

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 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 3

- (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 4

- (Exam Topic 1)

Create a playbook that changes the default target on all nodes to multi-user target. Do this in playbook file called target.yml in /home/sandy/ansible

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

- name: change default target hosts: all

tasks:

- name: change target file:

src: /usr/lib/systemd/system/multi-user.target dest: /etc/systemd/system/default.target state: link

NEW QUESTION 5

- (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 6

- (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 7

- (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 8

- (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 9

- (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 the users in the file usersjst.yml file provided. Do this in a playbook called users.yml located at /home/sandy/ansible. The passwords for these users should be set using the lock.yml file from TASK7. When running the playbook, the lock.yml file should be unlocked with secret.txt file from TASK 7.

All users with the job of 'developer' should be created on the dev hosts, add them to the group devops, their password should be set using the pw_dev variable. Likewise create users with the job of 'manager' on the proxy host and add the users to the group 'managers', their password should be set using the pw_mgr variable.

users_list.yml

```
users:
- username: bill
  job: developer
- username: chris
  job: manager
- username: dave
  job: test
- username: ethan
  job: developer
```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

ansible-playbook users.yml --vault-password-file=secret.txt

```
- name: create users
  hosts: all
  vars_files:
    - users_list.yml
    - lock.yml
  tasks:
    - name: create devops group nodes1
      group:
        name: devops
      when: ('dev' in group_names)
    - name: create manager group nodes45
      group:
        name: manager
      when: ('prod' in group_names)
    - name: create devs should happen on node1
      user:
        name: "{{item.username}}"
        groups: devops
        password: "{{ pw_dev | password_hash('sha512') }}"
      when: ('dev' in group_names) and ('developer' in item.job)
      loop: "{{users}}"
    - name: create managers on node45
      user:
        name: "{{item.username}}"
        groups: manager
        password: "{{ pw_mgr | password_hash('sha512') }}"
      when: ('prod' in group_names) and ('manager' in item.job)
      loop: "{{users}}"
```

NEW QUESTION 10

- (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 ntp 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 12

- (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.rpm>' 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.rpm
gpgcheck=no enabled=yes'
```

NEW QUESTION 17

- (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 18

.....

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