# First pipeline - check version

First, we install the necessary components in the setup directory:

cd /home/user/kubernetes-devops-security/setup/vm-install-script/bash install-script.sh

#### Check k8s resource:

kubectl get node -o wide kubectl run nginx-pod --image nginx kubectl expose po nginx-pod --port 80 --type NodePort kubectl get svc

# Check file config

[[!-z "\$KUBECONFIG"]] && echo"\$KUBECONFIG"|| echo"\$HOME/.kube/config"

## Install jenkins:

nano /etc/apt/sources.list.d/jenkins.list sudo apt-key del FCEF32E745F2C3D5

curl -fsSLhttps://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee\/usr/share/keyrings/jenkins-keyring.asc >/dev/null echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \https://pkg.jenkins.io/debian binary/ | sudo tee \/etc/apt/sources.list.d/jenkins.list >/dev/null

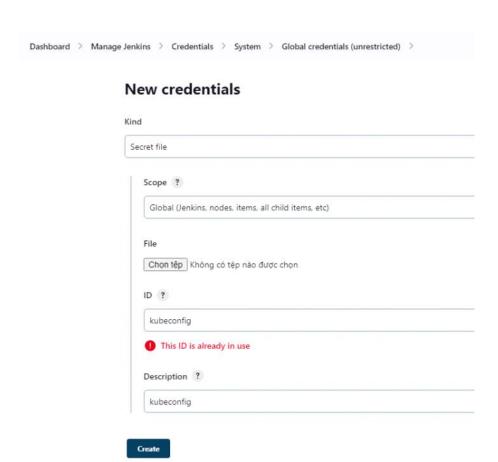
sudo apt-get update sudo apt-get install jenkins

## k8s credential

Export file kubeconfig:

cat /root/.kube/config

Go to Manage Jenkins - Credentials System - Global credentials Create kubeconfig and add file config above



This is can fix error: Unable to connect to the server: x509: certificate signed by unknown authority.

Now, we can create an example of a file pipeline for version checking:

```
pipeline {
agent any
stages {
 stage('git version') {
  steps {
    sh "git version"
 stage('maven version') {
  steps {
    sh "mvn -v"
 stage('docker version') {
   steps {
    sh "docker -v"
 stage('kubernetes version') {
   steps {
   withKubeConfig([credentialsId: 'kubeconfig']) {
     sh "kubectl version"
```

}

Command withKubeConfig([credentialsId: 'kubeconfig']) in pipeline will check credential to logic k8s, if you don't use ths command, you cannot run command about k8s because pipeline run with jenkins user and it is not have permission access k8s api by default. And to perform this command, you need install k8s cli plugins in jenkins.

Now, run build pipeline.

