RHCSAv9

==

*Important Instructions, read carefully.

* You will be given by 2 VMs

-hostname: node1.domainX.example.com (172.24.10.10)

-hostname: node2.domainX.example.com (172.24.10.11)

* Total number of Questions will be arround 22

* In one system root password is already set (no need to reset) but in second system password need to be recovered.

* In one system Network configuration is required but in another one networking is already done

* NTP need to be configured in only one system (not in both)

* YUM Repo need to configured in both systems.

* Firewall and SELinux both will be pre-enabled.

------

node1

------

your node1 by default is running.

through console:

Q1. Configure network and set the static parameters. ->consider machine configured as DHCP, nned to config it with static parameters.

IP-ADDRESS=	172.24.10.10
NETMASK=	255.255.255.0
GATEWAY=	172.24.10.254
Nameserver=	172.24.10.254
DomainName=	domainX.example.com
hostname=	node1.domainX.example.com

```
# ip a s

# cat /etc/resolv.conf

# ip route
# nmcli connection show

# nmcli connection modify "Wired connection 1" ipv4.address "172.24.10.10/24" ipv4.dns
"172.24.10.254" ipv4.gateway "172.24.10.254" ipv4.method manual

# nmcli connection reload

# hostname

# hostnamectl set-hostname node1.domainX.example.com

# vim /etc/ssh/sshd_config

41 PermitRootLogin yes

66 PasswordAuthentication yes

:wq!

# systemctl restart sshd.service
```

Note: make ssh connection from exam-host to node1.domainX.example.com and continue

```
==================================================================================
======================
```

Q2. Configure YUM repos with the given link (2repos: 1st is BaseOS and 2nd is AppStream)

BaseOS	http://content.example.com/rhel8.0/x86_64/dvd/BaseOS
AppSterm	http://content.example.com/rhel8.0/x86_64/dvd/AppStream

```
# vim /etc/yum.repos.d/rhel.repo

[BaseOs]

name=BaseOs

baseurl=http://content.example.com/rhel8.0/x86_64/dvd/BaseOS/

gpgcheck=0

enabled=1

[AppStream]

name=AppStream

baseurl=http://content.example.com/rhel8.0/x86_64/dvd/AppStream/

gpgcheck=0

enabled=1

:wq

# yum list autofs
```

===
====================

Q3. Debug SELinux

- A web server running on non standard port 82 is having issues serving content, Debug and fix the issues.

- The web server on your system can server all the existing HTML files from /var/www/html

- Web service should automatically start at boot time.

- Do not make any changes to these files

```
# yum list httpd

# systemctl status httpd.service

# systemctl enable httpd.service

# systemctl status httpd.service

# ls /var/www/html/

file1 dile2 file3
```

```
# cat /etc/httpd/conf/httpd.conf | grep -i "listen"

Listen 82
```

```
# firewall-cmd --permanent --add-port=82/tcp

# firewall-cmd --reload

# firewall-cmd --list-all
```

```
# semanage port -l | grep "http"

http_port_t tcp 80, 81, 443, 488, 8008, 8009, 8443, 9000

# semange port -a -t httpd_port_t -p tcp 82

# semanage port -l | grep "http"

http_port_t tcp 82, 80, 81, 443, 488, 8008, 8009, 8443, 9000
```

verify

```
# curl localhost:82/file1
```

==
=========================

Q4. Create User accounts with supplementry group.

-create the group a named "sysadms".

-create users as named "natasha" and "harry", will be the supplementry group "sysadms".

-cerate a user as named "sarah", should have non-interactive shell and it should be not the member of "sysadms".

-password for all users should be "trootent"

```
# groupadd sysadmin

# useradd -G sysadmin natasha

# useradd -G sysadmin harry

# useradd -s /sbin/nologin sarah

# tail /etc/etc/group

# tail /etc/etc/passwd

# echo "trootent" | passwd --stdin harry

# echo "trootent" | passwd --stdin natasha

# echo "trootent" | passwd --stdin sarah
```

==
=========================

Q5. Configure a cron job that runs every 2minutes and executes:

logger "EX200 in progress" as the user natasha.

```
# systemctl restart crond.service

# crontab -eu natasha

*/2 * * * * logger "EX200 in progress"

:wq!

# systemctl restart crond.service
```

==
=========================

Q6. Create a collaborative Directory.

-Create the Directory "/home/manager" with the following characteristics

-Group ownership of "/home/manager" should go to "sysadms" group

-The directory should have full permission for all members of "sysadms" group but not to the other users except "root"

-Files created in future under "/home/manager" should get the same group ownership

```
# mkdir /home/sysadms

# chgrp sysadms /home/sysadms

# ls -ld /home/sysadms

# chmod 2770 /home/sysadms

# ls -ld /home/sysadms verify

# touch /home/sysadms/file1.txt

# ls -l /home/sysadms/file1.txt
```

==

Q7. Configure NTP

-Synchronize time of your system with the server 'utility.example.com'

```
# timedatectl

# systemctl restart chronyd.service

# vim /etc/chrony.conf

server utility.example.com iburst

:wq!

# systemctl restart chronyd.service

# timedatectl

# chronyc sources -v
```

==

Q8. Configure AutoFS

- All remoteuserX home directory is exported via NFS, which is available on utility.example.com(172.24.10.100) and your NFS-exports directory is /home/remoteuserX for remoteuserX

-remoteuserX's home directory is utility.example.com:/rhome/remoteuserX, where X is your station number

-remoteuserX's home directory should be automounted autofs service.

-home directories must be writable by their users.

```
# getent passwd remoteuserX
# yum install autofs.x86_64 -y
# systemctl enable autofs.service
# systemctl start autofs.service
# vim /etc/auto.master
/rhome/remoteuserX /etc/auto.misc
:wq!
# vim /etc/auto.misc
remoteuserX -fstype=rw,nfs,soft,sync utility.example.com:/rhome/remoteuserX
:wq!
# systemctl restart autofs.service
```

verify

```
# su - remoteuserX
# pwd
/rhome/remoteuserX
$ touch file1.txt
```

NOTE: this solution is correct but doesnt work for me.

==
=====================

Q9. Create a container image from the provided link.

a. create a container image from "http://utility.example.com/container/Containerfile" name it as 'process_files'

login to 'registry.lab.example.com' through "admin" and "redhat321"
->find it out credentials from additional info page

```
# id athena
# ssh athena@localhost
```

```
$ podman login -u admin

$ password: redhat321

$ wget http://utility.example.com/container/Containerfile

$ podman build -t monitor -f .

$ podman images

localhost/process_files

$ exit

#
```

==
======================

Q10. Create rootless container and do volume mapping which they asked you in the question and run container as a service from normal user account, the service must be enable so it could start automatically after reboot.

a. Create a container named as 'ascii2pdf' using the previously created container image from previous question 'monitor'

b. Map the '/opt/files' to container '/opt/incoming'

c. Map the '/opt/processed' to container '/opt/outgoing'

d. Create systemd service as container-ascii2pdf.service

e. Make service active after all server reboots.

```
# mkdir /opt/files

# chown -R athena:athena /opt/files

# mkdir /opt/processed

# chown -R athena:athena /opt/processed

# ssh athen@localhost

$ podman run  -d --name ascii2pdf -v /opt/files:/opt/incoming:Z  -v /opt/processed:/opt/outgoing:Z localhost/monitor

$ podman ps

$ mkdir /home/athena/.config/systemd/user/

$ cd /home/william/.config/systemd/user/

$ podman generate systemd --name ascii2pdf --files --new

$ ls -lrt

container-ascii2pdf.service
```

```
$ systemctl --user daemon-reload
$ systemctl --user enable container-ascii2pdf.service
$ systemctl --user start container-ascii2pdf.service
$ loginctl enable-linger athena
$ loginctl show-user  athena
```

verify

```
$ systemctl --user stop container-ascii2pdf.service
$ podman ps
$ systemctl --user start container-ascii2pdf.service
$ podman ps
```

==

Q11. Create user 'alex' with 3456 uid and set the password 'trootent'

```
# useradd -u 3456 alex
# echo "trootent" | passwd --stdin alex
# id alex
```

==

Q12. Locate all files owned by user "harry" and copy it under /root/harry-files

```
# mkdir /root/harry-file
# find / -user harry -exec cp -rvfp {} /root/harry-files \;
# ls -a /root/harry-files
```

==

Q13. Find a string 'ich' from "/usr/share/dict/words" and put it into /root/lines file.

```
# grep "ich" /usr/lib/mem/ex200/samplefile.txt >/root/lines
# cat /root/lines
```

===

Q14. create an archive '/root/backup.tar.bz2' of /usr/local directory and compress it with bzip2

```
# yum install bzip2.x86_64 -y
# tar cfvj /root/backup.tar.bz2 /usr/local
```

Q15. script.

Store the search result of all files in the /usr/share directory that is greater than 30k and less than 50k in the /mnt/freespace/search.txt file

Ans:

```
# vim test.sh
#!/bin/bash
find /usr/share/ -uid 0 -size +30k -size -50k >/mnt/freespace/search.txt
:wq!
# chmod +x test.sh
# bash test.sh
# cat /mnt/freespace/search.txt
```

------

node2

===

Q15. Reset root user password and make it 'trootent'

===

Q16. Configure YUM Repos

BaseOS	http://content.example.com/rhel8.0/x86_64/dvd/BaseOS
AppSterm	http://content.example.com/rhel8.0/x86_64/dvd/AppStream

```
# vim /etc/yum.repos.d/rhel.repo

[BaseOs]

name=BaseOs

baseurl=http://content.example.com/rhel8.0/x86_64/dvd/BaseOS/

gpgcheck=0

enabled=1

[AppStream]

name=AppStream

baseurl=http://content.example.com/rhel8.0/x86_64/dvd/AppStream/

gpgcheck=0

enabled=1

:wq

# yum list autofs
```

==

Q17. Resize a logical Volume

-Resize the logical volume "mylv" so that after reboot size should be in between 290MB to 330MB

```
# lvs

# df -hT

# lvextend -L 310M /dev/myvg/mylv

# resize2fs /dev/mapper/myvg-mylv

# lvs

# df -hT
```

==

Q18. Add a swap partition of 512MB and mount it permanently.

```
# gdisk /dev/vdb

Command (? for help): n

Partition number (2-128, default 2):

First sector (34-2047, default = 34) or {+-}size{KMGTP}: Press Enter
```

```
Last sector (34-2047, default = 2047) or {+-}size{KMGTP}: +512M
Hex code or GUID (L to show codes, Enter = 8300): Press Enter
Command (? for help): t
Partition number (1-2): 2
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300): 8200
Changed type of partition to 'Linux swap'
Command (? for help): w
# udevadm settle
# reboot
# lsblk
# mkswap /dev/vdb2
copy UUID=e5a95dd4-0417-4229-a499-92b29fe9f201
# vim /etc/fstab
UUID=e5a95dd4-0417-4229-a499-92b29fe9f201 swap swap defaults 0 0
:wq!
# mount -a
# swapon /dev/vdb2
# swapon -s
# swapon -d
# free -m
==============================================================================
=====================
```

Q19. Create a logical volume of name "newlv" from a volume group name "newvg" physical extents of 16M and logical volume should have size of 50extents.

```
# gdisk /dev/vdb
Command (? for help): n
Partition number (3-128, default 3): 3
First sector (34-2047, default = 34) or {+-}size{KMGTP}: Press Enter
Last sector (34-2047, default = 2047) or {+-}size{KMGTP}: +1024M
```

```
Hex code or GUID (L to show codes, Enter = 8300): Press Enter

Command (? for help): t

Partition number (1-3): 3

Current type is 8300 (Linux filesystem)

Hex code or GUID (L to show codes, Enter = 8300): 8e00

Changed type of partition to 'Linux lvm'

Command (? for help): w

# udevadm settle

# pvcreate /dev/vdb3

# vgcreate -s 16M wgroup /dev/vdb3

# vgdisplay

# lvcreate -n wshare -l 50 wgroup

# lvs

# mkfs -t ext3 /dev/mapper/wgroup-wshare

# blkid

# mkdir /mnt/wshare

# echo "UUID=1902-BFCE /mnt/wshare ext3 defaults 0 0" >>/etc/fstab

# mount -a

# df -hT
```

==
=======================

Q20. Configure System Tuning:

-Choose the recommended 'tuned' profile for your system and set it as the default.

```
# yum list tuned

# syatemctl restart tuned.service

# tuned-adm active

Current active profile: balanced

# tuned-adm recommend

virtual-guest

# tuned-adm profile virtual-guest

# tuned-adm active
```

```
Current active profile: virtual-guest
```