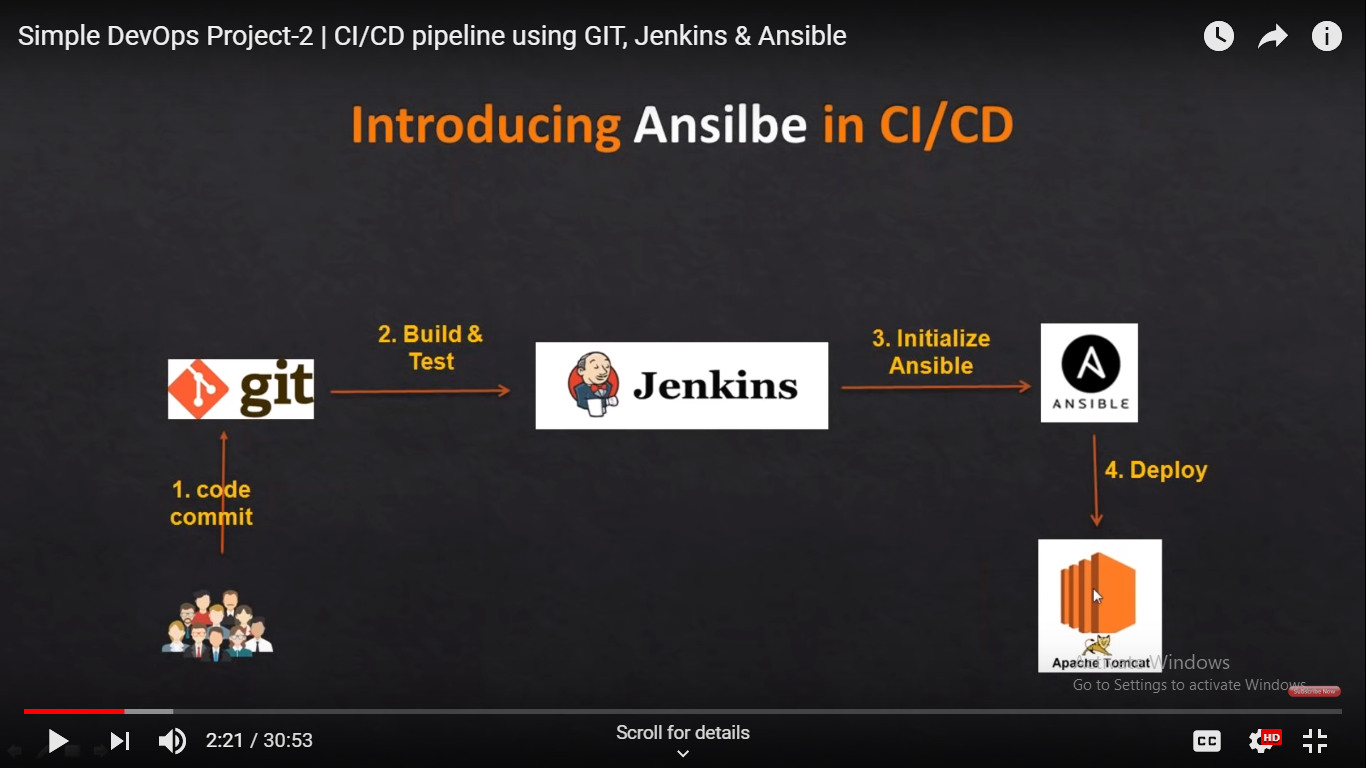
**Case study for Ansible Deployment**

**Case study workflow**



1. Pre-requites

* JDK 1.8 in control and slave machine
* Jenkins and ansible in Control machine
* Tomcat in Slave machine

1. #Install java 8 using below steps

**sudo su -**

**mkdir /opt/java/**

**cd /opt/java/**

**yum install wget**

**#edit the tar.gz link below , copy from** <https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

**wget --no-cookies --no-check-certificate --header "Cookie: oraclelicense=accept-securebackup-cookie" https://download.oracle.com/otn-pub/java/jdk/8u201-b09/42970487e3af4f5aa5bca3f542482c60/jdk-8u201-linux-x64.tar.gz**

**tar -zxvf jdk-8u201-linux-x64.tar.gz**

**cd jdk1.8.0\_201/**

**update-alternatives --install /usr/bin/java java /opt/java/jdk1.8.0\_201/bin/java 100**

**update-alternatives --config java**

select the new java option

**update-alternatives --install /usr/bin/javac javac /opt/java/jdk1.8.0\_201/bin/javac 100**

**update-alternatives --config javac**

select the new java option

**update-alternatives --install /usr/bin/jar jar /opt/java/jdk1.8.0\_201/bin/jar 100**

**update-alternatives --config jar**

enter 1

**export JAVA\_HOME=/opt/java/jdk1.8.0\_201/**

**export JRE\_HOME=/opt/java/jdk1.8.0\_201/jre**

**export PATH=$PATH:/opt/java/jdk1.8.0\_201/bin:/opt/java/jdk1.8.0\_181/jre/bin**

1. Install Jenkins in New machine or setup existing Jenkins

$ sudo wget -O /etc/yum.repos.d/jenkins.repo http://pkg.jenkins.io/redhat-stable/jenkins.repo

$ sudo rpm --import <http://pkg.jenkins.io/redhat-stable/jenkins.io.key>

$ sudo yum install jenkins –y

$ sudo service jenkins start

#update the inbound rules if you are using aws servers to access Jenkins from outside

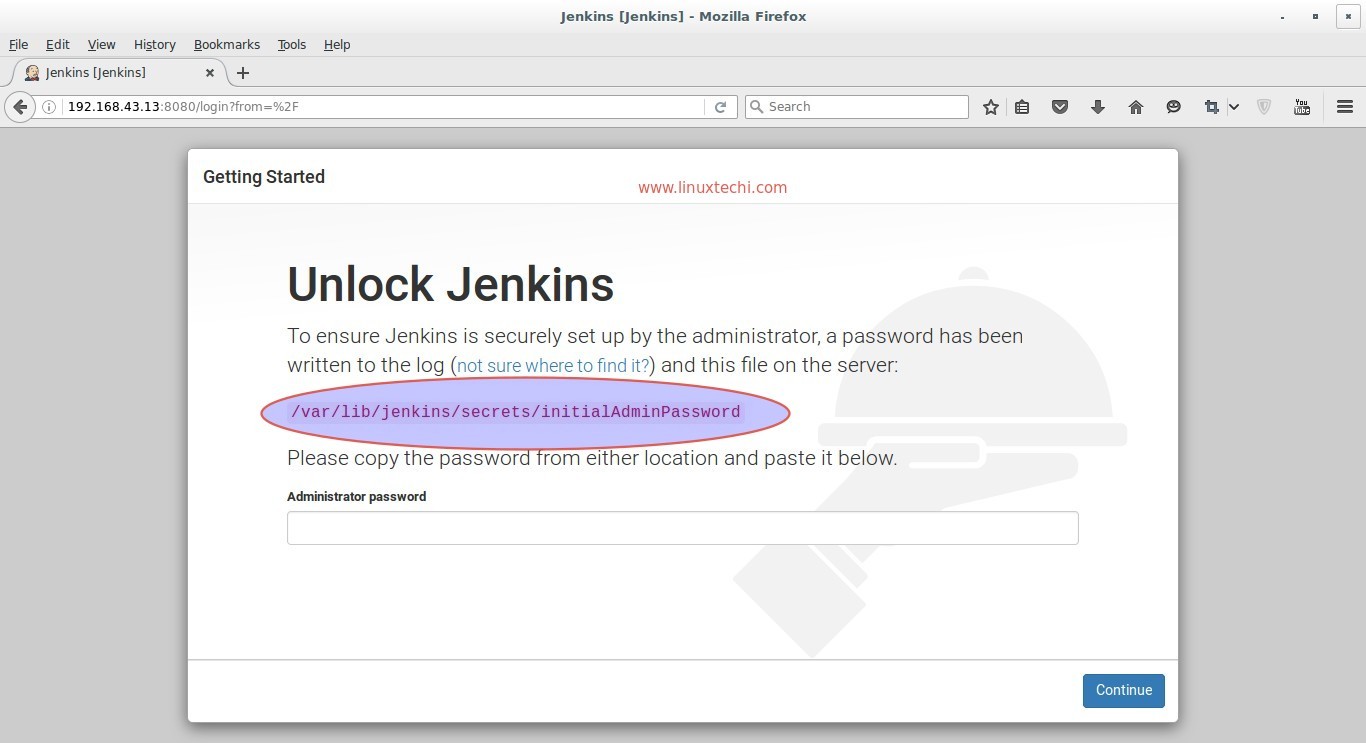
|  |  |  |  |
| --- | --- | --- | --- |
| All TCP | TCP | 0 - 65535 | 0.0.0.0/0 |

Security group – inbound rule – add like below

1. #Install git in Jenkins server

$ sudo yum install git

1. Configure Jenkins, access the Jenkins in the browser using ipaddress of the sytem and 8080 port



Follow the process to complete Jenkins initial setup

#Configure maven and jdk in global tool configuration

# install maven integration plugin

# install publish over ssh plugin

1. Configure the jdk and tomcat in slave machine

#Install the jdk using steps given in first page

#Login to slave machine and install the apache tomcat, use below link to download

<https://tomcat.apache.org/download-80.cgi>

Under /opt/tomcat/

$ sudo wget http://mirrors.estointernet.in/apache/tomcat/tomcat-8/v8.5.39/bin/apache-tomcat-8.5.39.tar.gz

$ tar –xzvf apache-tomcat-8.5.39.tar.gz

Tomcat-users.xml

<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<role rolename="manager-jmx"/>

<role rolename="manager-status"/>

<user username="ansadmin" password="Shiva@90" roles="manager-gui,manager-script,manager-jmx,manager-status" />

#start the tomcat from bin folder

./catalina.sh start

#verify that if you are able to access it from browser, enable inbound rules to open al ports to public if you are using aws server

http://<ipaddres>:8080

1. Login to control machine and write playbook for copying war to slave machine

---

- hosts: slave

become: true

tasks:

- name: task to copy war file

copy:

src: /home/ansadmin/webapp/CounterWebApp.war

dest: /opt/tomcat/apache-tomcat-8.5.39/webapps/

1. configure the Jenkins jobs to do git checkout, maven build and publish over ssh

