1. Install and configure Ansible install & config Ansible on the control node classroom.example.com

\* install the require packages

\* create a static file called /home/admin/ansible/inventory so that:

- system1 is a member of the dev host group

- system2 is member of the test host group

- system3 and system4 are member of the prod host group

- system5 is member of the balancers host group

- The prod group is a member of the webservers host group

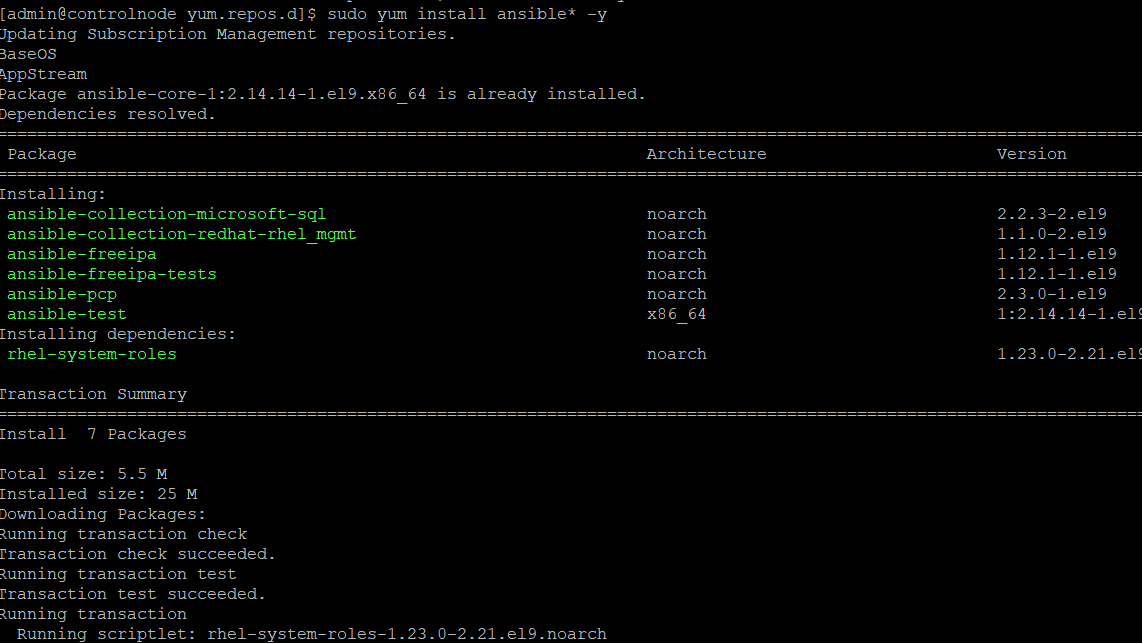
\* Create a configuration file calles /home/admin/ansible/ansible.cfg

- The host inventory file is /home/admin/ansible/inventory

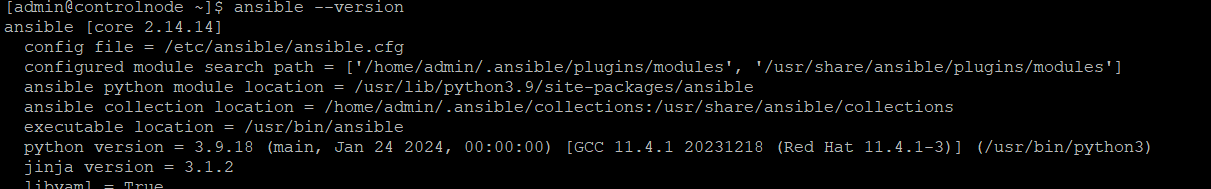
- The location of role used in playbooks include /home/admin/ansible/roles

Ans-

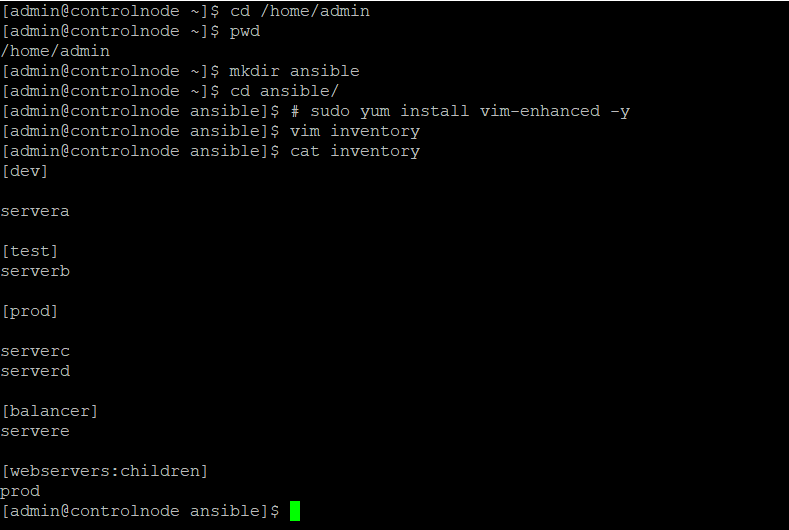
Steps 1. # sudo yum install ansible\* -y



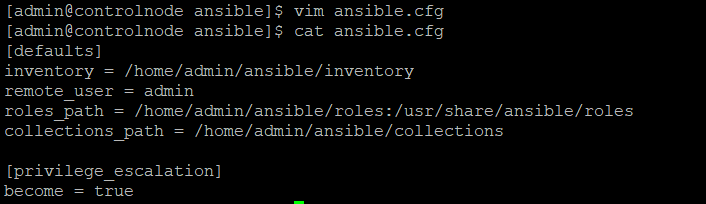
2. # asnsible –version

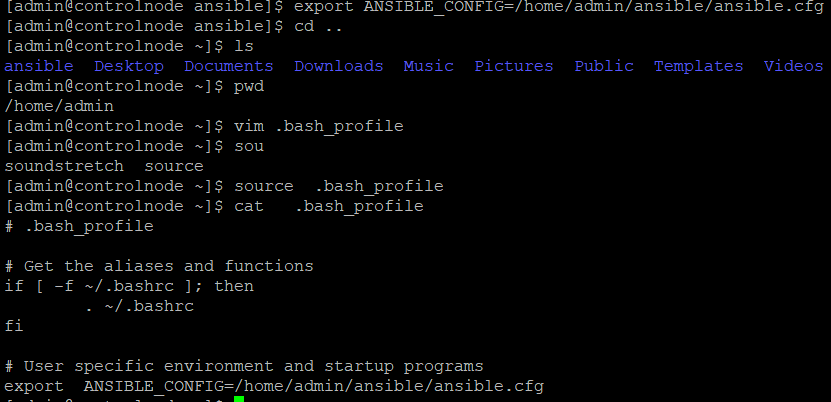


3. cd /home/admin -🡪 # mkdir ansible 🡪 # vim inventory

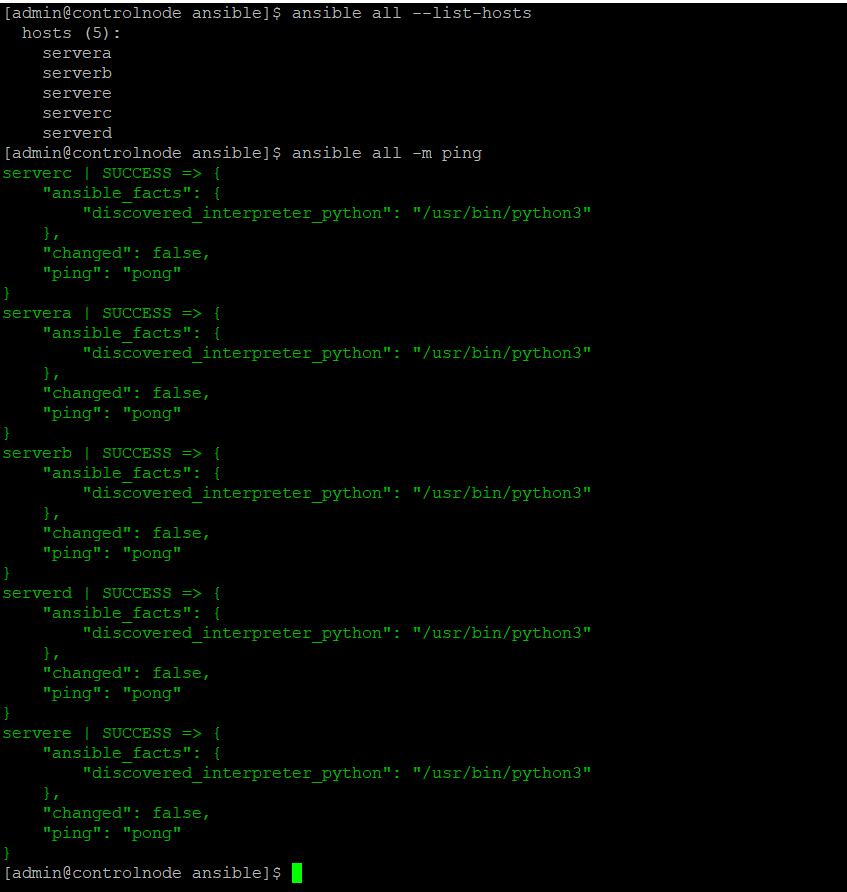


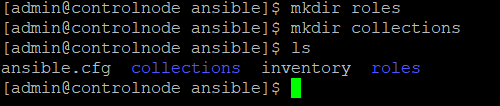
4. cd /home/admin/ansible 🡪 vim ansible.cfg



5. export the ansible.cfg file

6. check the all hosts list and also check the all server are pinging correctly.





Q2. Create and run Ansible ad-hoc commands

As a system administrator you will need install software on the managed nodes.

Create a yaml file called /home/admin/ansible/adhoc.yml the user ansible ad-hoc command to create a yum repository on each of the managed nodes as follows:

Repository 1: -

The name of the repository is EX294\_BASE

- The description is EX294 base software

- The base URL is <http://server.network.example.com/BaseOs>

- GPG signature checking is enabled

- The GPG Key URL is <http://server.network.example.com/RHEL/RPM-GPG-KEY-redhat-release>

- The repository is enabled

Repository 2:

- The name of the repository is EX294\_STREM

- The description is EX294 base software

- The base URL is <http://server.network.example.com/AppStream>

- GPG signature checking is enabled

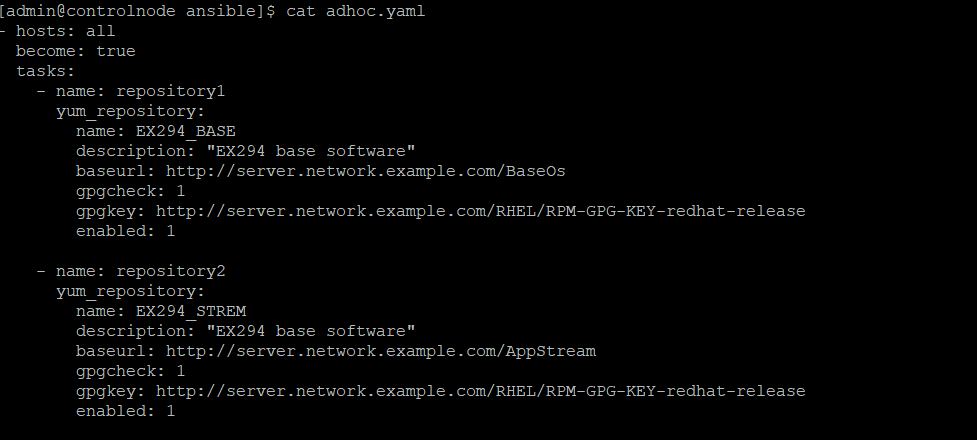
- The GPG Key URL is <http://server.network.example.com/RHEL/RPM-GPG-KEY-redhat-release>

- The repository is enabled

**Ans:**

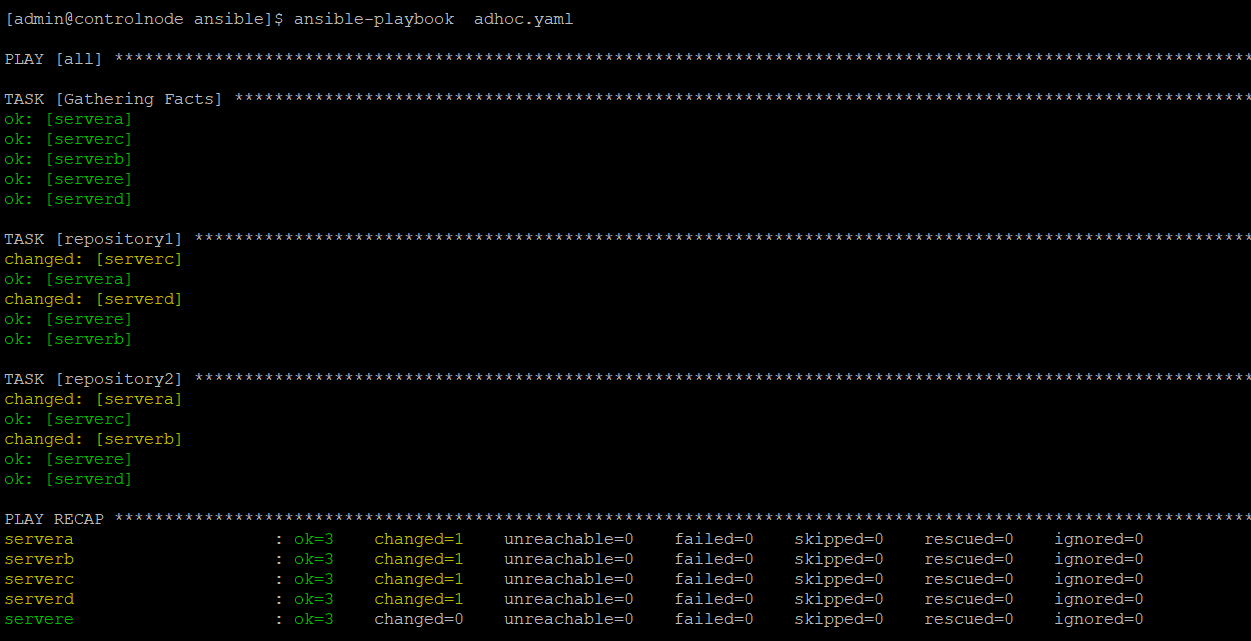
Step

1. Cd /home/admin/ansible/ -🡪 vim adhoc.yaml



1. Check the syntax with # ansible-playbook adhoc.yaml --syntax-check
2. Play the playbook

# ansible-playbook adhoch.yaml



1. Go to all server and check repository are configured or not



Q3 Install the ansible collections

A general.community

B ansible-posfix

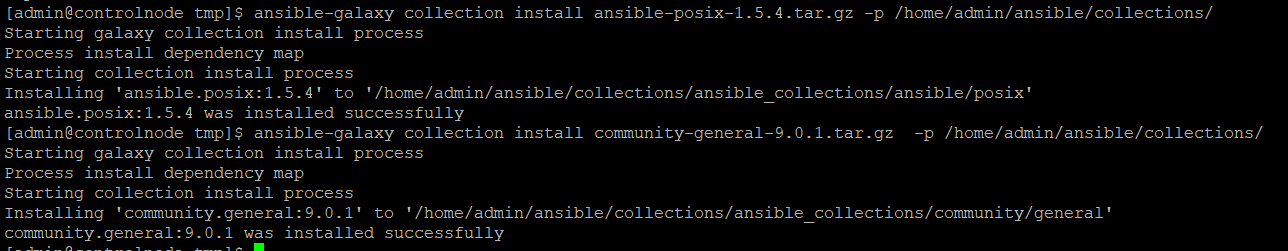
Ans 1 download the file from give path to control node with # wget command

# wget <http://given> path

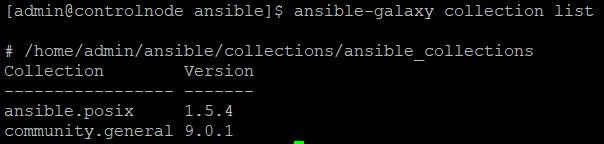
2. install the collection in given path

# ansible-galaxy collection install ansible-posix-1.5.4.tar.gz -p /home/admin/ansible/collections/

# ansible-galaxy collection install community-general-9.0.1.tar.gz -p /home/admin/ansible/collections/



Check the collections are installed or not



Q4 . Install packages

Create a playbook called /home/admin/ansible/packages.yml that:

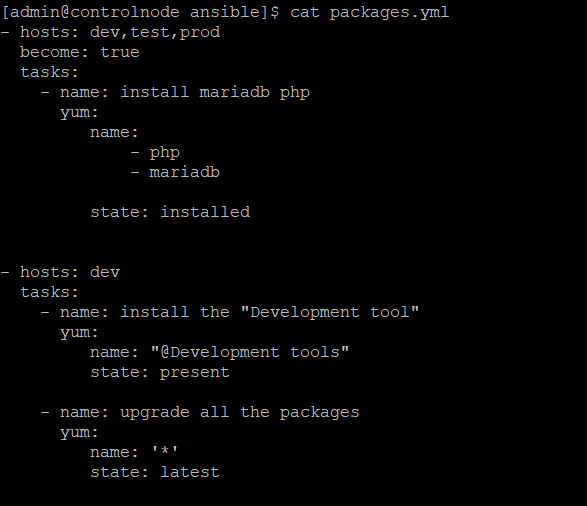
- Installs the php & mariadb on hosts in the dev,test and prod host group

- Installs the RPM Development Tools package group on host in the dev host group

- Update all packages to the latest version on host in the dev host group

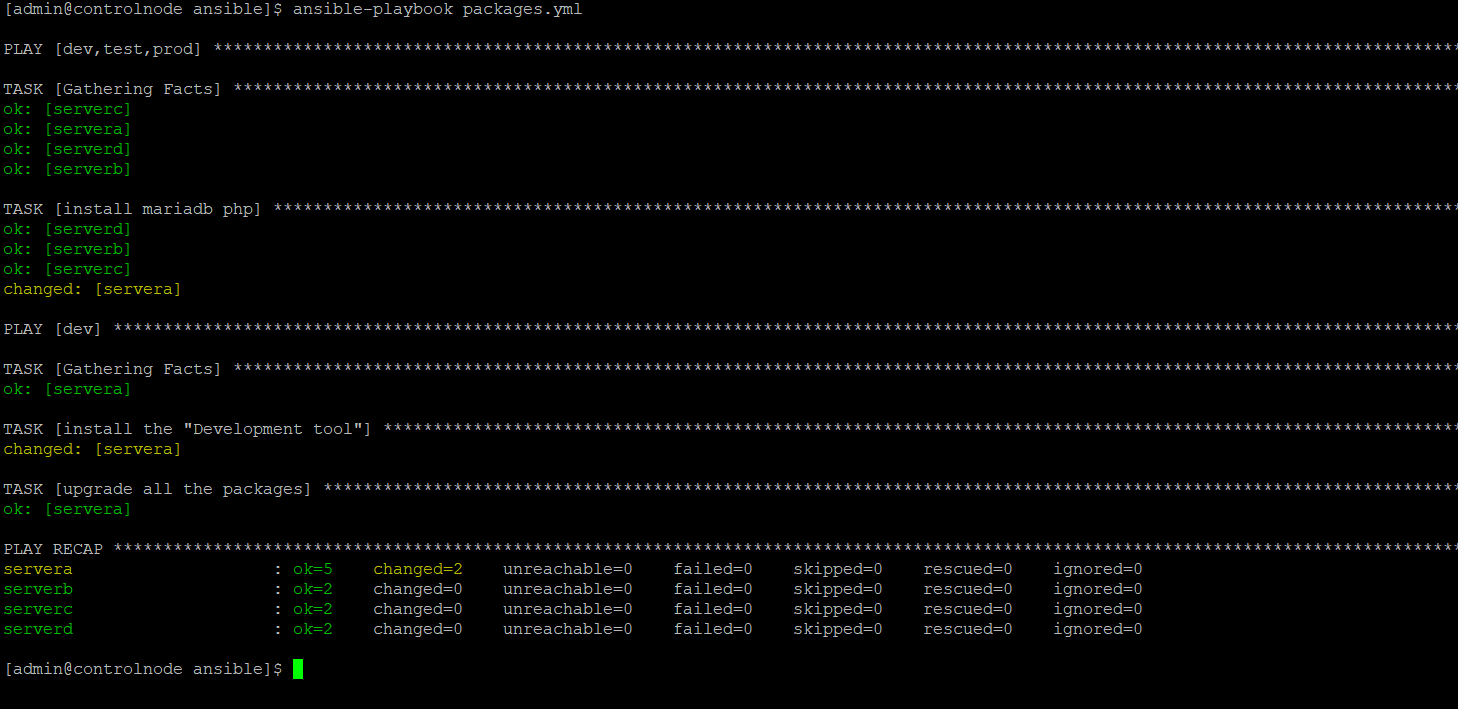
Ans:

Cd /home/admin/ansible -🡪 vim packages.yml



Check the syntax and run the playbook

# ansible-playbook packages.yml



Q 5 Use a RHEL system role

Install the RHEL System role package & create a playbook called /home/admin/ansible/timesync.yml that:

- Runs on all managed nodes

- Uses the timesync role

- Configures the role to use the currently active NTP provider

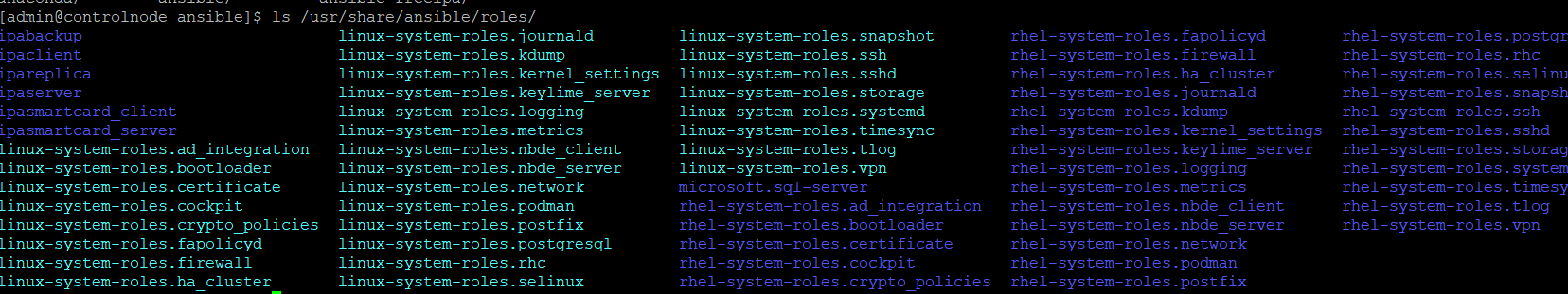
- Configure the role to use the time server 172.24.1.254

- Configure the role to enable the iburst parameter

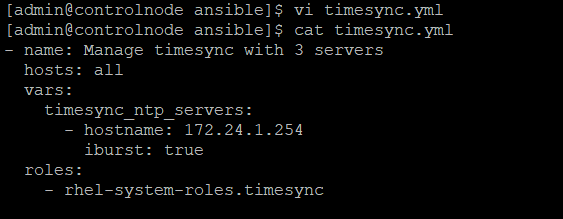
Ans:

1. Go to the /usr/share/ansible/roles and check the roles are present or not if not then

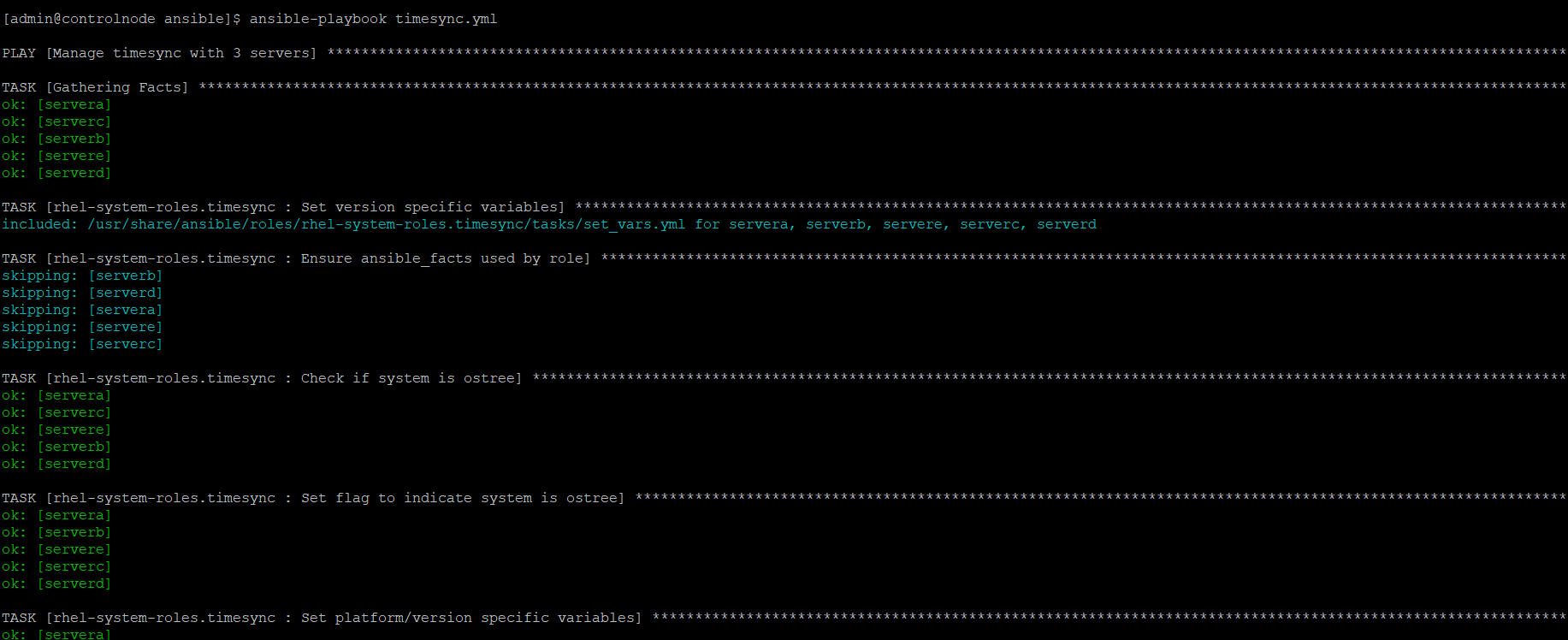
# yum install rhel-system-roles –y

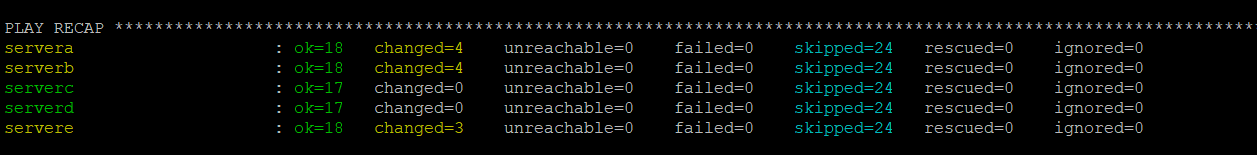


1. Go to the /cat /usr/share/ansible/roles/rhel-system-roles.timesync/README.md and copy the contented related to the timesync
2. Cd /home/admin/ansible/🡪 vim timesync.yml



1. Run the playbook



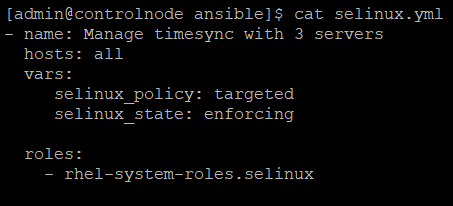


OR question selinux

Go to the cat /usr/share/ansible/roles/rhel-system-roles.selinux/README.md and copy the selinux content

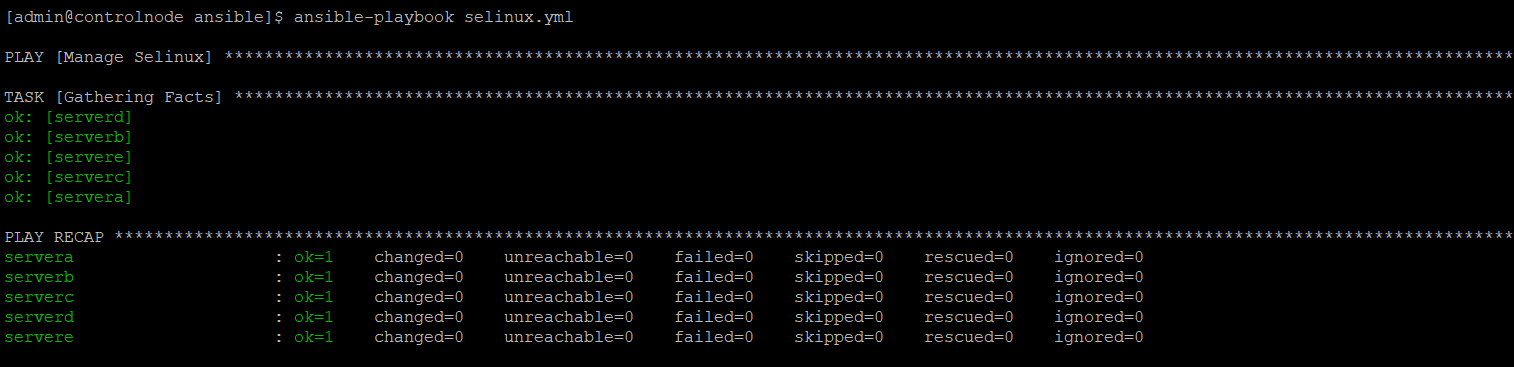


Cd /home/admin/ansible/ 🡪 vim selinux.yml



Run the playbook

# ansible-playbook selinux.yml



Q 6. . Install roles using Ansible Galaxy

use Ansible Galaxy with a requirements file called /hom/admin/ansible/roles/reuirement.yml to download and install role to /home/admin/ansible/roles from following URL:

\* <http://server.network.example.com/materials/haproxy.tar>

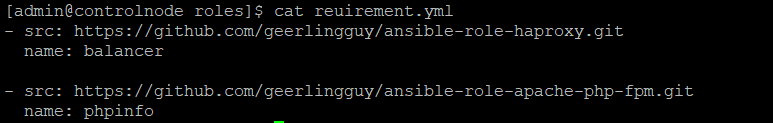
The Name of the role should be balancer

\* <http://server.network.example.com/materials/phpinfo.tar>

The name of this role should be phpinfo

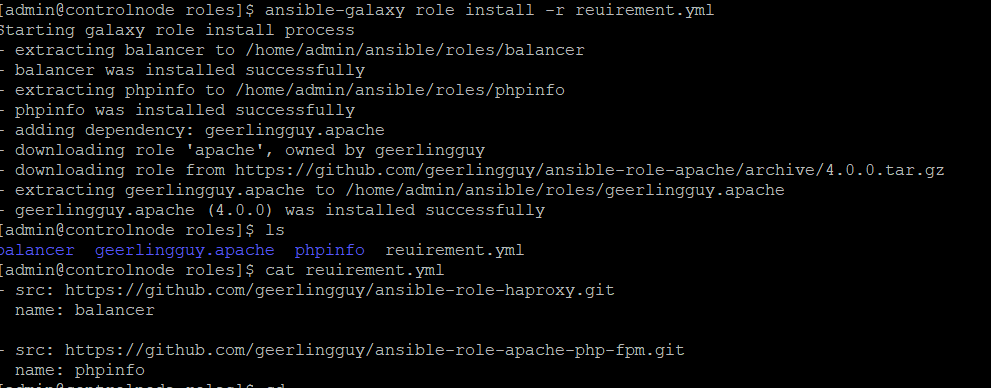
Ans:

1. Download and install the to /home/admin/ansible/roles from given url
2. Create yaml file cd /home/admin/ansible/roles 🡪 vim reuirement.yml



3 install the roles

# ansible-galaxy role install –r reuirement.yml



Q 7. Create & use a role

Create a role called apache in /home/admin/ansible/roles with the following requirement:

- The httpd package is install, enabled on boot and started

- The Firewall is enabled & running with a rule to allow access the web server

- A template file index.html.j2 exists and is used to create the file /var/www/html/index.html with the Welcome to HOSTNAME ON IPADDRESS

where HOSTNAME is the fqdn of the managed node and ip address is the ip address of managed node

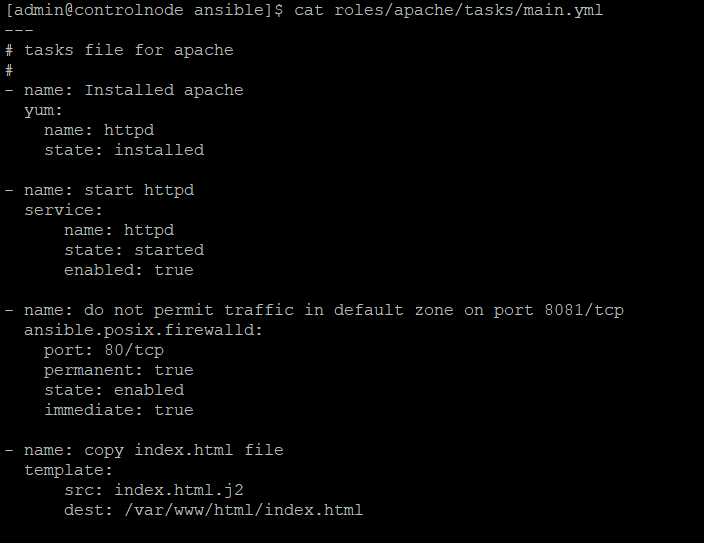
\* create a playbook called /home/admin/ansible/newrole.yml that uses this role as foloows: - The playbook runs on host in the webservers host group

Ans:

1 go to the cd /home/admin/ansible/roles and initialise apache

# ansible-galaxy init apache

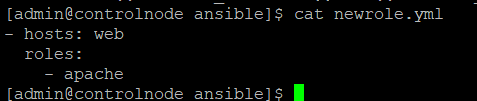
2 go to the cd /home/admin/ansible/roles/apache/tasks -- > vim main.yml



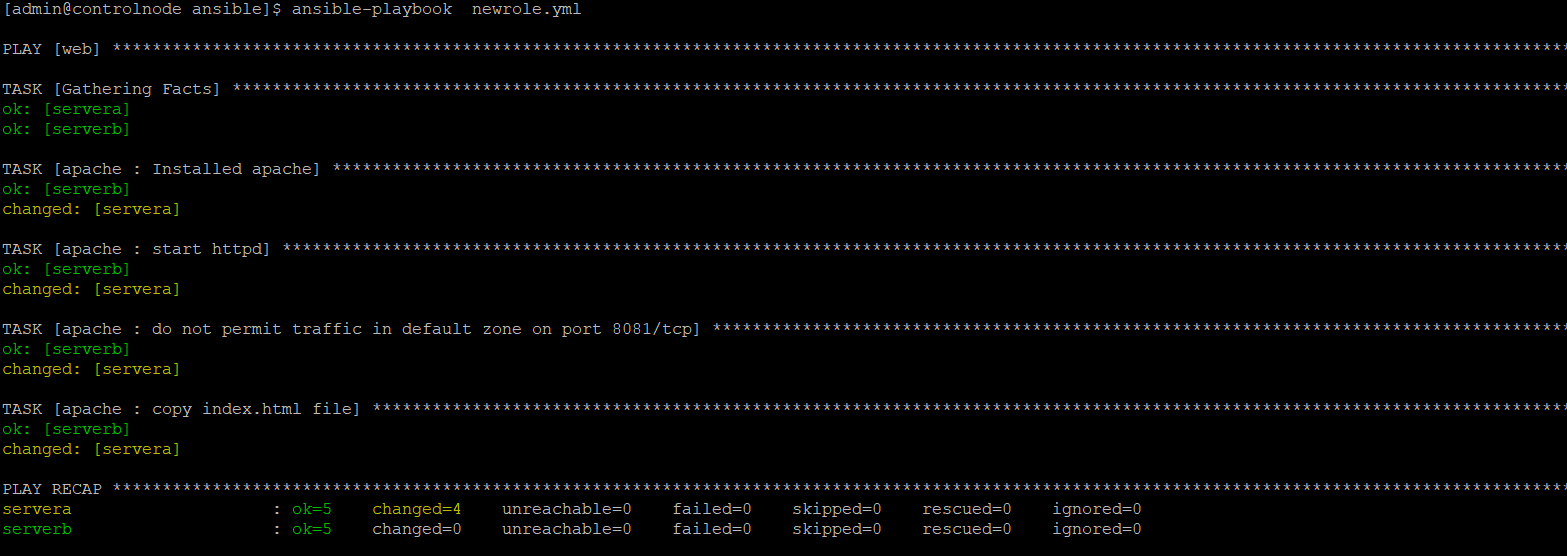
3 go to /home/admin/ansible/roles/apache/template 🡪 vim index.html.j2



4 go to /home/admin/ansible/ 🡪 vim newrole.yml



5 run the playbook



Q 8

Use roles from Ansible Galaxy

Create a playbook called /home/admin/ansible/roles.yml

\* The palybook contains a play that runs on host in the balances host group and uses the balances role.

- This role configures a service to load balance web server request between hosts in the webserver host group.

- Browsing to host in the balances host group ( for example http://system5.domain1.example.com) produces the following output:

Welcome to system3.domain1.example.com on 172.24.1.8

- Reloading the Browser produces output from the alternet web server:

Welcome to system4.domain1.example.com on 172.24.1.9

\* The Playbook contains a play the runs on hosts in webserver host group and user the phpinfo role. - Browsing to host in the webserver host group with the URL /hell.php produces the following out[ut : Hello PHP World from FQDN

- For example Browsing to http://system3.domain1.example.com/hello.php produces the following output:

Hello PHP World from system3.domain1.example.com

along with various details of the PHP configuration include the version of PHP that is installed.

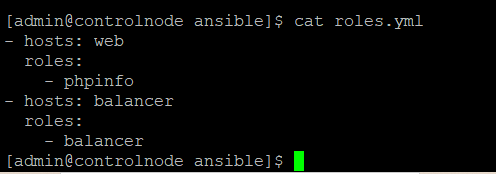
- Similarly, Browsing to http://system4.doamin1.example.com/hello.php, produces the following output:

Hello PHP World from system4.domain1.example.com

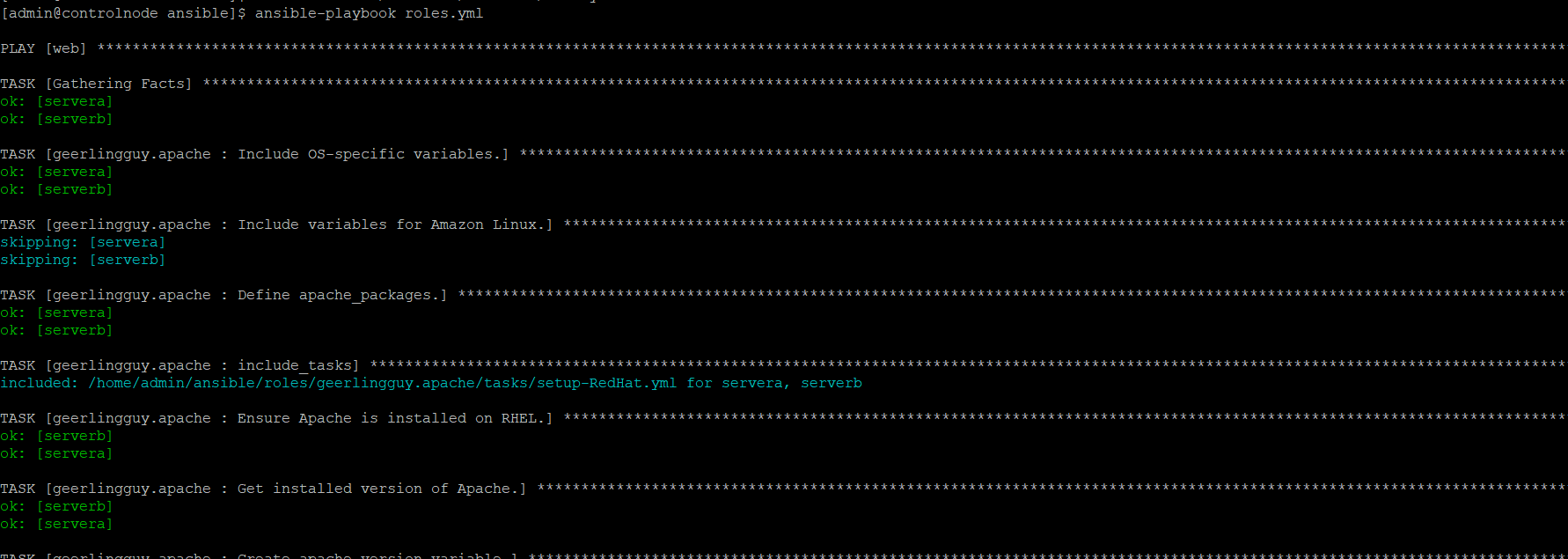
along with various details of the PHP configuration including the version of PHP that is installed.

Ans:

1 Go to the /home/admin/ansible 🡪 vim roles.yml

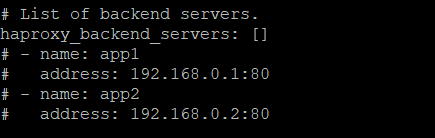


2 run the playbook



3 # curl <http://servere> and reload the page

If its not working proper then go to cd /home/admin/ansible/roles 🡪 vim balancer/defaults/main.yml and did enrtry of two servers



Q9. . Generate a host file

\* Download an initial template file from

http://server.network.example.com/materials/host.j2 to /home/admin/ansible

\* Complete the template so that it can he used to generate a file with a line for each host in the same format as /etc/hosts

\* Create a playbook call /home/admin/host.yml that uses this template to generate the file /etc/myhosts on hosts in the dev host group.

When the playbook is run, the file /etc/myhosts on host in the dev host group should have a line for each managed hosts:

127.0.0.1 localhost localhost.localdoamin localhost4 localhost4.localdomain

::1 localhost localhost.localdoamin localhost6 localhost6.localdomain

172.24.10.6 System1.domain1.example.com

172.24.10.7 System2.doamin1.example.com

172.24.10.8 System3.doamin1.example.com

172.24.10.9 System4.doamin1.example.com

172.24.10.10 system5.doamin1.example.com

Note: The order in which the inventory host names appear is not important

Ans:

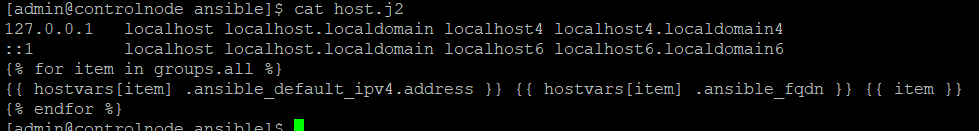
1 download the file from <http://server.network.example.com/materials/host.j2> to /home/admin/ansible with wget command

2 vim /home/admin/ansible/hosts.j2 and enter some variables

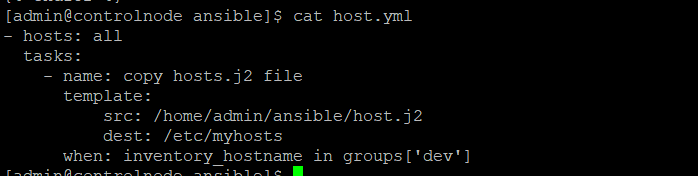
{% for item in groups.all %}

{{ hostvars[item] .ansible\_default\_ipv4.address }} {{ hostvars[item] .ansible\_fqdn }} {{ item }}

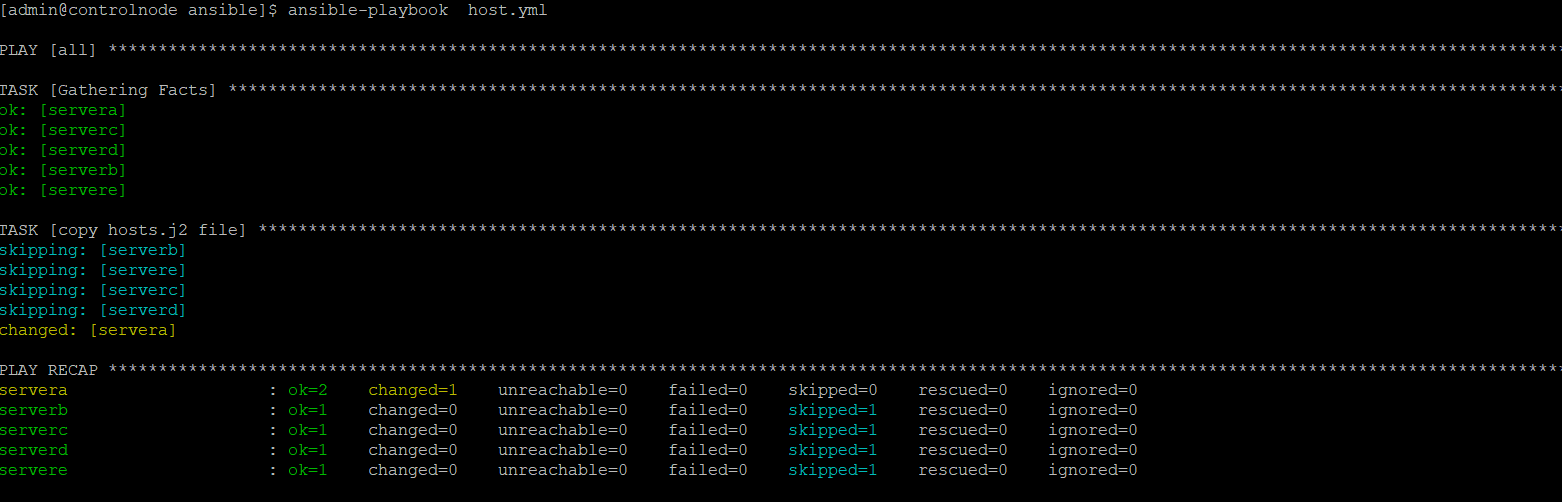
{% endfor %}

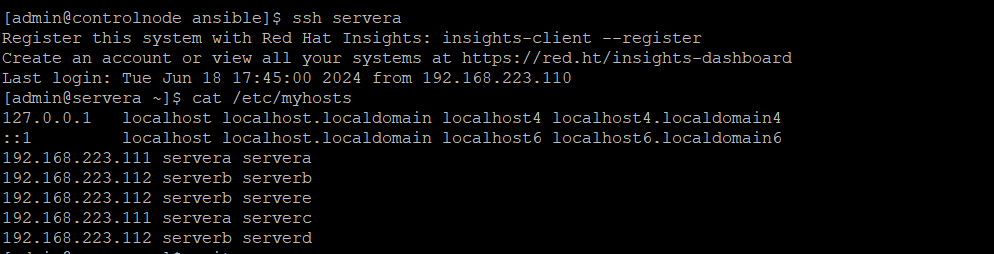


3 got to /home/admin/ansible and create host.yml file



4 run the playbook

5 go to the server which are present in dev group and verify.

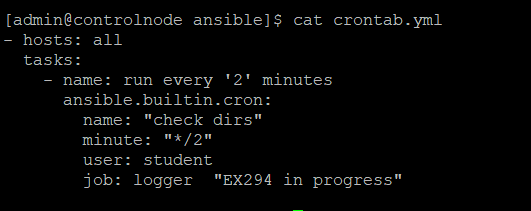


Q10 Create a cronjob for the user student on all nodes. Playbook name is crontab.yml and the jobe details are below

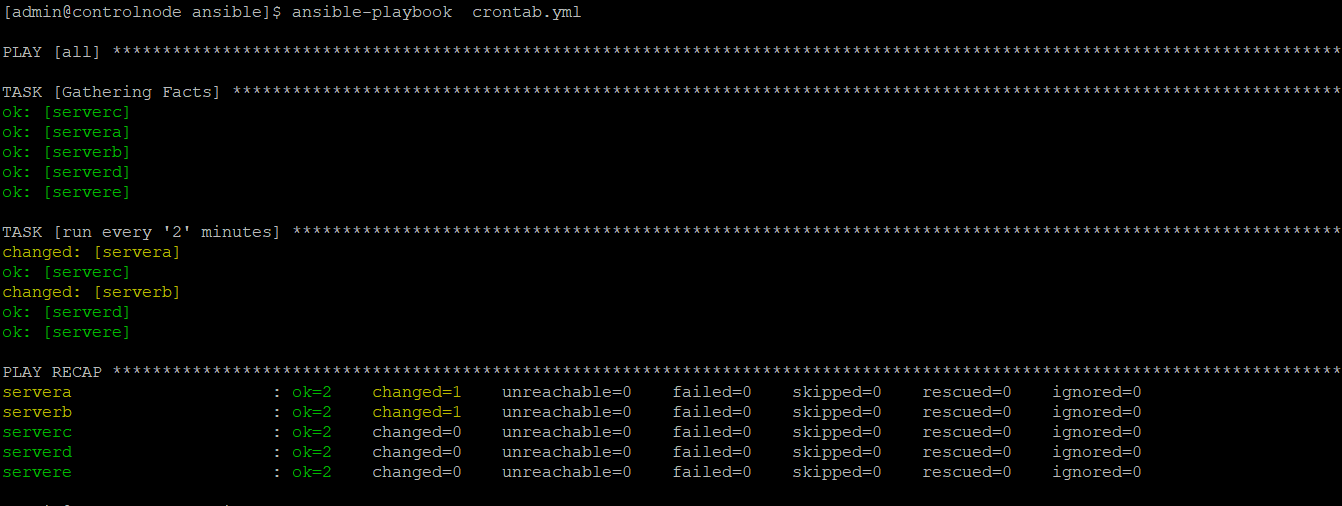
Every 2 min the jobe will execute logger “EX294 in progress”

Ans

1 Go to the /home/admin/ansible/-- > create crontab.yml file



2 run the playbook



3 got the servers and check

Crontabe –l –u student

11. Create a Web content directory Create a playbook called /home/admin/ansible/webcontent.yml as follows:

\* The playbook runs managed node in the dev host group

\* Create the directory /webdev with the following requirement:

- it is owner by the webdev group - it has regular permissions: owner=read+write+execute,group=r+w+x,other=r+x

- it has special permission: set group GID

\* Symbolically link /var/www/html/webdev to /webdev

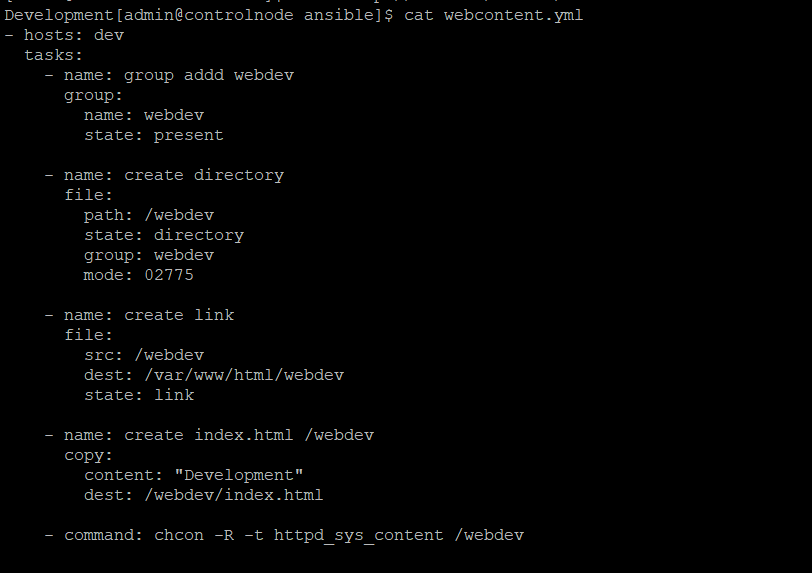
\* Create the file /web/index.html with a single line of text that reads: Development

\* Browsing this directory on host in the dev host group ( for example http://system1.domain1.example.com/webdev/) produces the following output:

Development

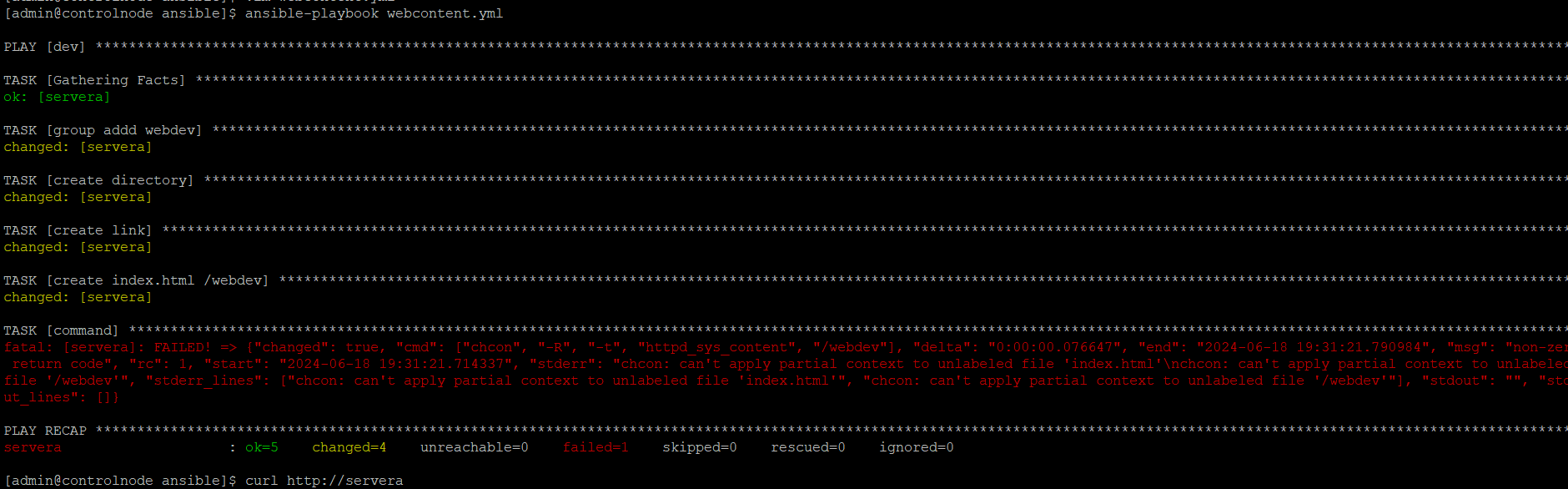
Ans

1 Go to the /home/admin/ansible -🡪 create webcontent.yml file



2 for selinux context check go to the server a and check the context of /var/www/html/ folder with # ls –ldZ /var/www/html/ and copy the context and past in yml file

3 run the playbook



4 verify the with curl <http://servera/webdev/>



12. Generate a hardware report

Create a playbook called /home/admin/ansible/hwreport.yml that produces an output file called /root/hwreport.txt on all managed nodes with the following information:

- Inventory hostname

- Total Memory

- BIOS version

- Size of disk device vda

- Size of disk device vdb

- Each line of the output file contains a single key=value pair.

\* your playbook should:

- Download the file from http://server.network.example.com/materials/hwreport.empty and save it is aas /root/hwreport.txt

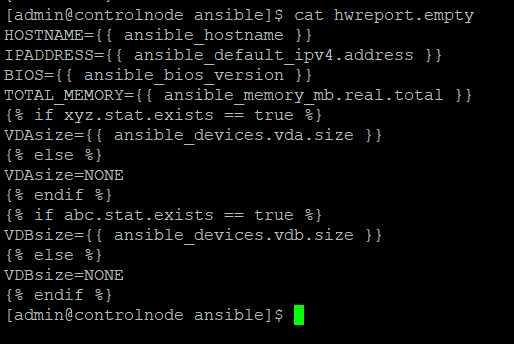
- Modify /root/hwreport.txt with the correct values

-- if a hardware item does not exist, the associated value should be set to NONE

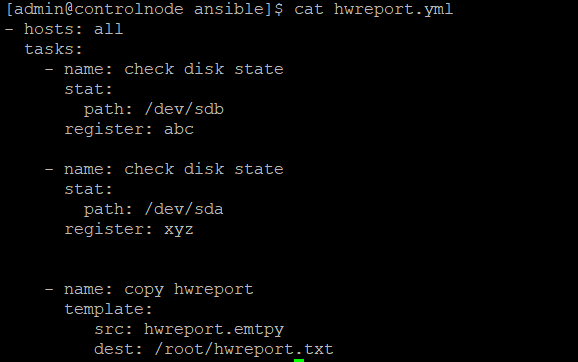
Ans:

1 Download the file from givent path <http://server.network.example.com/materials/hwreport.empty>

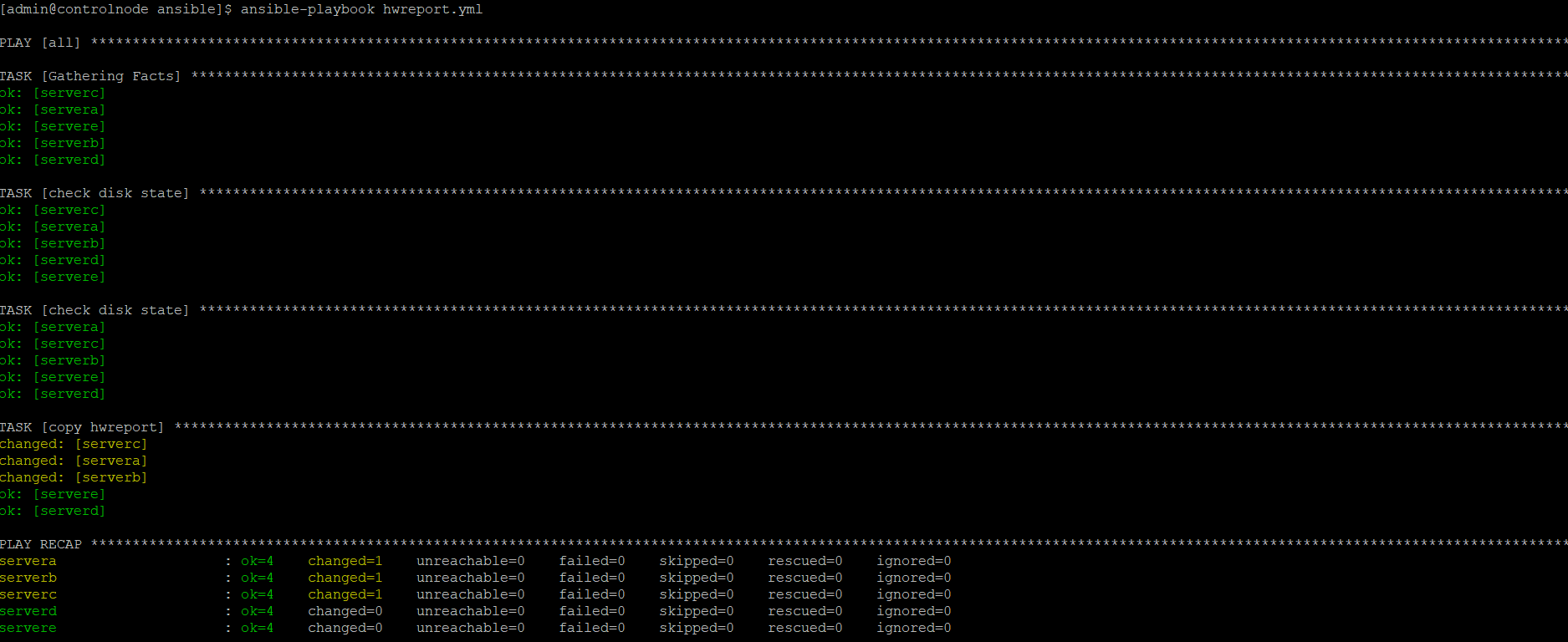
2 Modify the file with values.



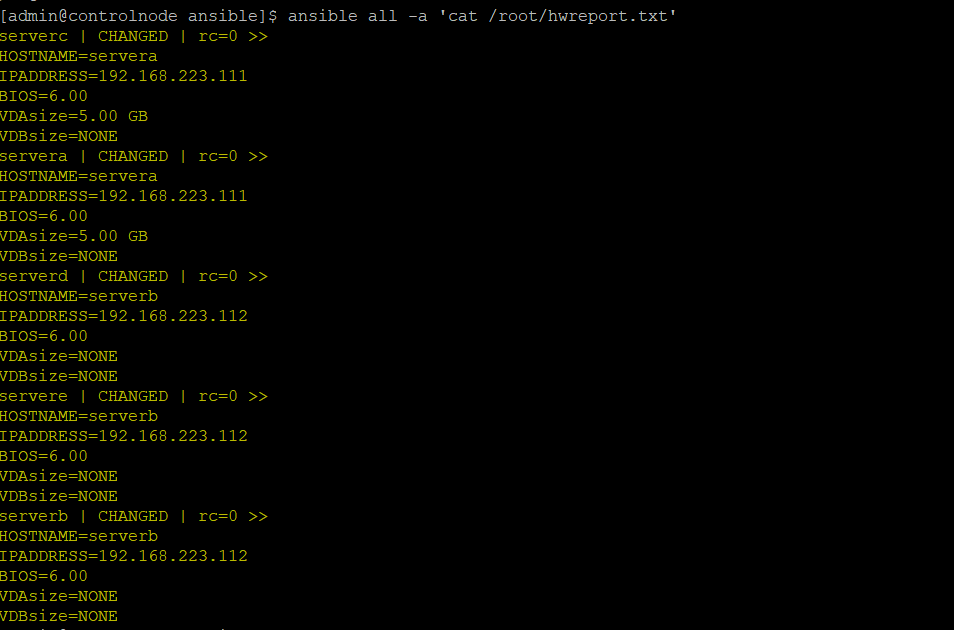
3 Create hwreport.yml file in /home/admin/ansible directory



4 Run the playbook



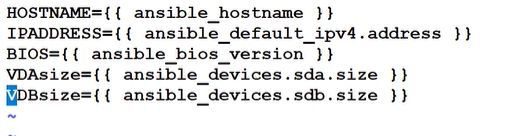
5 verify the /root/hwreport.txt file creted or not



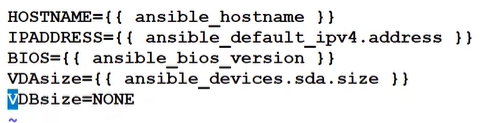
Second method

1 download the hwreport.emtpy file and create two files one hwreport.emtpy and second hwreport.emtpy2

2 in one file hwreport.emtpy



2 In second file hwreport.emtpy2



3 create hwreport.yml file



4 run the playbook.

Q13 Create a password vault Create an ansible vault to store user pasword as follows:

\* The name of the vault is /home/admin/ansible/locker.yml

\* The vault contains two variables with names:

- PW\_developer with value Imadev

- PW\_manager with value Imamgr

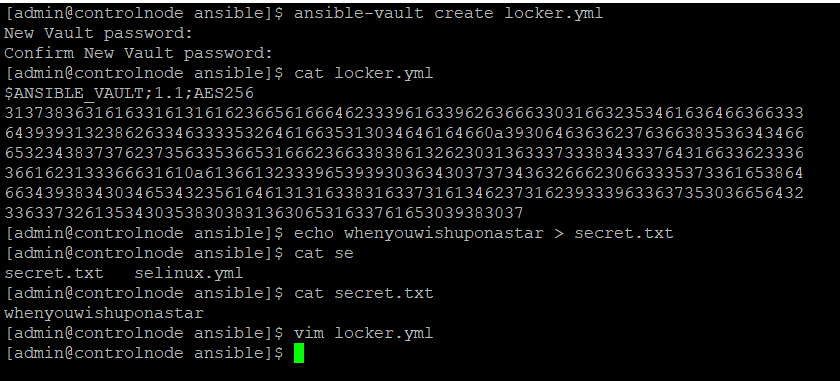
\* The Password to encrypt and decrypt the vault *is whenyouwishuponastar*

\* The password is stored in the file /home/admin/ansible/secret.txt

Ans:

1 Create locker.yml file with vault module

2 copy the password in secret.txt file



14. Create user accounts { I skip this question}

\* Download a list of user to be created from <http://server.network.example.com/materials/user_list.yml>

\* Using the password vault /home/admin/ansible/locker.yml create elsewhere in this exam, create a play book called /home/admin/ansible/users.yml that create user account as following:

\* User with job description of developer should be:

- create on managed node in the dev and test host group

- assigned the password from the pw\_devloper variable

- a member of supplementary group devops

\* User with a job description of manager should be:

- create on managed node in the prod host group

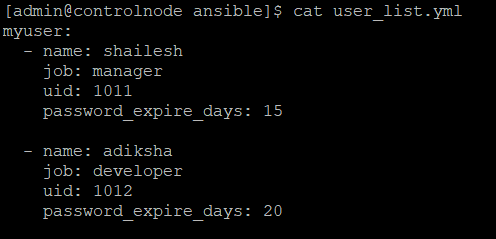
- assigned the password from pw\_manager variable

- a member of supplementary group opsmgr

\* password should use the SHA512 gas format

\* Your playbook should work using the vault password file /home/admin/ansible/secret.txt create elsewhere in this exam

Ans: 1 create user\_list.yml



2 create user.yml

ansible-playbook user.yml --vault-password-file=./secret.txt 

3 play the playbook with ansible-playbook user.yml --vault-password-file=./secret.txt command

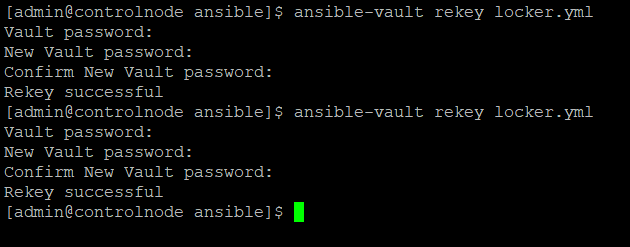
Q 15. ReKey an Ansible vault Rekey an existing Ansible vault as follows:

- Download the Ansible vault from [http://server.network.example.com/materials/salariries.yml to /home/admin/ansible/](http://server.network.example.com/materials/salariries.yml%20%20%20to%20/home/admin/ansible/)

- The current vault password is jaishreeram

- The new vault password is jaimatadi

- The vault remains in an encrypted state with the new password

Ans 

16. Modify file content

Create a playbook called /home/admin/ansible/issue.yml

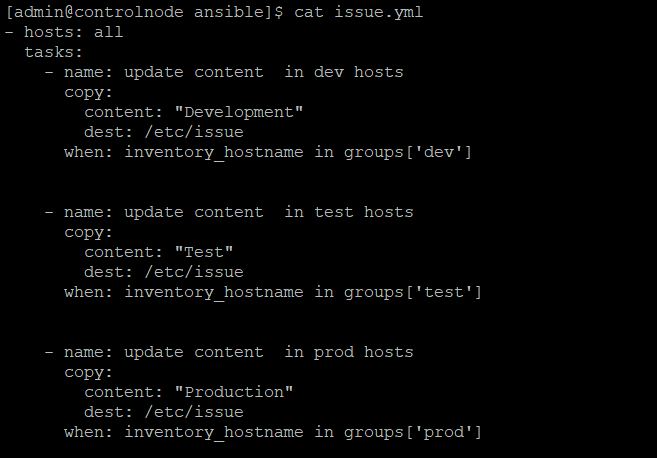
\* The playbook runs on all inventory hosts

\* The playbook replaces the content of /etc/issue with a single line of text as follows:

- On host is the dev host group, the line reads: Development

- On hosts in the test host group, the line read: Test

- On hosts in the prod hosts group, the line read: Production

Ans: 

Run the playbook and verify

ansible all –m shell –a ‘cat /etc/issue’

