



## Red-Hat

### Exam Questions EX294

Red Hat Certified Engineer (RHCE) exam

### NEW QUESTION 1

- (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 2

- (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 development 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 3

- (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 4

- (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 5

- (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](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 6**

- (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 7**

- (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: `node1.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`. `node2` is a member of group `test`, `node3` is a member of group `proxy`, `node4` 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 8

- (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 iwej2221. 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: iwej2221 Confirm password: iwej2221
```

### NEW QUESTION 9

- (Exam Topic 1)

Create a playbook called issue.yml in /home/sandy/ansible which changes the file /etc/issue on all managed nodes: If host is a member of (lev then write "Development" If host is a member of test then write "Test" If host is a member of prod then write "Production"

- A. Mastered
- B. Not Mastered

**Answer:** A

#### Explanation:

Solution as:

```
...
- name: issue file
  hosts: dev,test,prod
  tasks:
    - name: edit development node
      copy:
        content: Development
        dest: /etc/issue
        when: "dev" in group_names
    - name: edit test node
      copy:
        content: Test
        dest: /etc/issue
        when: "test" in group_names
    - name: edit development node
      copy:
        content: Production
        dest: /etc/issue
        when: "prod" in group_names
...

```

### NEW QUESTION 10

- (Exam Topic 1)

Create a file called specs.empty in home/bob/ansible on the local machine as follows: HOST= MEMORY= BIOS= VDA\_DISK\_SIZE= VDB\_DISK\_SIZE= Create the playbook /home/bob/ansible/specs.yml which copies specs.empty to all remote nodes' path /root/specs.txt. Using the specs.yml playbook then edit specs.txt on the remote machines to reflect the appropriate ansible facts.

- 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 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 15

- (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 20

- (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 25

.....

## Thank You for Trying Our Product

### We offer two products:

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

2nd - Questions 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 First Try
- \* EX294 Practice Test Questions in Multiple Choice Formats and Updates for 1 Year

**100% Actual & Verified — Instant Download, Please Click**  
[Order The EX294 Practice Test Here](#)