inventory được tính từ vị trí đầu repo





Bỏ cái disable check SSH key đi, Case này gặp r, issue viết trong file ansible

Copy cái gen vào step 2

File playbook.yml (apache.yml)

---

- hosts: webservers

  become: True

  tasks:

    - name: Install packages

      yum:

        name: "httpd"

        state: "present"

    - name: Start Apache server

      service:

        name: httpd

        state: started

        enabled: True

    - name: Deploy static website

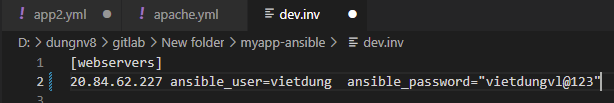
      copy:

        src: index.html

        dest: /var/www/html/

...

File inventory (dev.inv)



IP của con server web, con des để chạy

Oke build now



Oke web server đã chạy với nội dung từ file index.html của repo

# JenkinsFile

Tạo 1 thư mục code trên local