TestpassportQ&A

Higher Quality Better Service!

We offer free update service for one year

Http://www.testpassport.com

Exam : **EX294**

Title : Red Hat Certified Engineer (RHCE) exam for Red Hat Enterprise

Version : DEMO

1. Topic 1, LAB SETUP
You will need to set up your lab by creating 5 managed nodes and one control node.
So 6 machines total. Download the free RHEL8 iso from Red Hat Developers website.
Control node you need to set up
You need to create some static ips on your managed nodes then on the control node set them up in the /etc/hosts file as follows:
vim /etc/hosts
10.0.2.21 node1.example.com
10.0.2.22 node2.example.com
10.0.2.23 node3.example.com
10.0.2.24 node4.example.com
10.0.2.25 node5.example.com
yum -y install ansible
useradd ansible
echo password | passwd --stdin ansible
echo "ansible ALL=(ALL) NOPASSWD:ALL" > /etc/sudoers.d/ansible
su - ansible; ssh-keygen
ssh-copy-id node1.example.com
ssh-copy-id node2.example.com
ssh-copy-id node3.example.com
ssh-copy-id node4.example.com
ssh-copy-id node5.example.com
Each manage node setup
First, add an extra 2GB virtual harddisk to each control node 1,2,3. Then add an extra hard disk to control node 4. Do not add an extra hard disk to node 5. When you start up these machines the extra disks should be automatically located at /dev/sdb (or /dev/vdb depending on your hypervisor).
useradd ansible
echo password | passwd --stdin ansible
echo "ansible ALL=(ALL) NOPASSWD:ALL" > /etc/sudoers.d/ansible
Note python3 should be installed by default, however if it is not then on both the control node and managed nodes you can install it also set the default python3 if you are having trouble with python2 being the default.
yum -y install python3
alternatives --set python /usr/bin/python3
All machines need the repos available. You did this in RHSC
A. To set up locally you just need to do the same for each machine. Attach the rhel8 iso as a disk to virtualbox, kvm or whatever hypervisor you are using (this will be /dev/sr0). Then inside the machine:
mount /dev/sr0 to /mnt
Then you will have all the files from the iso in /mnt.
mkdir /repo
cp -r /mnt /repo
vim /etc/yum.repos.d/base.repo
Inside this file:
[baseos]

name=baseos
baseurl=file:///repo/BaseOS
gpgcheck=0
Also the appstream
vim /etc/yum.repos.d/appstream.repo
Inside this file:
[appstream]
name=appstream
baseurl=file:///repo/AppStream
gpgcheck=0

Install and configure ansible
User bob has been created on your control node. Give him the appropriate permissions on the control node. Install the necessary packages to run ansible on the control node.
Create a configuration file /home/bob/ansible/ansible.cfg to meet the following requirements:
• The roles path should include /home/bob/ansible/roles, as well as any other path that may be required for the course of the sample exam.
• The inventory file path is /home/bob/ansible/inventory.
• Ansible should be able to manage 10 hosts at a single time.
• Ansible should connect to all managed nodes using the bob user.
Create an inventory file for the following five nodes:
nodel.example.com
node2.example.com
node3.example.com
node4.example.com
node5.example.com
Configure these nodes to be in an inventory file where node1 is a member of group dev. nodc2 is a member of group test, nodc3 is a member of group proxy, nodc4 and node 5 are members of group prod. Also, prod is a member of group webservers.
Answer:
In/home/sandy/ansible/ansible.cfg
[defaults]
inventory=/home/sandy/ansible/inventory
roles_path=/home/sandy/ansible/roles
remote_user= sandy
host_key_checking=false
[privilegeescalation]
become=true
become_user=root
become_method=sudo
become_ask_pass=false
In /home/sandy/ansible/inventory
[dev]
node 1 .example.com

[test]
node2.example.com
[proxy]
node3.example.com
[prod]
node4.example.com
node5.example.com
[webservers: children]
prod

2.Create a file called adhoc.sh in /home/sandy/ansible which will use adhoc commands to set up a new repository. The name of the repo will be 'EPEL' the description 'RHEL8' the baseurl is 'https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rmp' there is no gpgcheck, but you should enable the repo.
* You should be able to use an bash script using adhoc commands to enable repos. Depending on your lab setup, you may need to make this repo "state=absent" after you pass this task.
Answer:
chmod 0777 adhoc.sh
vim adhoc.sh
#l/bin/bash
ansible all -m yum_repository -a 'name=EPEL description=RHEL8
baseurl=https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rmp gpgcheck=no
enabled=yes'

3.Create a file called packages.yml in /home/sandy/ansible to install some packages for the following hosts. On dev, prod and webservers install packages httpd, mod_ssl, and mariadb. On dev only install the development tools package. Also, on dev host update all the packages to the latest.
Answer:
Solution as:

```
---
- name: install pack
  hosts: dev,test,webservers
  become: true
  tasks:
    - name: install on all hosts in this play
      yum:
        name:
          - httpd
          - mod_ssl
          - mariadb
        state: latest
    - name: install on dev only
      yum:
        name:
          - '@Development tools'
        state: latest
      when: "dev" in group_names
~
```

** NOTE 1 a more acceptable answer is likely 'present' since it's not asking to install the latest state: present
** NOTE 2 need to update the development node
- name: update all packages on development node yum:
name: '*'
state: latest

4.Create a role called sample-apache in /home/sandy/ansible/roles that enables and starts httpd, enables and starts the firewall and allows the webserver service. Create a template called index.html.j2 which creates and serves a message from /var/www/html/index.html Whenever the content of the file changes, restart the webserver service.
Welcome to [FQDN] on [IP]
Replace the FQDN with the fully qualified domain name and IP with the ip address of the node using ansible facts. Lastly, create a playbook in /home/sandy/ansible/ called apache.yml and use the role to serve the index file on webserver hosts.
Answer:
/home/sandy/ansible/apache.yml

/home/sandy/ansible/roles/sample-apache/tasks/main.yml

```
---
# tasks file for sample-apache
- name: enable httpd
  service:
    name: httpd
    state: started
    enabled: true
- name: enable firewall
  service:
    name: firewalld
    state: started
    enabled: true
- name: firewall http service
  firewalld:
    service: http
    state: enabled
    permanent: yes
    immediate: yes
- name: index
  template:
    src: templates/index.html.j2
    dest: /var/www/html/index.html
  notify:
    - restart
```

/home/sandy/ansible/roles/sample-apache/templates/index.html.j2

```
Welcome to {{ansible_fqdn}} {{ansible_default_ipv4.address}}
```

In /home/sandy/ansible/roles/sample-apache/handlers/main.yml

```
- name: restart
  service:
      name: httpd
      state: restarted
```

5.Create a file called requirements.yml in /home/sandy/ansible/roles to install two roles. The source for the first role is geerlingguy.haproxy and geerlingguy.php. Name the first haproxy-role and the second php-role. The roles should be installed in /home/sandy/ansible/roles.

Answer:

in /home/sandy/ansible/roles

vim requirements.yml

```
- src: geerlingguy.haproxy
  name: haproxy-role
- src: geerlingguy. php_role
  name: php_role
```

Run the requirements file from the roles directory:

ansible-galaxy install -r requirements.yml -p /home/sandy/ansible/roles