

## Ansible roles

- Structure is important
- roles
  - create\_user
    - tasks
    - main.yaml
- Reference the role in a main.yaml in a folder called create\_user. In that main.yaml do what you want to execute in the playbook, so for example adding a user. To reference the actual task in the playbook you can simply call it by: create\_user, the same role name as the folder which was created
- Static files can be used by creating an extra folder in the roles/ start\_containers/files <add file>
- vars folder in the roles folder can be used to specify variables for that specific role also with main.yaml
- groups is a reserved word in ansible

## Ansible inventory

- file containing data about server being managed by ansible
- hosts meaning managed servers
- etc/ansible/hosts

Set host file: 157.245.75.110 ansible\_ssh\_private\_key\_file=~/.ssh/digital\_ocean\_macbook ansible\_user=root

- ansible all -i hosts -m ping / to verify connection can be made
- ssh-keyscan -H 128.199.62.148 >> ~/.ssh/known\_hosts (add key to known\_hosts)
- ssh-copy-id -i ~/.ssh/digital\_ocean\_macbook root@174.138.7.46 (this copies the local key to the server you want the ssh to be on)

— Variables

{{var}}

- Variable could be done via file or via CLI

—

.

chown -R (user)nexus:(group)nexus (folder)nexus-3.28.1-01

## Managing Host key checking

- ssh-keyscan -H <IP> >> ~/.ssh/known\_hosts this removed manual intervention and adds it to known hosts
- ssh-copy-id root@<host> this adds the ssh key to the server
- Static servers is where this is used often

————

- Dynamic infrastructure can disable hostKeyChecking
- ~/.ansible.cfg host\_key\_checking = False

## **Playbooks**

- Hosts should be defined in hostsfile you can use the [bracketName]
- name: Install nginx
- apt: is in this case the module name which resembles apt install
- <name of package>=1.24
- Collection = Playbook, Modules, Plugins