Configuring Ansible Managed Nodes

- **Create and distribute SSH keys to managed nodes**
- **Configure privilege escalation on managed nodes**
- **Validate a working configuration using ad hoc Ansible commands**

Task. Configure 'mhost4' to listen on non-default SSH port 555.

- **ansible** should be able to connect to **mhost4** on new SSH port as well as standard SSH port.
- Update the inventory file to tell ansible to use **port 555** to connect to **mhost4**.

Execute commands as root user :

```
ansible mhost4 -m lineinfile -a "path=/etc/ssh/sshd_config regexp='^#Port' line='Port 22'" -u root
ansible mhost4 -m lineinfile -a "path=/etc/ssh/sshd_config insertafter='^Port' line='Port 555'" -u root
ansible mhost4 -m seport -a "ports=555 proto=tcp setype=ssh_port_t state=present" -u root
ansible mhost4 -m firewalld -a "port=555/tcp state=enabled permanent=yes" -u root
ansible mhost4 -m service -a "name=firewalld state=reloaded" -u root
ansible mhost4 -m service -a "name=sshd state=restarted" -u root
```

Modify inventory file:

```
vim /home/ansible/tasks/mnodes
----
mhost4 ansible_port=555
---
:wq
```

Note: We need to install **policycoreutils*** package to use **seport module**. Install same using **dnf install policycoreutils*** after connecting VM to internet to use online repository.

Task. Generate SSH Keys for user 'ansible' on Ansible control node.

- Use ansible ad-hoc command to create user **ansible** on all managed nodes and copy the public key for **ansible** user to managed nodes.
- Execute this task as **root user**.
- Use password **password** for this user.

Execute command as ansible user:

```
ssh-keygen -t rsa
```

Execute commands as root user:

```
ansible all -m user -a "name=ansible state=present password='{{ 'password' | password_hash('sha256') }}'" -u root

ansible all -m authorized_key -a "user='ansible' state='present' key='{{ lookup('file', '/home/ansible/.ssh/id_rsa.pub') }}' path='/home/ansible/.ssh/authorized_keys'" -u root
```

Note: While creating user , never use plain text password instead use encrypted password creating using some hashing algorithm. Here we are using jinja2 filter **password_hash('sha256')** to generate encrypted password.

Task. Using ansible ad-hoc commands, Configure privilege escalation for user ‘ansible’ on all Managed hosts.

- User **ansible** should be able to use **sudo without providing password.**

Execute this command as root user:

```
ansible all -m lineinfile -a "path=/etc/sudoers state=present line='ansible ALL=(ALL) NOPASSWD: ALL' backup=yes
validate='/usr/sbin/visudo -cf %s'" -u root
```

Task. Use ansible ad-hoc command to configure MOTD on all managed hosts as “Welcome to Ansible managed host”

- Execute this command as **ansible** user.

Execute this command as ansible user:
ansible all -m copy -a “content=‘Welcome to Ansible managed host’ dest=/etc/motd’” --become

Task. Use ansible adhoc-commands to configure all managed nodes to use 'BaseOS' and 'AppStream' repos with following information.

For BaseOS Repository:

- name=BaseOS
- description= DNF BaseOS Repo
- baseurl=file:///root/BaseOS
- gpgcheck=1
- gpgkey=/etc/pki/rpm-gpg/RPM-GPG-KEY-centosoffcial
- enabled=1

For AppStream Repository:

- name=AppStream
- description= DNF AppStream Repo
- baseurl=file:///root/AppStream
- gpgcheck=1
- gpgkey=/etc/pki/rpm-gpg/RPM-GPG-KEY-centosoffcial
- enabled=1

Note: AppStream and BaseOS Repositories are already created at path **/root/AppStream** and **/root/BaseOS** on all Managed Nodes.

Execute commands as ansible user:

ansible all -m yum_repository -a “name=BaseOS description=‘DNF BaseOS Repo’ baseurl=file:///root/BaseOS gpgcheck=1 gpgkey=/etc/pki/rpm-gpg/RPM-GPG-KEY-centosoffcial enabled=1 file=BaseOS” --become

ansible all -m yum_repository -a “name=AppStream description=‘DNF AppStream Repo’ baseurl=file:///root/AppStream gpgcheck=1 gpgkey=/etc/pki/rpm-gpg/RPM-GPG-KEY-centosoffcial enabled=1 file=AppStream” --become

Note: Move all files already present at **/etc/yum.repos.d/** path to **/tmp** directory in case you need them to use online repos otherwise you can delete them**.**

To Disable GPG Key Check

Execute commands as ansible user:

ansible all -m yum_repository -a “name=BaseOS description=‘DNF BaseOS Repo’ baseurl=file:///root/BaseOS gpgcheck=0 enabled=1 file=BaseOS” --become

ansible all -m yum_repository -a “name=AppStream description=‘DNF AppStream Repo’ baseurl=file:///root/AppStream gpgcheck=0 enabled=1 file=AppStream” --become