

1. Check the current user and switch to Jenkins user

```
devops@ubuntu18:~$ whoami
devops
```

```
devops@ubuntu18:~$ hostname
ubuntu18
```

2. Switch to root user

```
devops@ubuntu18:~$ sudo su
[sudo] password for devops:
root@ubuntu18:/home/devops# whoami
root
root@ubuntu18:/home/devops# cd
root@ubuntu18:~# pwd
/root
```

3. Switch to Jenkins user

```
root@ubuntu18:~# su jenkins
jenkins@ubuntu18:/root$ whoami
jenkins
jenkins@ubuntu18:/root$ cd
jenkins@ubuntu18:~$ pwd
/var/lib/Jenkins
```

4. Returning to root

```
jenkins@ubuntu18:~$ exit
exit
root@ubuntu18:~#
```

```
Returning to main user account
root@ubuntu18:/home/devops# exit
exit
devops@ubuntu18:~$
```

5. Checking for Ansible availability:

```
jenkins@ubuntu18:~$ ansible --version
ansible 2.9.22
  config file = /etc/ansible/ansible.cfg
  configured module search path =
[u'/var/lib/jenkins/.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 = 2.7.17 (default, Feb 27 2021, 15:10:58) [GCC
7.5.0]
```

6. Connectivity ip address from current machine.

```
jenkins@ubuntu18:~$ cat /etc/hosts
127.0.0.1 localhost
```

7. Ping the local machine for connectivity

```
jenkins@ubuntu18:~$ ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.020 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.030 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.020 ms
^C
--- localhost ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2039ms
rtt min/avg/max/mdev = 0.020/0.023/0.030/0.006 ms
```

8. Use ansible command to push the ping module to local machine to check for connectivity

```
jenkins@ubuntu18:~$ ansible all -i 'localhost,' -m ping -c local
localhost | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

9. Ansible configuration and inventory files for connectivity

```
jenkins@ubuntu18:~$ ls /etc/ansible/
ansible.cfg  hosts  roles

jenkins@ubuntu18:~$ vi /etc/ansible/ansible.cfg
```

```
[defaults]

# some basic default values...

#inventory          = /etc/ansible/hosts
#library             = /usr/share/my_modules/
```

10. Uncomment the path of inventory file

```
root@ubuntu18:~# vi /etc/ansible/ansible.cfg
[defaults]
```

```
# some basic default values...

inventory      = /etc/ansible/hosts
#library       = /usr/share/my_modules/
```

11. Add following information in the inventory file

```
root@ubuntu18:~# vi /etc/ansible/hosts
# This is the default ansible 'hosts' file.
```

```
[local]
localhost
```

12. Use the ansible command without need for specifying the inventory in the command line.

```
jenkins@ubuntu18:~$ ansible all -m ping -c local
localhost | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

13. Add the connection information in the inventory file.

```
root@ubuntu18:~# vi /etc/ansible/hosts

# This is the default ansible 'hosts' file.

[local]
localhost ansible_connection=local
```

14. Use ansible command to push the ping module to all servers in the inventory file (only localhost as of now)

```
jenkins@ubuntu18:~$ ansible all -m ping
localhost | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

15. Use ansible command to push the ping module to all servers in the local group in inventory file (only localhost as of now)

```
jenkins@ubuntu18:~$ ansible local -m ping
localhost | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

16. Use ansible command to push the ping module to only localhost server in the inventory file (only localhost as of now)

```
jenkins@ubuntu18:~$ ansible localhost -m ping
localhost | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

17. Add the hostname in the /etc/hosts associating with ip address of server, which can be used to connect to local server through SSH

```
jenkins@ubuntu18:~$ hostname
ubuntu18
jenkins@ubuntu18:~$ cat /etc/hosts
127.0.0.1 localhost
192.168.56.102 ubuntu18
```

18. Ping the local server named as ubuntu18

```
jenkins@ubuntu18:~$ ping ubuntu18
PING ubuntu18 (192.168.56.102) 56(84) bytes of data.
64 bytes from ubuntu18 (192.168.56.102): icmp_seq=1 ttl=64
time=0.015 ms
64 bytes from ubuntu18 (192.168.56.102): icmp_seq=2 ttl=64
time=0.055 ms
^C
--- ubuntu18 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1011ms
rtt min/avg/max/mdev = 0.015/0.035/0.055/0.020 ms
```

19. SSH connect with the local server ubuntu18 through devops (or any local user whose password is known for ssh connectivity)

Ensure non availability of .ssh folder in the home directory

```
jenkins@ubuntu18:~$ rm -rf .ssh
jenkins@ubuntu18:~$ ls -a .ssh
ls: cannot access '.ssh': No such file or directory
```

Ensure ssh server is running the local server

```
jenkins@ubuntu18:~$ systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor
  preset: enabled)
   Active: active (running)
```

20. Following command creates .ssh folder if not there and will also create known_hosts file to remember ssh connections

```
jenkins@ubuntu18:~$ ssh devops@ubuntu18
The authenticity of host 'ubuntu18 (192.168.56.102)' can't be
established.
ECDSA key fingerprint is
SHA256:oqnaYoDQEA/bleR9j7rD4fofMuMn+Dr5mOQGXLQ0Mcs.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ubuntu18,192.168.56.102' (ECDSA) to the
list of known hosts.
devops@ubuntu18's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-147-generic x86_64)

Last login: Tue Jun 29 09:35:21 2021 from 192.168.56.1
devops@ubuntu18:~$ whoami
devops
devops@ubuntu18:~$ pwd
/home/devops
```

21. Returning Back to Jenkins user area, from ssh connectivity

```
devops@ubuntu18:~$ exit
logout
Connection to ubuntu18 closed.
jenkins@ubuntu18:~$ whoami
Jenkins
```

22. Add the ubuntu18 local server with its ip address in the inventory file

```
root@ubuntu18:~# vi /etc/ansible/hosts
# This is the default ansible 'hosts' file.

[local]
localhost ansible_connection=local
ubuntu18 ansible_host=192.168.56.102
```

23. Use ansible command to push ping module to local server ubuntu18 through devops user

```
jenkins@ubuntu18:~$ ansible ubuntu18 -m ping
ubuntu18 | UNREACHABLE! => {
  "changed": false,
  "msg": "Failed to connect to the host via ssh:
jenkins@192.168.56.102: Permission denied (publickey,password).",
  "unreachable": true
}
```

24. Above command fails because of need of user id and password as specified in the following command

```
jenkins@ubuntu18:~$ ansible ubuntu18 -m ping -u devops -k
SSH password:
ubuntu18 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

25. Add the user name in the inventory file

```
root@ubuntu18:~# vi /etc/ansible/hosts
# This is the default ansible 'hosts' file.

[local]
localhost ansible_connection=local
ubuntu18 ansible_host=192.168.56.102 ansible_user=devops
```

26. Redo the ansible command to push ping module without specifying devops user

```
jenkins@ubuntu18:~$ ansible ubuntu18 -m ping -k
SSH password:
ubuntu18 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

27. Generate the keypair using following commands

```
jenkins@ubuntu18:~$ ssh-keygen
```

```
jenkins@ubuntu18:~$ ls .ssh
id_rsa  id_rsa.pub  known_hosts
```

28. Push the public key to remote user area (devops in this case) associated with ubuntu18 local server

```
jenkins@ubuntu18:~$ ssh-copy-id -i /var/lib/jenkins/.ssh/id_rsa.pub
devops@ubuntu18
```

This would make ssh access passwordless, so that ansible command can be used as follows (avoiding -k)

```
jenkins@ubuntu18:~$ ansible ubuntu18 -m ping
ubuntu18 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
```

29. Connect remote machines “centclient” and “ubuntucient” in this case with their corresponding ip address which allow ssh access to those machines

```
jenkins@ubuntu18:~$ cat /etc/hosts
127.0.0.1 localhost
192.168.56.102 ubuntu18
192.168.56.131 centclient
192.168.56.132 ubuntucient
```

```
jenkins@ubuntu18:~$ ping centclient
PING centclient (192.168.56.131) 56(84) bytes of data.
64 bytes from centclient (192.168.56.131): icmp_seq=1 ttl=64
time=0.313 ms
64 bytes from centclient (192.168.56.131): icmp_seq=2 ttl=64
time=0.441 ms
^C
--- centclient ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1024ms
rtt min/avg/max/mdev = 0.313/0.377/0.441/0.064 ms
```

```
jenkins@ubuntu18:~$ ping ubuntucient
PING ubuntucient (192.168.56.132) 56(84) bytes of data.
64 bytes from ubuntucient (192.168.56.132): icmp_seq=1 ttl=64
time=0.392 ms
64 bytes from ubuntucient (192.168.56.132): icmp_seq=2 ttl=64
time=0.667 ms
^C
```

```
--- ubuntuclient ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1022ms
rtt min/avg/max/mdev = 0.392/0.529/0.667/0.139 ms
```

30. Push the public key into these remote machines to enable password less access.

```
jenkins@ubuntu18:~$ ssh-copy-id -i /var/lib/jenkins/.ssh/id_rsa.pub
devops@centclient
```

```
jenkins@ubuntu18:~$ ssh-copy-id -i /var/lib/jenkins/.ssh/id_rsa.pub
devops@ubuntuclient
```

31. Check the password-less ssh access to these remote machines

```
jenkins@ubuntu18:~$ ssh devops@centclient
Last login: Tue Jun 29 15:36:03 2021 from 192.168.56.102
[devops@centclient ~]$ exit
logout
Connection to centclient closed.
```

```
jenkins@ubuntu18:~$ ssh devops@ubuntuclient
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-142-generic x86_64)

Last login: Tue Jun 29 15:37:31 2021 from 192.168.56.102
devops@ubuntu16:~$ exit
logout
Connection to ubuntuclient closed.
jenkins@ubuntu18:~$
```

32. Make entries of these remote servers in the inventory file

```
jenkins@ubuntu18:~$ vi /etc/ansible/hosts
# This is the default ansible 'hosts' file.

[local]
localhost ansible_connection=local
ubuntu18 ansible_host=192.168.56.102 ansible_user=devops

[remote]
centclient ansible_host=192.168.56.131 ansible_user=devops
ubuntuclient ansible_host=192.168.56.132 ansible_user=devops
```

33. Push ping module to check connectivity to these remote servers

```
jenkins@ubuntu18:~$ ansible remote -m ping
centclient | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
```



```
    "changed": false,  
    "ping": "pong"  
}  
ubuntuclient | SUCCESS => {  
    "ansible_facts": {  
        "discovered_interpreter_python": "/usr/bin/python3"  
    },  
    "changed": false,  
    "ping": "pong"  
}
```