Practice Exam for RHCSA-EX200

This is a sample of RHCSA-EX200 exam that I've created to prepare for my exam. As with the real exam, no perfect answers to the sample exam questions will be provided, but more or less correct and accurate. Remember you want to cover all topics first before start dealing with questions.

Requirements

There are 18 questions in total could be more or less. You will be given 2 RHEL 8 virtual machines which you need to configure properly to be able to successfully complete all questions and pass the exam.

One VM will be configured as an Server-A and another would be Server-B. there will be a repository VM which will enable you to install packages to your VMs. The following FQDNs will be used throughout the sample exam.

FQDN	Description	IP Addresses	Network mask
node1.domain250.example.com	node1	172.25.250.100	255.255.255.0
node2.domain250.example.com	node2	172.25.250.101	255.255.255.0

Lab Setup

> you can create the lab setup manually, but instead i've `Vagrantfile` which you can use inorder to create this setup, please go to this website for more information regarding lab setup https://github.com/rdbreak/rhcsa8env

Question:1 Configure your Host Name, IP Address, Gateway and DNS.

Configure your Host Name, IP Address, Gateway and DNS.

- Host name: `mars.domain250.example.com`
- IP Address: `172.25.250.100/24`
- Network Mask: `255.255.255.0`
- Gateway: `172.25.250.254`
- DNS: `172.25.250.254`

Answer:1 Configure your Host Name, IP Address, Gateway and DNS.

> setting up hostname

```
[root@Server-A ~]# vim /etc/hostname
```

or

```
[root@Server-A ~]# hostnamectl set-hostname <hostname>
```

setting ipv4/GW/DNS

```
[root@Server-A ~]# nmcli connection modify enp0s3 autoconnect yes ipv4.method manual ipv4.ad
```

or

```
[root@Server-A ~]# vim /etc/sysconfig/network-scripts/ifcfg-enp0s3
IPADDR=172.25.250.100
GATEWAY=172.25.250.254
DNS1=172.25.250.254
```

Question:2 Configure your system to use the default repository

YUM repositories are already available from

- `http://foundation0.ilt.example.com/dvd/BaseOS`
- `http://foundation0.ilt.example.com/dvd/AppStream`
- Configure your system to use these locations as default repository

Answer:2 Configure your system to use the default repository

1. Install the yum-config-manager installation package

```
[root@clear ~]# rpm -ivh http://foundation0.ilt.example.com/dvd/BaseOS/Packages/yum-utils-4.
```

2. repository file download and install

```
# yum-config-manager command could be helpful to set a local repository quickly
# yum-config-manager -h command to look for some help
```

```
[root@clear ~]# yum-config-manager --add-repo http://foundation0.ilt.example.com/dvd/BaseOS
Adding repo from: http://foundation0.ilt.example.com/dvd/BaseOS
```

```
[root@clear ~]# yum-config-manager --add-repo http://foundation0.ilt.example.com/dvd/AppStre
Adding repo from: http://foundation0.ilt.example.com/dvd/AppStream
```

3. Check if the installation is successful

```
[root@clear ~]# cd /etc/yum.repos.d/
[root@clear yum.repos.d]# ls
foundation0.ilt.example.com_dvd_AppStream.repo  foundation0.ilt.example.com_dvd_BaseOS.repo
```

4. you should also ensure some parameters are correct

```
[root@clear yum.repos.d]# vim foundation0.ilt.example.com_dvd_BaseOS.repo
# enabled=1
# gpgcheck=0
```

Question:3 Debug SELinux (service)

A web server running on a non-standard port 82 is having trouble serving content. Debug and resolve the issue as necessary so that the following conditions are met:

- The web server on the system is able to serve all existing HTML files in `/var/www/html` (note: do not delete or otherwise alter the contents of existing files)
- The web server serves this content on port 82
- The web server starts automatically at system startup

Answer:3 Debug SELinux (service)

- semanage command is used to query and modify the security context of the SELinux default directory
- apache belongs to `httpd` service

1. View httpd service status

```
[root@node1 ~]# systemctl status httpd
Active: failed (Result: exit-code)
```

2. View the security context of the HTML file

```
ls -Z  #prints the security context of the file
```

```
[root@node1 html]# ls -Z /var/www/html/*
system_u:object_r:default_t:s0 /var/www/html/file1
system_u:object_r:httpd_sys_content_t:s0 /var/www/html/file2
system_u:object_r:httpd_sys_content_t:s0 /var/www/html/file3
```

3. Modify the security context of the original `/var/www/html/file1` file

```
man semange fcontext
# parameters to look for -a is replaced by -m (modify)
```

```
[root@node1 html]# semanage fcontext -m -t httpd_sys_content_t "/var/www/html/file1"
```

4. Refresh the security context

```
[root@node1 html]# restorecon -R -v /var/www/html/file1
Relabeled /var/www/html/file1 from system_u:object_r:default_t:s0 to system_u:object_r:httpc
```

5. Use semanage to release port 82

```
# man semanage port
[root@node1 ~]# semanage port -a -t http_port_t -p tcp 82
```

6. Check whether port 82 is allowed

```
# man semanage port
[root@node1 ~]# semanage port -l | grep http
http_port_t tcp 82, 80, 81, 443, 488, 8008, 8009, 8443, 9000
```

7. Restart the httpd service, set the boot to start automatically, and check whether the service is enabled

```
[root@node1 ~]# systemctl restart httpd
[root@node1 ~]# systemctl enable httpd
[root@node1 ~]# systemctl status httpd
```

8. Access verification

```
[root@node1 html]# curl http://172.25.250.100:82/file{1..3}
```

Question:4 create user account

Create the following users, groups, and group memberships: A group named sysmgrs

- User natasha , who also belongs to `sysmgrs` as a secondary group
- User harry , who also belongs to `sysmgrs` as a secondary group
- User sarah , does not have access to an interactive shell on the system and is not a member of `sysmgrs`
- passwords for natasha , harry and sarah should all be `password`

Answer:4 create user account

```
[root@clear ~]# groupadd sysmgrs
[root@clear ~]# useradd natasha -G sysmgrs
[root@clear ~]# useradd harry  -G sysmgrs
[root@clear ~]# useradd sarah --shell /sbin/nologin
[root@clear ~]# echo "password" | passwd --stdin natasha
Changing password for user natasha.
passwd: all authentication tokens updated successfully.
[root@clear ~]# echo "password" | passwd --stdin harry
Changing password for user harry.
passwd: all authentication tokens updated successfully.
[root@clear ~]# echo "password" | passwd --stdin sarah
Changing password for user sarah.
passwd: all authentication tokens updated successfully.

Check connectivity
[root@node1 ~]# ssh natasha@localhost
[harry@node1 ~]$ ssh sarah@localhost
[harry@node1 ~]$ ssh sarah@localhost    #no interactive shell allocated for sarah /sbin/nolo
```

Question:5 Configure a cron job (service)

Configure a cron job that runs every 2 minutes and executes the following command:

- logger "EX200 in progress", run as user natasha

Answer:5 Configure a cron job (service)

```
1.check if cron service is currently running
[root@node1 ~]# systemctl status crond.service
Active: active (running)

2. Edit scheduled tasks
[root@node1 ~]# crontab -u natasha -e
[root@node1 ~]# crontab -u natasha -l
*/2 * * * * logger "EX200 in progress"

3. enable and restart service
[root@node1 ~]# systemctl enable crond.service

4.Check if log message are being loggged
#verify
[root@node1 ~]# grep EX200 /var/log/messages
node1 natasha[28082]: EX200 in progress
node1 natasha[28086]: EX200 in progress
node1 natasha[28097]: EX200 in progress
```

Question:6 Create a collaboration directory

- `/home/managers` with the following characteristics:
- The group permissions for `/home/managers` are `sysmgrs`.
- The directory should be `read`, `write`, and `accessible by members of sysmgrs`, but not by `any other user`. (Of course, the root user has access to all files and directories on the system).
- Files created in `/home/managers` automatically set group ownership to the `sysmgrs group`.

Answer:6 Create a collaboration directory

```
Check if directory exists
[root@node1 ~]# ll -d /home/managers

1. Create the specified directory file
[root@node1 ~]# mkdir /home/managers
```

```
2.Modify the group permission of `/home/managers` to sysmgrs

[root@node1 ~]# ll -d /home/managers/
drwxr-xr-x. 2 root root 6 May 14 18:16 /home/managers/

[root@node1 ~]# chown root:sysmgrs /home/managers/

[root@node1 ~]# ll -d /home/managers/
drwxr-xr-x. 2 root sysmgrs 6 May 14 18:16 /home/managers/

3. Modify the directory file and the permissions of the group to which it belongs
[root@node1 ~]# chmod 070 /home/managers/
[root@node1 ~]# chmod g=rwx,o=- /home/managers

[root@node1 ~]# ll -d /home/managers/
d---rwx---. 2 root sysmgrs 6 May 14 18:16 /home/managers/

4. Set special permissions for the `/home/managers` directory so that its subdirectories inh
[root@node1 ~]# chmod g+s /home/managers/

[root@node1 ~]# ll -d /home/managers/
d---rws---. 2 root sysmgrs 6 May 14 18:16 /home/managers/

5. Verify the effect of g+s permission
[root@node1 managers]# touch file
[root@node1 managers]# ll
total 0
-rw-r--r--. 1 root sysmgrs 0 May 14 18:27 file
```

Question:7 Configure NTP (time synchronization service) (service)

- Configure your system to be an NTP client for `materials.example.com`. (Note: materials.example.com is a DNS alias for classroom.example.com)

Answer:7 Configure NTP (time synchronization service) (service)

```
# systemctl list-units lists all startup units
# Unit is the basic unit for Systemd to manage system resources

# The client side synchronizes with the server
# nodel is the client materials.example.com's NTP is the server

1.View the name of the main configuration file of the NTP service
```

```
[root@node1 ~]# systemctl list-units | grep NTP
chronyd.service

2. Confirm whether the service is started
[root@node1 ~]# systemctl status chronyd.service
Active: active (running)

3. Modify the service configuration file and specify the address of the upstream server to b
[root@node1 ~]# vim /etc/chrony.conf
#server_gateway iburst
server materials.example.com iburst

4. Restart the service, let the modification of the configuration file take effect, and set
[root@node1 ~]# systemctl restart chronyd.service
[root@node1 ~]# systemctl enable chronyd.service

5. Verify
method one:
[root@node1 ~]# chronyc sources -v #View synchronization time status
^* classroom.example.com

method two:
[root@node1 ~]# timedatectl
#Check whether NTP is active and whether the system clock is synchronized
System clock synchronized: yes
               NTP service: active
```

Question:8 Configure autofs (service)

- Configure autofs to automatically mount remote users' home directories as follows:
- materials.example.com (172.25.254.254) NFS exports /rhome to your system. This file system contains a preconfigured home directory for - user `remoteuser1`
- remoteuser1's home directory is `materials.example.com:/rhome/remoteuser1`
- The home directory of remoteuser1 should be automatically mounted to `/rhome/remoteuser1` under the local `/rhome`
- The home directory must be writable by its user
- The password for remoteuser1 is `password`

Answer:8 Configure autofs (service)

```
1. Find the main configuration file
[root@node1 ~]# rpm -qc autofs
/etc/auto.master # absolute path configuration file
/etc/auto.misc # relative path configuration file

2. Configure the main configuration file
[root@node1 ~]# vim /etc/auto.master
/misc /etc/auto.misc
/rhome /etc/auto.rhome #relative path configuration file

3. Copy the relative path configuration file to the path written by the main configuration f
[root@node1 ~]# cp /etc/auto.misc /etc/auto.rhome

4. Edit the relative path configuration file
[root@node1 ~]# vim /etc/auto.rhome
remoteuser1 -rw materials.example.com:/rhome/remoteuser1

5. restart service
[root@node1 ~]# systemctl restart autofs.service

check status
[root@node1 ~]# systemctl status autofs.service
Active: active (running)

Set up autostart
[root@node1 ~]# systemctl enable autofs.service
Created symlink /etc/systemd/system/multi-user.target.wants/autofs.service → /usr/lib/systen

6. Login test
[root@node1 ~]# ssh remoteuser1@localhost
[remoteuser1@clear ~]$ pwd
/rhome/remoteuser1
Check write permissions
[remoteuser1@node1 ~]$ touch 1.txt
[remoteuser1@node1 ~]$ ll
total 0
-rw-rw-r--. 1 devops devops 0 May 27 12:04 1.txt
```

Question:9 Configure /var/tmp/fstab permissions

- Copy the file `/etc/fstab` to `/var/tmp/fstab` . Configure the permissions of `/var/tmp/fstab` to meet the following conditions:
- File `/var/tmp/fstab` owned by `root` user
- File `/var/tmp/fstab` belongs to group `root`

- The file /var/tmp/fstab should not be executable by anyone.
- User natasha can read and write to /var/tmp/fstab.
- User harry cannot write or read /var/tmp/fstab.
- All other users (current or future) can read /var/tmp/fstab.

Answer:9 Configure /var/tmp/fstab permissions

```
1. File copy
[root@node1 ~]# cp /etc/fstab /var/tmp/fstab

2. Check whether the owner and the owner group meet the meaning of the question, and there i
[root@node1 ~]# ll -d /var/tmp/fstab
-rw-r--r--. 1 root root 534 May 23 12:27 /var/tmp/fstab

3. Set setfacl permission
man setfacl
[root@node1 ~]# setfacl -m u:natasha:rw-,u:harry:- /var/tmp/fstab

4. Verify
[root@node1 ~]# getfacl /var/tmp/fstab
user:natasha:rw-
user:harry:---
```

Question:10 Configure user accounts

- Configure user manalo with user ID 3533.

Answer:10 Configure user accounts

```
[root@clear ~]# useradd -u 3533 manalo
[root@clear ~]# echo "password" | passwd --stdin manalo
Changing password for user manalo.
passwd: all authentication tokens updated successfully.
```

Question:11 Find files

- Find all files owned by jacques and place a copy of them in the /root/findfiles directory.

Answer:11 Find files

```
1. Create a directory
[root@clear ~]# mkdir /root/findfiles

2. Find
[root@clear ~]# find / -user jacques -exec cp -a {} /root/findfiles/ \;
```

```
3. Verification
[root@clear ~]# ls /root/findfiles/
gamelan jacques libWedgeit.so.1.2.3
```

Question:12 Find a string

- Finds all lines in the file `/usr/share/xml/iso-codes/iso_639_3.xml` that contain the string `ng`.
- Put a copy of all these lines in the file `/root/list` in their original order.
- `/root/list` must not contain empty lines, and all lines must be exact copies of the original lines in `/usr/share/xml/iso-codes/iso_639_3.xml`.

Answer:12 Find a string

```
1. Find the string
[root@node1 ~]# grep ng /usr/share/xml/iso-codes/iso_639_3.xml | grep -v ^$ > /root/list

2. Verification
[root@node1 ~]# cat -n /root/list
```

Question:13 Create archive

- Create a tar archive named `/root/backup.tar.gz`, which should contain the tar archive of`/usr/local`
- Which should contain the contents of `/usr/local`. The tar archive must be compressed with `gzip` format.

Answer:13 Create archive

```
1. pack
[root@clear ~]# tar -czvf /root/backup.tar.gz /usr/local

2. Verification
[root@clear ~]# ls
anaconda-ks.cfg findfiles
backup.tar.gz original-ks.cfg

3. Verify whether it is gzip
[root@clear ~]# file backup.tar.gz
backup.tar.gz: gzip compressed data, last modified: Sun May 22 17:10:03 , from Unix, origina
```

Question:14 Add a swap partition

Adding an extra `512M` swap partition to your system, this swap partition should mount automatically when the system starts up. Don't remove and modify the

existing swap partitions on your system.

Answer:14 Add a swap partition

```
1. List all block devices
[root@node2~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
vda 252:0 0 10G 0 disk
 vda1 252:1 0 1M 0 part
 vda2 252:2 0 100M 0 part /boot/efi
 vda3 252:3 0 9.9G 0 part /
vdb 252:16 0 4G 0 disk
 vdb1 252:17 0 510M 0 part
   myvol-vo 253:1 0 232M 0 lvm /reports
 vdb2 252:18 0 512M 0 part
    vgroup-swap 253:0 0 256M 0 lvm [SWAP]
vdc 252:32 0 10G 0 disk

2. Create disk partitions
[root@node2 ~]# fdisk /dev/vdb

3. Formatting
[root@node2 ~]# mkswap /dev/vdb3
Setting up swapspace version 1, size = 512 MiB (792719360 bytes)
no label, UUID=ba522efa-8aa3-4e96-b8e2-39aafa20f3cb

4. Permanent mount
[root@node2 ~]# vim /etc/fstab
UUID=ba522efa-8aa3-4e96-b8e2-39aafa20f3cb none swap defaults 0 0

5. Verify
[root@node2 ~]# swapon -a #refresh
[root@node2 ~]# swapon
NAME TYPE SIZE USED PRIO
/dev/dm-0 partition 256M 0B -2
/dev/vdb3 partition 512M 0B -3
```

Question:15 Create a logical volume

Create a new logical volume as required:

- Name the logical volume as database, belongs to datastore of the volume group, size is `60 PE`.
- Expansion size of each volume in volume group datastore is `16MB`.
- Use ext3 to format this new logical volume, this logical volume should automatically mount to `/mnt/database`.

Answer:15 Create a logical volume

```
1. Create a disk partition
[root@node2 ~]# fdisk /dev/vdb

2. Create a physical group (pv)
[root@node2 ~]# pvcreate /dev/vdb4
   Physical volume "/dev/vdb4" successfully created.

3. Create a volume group (vg)
[root@node2 ~]# vgcreate qagroup -s 16M /dev/vdb4
# -s Extended block (PE) size of the physical volume on the volume group
   Volume group "qagroup" successfully created

4. Create a logical volume (lv)
[root@node2 ~]# lvcreate -n qa -l 60 /dev/qagroup
   Logical volume "qa" created.

5. Formatting
[root@node2 ~]# mkfs.ext3 /dev/qagroup/qa

6. View UUID
[root@node2 ~]# blkid /dev/qagroup/qa
/dev/qagroup/qa: UUID="5ad7f2df-9749-4a46-adb6-853f3805d795" SEC_TYPE="ext2" TYPE="ext3"

7. Create /mnt/qa directory
[root@node2 ~]# mkdir /mnt/qa

8. Make a permanent mount
[root@node2 ~]# vim /etc/fstab
UUID="5ad7f2df-9749-4a46-adb6-853f3805d795" /mnt/qa ext3 defaults 0 0

9. load
[root@node2 ~]# mount -a # Load all devices set in the file /etc/fstab

10.
[root@node2 ~]# df -h
/dev/mapper/qagroup-qa 929M 1.2M 880M 1% /mnt/qa
```

Question:16 Create a logical volume

- Set logical volume size
- Resize the logical volume vo and its file system to `230 MiB`. Make sure the filesystem contents remain unchanged. Note: The partition size is rarely exactly the requested size, so a range of `230 MiB` to `270 MiB` is acceptable.

Answer:16 Create a logical volume

```
1. Query logical volume vo
[root@node2 ~]# df -h
/dev/mapper/myvol-vo 175M 1.6M 160M 1% /reports

2. Expansion
[root@node2 ~]# lvextend -L 250M /dev/mapper/myvol-vo #Expand logical volume space
   Logical volume myvol/vo successfully resized.

3. Query formatting type (ext4)
[root@node2 ~]# blkid | grep vo
/dev/mapper/myvol-vo: UUID="67994f68-d3e1-4686-8393-8df05149883f" TYPE="ext4"


5. Refresh according to type
[root@node2 ~]# resize2fs /dev/mapper/myvol-vo

6. Verify
[root@node2 ~]# df -h
/dev/mapper/myvol-vo 240M 2.1M 204M 1% /reports
```

Question:17 Permissions

1. Find all sizes of 10k file or directory under the `/etc` directory, and copy to `/tmp/findfiles` directory.
2. Find all the files or directories with `Lucy as the owner`, and copy to `/tmp/findfiles` directory.

Answer:17 Permissions

```
1. File copy
[root@node1 ~]# cp /etc/fstab /var/tmp/fstab

2. Check whether the owner and the owner group meet the meaning of the question, and there i
[root@node1 ~]# ll -d /var/tmp/fstab
-rw-r--r--. 1 root root 534 May 23 12:27 /var/tmp/fstab

3. Set setfacl permission
man setfacl
[root@node1 ~]# setfacl -m u:natasha:rw-,u:harry:- /var/tmp/fstab

4. Verify
[root@node1 ~]# getfacl /var/tmp/fstab
user:natasha:rw-
user:harry:---
```

Question:18 Create VDO volume (service)

- Create VDO volumes
- Create a new VDO volume with the following requirements:
- use unpartitioned disk
- The name of the volume is `vdough`
- The logical size of the volume is `50G`
- The volume is formatted with the `xfs` file system
- The volume is mounted (at system boot) under `/vbread`

```
1. Search for the installation package
[root@node2 ~]# yum search vdo
vdo.x86_64
kmod-kvdo.x86_64

2. Check the installation
[root@node2 ~]# rpm -q vdo kmod
package vdo is not installed
kmod-25-16.el8.x86_64

3. Install
[root@node2 ~]# yum install -y vdo.x86_64 kmod-kvdo.x86_64
Installed:
   kmod-kvdo-6.2.2.117-65.el8.x86_64
   vdo-6.2.2.117-13.el8.x86_64

Complete!

4.man vdo view format
[root@node2 ~]# vdo create --name=vdough --device=/dev/vdc --vdoLogicalSize=50G

5. Formatting
# -K quick format (uppercase)

[root@node2 ~]# mkfs.xfs -K /dev/mapper/vdough

6. Create a mount point
[root@node2 ~]# mkdir /vbread

7. Find the UUID
[root@node2 ~]# blkid
or
[root@node2 ~]# blkid /dev/mapper/vdough
```

```
/dev/mapper/vdough: UUID="a1f68c65-cf38-4cc8-b508-860a2e90397c" TYPE="xfs"

8. Make a permanent mount, and then mount it through the network
[root@node2 ~]# vim /etc/fstab
UUID="a1f68c65-cf38-4cc8-b508-860a2e90397c" /vbread xfs _netdev 0 0

9. load
[root@node2 ~]# mount -a # Load all devices set in the file /etc/fstab

10.
[root@node2 ~]# df -h
/dev/mapper/vdough 50G 390M 50G 1% /vbread

11. vdo is a service that needs to be set to start automatically at boot
[root@node2 ~]# systemctl restart vdo
[root@node2 ~]# systemctl enable vdo
[root@node2 ~]# systemctl status vdo
```

Question:20 Create a script for locating files

- Create a script named `/usr/local/bin/file.sh`.
- Find all files under `/usr` that are less than 10M and have `sgid` permissions set.
- Save the found file list to `/root/myfile`.

Answer:20 Create a script for locating files

```
[root@servera ~]# vim /usr/local/bin/file.sh
#!/bin/bash
find /usr -size -10M -perm -2000 > /root/myfile
[root@servera ~]# chmod a+x /usr/local/bin/file.sh
[root@servera ~]# file.sh
[root@servera ~]# cat /root/myfile
/usr/bin/write
/usr/bin/locate
/usr/libexec/utempter/utempter
/usr/libexec/openssh/ssh-keysign
```

Question:21 Creating scripts

- Create a newsearchscript called
- The script is placed `/usr/binunder`
- This script is used to find /usrall files that are greater than `30k`, but less than `50k` and have SUIDpermissions, and place these file names in the `/root/newfilesfile`

Answer:21 Creating scripts

```
# vim /usr/bin/newsearch

#!/bin/bash
touch /root/newfiles
find /usr -size +30k -size -50k -perm /u=s > /root/newfiles
```

```
# chmod +x /usr/bin/newsearch

# ./usr/bin/newsearch

# cat /root/newfiles
```

Question:22 Creating scripts

create script

- Create a myresearchscript called
- The script is placed `/usr/binunder`
- This script is used to find `/usrall` files that are smaller than `10m` and have modification Group ID permissions `s`, and place these files `/root/myfilesunder`

Answers:22 Creating scripts

```
# vim /usr/bin/newsearch

#!/bin/bash
mkdir /root/newfiles
find /usr -size -10M -perm /u=s cp -a {} /root/newfiles \;
```

```
# chmod +x /usr/bin/newsearch

# ./usr/bin/newsearch

# cat /root/newfiles
```