

RHEL-9 RHCE EXAM MODEL PAPER
EX294

Duration: 4Hrs
Total Marks: 300

Instructions:

control node: workstation

managed node: servera,
 serverb.

- * All node root password is 'redhat' and Ansible control node user name is student.
- * Create a directory 'ansible' under the path /home/student and all the playbook should be under /home/student/ansible.
- * All playbook should be owned/executed by student.
- * Ansible control node user password is student
- * Unless advised password should be 'redhat' for all users

Note: In Exam, If they not given the Managed node user use the control node user as remote user

ssh student@localhost (in workstation)

1. Install and Configure Ansible on the control node as follows:

- * Install the required packages.
- * Create a static inventory file called /home/student/ansible/inventory as follows:
 - servera is a member of the dev host group
 - serverb is a member of the test host group
 - servera is a member of the prod host group
 - serverb is a member of the balancers host group
 - The prod group is a member of the webserver host group
- * Create a configuration file called ansible.cfg as follows:
 - The host inventory file /home/student/ansible/inventory is defined
 - The location of roles used in playbooks is defined as /home/student/ansible/roles
 - The location of collections used in playbooks is defined as /home/student/ansible/collections

2. Create a playbook adhoc.yml for configuring repository in all nodes.

i) Name = baseos

Description = Baseos Description

Url = http://content/rhel9.0/x86_64/dvd/BaseOS

GPG is enabled.

Gpgkey = http://content.example.com/rhel9.0/x86_64/dvd/RPM-GPG-KEY-redhat-release

Repository is enabled.

ii) Name = appstream

Description = App Description

Url = http://content/rhel9.0/x86_64/dvd/AppStream

GPG is enabled.

Gpgkey = http://content.example.com/rhel9.0/x86_64/dvd/RPM-GPG-KEY-redhat-release

Repository is enabled.

3. Installing the Collection.

i) Create a directory "collections" under the /home/student/ansible.

ii) install the ansible.posix and the community.general collections under collections directory.

iii) install the system roles collection under collections directory. (yum then cp!)

4. installing the roles.

i) Create a directory 'roles' under /home/student/ansible

ii) Create a playbook called requirements.yml under the roles directory and download the given roles under the 'roles' directory using galaxy command under it.

iii) Role name should be balancer and download using this url [WesleyMiler.balancer](#)

iv) Role name phpinfo and download using this url [JelleBrouwer02.php](#)

5. Create offline role named apache under roles directory.

i) Install httpd package and the service should be start and enable the httpd service.

ii) Host the web page using the template.j2

iii) The template.j2 should contain i

My host is HOSTNAME on IPADDRESS

Where HOSTNAME is fully qualified domain name.

iv) Create a playbook named apache_role.yml and run the role in dev group.

6. Create a Playbook roles.yml for using the roles

1) The playbook contains the balancer hosts for use balancer role

- a) browsing, the balancers host group with url `http://serverd.lab.example.com` that produce the output
`"Welcome to servera.lab.example.com, (version 1.0)"`
 - b) Refreshing, the balancers host group with the same url the output should be change
`"Welcome to serverc.lab.example.com, (version 1.0)"`
- 2) The playbook contains the webserver host group for using the role `phpinfo`
- a) browsing, the webserver host group name that provides the output
`"Welcome to serverc.lab.example.com, (version 1.0)"`
 and the output comes with various php contents
 - b) For example, the webserver hostgroup `http://serverc.lab.example.com` That provides the output
`"Welcome to serverc.lab.example.com, (version 1.0)"`
 and the output comes with various php contents
 - c) Similarly, the webserver hostgroup `http://servera.lab.example.com` that provides the output
`"My host is servera.lab.example.com on 172.25.250.10"`

7.1 Create a playbook name `timesync.yml` and use system roles

- i) Use ntp server `172.25.254.254` and enable `iburst`.
- ii) Run this playbook on all the managed nodes.

7.2 Create a playbook name `selinux.yml` and use system roles

- i) Set selinux mode as enforcing in all manage node

8. Install packages in multiple group.

- i) Install `vsftpd` and `mariadb-server` packages in dev and test group.
- ii) Install "RPM Development Tools" group package in prod group.
- iii) Update all packages in dev group.
- iv) Use separate play for each task and playbook name should be `packages.yml`.

9. Create a playbook `webcontent.yml` and it should run on dev group.

- i) Create a directory `/devweb` and it should be owned by devops group.
 - ii) `/devweb` directory should have context type as `"httpd"`
 - iii) Assign the permission for `user=rwx,group=rwx,others=rx` and group special permission should be applied to `/devweb`.
 - iv) Create an `index.html` file under `/devweb` directory and the file should have the content `"Develoement"`.
 - v) Link the `/devweb` directory to `/var/www/html/devweb`.
10. Collect hardware report using playbook in all nodes.

i) create /root/hwreport.txt that should have the content with node informations as key=value.

```
#hwreport
HOSTNAME=
MEMORY=
BIOS=
CPU=
DISK_SIZE_VDA=
DISK_SIZE_VDB=
```

ii) If there is no information it have to show "NONE".

iii) playbook name should be hwreport.yml.

11. Replace the file /etc/issue on all managed nodes.

i) In dev group /etc/issue should have the content "Developement".

ii) In test group /etc/issue should have the content "Test".

iii) In prod group /etc/issue should have the content "Production".

iv) Playbook name issue.yml and run in all managed nodes.

12. Download the file <http://content.example.com/Rhce/myhosts.j2>.

i) myhosts.j2 is having the content.

```
127.0.0.1 localhost.localdomain localhost
192.168.0.1 localhost.localdomain localhost
```

ii) The file should collect all node information like ipaddress,fqdn,hostname and it should be the same as in the /etc/hosts file, if playbook run in all the managed node it must store in /etc/myhosts.

Finally /etc/myhosts file should contains like.

```
127.0.0.1 localhost.localdomain localhost
192.168.0.1 localhost.localdomain localhost
```

```
172.25.250.10 servera.lab.example.com servera
172.25.250.11 serverb.lab.example.com serverb
172.25.250.12 serverc.lab.example.com serverc
172.25.250.13 serverd.lab.example.com serverd
```

iii) playbook name hosts.yml and run in dev group.

13. Create a variable file vault.yml and that file should contains the variable and its value.

pw_developer is value lamdev

pw_manager is value lammgr

- i) vault.yml file should be encrypted using the password "P@sswOrd".
- ii) Store the password in secret.txt file and which is used for encrypt the variable file.

14. create the file user_list.yml and its content is:

users:

- name: natasha
uid: 2222
password_expire_days: 3
job: manager
- name: adam
uid: 2223
password_expire_days: 5
job: developer

and create a Playbook named users.yml and run it in all nodes using two variable files user_list.yml and vault.yml

- i) * Create a group opsdev
 - * Create user from users variable who's job is equal to developer and need to be in opsdev group
 - * Assign a password using SHA512 format and run the playbook on dev and test.
 - * User password is {{ pw_developer }}
- ii) * Create a group opsmgr
 - * Create user from users variable who's job is equal to manager and need to be in opsmgr group
 - * Assign a password using SHA512 format and run the playbook on test.
 - * User password is {{ pw_manager }}
- iii)* Use when condition for each play.

15. Rekey the variable file vault.yml

- i) Old password: P@sswOrd
- ii) New password: redhat

16. Create a cronjob for user student in all nodes, the playbook name crontab.yml and the job details are below

- i) Every 2 minutes the job will execute logger "EX294 in progress"

17. Create a logical volume named data of 1500M size from the volume group research and if 1500M size is not created, then atleast it should create 800M size.

- i) Verify if vg not exist, then it should debug msg "vg not found" .
- ii) 1500M lv size is not created, then it should debug msg "Insufficient size of vg" .

- iii) If Logical volume is created, then assign file system as "ext3" .
- iv) Do not perform any mounting for this LV.
- iv) The playbook name lvm.yml and run the playbook in all nodes.