

Exam Questions EX294

Red Hat Certified Engineer (RHCE) exam

<https://www.2passeasy.com/dumps/EX294/>



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 directory 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 firewallld:
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 webserver 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: firewallld http_srv: httpd firewall_srv: firewallld 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 firewallld:
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)

Create a playbook called packages.yml that:

--> Installs the php and mariadb packages on hosts in the dev, test, and prod host groups.

--> Installs the Development Tools package group on hosts in the dev host group.

--> Updates all packages to the latest version on hosts in the dev host group.

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Solution as:

```
# pwd home/admin/ansible/
# vim packages.yml
--
- name: Install the packages hosts: dev,test,prod
vars:
- php_pkg: php
- mariadb_pkg: mariadb tasks:
- name: install the packages yum:
name:
- "{{ php_pkg }}"
- "{{ mariadb_pkg }}"
state: latest
- name: install the devops tool packages hosts: dev
tasks:
- name: install devepment tools yum:
name: "@Development Tools" state: latest
- name: upgrade all the packages yum:
name: "*" state: latest
exclude: kernel*
!wq
# ansible-playbook package.yml --syntax-check
# ansible-playbook package.yml
```

NEW QUESTION 6

- (Exam Topic 2)

Modify file content.

Create a playbook called /home/admin/ansible/modify.yml as follows:

* The playbook runs on all inventory hosts

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

--> On hosts in the dev host group, the line reads: "Development"

--> On hosts in the test host group, the line reads: "Test"

--> On hosts in the prod host group, the line reads: "Production"

A. Mastered

B. Not Mastered

Answer: A

Explanation:

Solution as:

```
# pwd
/home/admin/ansible
# vim modify.yml
--
- name: hosts: all tasks:
- name: copy:
content: "Development" dest: /etc/issue
when: inventory_hostname in groups['dev']
- name: copy:
content: "Test" dest: /etc/issue
when: inventory_hostname in groups['test']
- name: copy:
content: "Production" dest: /etc/issue
when: inventory_hostname in groups['prod'] wq
# ansible-playbook modify.yml --syntax-check
# ansible-playbook modify.yml
```

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 serevrblab.example.com serverb

* 172.25.250.12 serevrc.lab.example.com serverc

* 172.25.250.13 serevrld.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 playbook called regulartasks.yml which has the system that append the date to /root/datefile every day at noon. Name is job 'datejob'

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Solution as:

```
- name: Creates a cron file under /etc/cron.d
cron:
  name: datejob
  hour: "12"
  user: root
  job: "date >> /root/ datefile"
```

NEW QUESTION 10

- (Exam Topic 1)

Create a file in /home/sandy/ansible/ called report.yml. Using this playbook, get a file called report.txt (make it look exactly as below). Copy this file over to all remote hosts at /root/report.txt. Then edit the lines in the file to provide the real information of the hosts. If a disk does not exist then write NONE.

report.txt

```
HOST=inventory hostname
MEMORY=total memory in mb
BIOS=bios version
VDA_DISK_SIZE=disk size
VDB_DISK_SIZE=disk size
```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Solution as:

```
- name: edit file
hosts: all
tasks:
  - name: copy file
    copy: report.txt
    dest: /root/report.txt
  - name: change host
    lineinfile:
      regex: ^HOST
      line: HOST={{ansible_hostname}}
      state: present
      path: /root/report.txt
  - name: change mem
    lineinfile:
      line: MEMORY={{ansible_memtotal_mb}}
      regex: ^MEMORY
      state: present
      path: /root/report.txt
```

```
- name: change bios
  lineinfile:
    line: BIOS={{ansible_bios_version}}
    regex: ^BIOS
    state: present
    path: /root/report.txt
- name: change vda
  lineinfile:
    line: VDA_DISK_SIZE ={%if ansible_devices.vda is defined%}{{ansible_devices.vda.size}}{%else%}NONE{%endif%}
    regex: ^VDA_DISK_SIZE
    state: present
    path: /root/report.txt
- name: change vdb
  lineinfile:
    line: VDB_DISK_SIZE ={%if ansible_devices.vdb is defined%}{{ansible_devices.vdb.size}}{%else%}NONE{%endif%}
    regex: ^VDB_DISK_SIZE
    state: present
    path: /root/report.txt
```

NEW QUESTION 10

- (Exam Topic 1)

Create a file called packages.yml in /home/sandy/ansible to install some packages for the following hosts. On dev, prod and webserver 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.

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Solution as:

```
---
- name: install pack
  hosts: dev,test,webserver
  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

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 18

- (Exam Topic 1)

Create a playbook /home/bob /ansible/motd.yml that runs on all inventory hosts and docs the following: The playbook should replace any existing content of/etc/motd in the following text. Use ansible facts to display the FQDN of each host

On hosts in the dev host group the line should be "Welcome to Dev Server FQDN".

On hosts in the webserver host group the line should be "Welcome to Apache Server FQDN". On hosts in the database host group the line should be "Welcome to MySQL Server FQDN".

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

/home/sandy/ansible/apache.yml

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

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

NEW QUESTION 20

- (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 22

- (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 27

.....

THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual EX294 Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the EX294 Product From:

<https://www.2passeasy.com/dumps/EX294/>

Money Back Guarantee

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