

# How to Jenkins-Ansible integration.

## Ansible configurations:-

Agenda: Create a ansible role for deploying the artifacts from nexus to destination.

1. first we need to install the ansible in the linux machine i.e ubuntu/centos, for package mangement may vary based on flavours of linux. Ansible is not supporting for windows.

```
On ubuntu
$ sudo apt update
$ sudo apt install software-properties-common
$ sudo apt-add-repository --yes --update ppa:ansible/ansible
$ sudo apt install ansible
On Fedora:
$ sudo dnf install ansible
On RHEL and CentOS:
$ sudo yum install ansible
```

2. Check the version.

```
root@ip-address:/home/ubuntu# ansible --version
ansible 2.8.6
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
  ansible python module location =
/usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.12 (default, Oct  8 2019, 14:14:10) [GCC 5.4.0
20160609]
```

3. Please install the supported python libraries for Ansible, i.e lxml etc.

4. Ansible related folders.

```
$ whereis ansible
ansible: /usr/bin/ansible ---> bin folder
/etc/ansible ----> ansible home deafulst directory
/usr/share/man/man1/ansible.1.gz ---> check the sub commands.
```

5. Go to ansible home directory.

```

ansible.cfg  ---> configuration related to ansible configurations like
below
#inventory      = /etc/ansible/hosts
#library        = /usr/share/my_modules/
#module_utils   = /usr/share/my_module_utils/
#remote_tmp     = ~/.ansible/tmp
#local_tmp      = ~/.ansible/tmp
#plugin_filters_cfg = /etc/ansible/plugin_filters.yml
#forks          = 5
#poll_interval  = 15
#sudo_user      = root
#ask_sudo_pass  = True
#ask_pass       = True
#transport      = smart
#remote_port    = 22
#module_lang    = C
#module_set_locale = False
-----
-----
hosts          ---> you can store the hosts here.
roles          ----> defaults role can install here.

```

6. Create a file with naming playbook.yml and structure is below.

```

---
- hosts: localhost
  name: nexus-role
  gather_facts: true
  roles:
    - nexus-role
  vars:
    versioning: latest

```

7. Create a role and struture is below.

```
$ ansible-galaxy init nexus-role.  
$ tree  
.  
 defaults  
   main.yml  
 files  
 handlers  
   main.yml  
 meta  
   main.yml  
 README.md  
 tasks  
   main.yml  
 templates  
 tests  
   inventory  
   test.yml  
 vars  
   main.yml  
this yml files are defaults file for placing the tasks, vars, templates,  
handlers.
```

8. Try to run the playbook with mapping role.

```
PLAY [nexus-role] *****  
TASK [Gathering Facts] *****  
ok: [localhost]  
TASK [nexus-role : nexusurl] *****  
ok: [localhost]  
PLAY RECAP *****  
localhost : ok=2  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0  
root@ip-172-31-35-91:/home/ubuntu/ansible-roles#
```

9. I'm trying to deploy the nexus artifacts from nexus server to destination.

```

task main.yml:-
---
-
  maven_artifact:
    artifact_id: consumerBanking
    dest: /home/ubuntu/
    group_id: com.companyname.bank
    password: {{ servicetoken }}
    repository_url: "http://localhost/repository/novartis-maven"
    username: admin
    version: {{ versioning }}
  name: nexusurl
- name: deploying
  shell: java -jar consumerBanking-latest.jar

var main.yml:-
---
servicetoken: XXXXXXXXXXXXXXXXXXXX

```

10. Now our task is done, next we can integrate with Jenkins and implement the same.

### Jenkins configurations:-

#### Jenkins-Ansible(Master node) configuration.

**Note: Ansible should install in master node. Ansible-playbook will run in master node.**

1. start the Jenkins server and install the Ansible plugin.
2. Configure the Ansible as below pic.

The screenshot displays the Jenkins 'Ansible installations' configuration page. It features a table with one entry named 'ansible'. The configuration for this entry includes the path to the ansible executables directory set to '/usr/bin/' and the 'Install automatically' checkbox checked. Below the table, there is an 'Add Installer' dropdown menu with options: 'Extract \*.zip/\*.tar.gz', 'Run Batch Command', and 'Run Shell Command'. A 'Delete Ansible' button is visible on the right side of the page.

3. Create a pipeline job and add the declarative pipeline , add repository, map the jenkinsfile.

General Datadog Tagging Build Triggers Advanced Project Options **Pipeline** Advanced...

## Pipeline

Definition Pipeline script from SCM

SCM Git

Repositories

Repository URL

Credentials  Add

Advanced...

Add Repository

Branches to build

Branch Specifier (blank for 'any')  Add Branch

Repository browser (Auto)

Additional Behaviours Add

Script Path

Lightweight checkout ☐

Save Apply

4. For declarative syntax, add the ansible playbook step and generate the syntax, add in jenkinsfile.

Sample Step ansiblePlaybook: Invoke an ansible playbook

Ansible tool

Playbook file path in workspace

Inventory file path in workspace

SSH connection credentials  Add

Vault credentials  Add

Use become ☐

Become username

Use sudo (deprecated) ☐

Sudo username (deprecated)

Host subset

Tags

Tags to skip

Task to start at

Number of parallel processes to use

Disable the host SSH key check ☐

Colorized output ☐

Extra parameters

6. Declarative pipeline for the Ansible-playbook.

```

node {
  stage("checkout SCM") {
    checkout changelog: false, scm: [$class: 'GitSCM', branches: [[name:
    '*/master']], doGenerateSubmoduleConfigurations: false, extensions: [],
    submoduleCfg: [], userRemoteConfigs: [[credentialsId: 'BitCred', url:
    'http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git']]
  }
  stage("ansible-roles") {
    ansiblePlaybook extras: 'versioning: "latest"', installation:
    'ansible', inventory: 'inverntory_directory', playbook: 'playbook.yml'
  }
}

```

6. Build the pipeline job and see the output, check the logs by going to corresponding logs.

The screenshot displays the Jenkins Pipeline interface for 'Pipeline ND-63- ansible-nexus'. On the left, a sidebar contains navigation links: Back to Dashboard, Status, Changes, Build Now, Delete Pipeline, Configure, Move, Full Stage View, Open Blue Ocean, Rename, and Pipeline Syntax. The main area shows the 'Stage View' with a table of stage times and a 'Recent Changes' section. Below the table, the 'Stage Logs (ansible-roles)' are displayed, showing the output of the 'Invoke Ansible Playbook' stage.

**Pipeline ND-63- ansible-nexus**

**Recent Changes**

**Stage View**

	Declarative: Checkout SCM	checkout SCM	ansible-roles
Average stage times: (Average full run time: ~12s)	3s	287ms	3s
#55 Oct 25 11:40 No Changes	3s	288ms	3s
#54 Oct 25 11:37 1 commit	3s	276ms	4s

**Stage Logs (ansible-roles)**

Invoke Ansible Playbook (self time 3s)

```

[ND-63- ansible-nexus] $ ansible-playbook playbook.yml -f 5
[WARNING]: provided hosts list is empty, only localhost is available

PLAY [nexus-role] *****

TASK [setup] *****
ok: [localhost]

TASK [nexus-role : nexusurl] *****
changed: [localhost]

PLAY RECAP *****
localhost                : ok=2   changed=1   unreachable=0   failed=0

```

7. The output of the build stage can be observe in the job dashboard.

#### Jenkins-Ansible(Docker slave node) configuration.

**Note:** Ansible should install in Docker node. Ansible-playbook will run in container node, JDK is mandatory in order to run the slaves in any nodes.

1. There are few plugins need to install in Jenkins to use Docker within Jenkins jobs and Pipeline. Go to Manage Jenkins >> Manage Plugins >> Available tab >> search for Docker then install Docker Plugin, Docker Pipeline and Docker Slave Plugin.
2. To use the docker within Jenkins jobs and pipeline; need to configure docker in Jenkins as other tools.

Go to configure security settings "Enable Agent Master Access Control".

3. Go to Manage Jenkins >> Configure System >> Search for an "Add a new cloud" >> Docker.

Mentioned any Name to save this configuration. However Docker Host URI needs to mention, there are three types of Docker Host URI can be given like fd://, unix:// and tcp:// along with IP address (can be given 0.0.0.0) and port numbers (2375, 2376 and 4243). Then click on "Test Connection", you should be able to see text like "Version = 19.03.2, API Version = 1.40" if connection is successful.

If any error is coming then need to investigate further, however one probable reason could be to assign the permission of "/var/run/docker.sock" in host machine.

4. Need to configure the slave container under "Docker Agent Templates" in Docker Cloud.

- Labels: How you will identify this image in your pipelines or job.
- Docker image: The name of the image related to this template.
- Registry Authentication: The credentials used to log in to your registry account(DockerHub, ECR).
- Remote Filing System Root: The home folder for the user we have created in the image, in this case, /home/jenkins.
- Usage: Few options are available, however we have selected "Only build jobs with label expression matching this mode" for our use.
- Connect Method: Few options are available "Attach Docker Container", "Connect with SSH" and "Connect with JNLP" like Select "Attach Docker Container" which our images were built to be accessed with.
- User: Name of the user used to connect, so, jenkins.
- Pull Strategy: Options are available as "Never Pull", "Pull all images every time", "Pull once and update latest". So "Pull once and update latest" option is most suitable to choose.
- **Note:** if you are using public docker image just mentions name, else you can create customized Dockerfile, please test it and

push to registry, right-now im using cutomed docker file in my local, find below code.

```
FROM jenkinsci/slave:3.35-3
USER root
RUN apt-get update && \
    apt-get install --no-install-recommends -y \
    software-properties-common \
    ansible \
    && rm -rf /tmp/downloaded_packages/ /tmp/*.rds \
    && rm -rf /var/lib/apt/lists/* \
    && apt-get install -y python-lxml

USER jenkins
```

**Jenkins-job execution:**

- Create a pipeline job and map the repository and jenkinsfile as i show above.
- Write the Jenkins file as below, mention the node label.



```

pipeline {
  agent { label 'ansible-slave' } -----> docker slave which i
mentioned in docker node template.
  stages {
    stage("checkout SCM") {
      steps {
        checkout changelog: false, scm: [$class: 'GitSCM',
branches: [[name: '*/master']],
doGenerateSubmoduleConfigurations: false, extensions: [],
submoduleCfg: [], userRemoteConfigs: [[credentialsId:
'BitCred', url:
'http://novartis.devops.altimetrik.io:7990/scm/son/ansible-role
s.git']]
      }
    }
    stage("ansible-roles") {
      steps {
        step([$class: 'AnsiblePlaybookBuilder',
additionalParameters: '', ansibleName: 'ansibles', becomeUser:
'', credentialsId: '', forks: 5, inventory: [$class:
'InventoryDoNotSpecify'], limit: '', playbook: 'playbook.yml',
skippedTags: '', startAtTask: '', sudoUser: '', tags: '',
vaultCredentialsId: ''])
          //step {
            //sh 'ansible-playbook playbook.yml'
          //}
      }
    }
  }
}

```

see the outputs in the build section, console output.

## Console Output

```
Started by user admin
Checking out git http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git into /var/jenkins_home/workspace/ND-63- ansible-nexus@script to read jenkinsfile
using credential BitCred
> git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git # timeout=10
Fetching upstream changes from http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git
> git --version # timeout=10
using GIT_ASKPASS to set credentials Credential for logging into BitBucket
> git fetch --tags --progress -- http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git +refs/heads/*:refs/remotes/origin/*
> git rev-parse origin/master^{commit} # timeout=10
Checking out Revision 46a80cbce16d204c6593df9c3e9faad0835dc926 (origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f 46a80cbce16d204c6593df9c3e9faad0835dc926
Commit message: "main.yml edited online with Bitbucket"
> git rev-list --no-walk 46a80cbce16d204c6593df9c3e9faad0835dc926 # timeout=10
Running in Durability level: MAX_SURVIVABILITY
[Pipeline] Start of Pipeline
[Pipeline] node
Running on ansible-slave-00003x5nd4ps on docker1 in /home/jenkins/workspace/ND-63- ansible-nexus
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
using credential BitCred
Cloning the remote Git repository
Cloning repository http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git
> git init /home/jenkins/workspace/ND-63- ansible-nexus # timeout=10
Fetching upstream changes from http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git
> git --version # timeout=10
using GIT_ASKPASS to set credentials Credential for logging into BitBucket
> git fetch --tags --progress -- http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git +refs/heads/*:refs/remotes/origin/*
> git config remote.origin.url http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git # timeout=10
Fetching upstream changes from http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git
using GIT_ASKPASS to set credentials Credential for logging into BitBucket
> git fetch --tags --progress -- http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git +refs/heads/*:refs/remotes/origin/*
Checking out Revision 46a80cbce16d204c6593df9c3e9faad0835dc926 (origin/master)
Commit message: "main.yml edited online with Bitbucket"
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (checkout SCM)
[Pipeline] checkout
using credential BitCred
Fetching changes from the remote Git repository
Checking out Revision 46a80cbce16d204c6593df9c3e9faad0835dc926 (refs/remotes/origin/master)
> git rev-parse origin/master^{commit} # timeout=10
> git config core.sparsecheckout # timeout=10
> git checkout -f 46a80cbce16d204c6593df9c3e9faad0835dc926
Commit message: "main.yml edited online with Bitbucket"
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (ansible-roles)
[Pipeline] step
[ND-63- ansible-nexus] $ ansible-playbook playbook.yml -f 5
> git rev-parse --is-inside-work-tree # timeout=10
> git config remote.origin.url http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git # timeout=10
Fetching upstream changes from http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git
> git --version # timeout=10
using GIT_ASKPASS to set credentials Credential for logging into BitBucket
> git fetch --tags --progress -- http://novartis.devops.altimetrik.io:7990/scm/son/ansible-roles.git +refs/heads/*:refs/remotes/origin/*
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
> git rev-parse refs/remotes/origin/origin/master^{commit} # timeout=10
> git config core.sparsecheckout # timeout=10
> git checkout -f 46a80cbce16d204c6593df9c3e9faad0835dc926
[WARNING]: provided hosts list is empty, only localhost is available

PLAY [nexus-role] *****

TASK [setup] *****
ok: [localhost]

TASK [nexus-role : nexusurl] *****
changed: [localhost]

PLAY RECAP *****
localhost                : ok=2   changed=1   unreachable=0   failed=0

[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withinv
[Pipeline] }
[Pipeline] }
[Pipeline] // node
-- -- -- -- --
```

[Back to Dashboard](#)  
[Status](#)  
[Changes](#)  
[Build Now](#)  
[Delete Pipeline](#)  
[Configure](#)  
[Move](#)  
[Full Stage View](#)  
[Open Blue Ocean](#)  
[Rename](#)  
[Pipeline Syntax](#)

## Pipeline ND-63- ansible-nexus

[Recent Changes](#)

### Stage View

Average stage times:

(Average full run time: ~12s)

	Declarative: Checkout SCM	checkout SCM	ansible-roles
PS5	3s	267ms	3s
PS4	3s	276ms	4s

Build History

#55	Oct 25, 2019 6:10 AM
#54	Oct 25, 2019 6:07 AM

check the destination server for deployment to conform.

```

root@ip-172-31-35-91:/home/ubuntu# ls
: ansible-roles consumerBanking-latest.jar Dockerfile
root@ip-172-31-35-91:/home/ubuntu# java -jar consumerBanking-latest.jar
no main manifest attribute, in consumerBanking-latest.jar
root@ip-172-31-35-91:/home/ubuntu# cat playbook.yml

```

that's all !! done.