CI-CD Lab by integrating GIT, Jenkins, Ansible and Tomcat:

Task 1: Install and Configure Ansible

Task 2: Install and Configure Jenkins

Task 3: Integrate GIT, Jenkins, Ansible and Tomcat

Task 4: Setup CI/CD pipeline

Task 1: Install and Configure Ansible: Create 2 Redhat VM and follow steps

Step 1.1: **Make sure you have 2 Redhat VM, 1st Instance as Ansible Master and 2nd Instance as Ansible Node or Ansible client VM**

Step 1.2: Created user on all the EC2 instance, password should match password on the all the instances (Master & Node) {Note: IF you already have integrated with LDAP/AD then create LDAP account instead}

```
useradd ansadmin

passwd ansadmin
```

Step 1.3: Add ansadmin to the sudoers file to have admin prviledges (Master and Node)

```
visudo

ansadmin      ALL=(ALL)      NOPASSWD: ALL
```

```
## Allows members of the users group to shutdown this system
# %users  localhost=/sbin/shutdown -h now

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)
#includedir /etc/sudoers.d
ec2-user        ALL=(ALL)       NOPASSWD: ALL
ansadmin        ALL=(ALL)       NOPASSWD: ALL
-- INSERT --
```

Also we have to allow our user to run playbook on client VM so add user.

```
## Allow root to run any commands anywhere
root    ALL=(ALL)       ALL
ansadmin        ALL=(ALL)       NOPASSWD: ALL
```

Step 1.4: **Enable user login on all EC2 Instance** **(Master & Node)**

```
cd /etc/ssh

vi /etc/ssh/sshd_config
```

uncomment PasswordAuthentication yes

comment PasswordAuthentication no

```
# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication yes
#PermitEmptyPasswords no
#PasswordAuthentication no
```

```
service sshd restart
```
(Master & Node)

Step 1.5: Login as ansadmin user on master and generate ssh key (Master)

Disconnect and login as ansadmin then provide your password

```
login as: ansadmin
Server refused our key
ansadmin@52.59.245.148's password:
[ansadmin@ip-172-31-42-72 ~]$
```

```
ssh-keygen                                  (Master)

cd /home/ansadmin/.ssh                      (Master)
```

Step 1.6: Copy your public key (ansadmin user of master) and paste on client vm. You also can copy paste manually. (You also can use scrip https://www.ssh.com/ssh/copy-id)

(Master)

ssh-copy-id 172.31.38.190 (172.31.38.190 is private ip of Node , Make sure you update IP)

ssh 172.31.38.190 (you should be able to ssh your node or client VM)

then exit from node(client)

sudo vi /etc/ansible/hosts (update private IPs of client VM under hosts file) **(Master)**

```
[amazonweb]
172.31.38.190
```

ansible all -m ping

ansible -m ping amazonweb

Step 1.7: Create a Playbooks directory just for testing. **(Master)**

mkdir -p /etc/ansible/playbooks (Create playbooks directory if doesn't exist)

Task 2: Install and Configure Jenkins:

Step 2.1: **Create redhat instance for jenkins and follow jenkins installations steps (refer jenkins ppt page no 24)**

Task 3: Integrate GIT, Jenkins, Ansible and Tomcat:

Step 3.1: Install following plugins

"publish over ssh" , "Maven Integration"

Step 3.2: **Manage Jenkins ->configuration system -> publish over ssh**

(Here we have 2 option for Jenkins Ansible Authentication

a: key based-(pub key of ansadmin user)

b: password based (password of ansadmin user which can be integrated with LDAP/Active Directory)

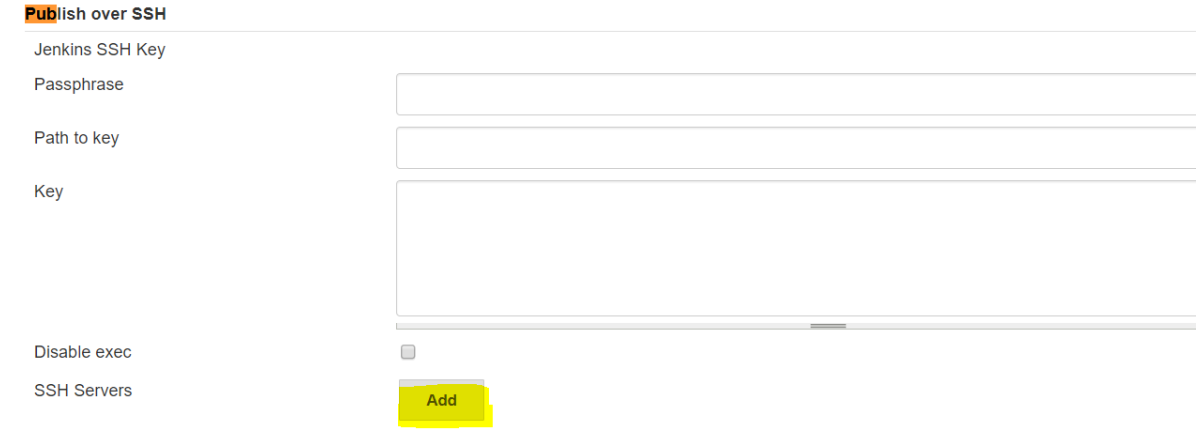
(We will be using password method for authentication)

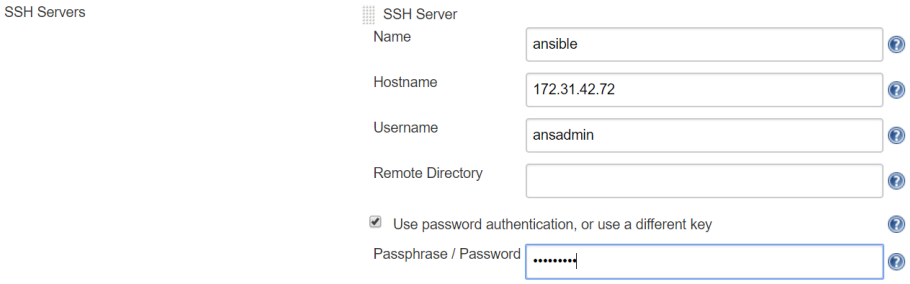
select Add

Name: ansible_server

Host Name: private ip of Ansible Master

User name: ansadmin (for password click Advanced then password)





Then click on Test connection.

If succeed then we are good to follow remaining steps.

Task 4: Setup CI/CD Projects:

Step 4.1: Change permissions: **(Master)**

```
sudo chmod 777 /etc/ansible/playbooks .
```

Change Permission **(Node)**

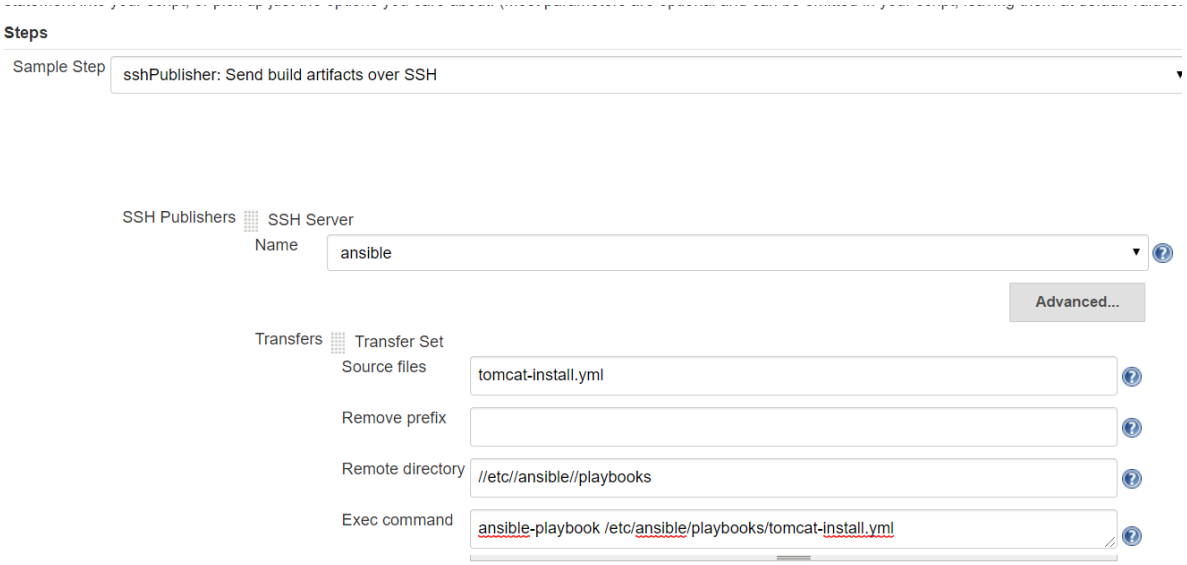
```
sudo chmod 777 /opt .
```

Step 4.2: Create Jenkins pipeline job.

Step 1: Write a jenkins file and add stages like “scm checkout”, “maven test” , “maven build”

Then test it before you deploy package to tomcat server.

Step 2: Add a new stage called “tomcat install” and execute pipeline. Refer below yaml, use pipeline syntax generate for script. (copying tomcat-install.yml from Jenkins’s workspace are to Ansible Master’s playbook folder, once we copied successfully then run the playbook)



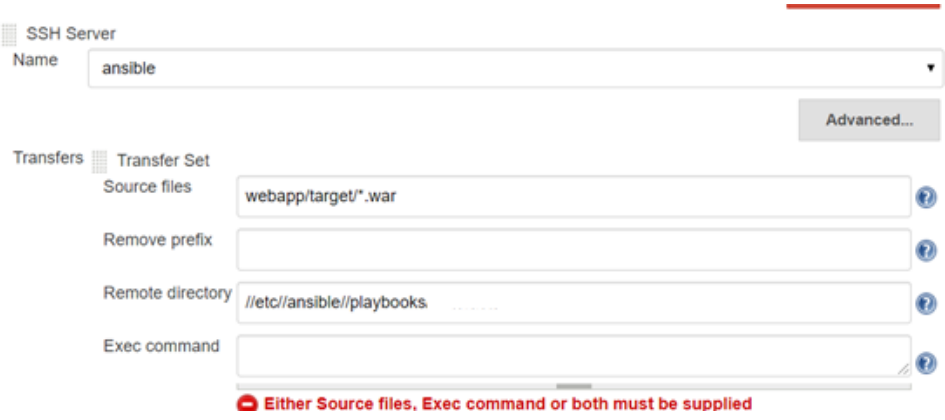
https://github.com/prakashk0301/maven-project/blob/ci-cd-ansible/tomcat-install.yml

Run the pipeline and verify, playbook should install tomcat on Ansible Node(Tomcat)

<public IP of ansible Node>:8080

Step 3: Once you install tomcat on Ansible node, now its time to send artifacts to Ansible Master’s playbooks directory or Nexus Snapshot Repo.

Generate Pipeline script from pipeline syntax generator.

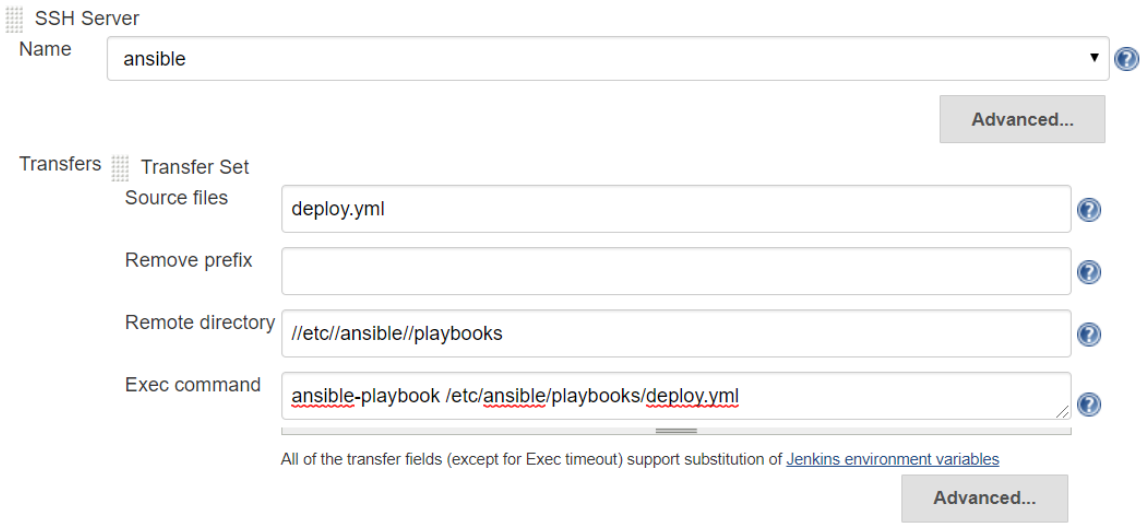


Reference Link: stage ('send artifacts to /etc/ansible/playbooks folder')

https://github.com/prakashk0301/maven-project/blob/ci-cd-ansible/Jenkinsfile

Step 4: Now its time to run deploy.yml

First you need to send deploy.yml from jenkins's workspace area to Ansible Master's playbooks folder using publish over ssh plugins.



Reference Link of deploy.yml file and Jenkins file.

https://github.com/prakashk0301/maven-project/blob/ci-cd-ansible/deploy.yml

https://github.com/prakashk0301/maven-project/blob/ci-cd-ansible/Jenkinsfile

Once you add all the stages then run your jenkins job. It should deploy artifacts to tomcat's webapp folder. You can verify by access from browser.

<Tomcat public IP>:8080/webapp

Pipeline ci-cd-pipeline-ansible

add description

Disable Project

Recent Changes

Stage View

	Declarative: Checkout SCM	Compile Stage	Testing Stage	install Stage	Jenkins-Ansible-CI-CD-Pipeline
Average stage times: (Average full run time: ~35s)	1s	3s	2s	2s	7s
#2 Dec 25 12:35 1 commit	1s	5s	5s	5s	15s