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 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 4
- (Exam Topic 2)
Create a playbook called balance.yml as follows:
* The playbook contains a play that runs on hosts in balancers host group and uses the balancer role.
--> This role configures a service to loadbalance webserver requests between hosts in the webservers host group.curl
--> When implemented, browsing to hosts in the balancers host group (for example
http://node5.example.com)
should produce the following output:
Welcome to node3.example.com on 192.168.10.z
--> Reloading the browser should return output from the alternate web server: Welcome to node4.example.com on 192.168.10.a
* The playbook contains a play that runs on hosts in webservers host group and uses the phphello role.
--> When implemented, browsing to hosts in the webservers host group with the URL / hello.php should produce the following output:

Hello PHP World from FQDN
--> where FQDN is the fully qualified domain name of the host. For example, browsing
to http://node3.example.com/hello.php, should produce the following output: Hello PHP World from node3.example.com
*
Similarly, browsing to http://node4.example.com/hello.php, should produce the following output:
Hello PHP World from node4.example.com

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
pwd
/home/admin/ansible/
vim balancer.yml
--
- name: Including phphello role hosts: webservers
roles:
- ./roles/phphello
- name: Including balancer role hosts: balancer
roles:
- ./roles/balancer wq!
ansible-playbook balancer.yml --syntax-check
ansible-playbook balancer.yml

NEW QUESTION 5
- (Exam Topic 2)
Create a playbook called hwreport.yml that produces an output file called /root/ hwreport.txt on all managed nodes with the following information:

--> Inventory host name
--> Total memory in MB
--> 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 hwreport.empty from the URL http://classroom.example.com/ hwreport.empty and
save it as /root/hwreport.txt
--> Modify with the correct values.
note: If a hardware item does not exist, the associated value should be set to NONE
----
while practising you to create these file hear. But in exam have to download as per questation.
hwreport.txt file consists. my_sys=hostname
my_BIOS=biosversion my_MEMORY=memory my_vda=vdasize my_vdb=vdbsize

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
pwd
/home/admin/ansible
vim hwreport.yml
- name: hosts: all
ignore_errors: yes tasks:
- name: download file get_url:
url: http://classroom.example.com/content/ex407/hwreport.empty dest: /root/hwreport.txt
- name: vdasize replace:
regexp: "vdasize"
replace: "{{ ansible_facts.devices.vda.size }}" dest: /root/hwreport.txt
register: op1
- debug:
var: op1
- name: none replace:
regexp: "vdasize" replace: NONE
dest: /root/hwreport.txt when:
op1.failed == true
- name: vdbsize replace:
regexp: "vdbsize"
replace: "{{ ansible_facts.devices.vdb.size }}" dest: /root/hwreport.txt
register: op2
- debug: var: op2
- name: none replace:
regexp: "vdbsize" replace: NONE
dest: /root/hwreport.txt when:
op2.failed == true

```
- name: sysinfo replace:
regexp: "{{item.src}}"
replace: "{{item.dest}}" dest: /root/hwreport.txt loop:
- src: "hostname"
dest: "{{ ansible_facts.fqdn }}"
- src: "biosversion"
dest: "{{ ansible_facts.bios_version }}"
- src: "memory"
dest: "{{ ansible_facts.memtotal_mb }}" wq!
# ansible-playbook hwreport.yml ---syntax-check
# ansible-playbook hwreport.yml
```

NEW QUESTION 6
- (Exam Topic 2)
Rekey an existing Ansible vault as follows:
----
*
Download Ansible vault from http:// classroom.example.com /secret.yml to /home/ admin/ansible/
* The current vault password is curabete
* The new vault password is newvare
* The vault remains in an encrypted state with the new password

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:
```
# pwd
/home/admin/ansible/
#
wget http://classroom.example.com/secret.yml
# chmod 0600 newpassword.txt
# ansible-vault rekey vault.yml --new-vault-password-file=newpassword.txt
```

NEW QUESTION 7
- (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 8
- (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 9
- (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 10
- (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 14
- (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 15
- (Exam Topic 1)
Create an empty encrypted file called myvault.yml in /home/sandy/ansible and set the password to
notsafepw. Rekey the password to iwejfj2221. See the

A. Mastered
B. Not Mastered

Answer: A

Explanation:
ansible-vault create myvault.yml
Create new password: notsafepw Confirm password: notsafepw ansible-vault rekey myvault.yml
Current password: notsafepw New password: iwejfj2221 Confirm password: iwejfj2221

NEW QUESTION 18
- (Exam Topic 1)
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.

A. Mastered
B. Not Mastered

Answer: A

Explanation:

chmod 0777 adhoc.sh
vim adhoc.sh
#!/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'

NEW QUESTION 23
- (Exam Topic 1)
Create a file called requirements.yml in /home/sandy/ansible/roles a file called role.yml in
/home/sandy/ansible/. The haproxy-role should be used on the proxy host. And when you curl http://node3.example.com it should display "Welcome to
node4.example.com" and when you curl again "Welcome to node5.example.com" The php-role should be used on the prod host.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:

```
- name: install haproxy and php roles
  hosts: all
  vars:
    haproxy_backend_servers:
    - name: web1
      address: node4.example.com
    - name: web2
      address: node5.example.com
  tasks:
    - name: import haproxy
      include_role: haproxy-role
      when: "proxy" in group_names
    - name: import php
      include_role: php-role
      when: "prod" in group_names
```

Check the proxy host by curl http://node3.example.com

NEW QUESTION 27
- (Exam Topic 1)
Create a playbook called webdev.yml in 'home/sandy/ansible. The playbook will create a directory Avcbdev on dev host. The permission of the directory are 2755
and owner is webdev. Create a symbolic link from
/Webdev to /var/www/html/webdev. Serve a file from Avebdev7index.html which displays the text "Development" Curl http://node1.example.com/webdev/index.html

to test

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:

```
- name: webdev
  hosts: dev
  tasks:
    - name: create webdev user
      user:
        name: webdev
        state: present
    - name: create a directory
      file:
        mode: '2755'
        path: /webdev
        state: directory
    - name: create symbolic link
      file:
        src: /webdev
        path: /var/www/html/webdev
        state: link
    - name: create index.html
      copy:
        content: Development
        dest: /webdev/ index.html
    - name: Install selinux policies
      yum:
        name: python3-policycoreutils
        state: present
    - name: allow httpd from this directory
      sefcontext:
        target: '/webdev(/.*)?'
        setype: httpd_sys_content_t
        state: present
    - name: restore the context
      shell: restorecon -vR /webdev
```

NEW QUESTION 32
- (Exam Topic 1)
Create a jinja template in /home/sandy/ansible/ and name it hosts.j2. Edit this file so it looks like the one below. The order of the nodes doesn't matter. Then create a playbook in /home/sandy/ansible called hosts.yml and install the template on dev node at /root/myhosts

```
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1          localhost localhost.localdomain localhost6 localhost6.localdomain6

10.0.2.1        node1.example.com        node1
10.0.2.2        node2.example.com        node2
10.0.2.3        node3.example.com        node3
10.0.2.4        node4.example.com        node4
10.0.2.5        node5.example.com        node5
```

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution as:

in /home/sandy/ansible/hosts.j2

{%for host in groups['all']%}
{{hostvars[host]['ansible_default_ipv4']['address']}} {{hostvars[host]['ansible_fqdn']}}
{{hostvars[host]['ansible_hostname']}}
{%endfor%}

in /home/sandy/ansible/hosts.yml

 - name: use template
   hosts: all
   template:
       src: hosts.j2
       dest: /root/myhosts
   when: "dev" in group_names

NEW QUESTION 35
- (Exam Topic 1)
Create an ansible vault password file called lock.yml with the password reallysafepw in the
/home/sandy/ansible directory. In the lock.yml file define two variables. One is pw_dev and the password is 'dev' and the other is pw_mgr and the password is 'mgr'
Create a regular file called secret.txt which contains the password for lock.yml.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
ansible-vault create lock.yml
New Vault Password: reallysafepw Confirm: reallysafepw
In File:

```
pw_dev: dev
pw_mgr: mgr
```

NEW QUESTION 37
......

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

100% Actual & Verified — Instant Download, Please Click
Order The EX294 Practice Test Here