Red-Hat

Exam Questions EX294

Red Hat Certified Engineer (RHCE) exam

NEW QUESTION 1
- (Exam Topic 2)
Create user accounts

--> A list of users to be created can be found in the file called user_list.yml
which you should download from http://classroom.example.com/user_list.yml and
save to /home/admin/ansible/
--> Using the password vault created elsewhere in this exam, create a playbook called create_user.yml
that creates user accounts as follows:
--> Users with a job description of developer should be:
--> created on managed nodes in the "dev" and "test" host groups assigned the password from the "dev_pass"
variable and these user should be member of supplementary group "devops".
--> Users with a job description of manager should be:
--> created on managed nodes in the "prod" host group assigned the password from the "mgr_pass" variable
and these user should be member of supplementary group "opsmgr"
--> Passwords should use the "SHA512" hash format. Your playbook should work using the vault password file created elsewhere in this exam.
while practising you to create these file hear. But in exam have to download as per questation.
user_list.yml file consist:
--
user:
- name: user1 job: developer
- name: user2 job: manager

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
pwd
/home/admin/ansible
#
wget http://classroom.example.com/user_list.yml
cat user_list.yml
vim create_user.yml
--
- name: hosts: all vars_files:
- ./user_list.yml
- ./vault.yml tasks:
- name: creating groups group:
name: "{{ item }}" state: present
loop:
- devops
- opsmgr
- name: creating user user:
name: "{{ item.name }}" state: present
groups: devops
password: "{{ dev_pass|password_hash ('sha512') }}" loop: "{{ user }}"
when: (inventory_hostname in groups['dev'] or inventory_hostname in groups['test']) and item.job == "developer"
- name: creating user user:
name: "{{ item.name }}" state: present
groups: opsmgr
password: "{{ mgr_pass|password_hash ('sha512') }}" loop: "{{ user }}"
when: inventory_hostname in groups['prod'] and item.job == "manager" wq!
ansible-playbook create_user.yml -–vault-password-file=password.txt -–syntax-check
ansible-playbook create_user.yml -–vault-password-file=password.txt

NEW QUESTION 2
- (Exam Topic 2)
Create a playbook called web.yml as follows:
* The playbook runs on managed nodes in the "dev" host group
* Create the directory /webdev with the following requirements:
--> membership in the apache group
--> regular permissions: owner=r+w+execute, group=r+w+execute, other=r+execute s.p=set group-id
* Symbolically link /var/www/html/webdev to /webdev
* Create the file /webdev/index.html with a single line of text that reads: “Development”
-->
it should be available on http://servera.lab.example.com/webdev/index.html

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
pwd
/home/admin/ansible/
vim web.yml

--
- name: hosts: dev tasks:
- name: create group yum:
name: httpd state: latest
- name: create group group:
name: apache state: present
- name: creating directiory file:
path: /webdev state: directory mode: '2775' group: apache
- sefcontext:
target: '/webdev/index.html' setype: httpd_sys_content_t state: present
- name: Apply new SELinux file context to filesystem command: restorecon -irv
- name: creating symbolic link file:
src: /webdev
dest: /var/www/html/webdev state: link
force: yes
- name: creating file file:
path: /webdev/index.html
sate: touch
- name: Adding content to index.html file copy:
dest: /webdev/index.html content: "Development"
- name: add service to the firewall firewalld:
service: http permanent: yes state: enabled immediate: yes
- name: active http service service:
name: httpd state: restarted enabled: yes wq
ansible-playbook web.yml -–syntax-check
ansible-playbook web.yml

NEW QUESTION 3
- (Exam Topic 2)
Create a role called apache in "/home/admin/ansible/roles" with the following requirements:
--> The httpd package is installed, enabled on boot, and started.
--> The firewall is enabled and running with a rule to allow access to the web server.
--> template file index.html.j2 is used to create the file /var/www/html/index.html with the output:
Welcome to HOSTNAME on IPADDRESS
--> Where HOSTNAME is the fqdn of the managed node and IPADDRESS is the IP-Address of the managed node.
note: you have to create index.html.j2 file.
--> Create a playbook called httpd.yml that uses this role and the playbook runs on hosts in the webservers host group.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
---
pwd
/home/admin/ansible/roles/
ansible-galaxy init apache
vim apache/vars/main.yml
--
vars file for apache http_pkg: httpd firewall_pkg: firewalld http_srv: httpd firewall_srv: firewalld rule: http
webpage: /var/www/html/index.html template: index.html.j2
wq!
vim apache/tasks/package.yml
--
- name: Installing packages yum:
name:
- "{{http_pkg}}"
- "{{firewall_pkg}}" state: latest
wq!
vim apache/tasks/service.yml
--
- name: start and enable http service service:
name: "{{http_srv}}"
enabled: true state: started
- name: start and enable firewall service service:
name: "{{firewall_srv}}" enabled: true
state: started wq!
vim apache/tasks/firewall.yml
--
- name: Adding http service to firewall firewalld:
service: "{{rule}}" state: enabled permanent: true immediate: true wq!
vim apache/tasks/webpage.yml
--
- name: creating template file template:
src: "{{template}}"
dest: "{{webpage}}" notify: restart_httpd
!wq
vim apache/tasks/main.yml
tasks file for apache
- import_tasks: package.yml

- import_tasks: service.yml
- import_tasks: firewall.yml
- import_tasks: webpage.yml wq!
vim apache/templates/index.html.j2
Welcome to {{ ansible_facts.fqdn }} on {{ ansible_facts.default_ipv4.address }}
vim apache/handlers/main.yml
--
handlers file for apache
- name: restart_httpd service:
name: httpd state: restarted wq!
cd ..
pwd
/home/admin/ansible/
vim httpd.yml
--
- name: Including apache role hosts: webservers
pre_tasks:
- name: pretask message
debug:
msg: 'Ensure webserver configuration' roles:
- ./roles/apache post_tasks:
- name: Check webserver uri:
url: "http://{{ ansible_facts.default_ipv4.address }} "
return_content: yes status_code: 200 wq!
ansible-playbook httpd.yml –-syntax-check
ansible-playbook httpd.yml
#
curl http://serverx

NEW QUESTION 4
- (Exam Topic 2)
Create Logical volumes with lvm.yml in all nodes according to following requirements.
---
* Create a new Logical volume named as 'data'
* LV should be the member of 'research' Volume Group
* LV size should be 1500M
* It should be formatted with ext4 file-system.
--> If Volume Group does not exist then it should print the message "VG Not found"
--> If the VG can not accommodate 1500M size then it should print "LV Can not be created with following size", then the LV should be created with 800M of size.
--> Do not perform any mounting for this LV.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
pwd
/home/admin/ansible
vim lvm.yml
--
- name: hosts: all
ignore_errors: yes tasks:
- name: lvol: lv: data
vg: research size: "1500"
- debug:
msg: "VG Not found"
when: ansible_lvm.vgs.research is not defined
- debug:
msg: "LV Can not be created with following size" when: ansible_lvm.vgs.research.size_g < "1.5"
- name: lvol: lv: data
vg: research size: "800"
when: ansible_lvm.vgs.research.size_g < "1.5"
- name:
filesystem: fstype: ext4
dev: /dev/research/data wq!
ansible-playbook lvm.yml –-syntax-check
ansible-playbook lvm.yml

NEW QUESTION 5
- (Exam Topic 2)
Install and configure Ansible on the control-node control.realmX.example.com as follows:

--> Install the required packages
--> Create a static inventory file called /home/admin/ansible/inventory as follows: node1.realmX.example.com is a member of the dev host group node2.realmX.example.com is a member of the test host group node3.realmX.example.com & node4.realmX.example.com are members of the prod host group node5.realmX.example.com is a member of the balancers host group. prod group is a member of the webservers host group
--> Create a configuration file called ansible.cfg as follows:
--> The host inventory file /home/admin/ansible/inventory is defined

--> The location of roles used in playbooks is defined as /home/admin/ansible/ roles

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
Through physical host, login to workstation.lab.example.com with user root.
ssh root@workstation.lab.example.com
hostname workstation.lab.example.com
yum install platform-python*
su - admin
pwd
/home/admin/
vim .vimrc
mkdir -p ansible/roles
cd ansible
vim inventory [dev]
servera.lab.example.com [test] serverb.example.com [prod] serverc.example.com serverd.example.com [balancer] serverd.lab.example.com [webservers:children]
prod
!wq
# vim ansible.cfg [defaults]
inventory = ./inventory
role_path = ./roles remote_user = admin ask_pass = false [privilege_escalation] become = true become_method = sudo become_user = root become_ask_pass =
false
!wq
ansible all -–list-hosts

NEW QUESTION 6
- (Exam Topic 2)
Install the RHEL system roles package and create a playbook called timesync.yml that:
--> Runs over all managed hosts.
--> Uses the timesync role.
--> Configures the role to use the time server 192.168.10.254 (Hear in redhat lab use "classroom.example.com")
--> Configures the role to set the iburst parameter as enabled.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
pwd home/admin/ansible/
sudo yum install rhel-system-roles.noarch -y
cd roles/
ansible-galaxy list
cp -r /usr/share/ansible/roles/rhelsystem-roles.timesync .
vim timesync.yml
--
- name: timesynchronization hosts: all
vars:
timesync_ntp_provider: chrony timesync_ntp_servers:
- hostname: classroom.example.com _ in exam its ip-address iburst: yes
timezone: Asia/Kolkata roles:
- rhel-system-roles.timesync tasks:
- name: set timezone timezone:
name: "{{ timezone }}" wq!
timedatectl list-timezones | grep india
ansible-playbook timesync.yml --syntax-check
ansible-playbook timesync.yml
ansible all -m shell -a 'chronyc sources -v'
ansible all -m shell -a 'timedatectl'
ansible all -m shell -a 'systemctl is-enabled chronyd'

NEW QUESTION 7
- (Exam Topic 2)
Generate a hosts file:
*
Download an initial template file hosts.j2 from http://classroom.example.com/ hosts.j2 to
/home/admin/ansible/ Complete the template so that it can be used to generate a file with a
line for each inventory host in the same format as /etc/hosts: 172.25.250.9 workstation.lab.example.com workstation
* Create a playbook called gen_hosts.yml that uses this template to generate the file
/etc/myhosts on hosts in the dev host group.
* When completed, the file /etc/myhosts on hosts in the dev host group should have a line for
each managed host:
* 127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
* 172.25.250.10 serevra.lab.example.com servera

* 172.25.250.11 serevrb.lab.example.com serverb
* 172.25.250.12 serevrc.lab.example.com serverc
* 172.25.250.13 serevrd.lab.example.com serverd
---
while practising you to create these file hear. But in exam have to download as per questation.
hosts.j2 file consists.
localhost localhost.localdomain localhost4 localhost4.localdomain4
::1
localhost localhost.localdomain localhost6 localhost6.localdomain6

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
pwd
/home/admin/ansible
#
wget http://classroom.example.com/hosts.j2
vim hosts.j2
* 127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4 ::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
{% for host in groups['all'] %}
{{ hostvars[host]['ansible_facts']['default_ipv4']['address'] }} {{ hostvars[host] ['ansible_facts']['fqdn'] }} {{ hostvars[host]['ansible_facts']['hostname'] }}
{% endfor %} wq!
vim gen_hosts.yml
--
- name: collecting all host information hosts: all
tasks:
- name: template: src: hosts.j2
dest: /etc/myhosts
when: inventory_hostname in groups['dev'] wq
ansible-playbook gen_hosts.yml -–syntax-check
ansible-playbook gen_hosts.yml

NEW QUESTION 8
- (Exam Topic 2)
Create and run an Ansible ad-hoc command.
--> As a system administrator, you will need to install software on the managed nodes.
--> Create a shell script called yum-pack.sh that runs an Ansible ad-hoc command to create yum-repository on each of the managed nodes as follows:
--> repository1
---
* 1. The name of the repository is EX407
* 2. The description is "Ex407 Description"
* 3. The base URL is http://content.example.com/rhel8.0/x86_64/dvd/BaseOS/
* 4. GPG signature checking is enabled
* 5. The GPG key URL is http://content.example.com/rhel8.0/x86_64/dvd/RPM-GPG-KEYredhat- release
* 6. The repository is enabled
--> repository2
---
* 1. The name of the repository is EXX407
* 2. The description is "Exx407 Description"
* 3. The base URL is http://content.example.com/rhel8.0/x86_64/dvd/AppStream/
* 4. GPG signature checking is enabled
* 5. The GPG key URL is http://content.example.com/rhel8.0/x86_64/dvd/ RPM-GPG-KEYredhat- release
* 6. The repository is enabled

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
pwd
/home/admin/ansible
vim yum-pack.sh
#!/bin/bash
ansible all -m yum_repository -a 'name=EX407 description="Ex407 Description"
baseurl=http://content.example.com/rhel8.0/x86_64/dvd/BaseOS/
gpgcheck=yes
gpgkey=http://content.example.com/rhel8.0/x86_64/dvd/RPM-GPG-KEY-redhat-release
enabled=yes'
ansible all -m yum_repository -a 'name=EXX407 description="Exx407 Description"
baseurl=http://content.example.com/rhel8.0/x86_64/dvd/AppStream/
gpgcheck=yes
gpgkey=http://content.example.com/rhel8.0/x86_64/dvd/RPM-GPG-KEY-redhat-release
enabled=yes'
!wq
chmod +x yum-pack.sh

bash yum-pack.sh
ansible all -m command -a 'yum repolist all'

NEW QUESTION 9
- (Exam Topic 1)
Create a Shell script /root/program:
The shell script will come back to ”user” parameter when you are entering ”kernel” parameter.
The shell script will come back to ”kernel” when you are entering ”user” parameter.
It will output the standard error when this script ”usage:/root/program kernel|user” don’t input any parameter or the parameter you inputted is entered as the requirements.

A. Mastered
B. Not Mastered

Answer: A

Explanation:

```
[root@server1 virtual]# cat/root/program
#!/bin/bash
param1="$1"
if [ "$param1" == "kernel" ]; then
echo "user"
elif [ "$param1" == "user" ]; then
echo "kernel"
else
echo "usage:/root/program kernel|user"
if
[root@server1 ~]# chmod +x /root/program
```

NEW QUESTION 10
- (Exam Topic 1)
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.

A. Mastered
B. Not Mastered

Answer: A

Explanation:

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

NEW QUESTION 10
- (Exam Topic 1)
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.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
/home/sandy/ansible/apache.yml

```
---
- name: http
  hosts: webservers
  roles:
    - sample-apache
```

/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
```

NEW QUESTION 13
- (Exam Topic 1)
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.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
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

NEW QUESTION 16
- (Exam Topic 1)
Create a playbook called timesvnc.yml in /home/sandy/ansible using rhel system role timesync. Set the time to use currently configured nip with the server 0.uk.pool.ntp.org. Enable burst. Do this on all hosts.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:

```
- name: use rhel system role
  hosts: all
  roles:
   - rhel-system-roles.timesync
  timesync_ntp_servers:
    - hostname: 0.uk.pool.ntp.org
      iburst: yes
```

NEW QUESTION 21
- (Exam Topic 1)
In /home/sandy/ansible/ create a playbook called logvol.yml. In the play create a logical volume called lv0 and make it of size 1500MiB on volume group vgO If there is not enough space in the volume group print a message "Not enough space for logical volume" and then make a 800MiB lv0 instead. If the volume group still doesn't exist, create a message "Volume group doesn't exist" Create an xfs filesystem on all lv0 logical volumes. Don't mount the logical volume.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:

```
- name: hosts
 hosts: all
 tasks:
 -  name: create partition
   parted:
     device: /dev/vdb
     number: 1
     flags: [ lvm ]
     state: present
- name: create vg
  lvg:
     vg: vg0
     pvs: /dev/vdb1
  when: ansible_devices.vdb.partitions.vdb1 is defined
- name: create logical volume
  lvol:
     vg: vg0
     lv: lv0
     size: 1500m
  when: ansible_lvm.vgs.vg0 is defined and ( (ansible_lvm.vgs.vg0.size_g | float ) > 1.5)
- name: send message if  volume group not large enough
  debug:
      msg: Not enough space for logical volume
  when: ansible_lvm.vgs.vg0 is defined and ( (ansible_lvm.vgs.vg0.size_g | float ) < 1.5)
- name: create a smaller logical volume
  lvol:
     vg: vg0
     lv: lv0
     size: 1500m
  when: ansible_lvm.vgs.vg0 is defined and ( (ansible_lvm.vgs.vg0.size_g | float ) < 1.5)
- name: create fs
 filesystem:
    dev: /dev/vg0/lv0
    fstype: xfs
 when: ansible_lvm.vgs.vg0 is defined
```

NEW QUESTION 25

.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questons and Answers in PDF Format

EX294 Practice Exam Features:

* EX294 Questions and Answers Updated Frequently

* EX294 Practice Questions Verified by Expert Senior Certified Staff

* EX294 Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* EX294 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year