Setup an ip address for [servera] virtual machine
Password: redhat
IP: 172.25.250.10/24
GW: 172.25.250.254
DNS: 172.25.250.254
NB: All partition should be created on /dev/vdb

Root Password: TomBigBee

****************Network configuration***********************
-check your physical or virtual interface name and ip address

```
#nmcli device show

# nmcli connection add con-name lan1 ifname enp2s0 type ethernet ipv4.method manual
ipv4.addresses 172.25.250.10/24 ipv4.gateway 172.25.250.254 ipv4.dns 172.25.250.254 autoconnect
yes
# nmcli connection up lan1
# nmcli connection show
```

*************Selinux mode***************

```
#vim /etc/config

SELINUX=enforcing
```

*********SSH permission************

```
# PermitRootLogin yes
# systemctl restart sshd.service
```

01: Set the hostname on your virtual machine: nodea.lab.example.com

Answer:

```
# hostnamectl set-hostname nodea.lab.example.com
# hostname
```

02: Yum repository configuration on node1 machine:

- Packages are available at: url1= http://content.example.com/rhel9.0/x86_64/dvd/AppStream/
- Packages are available at: url2= http://content.example.com/rhel9.0/x86_64/dvd/BaseOS/

Answer:

```
# vim /etc/yum.repos.d/appstream.repo
[app]
name=Appstream
baseurl=http://content.example.com/rhel9.0/x86_64/dvd/AppStream/
gpgcheck=0

[Base]
name=BaseOS
baseurl=http://content.example.com/rhel9.0/x86_64/dvd/BaseOS/
gpgcheck=0
```

Test:

```
#yum clean all
```

```
#yum repolist all
```

03: Configure a cron job on Primary machine:

▪ a. The user natasha must configure a cron job that runs daily at 14:23 local time & executes /bin/echo "hi alex"

Answer:

```
# yum install cronie
# systemctl enable crond --now
# systemctl status crond

# crontab -eu natasha
23 14 * * * /bin/echo "hi alex"
```

verification:

```
# crontab -u -l natasha
```

▪ b. The user harry must configure a cron job that runs daily at every 3 minute local time & executes /bin/echo I got RHCE certificate.

Answer:

```
# crontab -e -u harry
# */3 * * * * /bin/echo "I got RHCE certificate."
```

verification:

```
# crontab -u -l harry
```

04: Debug Selinux:

Fixed the HTTP service, the page isn't provived nodea machine by this link=http://172.25.250.10:82
SELinux must be running in the Enforcing mode.

Answer:

```
# yum install httpd
# systemctl enable httpd
# systemctl restart httpd
# vim /etc/httpd/conf/httpd.conf
listen on 82
```

## This part is already done in the exam & document root is aslo set.

Frist you check the service is running or not

```
# systemctl status httpd
```

or you can restart the service.
then it's show [journalctl -xe]

```
# journalctl -xe                    [you can check the log.]


# semanage port -l|grep http        [Check the port is here or not.]
# man semanage port                 [for manual to see the example & simply copy the example &
                                    change the port no.]
# semanage port -a -t http_port_t -p tcp 82
# curl http://172.25.250.10:82
```

05: Create the following users, groups, and group memberships:

- A group named sysadmin
- A user natasha who belongs to sysadmin as a secondary group.

- A user sarah who also belongs to sysadmin as a secondary group.
- A user harry who does not have access to an interactive shell on the system & who is not a member of sysadmin.
-natasha, sarah & harry should all have the password of password.

Answer:

A group named sysadmin

```
# groupadd sysadmin
```

-A user natasha who belongs to sysadmin as a secondary group.

```
# useradd natasha
# usermod -G sysadmin natasha
```

-A user sarah who also belongs to sysadmin as a secondary group.

```
# useradd sarah
# usermod -G sysadmin sarah
```

-A user harry who does not have access to an interactive shell on the system & who is not a member of sysadmin.

```
#useradd harry
# usermod -s /sbin/nologin harry
```

-natasha, sarah & harry should all have the password of password.

```
# passwd sarah
# passwd harry
# passwd natasha
```

Or

```
# echo password |passwd --stdin natasha
# echo password |passwd --stdin natash
# echo password |passwd --stdin natasha
```

06: Create a collaborative directory "/common/admin" with the following characteristics:
- Group ownership of "/common/admin/" is sysadmin.
- The directory should be readable, writable & accessible to members of sysadmin, but not to any other users. (It is understood that root has access to all files & directories on the system.)
- Files created in "/common/admin/" automatically have group ownership set to the sysadmin.

Answer:

```
# mkdir /common/admin -p
```

-Group ownership of "/common/admin/" is sysadmin.

```
# chgrp sysadmin /common/admin
```

-The directory should be readable, writable & accessible to members of sysadmin, but not to any other users. (It is understood that root has access to all files & directories on the system.)
Files created in "/common/admin/" automatically have group ownership set to the sysadmin.

```
# chmod 2770 /common/admin
```

or

```
# chmod o-rwx /common/admin/
# chmod g+s /common/admin/
```

verification:

```
# getfacl /common/admin/
# ls -ld /common/admin
```

[X]   07: Copy the file "/etc/passwd" to "/var/tmp". Configure the permissions of "/var/tmp/passwd" so that:
- The file "/var/tmp/passwd" is owned by the root user.
- The file "/var/tmp/passwd" belong to the group root.
- The file "/var/tmp/passwd" should not be executable by anyone.
- The user harry is able to read and write "/var/tmp/passwd".
- The user sarah can neither write nor read "/var/tmp/passwd". [Note that: all other users (current or future) have the ability to read "/var/tmp/passwd".]

Answer:

```
#cp /etc/passwd /var/tmp
                                [The file "/var/tmp/passwd" is owned by the root user.]
                                [The file "/var/tmp/passwd" belong to the group root.]
# getfacl /var/tmp/passwd       [The file "/var/tmp/passwd" should not be executable by anyone.]
```

-The user harry is able to read and write "var/tmp/passwd". [ACL]

```
# setfacl -m u:harry:rw- /var/tmp/passwd
```

-The user sarah can neither write nor read "/var/tmp/passwd". [Note that: all other users (current or future) have the ability to read "/var/tmp/passwd".]

```
# setfacl -m u:sarah:--- /var/tmp/passwd
```

verification:

```
 #getfacl /var/tmp/passwd
```

08: Syncronise your system time with the classroom.example.com.

Answer:

```
#yum install chrony -y
# vim /etc/chrony.conf
server classroom.example.com iburst

# systemctl enable chronyd
# systemctl restart chronyd
```

verification:

```
# chronyc tracking
```

09: Configure AutoFS.
All remote users home directory is exported via NFS, which is available on workstation.lab.example.com or 172.25.250.9 and your NFS-exports directory is /home/guests/ for remote5.
- Remote home directory is workstation.lab.example.com:/home/guests/
- Remote home directory should be automount autofs service.
- Home directories must be writable by their users.

- when you are able to log in as remote5 user it's found home directory as /home/guests/remote5.
- Ensure that remote5 user can read, write on his home directory

Answer:

```
# yum install autofs -y
# systemctl enable autofs.service
# systemctl restart autofs.service

# Showmount -e 172.25.250.9

# vim /etc/auto.master
/home/guests   /etc/auto.misc

# vim /etc/auto.misc
remote5       172.25.250.9:/home/guests/remote5
```

2nd way for auto.misc file

```
# vim /etc/auto.misc
*         172.25.250.9:/home/guests/&
```

10: Create a backup.tar.(bz2 and gz) of /etc directory in /home location.

Answer:

```
# tar -cvjf /home/backup.tar.bz2 /etc
# file /home/backup.tar.bz2

# tar -cvzf /home/backup.tar.gz /etc
# file /home/backup.tar.gz
```

11: Deny cronjob for user susan so that other user for this system are not effected for this cronjob.

Answer:

```
# vim /etc/cron.deny
susan
```

12: Find all files owned by user brian and put them into /root/brian.

Answer:

```
# find / -user brain -exec cp -frvp {} /root/brain/ \;
```

13: Download a file word.dict from http://content.example.com & put it to "/root". Copy all the lines from /root/word.dict files that contains the word "mail" and put those lines in /root/sorted.dict

Answer:

```
# cd /root
# wget http://classroom.example.com/content/word.dict
# grep mail word.dict > /root/sorted.dict
```

or

```
# wget -O /root/word.dict http://classroom.example.com/content/word.dict
# grep mail word.dict > /root/sorted.dict
```

14. write a shell script /root/program1 which will search the file from 10MB to 20MB. and copy those files to /tmp/ex200 directory.

Answer:

```
# Vim /root/program1
!# /bin/bash
mkdir  /tmp/ex200
find  / -size +10M -size -20M -exec cp -frvp {} /tmp/ex200/ \;
```

15. Customize user environment so that when user “bob” create a directory its defaultspermissions set as: “user=rwx”,”group=rwx” ,”others=---” and createing a files set as “user=rw-”, “group=rw-” ,”others=---”

Answer:

```
#su  bob
#vim  .bashrc
umask =007
```