











**encrypt**

ansible-vault encrypt yamlfilename

**DECRYPT**:

ANSIBLE-VAULT.DECRYPT YAMLFILE

**ansible playbook command'**

ansible-playbook yaml file name

**slve misson connectivity**

ansible all -i slave.txt -m ping

**seting permission**

ansible-playbook yml filename

set a permission using ansible nplaybook

firt chnge to root user

create a yaml file

root@linuxhelp:~# vi test.yml

- hosts: localhost

gather\_facts: true

tasks:

- name: Create a file named Linuxhelp

file:

path: /Linuxhelp

state: touch

- name: Changing the permissions

file:

path: /Linuxhelp

owner: root

group: linuxhelp

mode: 0755

finaly run this command

ansible-playbook test.yml

**master slave method in jenkins**

Install Java on master node.

Install Jenkins on master node.

Install java on slave node.

Create a user and ssh keys on slave node. - SSH-KEYGEN

Setting up the remote server - ssh-copy-id (ne slave ip addr5ess)

copy the public key on master node

Join slave node to master.

Test the setup.

**ANSIBLE HOST FILE LOCATION:**

ETC/HOST

* **folder path - uploading the deployment package**

the folder path for uploading a deployment package depends on the repository format

By default, Nexus supports several repository formats such as Maven, npm, NuGet, and others.

For example : For Maven repositories

<repository-name>/<group-id>/<artifact-id>/<version>/

like this : releases/com/example/my-project/1.0.0/

* **login using service account.**
* Service account is like a admin account.
* Using the service account details, we can login to nexus.
* Also, we can modify changes in nexus repo using the service account cred.
* **Nexus components**

Artifact ID It represents the name of the project.

For example: if we’re building a library called “my-library”, the artifice id is “my-library’.

Version refers to specific release of a software artifact within a repo. It follows a structured format such as numeric, combination of numbers, letters..

For example: 1.0.1-snapshots

Group ID Unique identifier helps to organize software artifact within a repo.

Nexus URL default URL for nexus is <https://repo1.maven.org/maven2/>

Using all the combination, we will create curl command to upload r download artifacts from nexus.

Curl command :

curl -k -u ${user}:${password} -X POST ${nexus URL} for uploading

curl -k -u ${user}:${password} -X GET ${nexus URL} for downloading

Default login for Nexus is

Username : admin

Password : admin123