pb186007@WINPB186007-S9F MINGW64 ~/Desktop

$ vi /etc/ssh/sshd\_config^C

pb186007@WINPB186007-S9F MINGW64 ~/Desktop

$ ssh -i "redhat1\_key.pem" ec2-user@ec2-3-16-167-196.us-east-2.compute.amazonaws.com

Last login: Sun Dec 30 10:45:42 2018 from 27.6.44.45

[ec2-user@controller ~]$ ====================================

-bash: ====================================: command not found

[ec2-user@controller ~]$ \*\*\*\*\*\*\*\* Ansible Practise \*\*\*\*\*\*\*\*\*\*\*

-bash: \*\*\*\*\*\*\*\*: command not found

[ec2-user@controller ~]$ ====================================

-bash: ====================================: command not found

[ec2-user@controller ~]$

[ec2-user@controller ~]$

[ec2-user@controller ~]$ yum update

Loaded plugins: amazon-id, rhui-lb, search-disabled-repos

Repo rhui-REGION-client-config-server-7 forced skip\_if\_unavailable=True due to: /etc/pki/rhui/cdn.redhat.com-chain.crt

Repo rhui-REGION-client-config-server-7 forced skip\_if\_unavailable=True due to: /etc/pki/rhui/product/rhui-client-config-server-7.crt

Repo rhui-REGION-client-config-server-7 forced skip\_if\_unavailable=True due to: /etc/pki/rhui/rhui-client-config-server-7.key

Repo rhui-REGION-rhel-server-releases forced skip\_if\_unavailable=True due to: /etc/pki/rhui/cdn.redhat.com-chain.crt

Repo rhui-REGION-rhel-server-releases forced skip\_if\_unavailable=True due to: /etc/pki/rhui/product/content-rhel7.crt

Repo rhui-REGION-rhel-server-releases forced skip\_if\_unavailable=True due to: /etc/pki/rhui/content-rhel7.key

Repo rhui-REGION-rhel-server-rh-common forced skip\_if\_unavailable=True due to: /etc/pki/rhui/cdn.redhat.com-chain.crt

Repo rhui-REGION-rhel-server-rh-common forced skip\_if\_unavailable=True due to: /etc/pki/rhui/product/content-rhel7.crt

Repo rhui-REGION-rhel-server-rh-common forced skip\_if\_unavailable=True due to: /etc/pki/rhui/content-rhel7.key

You need to be root to perform this command.

[ec2-user@controller ~]$ sudo -i

[root@controller ~]# yu, update

-bash: yu,: command not found

[root@controller ~]# yum update

Loaded plugins: amazon-id, rhui-lb, search-disabled-repos

No packages marked for update

[root@controller ~]# yum install ansible

Loaded plugins: amazon-id, rhui-lb, search-disabled-repos

Package ansible-2.7.5-1.el7.noarch already installed and latest version

Nothing to do

[root@controller ~]#

[root@controller ~]#

[root@controller ~]# ansible --version

ansible 2.7.5

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/site-packages/ansible

executable location = /bin/ansible

python version = 2.7.5 (default, Sep 12 2018, 05:31:16) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]

[root@controller ~]#

[root@controller ~]#

[root@controller ~]# yum update

Loaded plugins: amazon-id, rhui-lb, search-disabled-repos

No packages marked for update

[root@controller ~]#

[root@controller ~]#

[root@controller ~]# adduser homeuser

[root@controller ~]# passwd homeuser

Changing password for user homeuser.

New password:

BAD PASSWORD: The password is shorter than 8 characters

Retype new password:

passwd: all authentication tokens updated successfully.

[root@controller ~]#

[root@controller ~]#

[root@controller ~]# sudo vi /etc/sudoers

[root@controller ~]# vi /etc/sudoers

[root@controller ~]# tail -n 5 /etc/sudoers

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)

#includedir /etc/sudoers.d

ec2-user ALL=(ALL) NOPASSWD: ALL

demouser ALL=(ALL) NOPASSWD: ALL

[root@controller ~]# vi /etc/sudoers

[root@controller ~]#

[root@controller ~]# tail -n 5 /etc/sudoers

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)

#includedir /etc/sudoers.d

ec2-user ALL=(ALL) NOPASSWD: ALL

demouser ALL=(ALL) NOPASSWD: ALL

homeuser ALL=(ALL) NOPASSWD: ALL

[root@controller ~]#

[root@controller ~]#

[root@controller ~]# vi /etc/ssh/sshd\_config

[root@controller ~]# cd /etc/ssh/

[root@controller ssh]# ls -lrt ssh\*

-rw-r--r--. 1 root root 2276 Nov 24 2017 ssh\_config

-rw-r--r--. 1 root root 382 Dec 18 02:17 ssh\_host\_rsa\_key.pub

-rw-r-----. 1 root ssh\_keys 1679 Dec 18 02:17 ssh\_host\_rsa\_key

-rw-r--r--. 1 root root 162 Dec 18 02:17 ssh\_host\_ecdsa\_key.pub

-rw-r-----. 1 root ssh\_keys 227 Dec 18 02:17 ssh\_host\_ecdsa\_key

-rw-r--r--. 1 root root 82 Dec 18 02:17 ssh\_host\_ed25519\_key.pub

-rw-r-----. 1 root ssh\_keys 387 Dec 18 02:17 ssh\_host\_ed25519\_key

-rw-------. 1 root root 3955 Dec 27 03:38 sshd\_config

[root@controller ssh]# vi sshd\_config

[root@controller ssh]#

[root@controller ssh]# su - homeuser

[homeuser@controller ~]$ ssh-keygen

Generating public/private rsa key pair.

Enter file in which to save the key (/home/homeuser/.ssh/id\_rsa):

Created directory '/home/homeuser/.ssh'.

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /home/homeuser/.ssh/id\_rsa.

Your public key has been saved in /home/homeuser/.ssh/id\_rsa.pub.

The key fingerprint is:

SHA256:ixo9KkmsyYR0kMH0dwX3HiVqbXI8ginhjhNJtxa4zT8 homeuser@controller

The key's randomart image is:

+---[RSA 2048]----+

|ooo..+ ..o . . |

| +o.+ + = = o |

| .++\* + = O |

| . o\*oo . \* o |

|o..o .. S . |

|..o .. E . |

|o+ .. + o |

|o.o + . |

| .o |

+----[SHA256]-----+

[homeuser@controller ~]$ su - homeuser

Password:

[homeuser@controller ~]$ ssh-copy-id homeuser@172.31.16.187

/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/homeuser/.ssh/id\_rsa.pub"

The authenticity of host '172.31.16.187 (172.31.16.187)' can't be established.

ECDSA key fingerprint is SHA256:eYswYlId7zTZuEpAoNrMp48HWu4D5qAo5+Rw7HCkK90.

ECDSA key fingerprint is MD5:a4:05:a4:09:4d:e3:fe:4e:d8:22:2f:30:1e:2f:0b:eb.

Are you sure you want to continue connecting (yes/no)? yes

/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed

/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys

homeuser@172.31.16.187's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'homeuser@172.31.16.187'"

and check to make sure that only the key(s) you wanted were added.

[homeuser@controller ~]$ ssh 'homeuser@172.31.16.187'

Last login: Sun Dec 30 11:04:59 2018

[homeuser@remote\_machine1 ~]$ exit

logout

Connection to 172.31.16.187 closed.

[homeuser@controller ~]$ pwd

/home/homeuser

[homeuser@controller ~]$ cd /etc/ansible/

[homeuser@controller ansible]$ ls

28th-nov.rtf copy\_playbook.yaml inventoryvar\_playbook.yaml

5th\_dec.rtf file1.txt playbook1.yaml

5th\_dec\_sql\_assignment.rtf Git roledemo.yaml

6th Dec group\_vars roles

ansible hosts tomcat\_palybook.yaml

ansible.cfg hosts.org

chef index.html

[homeuser@controller ansible]$ vi hosts

[homeuser@controller ansible]$ sudo hosts hosts\_demouser

sudo: hosts: command not found

[homeuser@controller ansible]$ sudo cp hosts hosts\_demouser

[homeuser@controller ansible]$ sudo vi hosts

[homeuser@controller ansible]$ rm -rf 28th-nov.rtf 5th\_dec\* 6th\ Dec/ ansible ansible.cfg chef/ file1.txt Git/

rm: cannot remove ‘28th-nov.rtf’: Permission denied

rm: cannot remove ‘5th\_dec.rtf’: Permission denied

rm: cannot remove ‘5th\_dec\_sql\_assignment.rtf’: Permission denied

rm: cannot remove ‘6th Dec/6th\_dec\_playground\_leader.docx’: Permission denied

rm: cannot remove ‘6th Dec/6th\_dec\_playground\_worker1.docx’: Permission denied

rm: cannot remove ‘6th Dec/terminal output.rtf’: Permission denied

rm: cannot remove ‘6th Dec/~$h\_dec\_playground\_worker1.docx’: Permission denied

rm: cannot remove ‘ansible/27th dec/controller.docx’: Permission denied

rm: cannot remove ‘ansible/27th dec/remote node.docx’: Permission denied

rm: cannot remove ‘ansible/28th dec/controller.docx’: Permission denied

rm: cannot remove ‘ansible/28th dec/work station.docx’: Permission denied

rm: cannot remove ‘ansible/29th dec/controller.docx’: Permission denied

rm: cannot remove ‘ansible/Ansible\_Install (1).docx’: Permission denied

rm: cannot remove ‘ansible/New Text Document.txt’: Permission denied

rm: cannot remove ‘ansible.cfg’: Permission denied

rm: cannot remove ‘chef/18th dec/chef server.txt’: Permission denied

rm: cannot remove ‘chef/18th dec/cygtermd.exe.stackdump’: Permission denied

rm: cannot remove ‘chef/18th dec/node1 output for httpd service.rtf’: Permission denied

rm: cannot remove ‘chef/18th dec/workstation output.rtf’: Permission denied

rm: cannot remove ‘chef/19th dec/node1.rtf’: Permission denied

rm: cannot remove ‘chef/19th dec/workstation.rtf’: Permission denied

rm: cannot remove ‘chef/20th dec/node1 readhat.docx’: Permission denied

rm: cannot remove ‘chef/20th dec/node2 ubuntu.docx’: Permission denied

rm: cannot remove ‘chef/20th dec/workstation.docx’: Permission denied

rm: cannot remove ‘chef/21st dec/New Text Document.txt’: Permission denied

rm: cannot remove ‘chef/21st dec/node.docx’: Permission denied

rm: cannot remove ‘chef/22nd dec/Kitchen Setup (1).docx’: Permission denied

rm: cannot remove ‘chef/22nd dec/Node 1.docx’: Permission denied

rm: cannot remove ‘chef/22nd dec/node created by kitchen.docx’: Permission denied

rm: cannot remove ‘chef/22nd dec/workstation.docx’: Permission denied

rm: cannot remove ‘chef/chef installation guide.docx’: Permission denied

rm: cannot remove ‘file1.txt’: Permission denied

rm: cannot remove ‘Git/11th dec.docx’: Permission denied

rm: cannot remove ‘Git/13th dec.docx’: Permission denied

rm: cannot remove ‘Git/17th dec.docx’: Permission denied

rm: cannot remove ‘Git/git 1st day.docx’: Permission denied

rm: cannot remove ‘Git/sites.txt’: Permission denied

[homeuser@controller ansible]$ sudo rm -rf 28th-nov.rtf 5th\_dec\* 6th\ Dec/ ansible ansible.cfg chef/ file1.txt Git/

[homeuser@controller ansible]$ ls -lrt

total 36

-rw-r--r--. 1 root root 1016 Dec 28 01:58 hosts.org

-rw-r--r--. 1 root root 47 Dec 28 03:04 index.html

-rw-r--r--. 1 root root 324 Dec 28 03:06 playbook1.yaml

-rw-r--r--. 1 root root 443 Dec 28 20:43 tomcat\_palybook.yaml

-rw-r--r--. 1 root root 217 Dec 29 03:54 copy\_playbook.yaml

-rw-r--r--. 1 root root 143 Dec 29 04:00 inventoryvar\_playbook.yaml

drwxr-xr-x. 2 root root 28 Dec 29 04:17 group\_vars

drwxr-xr-x. 3 root root 26 Dec 29 05:24 roles

-rw-r--r--. 1 root root 46 Dec 29 05:48 roledemo.yaml

-rw-r--r--. 1 root root 106 Dec 30 11:08 hosts\_demouser

-rw-r--r--. 1 root root 104 Dec 30 11:08 hosts

[homeuser@controller ansible]$ mkdir demouser\_contents

mkdir: cannot create directory ‘demouser\_contents’: Permission denied

[homeuser@controller ansible]$ sudo mkdir demouser\_contents

[homeuser@controller ansible]$ sudo \*yaml ./demouser\_contents/

sudo: copy\_playbook.yaml: command not found

[homeuser@controller ansible]$ sudo mv \*yaml ./demouser\_contents/

[homeuser@controller ansible]$ ls -lrt

total 16

-rw-r--r--. 1 root root 1016 Dec 28 01:58 hosts.org

-rw-r--r--. 1 root root 47 Dec 28 03:04 index.html

drwxr-xr-x. 2 root root 28 Dec 29 04:17 group\_vars

drwxr-xr-x. 3 root root 26 Dec 29 05:24 roles

-rw-r--r--. 1 root root 106 Dec 30 11:08 hosts\_demouser

-rw-r--r--. 1 root root 104 Dec 30 11:08 hosts

drwxr-xr-x. 2 root root 137 Dec 30 11:10 demouser\_contents

[homeuser@controller ansible]$ sudo mv hosts\_demouser ./demouser\_contents/

[homeuser@controller ansible]$ ls

demouser\_contents group\_vars hosts hosts.org index.html roles

[homeuser@controller ansible]$ tree

.

├── demouser\_contents

│   ├── copy\_playbook.yaml

│   ├── hosts\_demouser

│   ├── inventoryvar\_playbook.yaml

│   ├── playbook1.yaml

│   ├── roledemo.yaml

│   └── tomcat\_palybook.yaml

├── group\_vars

│   └── demogroup.yaml

├── hosts

├── hosts.org

├── index.html

└── roles

└── myhttpd\_role

├── defaults

│   └── main.yml

├── files

├── handlers

│   └── main.yml

├── meta

│   └── main.yml

├── README.md

├── tasks

│   ├── install.yaml

│   ├── main.yml

│   └── start.yaml

├── templates

├── tests

│   ├── inventory

│   └── test.yml

└── vars

└── main.yml

12 directories, 20 files

[homeuser@controller ansible]$ pwd

/etc/ansible

[homeuser@controller ansible]$ more hosts

[mygroup]

172.31.16.187

#[demogroup:vars]

# my\_source=/etc/ansible/file1.txt

# my\_dest=/home/demouser

[homeuser@controller ansible]$

[homeuser@controller ansible]$ ansible wegserver -m ping

[WARNING]: Could not match supplied host pattern, ignoring: wegserver

[WARNING]: No hosts matched, nothing to do

[homeuser@controller ansible]$ ansible webserver -m ping

[WARNING]: Could not match supplied host pattern, ignoring: webserver

[WARNING]: No hosts matched, nothing to do

[homeuser@controller ansible]$ ansible homeuser -m ping

[WARNING]: Could not match supplied host pattern, ignoring: homeuser

[WARNING]: No hosts matched, nothing to do

[homeuser@controller ansible]$ ansible mygroup -m ping

172.31.16.187 | SUCCESS => {

"changed": false,

"ping": "pong"

}

[homeuser@controller ansible]$ pwd

/etc/ansible

[homeuser@controller ansible]$ ls -lrt

total 12

-rw-r--r--. 1 root root 1016 Dec 28 01:58 hosts.org

-rw-r--r--. 1 root root 47 Dec 28 03:04 index.html

drwxr-xr-x. 2 root root 28 Dec 29 04:17 group\_vars

drwxr-xr-x. 3 root root 26 Dec 29 05:24 roles

-rw-r--r--. 1 root root 104 Dec 30 11:08 hosts

drwxr-xr-x. 2 root root 159 Dec 30 11:11 demouser\_contents

[homeuser@controller ansible]$ cd demouser\_contents/

[homeuser@controller demouser\_contents]$ ls

copy\_playbook.yaml inventoryvar\_playbook.yaml roledemo.yaml

hosts\_demouser playbook1.yaml tomcat\_palybook.yaml

[homeuser@controller demouser\_contents]$ more hosts\_demouser

[demogroup]

172.31.16.187

#[demogroup:vars]

# my\_source=/etc/ansible/file1.txt

# my\_dest=/home/demouser

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ rpmquery

rpmquery: no arguments given for query

[homeuser@controller demouser\_contents]$ rpmquery ansible

ansible-2.7.5-1.el7.noarch

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ pwd

/etc/ansible/demouser\_contents

[homeuser@controller demouser\_contents]$ ls

copy\_playbook.yaml inventoryvar\_playbook.yaml roledemo.yaml

hosts\_demouser playbook1.yaml tomcat\_palybook.yaml

[homeuser@controller demouser\_contents]$ cat \*

---

- hosts: demogroup

vars:

my\_source: /etc/ansible/file1.txt

my\_dest: /home/demouser

tasks:

- name: example copying file for demonstrating variable concept

copy: src={{my\_source}} dest={{my\_dest}}

[demogroup]

172.31.16.187

#[demogroup:vars]

# my\_source=/etc/ansible/file1.txt

# my\_dest=/home/demouser

---

- hosts: demogroup

tasks:

- name: example copying file for demonstrating variable concept

copy: src={{my\_source}} dest={{my\_dest}}

---

- hosts: demogroup

tasks:

- name: Installing httpd

yum:

name: httpd

state: present

- name: starting httpd

service:

name: httpd

state: started

- name: example copying file with owner and permissions

copy:

src: /etc/ansible/index.html

dest: /var/www/html/index.html

---

- hosts: all

roles:

- myhttpd\_role

---

- hosts: demogroup

tasks:

- name: Installing tomcat

yum:

name: tomcat

state: present

- name: starting tomcat

service:

name: tomcat

state: started

- name: Downloading war file to /usr/share/tomcat/webapps/benefits.war

get\_url:

url: 'https://www.oracle.com/webfolder/technetwork/tutorials/obe/fmw/wls/12c/03-DeployApps/files/benefits.war'

dest: /usr/share/tomcat/webapps/benefits.war

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ pwd

/etc/ansible/demouser\_contents

[homeuser@controller demouser\_contents]$ cd ..

[homeuser@controller ansible]$ ls

demouser\_contents group\_vars hosts hosts.org index.html roles

[homeuser@controller ansible]$ cat \*

cat: demouser\_contents: Is a directory

cat: group\_vars: Is a directory

[mygroup]

172.31.16.187

#[demogroup:vars]

# my\_source=/etc/ansible/file1.txt

# my\_dest=/home/demouser

# This is the default ansible 'hosts' file.

#

# It should live in /etc/ansible/hosts

#

# - Comments begin with the '#' character

# - Blank lines are ignored

# - Groups of hosts are delimited by [header] elements

# - You can enter hostnames or ip addresses

# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com

## blue.example.com

## 192.168.100.1

## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

## [webservers]

## alpha.example.org

## beta.example.org

## 192.168.1.100

## 192.168.1.110

# If you have multiple hosts following a pattern you can specify

# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]

##

## db01.intranet.mydomain.net

## db02.intranet.mydomain.net

## 10.25.1.56

## 10.25.1.57

# Here's another example of host ranges, this time there are no

# leading 0s:

## db-[99:101]-node.example.com

This is the index page from controller machine

cat: roles: Is a directory

[homeuser@controller ansible]$

[homeuser@controller ansible]$

[homeuser@controller ansible]$

[homeuser@controller ansible]$ cd roles

[homeuser@controller roles]$ ls

myhttpd\_role

[homeuser@controller roles]$ cd myhttpd\_role/

[homeuser@controller myhttpd\_role]$

[homeuser@controller myhttpd\_role]$

[homeuser@controller myhttpd\_role]$

[homeuser@controller myhttpd\_role]$ pwd

/etc/ansible/roles/myhttpd\_role

[homeuser@controller myhttpd\_role]$ ls

defaults files handlers meta README.md tasks templates tests vars

[homeuser@controller myhttpd\_role]$ pwd

/etc/ansible/roles/myhttpd\_role

[homeuser@controller myhttpd\_role]$

[homeuser@controller myhttpd\_role]$

[homeuser@controller myhttpd\_role]$ tree

.

├── defaults

│   └── main.yml

├── files

├── handlers

│   └── main.yml

├── meta

│   └── main.yml

├── README.md

├── tasks

│   ├── install.yaml

│   ├── main.yml

│   └── start.yaml

├── templates

├── tests

│   ├── inventory

│   └── test.yml

└── vars

└── main.yml

8 directories, 10 files

[homeuser@controller myhttpd\_role]$ cat defaults/main.yml

---

# defaults file for myhttpd\_role[homeuser@controller myhttpd\_role]$

[homeuser@controller myhttpd\_role]$ cd tasks/

[homeuser@controller tasks]$ cat install.yaml

- name: Installing httpd

yum:

name: httpd

state: present

[homeuser@controller tasks]$ cat main.yml

---

# tasks file for myhttpd\_role

- import\_tasks: install.yaml

- import\_tasks: start.yaml

[homeuser@controller tasks]$ cat start.yaml

- name: Starting httpd

service:

name: httpd

state: started

[homeuser@controller tasks]$ cd ../te

templates/ tests/

[homeuser@controller tasks]$ cd ../tests/

[homeuser@controller tests]$ cat inventory

localhost

[homeuser@controller tests]$ cat test.yml

---

- hosts: localhost

remote\_user: root

roles:

- myhttpd\_role[homeuser@controller tests]$

[homeuser@controller tests]$ cd ../vars/

[homeuser@controller vars]$ cat main.yml

---

# vars file for myhttpd\_role[homeuser@controller vars]$

[homeuser@controller vars]$

[homeuser@controller vars]$ pwd

/etc/ansible/roles/myhttpd\_role/vars

[homeuser@controller vars]$ cd ../..

[homeuser@controller roles]$ ls

myhttpd\_role

[homeuser@controller roles]$ cd ..

[homeuser@controller ansible]$ ls

demouser\_contents group\_vars hosts hosts.org index.html roles

[homeuser@controller ansible]$ cd demouser\_contents/

[homeuser@controller demouser\_contents]$ ls

copy\_playbook.yaml inventoryvar\_playbook.yaml roledemo.yaml

hosts\_demouser playbook1.yaml tomcat\_palybook.yaml

[homeuser@controller demouser\_contents]$ cat copy\_playbook.yaml

---

- hosts: demogroup

vars:

my\_source: /etc/ansible/file1.txt

my\_dest: /home/demouser

tasks:

- name: example copying file for demonstrating variable concept

copy: src={{my\_source}} dest={{my\_dest}}

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ cat inventoryvar\_playbook.yaml

---

- hosts: demogroup

tasks:

- name: example copying file for demonstrating variable concept

copy: src={{my\_source}} dest={{my\_dest}}

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ cat roledemo.yaml

---

- hosts: all

roles:

- myhttpd\_role

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ cat hosts\_demouser

[demogroup]

172.31.16.187

#[demogroup:vars]

# my\_source=/etc/ansible/file1.txt

# my\_dest=/home/demouser

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ cat playbook1.yaml

---

- hosts: demogroup

tasks:

- name: Installing httpd

yum:

name: httpd

state: present

- name: starting httpd

service:

name: httpd

state: started

- name: example copying file with owner and permissions

copy:

src: /etc/ansible/index.html

dest: /var/www/html/index.html

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ cat tomcat\_palybook.yaml

---

- hosts: demogroup

tasks:

- name: Installing tomcat

yum:

name: tomcat

state: present

- name: starting tomcat

service:

name: tomcat

state: started

- name: Downloading war file to /usr/share/tomcat/webapps/benefits.war

get\_url:

url: 'https://www.oracle.com/webfolder/technetwork/tutorials/obe/fmw/wls/12c/03-DeployApps/files/benefits.war'

dest: /usr/share/tomcat/webapps/benefits.war

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$

[homeuser@controller demouser\_contents]$ pwd

/etc/ansible/demouser\_contents

[homeuser@controller demouser\_contents]$ cd ..

[homeuser@controller ansible]$ ls

demouser\_contents group\_vars hosts hosts.org index.html roles

[homeuser@controller ansible]$ cd group\_vars/

[homeuser@controller group\_vars]$ ls

demogroup.yaml

[homeuser@controller group\_vars]$ cat demogroup.yaml

my\_source: /etc/ansible/file1.txt

my\_dest: /home/demouser

[homeuser@controller group\_vars]$

[homeuser@controller group\_vars]$

[homeuser@controller group\_vars]$ pwd

/etc/ansible/group\_vars

[homeuser@controller group\_vars]$ cd ..

[homeuser@controller ansible]$ cat hosts

[mygroup]

172.31.16.187

#[demogroup:vars]

# my\_source=/etc/ansible/file1.txt

# my\_dest=/home/demouser

[homeuser@controller ansible]$

[homeuser@controller ansible]$ cat hosts.org

# This is the default ansible 'hosts' file.

#

# It should live in /etc/ansible/hosts

#

# - Comments begin with the '#' character

# - Blank lines are ignored

# - Groups of hosts are delimited by [header] elements

# - You can enter hostnames or ip addresses

# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com

## blue.example.com

## 192.168.100.1

## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

## [webservers]

## alpha.example.org

## beta.example.org

## 192.168.1.100

## 192.168.1.110

# If you have multiple hosts following a pattern you can specify

# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]

##

## db01.intranet.mydomain.net

## db02.intranet.mydomain.net

## 10.25.1.56

## 10.25.1.57

# Here's another example of host ranges, this time there are no

# leading 0s:

## db-[99:101]-node.example.com

[homeuser@controller ansible]$

[homeuser@controller ansible]$ cat index.html

This is the index page from controller machine

[homeuser@controller ansible]$

[homeuser@controller ansible]$

[homeuser@controller ansible]$ history

1 ssh-keygen

2 su - homeuser

3 ssh-copy-id homeuser@172.31.16.187

4 ssh 'homeuser@172.31.16.187'

5 pwd

6 cd /etc/ansible/

7 ls

8 vi hosts

9 sudo hosts hosts\_demouser

10 sudo cp hosts hosts\_demouser

11 sudo vi hosts

12 rm -rf 28th-nov.rtf 5th\_dec\* 6th\ Dec/ ansible ansible.cfg chef/ file1.txt Git/

13 sudo rm -rf 28th-nov.rtf 5th\_dec\* 6th\ Dec/ ansible ansible.cfg chef/ file1.txt Git/

14 ls -lrt

15 mkdir demouser\_contents

16 sudo mkdir demouser\_contents

17 sudo \*yaml ./demouser\_contents/

18 sudo mv \*yaml ./demouser\_contents/

19 ls -lrt

20 sudo mv hosts\_demouser ./demouser\_contents/

21 ls

22 tree

23 pwd

24 more hosts

25 ansible wegserver -m ping

26 ansible webserver -m ping

27 ansible homeuser -m ping

28 ansible mygroup -m ping

29 pwd

30 ls -lrt

31 cd demouser\_contents/

32 ls

33 more hosts\_demouser

34 rpmquery

35 rpmquery ansible

36 pwd

37 ls

38 cat \*

39 pwd

40 cd ..

41 ls

42 cat \*

43 cd roles

44 ls

45 cd myhttpd\_role/

46 pwd

47 ls

48 pwd

49 tree

50 cat defaults/main.yml

51 cd tasks/

52 cat install.yaml

53 cat main.yml

54 cat start.yaml

55 cd ../tests/

56 cat inventory

57 cat test.yml

58 cd ../vars/

59 cat main.yml

60 pwd

61 cd ../..

62 ls

63 cd ..

64 ls

65 cd demouser\_contents/

66 ls

67 cat copy\_playbook.yaml

68 cat inventoryvar\_playbook.yaml

69 cat roledemo.yaml

70 cat hosts\_demouser

71 cat playbook1.yaml

72 cat tomcat\_palybook.yaml

73 pwd

74 cd ..

75 ls

76 cd group\_vars/

77 ls

78 cat demogroup.yaml

79 pwd

80 cd ..

81 cat hosts

82 cat hosts.org

83 cat index.html

84 history

[homeuser@controller ansible]$

[homeuser@controller ansible]$

[homeuser@controller ansible]$