RHEL 9 Dump Questions:

- Node1.example.com

Q1: Configure TCP/IP and "Hostname" as following.

Hostname: ***node1.domain.example.com***

IP Address: ***192.168.71.240***

Netmask: ***255.255.255.0***

Gateway: ***192.168.71.2***

DNS: ***192.168.71.2***

Answer:

[root@node1.com]# *hostnamectl set-hostname node1.domain.example.com*

[root@node1.com]# *nmcli con show >> (ens160)*

[root@node1.com]# *nmcli con mod ens160 ipv4 192.168.71.240/24 ipv4.gateway 192.168.71.2 ipv4.dns 192.168.71.2*

[root@node1.com]# nmcli con up ens160

(USE "nmtui" instead most probably nmcli will not be available in the exam)

Verification:

[root@node1.com]# *ifconfig*

[root@node1.com]# *cat /etc/resolv.conf*

Q2: Configure your Red hat VM repository installed the packages distribution is available via YUM:

1. BaseOS url= http://content.example.com/rhel9/x86_64/dvd/BaseOS
2. AppStream url= http://content.example.com/rhel9/x86_64/dvd/AppStream

Answer:

[root@node1.com]# *vi /etc/yum.repos.d/yum.repo*

[Server-1]

name=baseos

baseurl=http://content.example.com/rhel9/x86_64/dvd/BaseOS

enabled=1

gpgcheck=0

[Server-2]

name=appstream

baseurl=http://content.example.com/rhel9/x86_64/dvd/AppStream

enabled=1

gpgcheck=0

:wq!

Verification:

[root@node1.com]# *yum clean all*

[root@node1.com]# *yum update*

[root@node1.com]# *yum repolist*

Q3: Debug SELinux:

Web server running on non-standard port “82” is having issues serving content, Debug and fix the issues.

- The web server can serve all the existing HTML files from ***‘/var/www/html’***, Don’t make any changes to these files.
- Web service should automatically start at boot time.

Answer:

Check if SELinux is enforced.

[root@node1.com]# *getenforce*

Enforcing

httpd should be installed by default but you can check using (#rpm -qa | grep httpd)

Check firewalld

[root@node1.com]# *firewall-cmd --list-all*
Ports: 82/tcp

If 82/tcp is not shown at port field, then you will need to add manually
[root@node1.com]# *firewall-cmd --permanent --add-service=http*
[root@node1.com]# *firewall-cmd --permanent --add-port=82/tcp*
[root@node1.com]# *firewall-cmd --reload*

Verify the configurations.
[root@node1.com]# *firewall-cmd --list-all*

Check SELinux:
[root@node1.com]# *semanage port -l | grep “http”*
http_port_t tcp 80,81,443,488,8008,8009,8443,9000

[root@node1.com]# *semanage port -a -t http_port_t -p tcp 82*

[root@node1.com]# *systemctl start httpd.service*
[root@node1.com]# *systemctl enable httpd.service*

Verify the configurations.

[root@node1.com]# *semanage port -l | grep "http"*

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

Verification:

- Open on Firefox browser, on the address-bar and type:
  hostname:82 >> node1.domain.example.com:82

  On this page you should see: This is my web server.

- [root@node1.com]# *curl node1.domain.example.com:82*
- *User the browser: http://<ipaddress>:82*

Q4: Create the following users, groups, and group membership:

- A group named ***adminuser***
- A user "***harry***" who belongs to ***adminuser*** as a secondary group.
- A user "***natasha***" who belongs to ***adminuser*** as a secondary group.
- A user "***sarah***" who doesn't have access to an interactive shell and who is not a member of ***adminuser*** group.
- All users should have the password of "***centtered***".

Answer:

[root@node1.com]# *groupadd adminuser*

[root@node1.com]# *useradd -G adminuser harry*

[root@node1.com]# *useradd -G adminuser natasha*

[root@node1.com]# *useradd -s /sbin/nologin sarah*

[root@node1.com]# *passwd harry*

[root@node1.com]# *passwd natasha*

[root@node1.com]# *passwd sarah*

Verification:

[root@node1.com]# *cat /etc/passwd*

harry:x:1001:1002::/home/harry:/bin/bash

natasha:x:1002:1003::/home/natasha:/bin/bash

sarah:x:1003:1004::/home/sarah:/sbin/nologin

Q5: Cron job

- *Configure a cron job for the user "**natasha**" that runs daily every **14:23** minute local time executes "**Ex200 is processing**" with logger.*
- Configure a cron job for the user "**natasha**" that runs daily every **1-minute** local time executes "**Ex200 is processing**" with logger.

Answer:

[root@node1.com]# *crontab -e -u natasha*

*23 14 * * * echo "Ex200 is processing."*

**/1 * * * * echo "Ex200 is processing."*

[root@node1.com]# *systemctl restart crond.service*

Verification

[root@node1.com]# *tail -f /var/log/cron*

Q6: Create a collaborative directory.

- Create a directory ***/home/admin*** with the following characteristics.
- Group ownership of ***/home/admin*** is ***adminuser***.
- The directory should be readable, writable, and accessible to member of ***adminuser***, but not any other user.
- Files created in ***/home/admin*** automatically have group ownership set to the ***adminuser*** group.

Answer:

[root@node1.com]# *mkdir /home/admin*

[root@node1.com]# *chgrp adminuser /home/admin*

[root@node1.com]# *chmod 2770 /home/admin*

Verification

[root@node1.com]# *cd /home/admin*

[root@node1.com]# *touch file1*

[root@node1.com]# *mkdir dir1*

Q7: Create user '***alex'*** with ***3456 uid*** and set password to "***centtered***".

Answer:

[root@node1.com]# *useradd -u 3456 alex*

[root@node1.com]# *passwd alex*

Q8: Locate all the files owned by user "***natasha***" and copy them under ***/root/locatedfiles.***

Answer:

[root@node1.com]# *mkdir /root/locatedfiles*

[root@node1.com]# *find / -user natasha -type f -exec cp -rvp {} /root/locatedfiles \;*

Q9: Find a string '***strato'*** from ***/usr/share/dict/words*** and put it into ***/root/lines*** file.

Answer:

[root@node1.com]# *cat /usr/share/dict/words | grep strato > /root/lines*

Q10: Configure AutoFS

Configure **autofs** to automount the home directories of remote users. NFS export **/home** on your system. In system have preconfigured for **remoteuser20**

Note: the following requirements automount remoteuser20 home directory:

- remoteuser20 is exported on ***classroom.example.com(192.168.71.254):/home/remoteuser20***
- remoteuser20 home directory should be automounted locally beneath ***/home*** as ***/home/remoteuser20***.
- home directories must be writable by their users.

Answer:

[root@node1.com]# *yum install autofs* nfs* -y (should be installed by default)*

[root@node1.com]# *systemctl start autofs*

[root@node1.com]# *systemctl enable autofs*

[root@node1.com]# *showmount -e 192.168.71.254*

Output: /home/remoteuser20 (could be different but use the output in the next steps as it comes)

[root@node1.com]# *vi /etc/auto.mster.d/remoteuser.autofs*

/home/remoteuser20 /etc/auto.remoteuser

[root@node1.com]# *vi /etc/auto.remoteuser*

**** -rw,sync,fstype=nfs4 192.168.71.254:/home/remoteuser20/&***

[root@node1.com]# *systemctl restart autofs*

Verification:

[root@node1.com]# *su – remoteuser20*

[root@node1.com]# *cd*

[root@node1.com]# *pwd*

It should be /home/remoteuser20

Q11: Create an archive.

a. Create an archive ***'/root/backup.tar.bz2'*** of ***/usr/local*** directory and compress it with bzip2.

Answer:
[root@node1.com]# *tar -cvf /root/backup.tar /usr/local*
[root@node1.com]# *bzip2 /root/backup.tar*

Verification:

[root@node1.com]# *ls -l*

b. Create an archive ***'/root/myetcbackup.tgz'*** of ***/etc*** directory.
Answer:
[root@node1.com]# *tar -czvf /root/myetcbackup.tgz /etc*

Verification:
[root@node1.com]# *ls -l*

Q12: Create a container for alth user

- Use this link: http://domain.exam.com/rhel9/Containerfile build image named monitor.
- Don't change anything in Container file.

Answer:

[root@node1.com]# *ssh alth@localhost*
[alth@node1.com]# *wget http://domain.exam.com/rhel9/Containerfile*
[alth@node1.com]# *podman build -t monitor -f Containerfile*

Q13: Create a rootless container.

- Create a container name asciipdf
- Use monitor image for asciipdf which you previously created.
- Create a system service named container-asciipdf for alth user only.
- Servie will automatically started across reboot.
- Local directory /opt/files attached to container directory /opt/incoming.
- Local directory /opt/processed attached to container directory /opt/outgoing.
- If the service work properly, when you place any plain text file in /opt/file, then this file automatically converted into pdf and also placed under /opt/processed.

Answer:

[root@node1~]# *mkdir /opt/files /opt/processed*
[root@node1~]# *chown alth:alth /opt/files /opt/processed*
[root@node1~]# *ssh alth@localhost*
[alth@node1~]# *loginctl enable-linger*
[alth @node1~]# *podman run -d --name ascii2pdf -v /opt/files:/opt/incoming:z -v /opt/processed:/opt/outgoing:z monitor*
[alth @node1~]# *podman ps (verification step)*
[alth @node1~]# *mkdir -p .config/systemd/user*
[alth @node1~]# *cd .config/systemd/user*
[alth @node1]# *podman generate systemd --name ascii2pdf --files --new*
[alth @node1]# *systemctl --user daemon-reload*
[alth @node1]# *systemctl --user start container-ascii2pdf.services*
[alth @node1]# *systemctl --user enable container-ascii2pdf.services*
[alth @node1]# *exit*

[root@node1~]# *touch /opt/files/testfile (it should appear as pdf in /opt/processed)*

Q14: Make a simple script. ***<To be Checked>***

- Create **myscript** file to locate all files under ***/usr/*** of less than 10MB with permissions user identifier (SGID).
- Save all these files in list ***/root/script***.
- Copy script file in ***/usr/local/bin***.
- Make sure that the script run at any location.

Answer:

[root@node1.com]# *vi /usr/local/bin/myscript.sh*
#!/bin/bash
find /usr -size -10M -type f -perm -2000 -exec ls -ltr {} \; > /root/script

[root@node1.com]# *chmod 2775 myscript.sh*

Verification:

[root@node1.com]# *cd /tmp*

[root@node1.com]# *myscript.sh*

Q15: Synchronize the time of your system from "3.asia.ntp.org"

Answer:

[root@node1.com]# *rpm -qa | grep chrony*
[root@node1.com]# *systemctl status chronyd.service*
[root@node1.com]# *systemctl start chronyd.service*

[root@node1.com]# *systemctl enable chronyd.service*
[root@node1.com]# *vi /etc/chrony.conf*
server 3.asia.ntp.org iburst

:wq!

[root@node1.com]# *systemctl restart chronyd.service*

Verification:
[root@node1.com]# *chronyc sources -v*

Q16: Set the permissions:

a. All the new creating files for user "***natasha***" as ***-r-- --- ---*** as default permission.
b. All the new creating directories for user "***natasha***" as ***dr-x --- ---*** as default permission.
c. **Set the password expire date:** The password for all new users in 1st server should expires after 20 days.
d. **Assign sudo Privilege:** Assign the sudo Privilege for Group "***adminuser***" and Group members can administrate without any password.

Answer:

- **Q: A and B:**
  [root@node1.com]# *echo "umask 277" >> /home/natsha/.bashrc*

  Verification:
  [root@node1.com]# *su - natsha*
  [root@node1.com]# *umask*
  277
  [root@node1.com]# *cd /home/natasha*
  [root@node1.com]# *mkdir dir1*
  [root@node1.com]# *touch file1*
  [root@node1.com]# *ls -l*

- **Q: C:**
  [root@node1.com]# *vi /etc/login.defs*

PASS_MAX_DAYS 20

:wq!

- **Q: D:**
  [root@node1.com]# *visudo*
  Under this line: # %wheel ALL=(ALL) NOPASSWD: ALL
  Put the following line: %adminuser ALL=(ALL) NOPASSWD: ALL

- Node2.example.com

Q17: Reset root password and make it "***centtered***". ***<You should take care to enter the GRUB mode of the Rescue part and not from the default part>***

Answer:

1. Reboot the system.
2. Interrupt the boot-loader countdown by pressing any key, except Enter.
3. Move the cursor to the rescue kernel entry to boot (the one with the word rescue in its name).
4. Press e to edit the selected entry.
5. Move the cursor to the kernel command line (the line that starts with Linux).
6. Move with the right arrow from there until you find "ro" and replace it with "rw init=/sysroot/bin/sh"
7. Hit ctrl + x
8. Hit enter when it asks for a password without typing any passwords (exam environment only)
9. Chroot /sysroot
10. Passwd root
11. Touch /.autorelabel
12. Exit
13. Reboot

Q18: Configure your Red hat VM repository installed the packages distribution is available via YUM:

1. BaseOS url= http://domain10.example.com/rhel9/x86_64/dvd/BaseOS
2. AppStream url= http://domain10.example.com/rhel9/x86_64/dvd/AppStream

Answer:

[root@node2.com]# *cd /etc/yum.repos.d*

[root@node2.com]# *vi yum.repo*

[Server-1]

name=baseos

baseurl=http://domain10.example.com/rhel9/x86_64/dvd/BaseOS

enabled=1

gpgcheck=0

[Server-2]

name=appstream

baseurl=http://domain10.example.com/rhel9/x86_64/dvd/AppStream

enabled=1

gpgcheck=0

:wq!

Verification:

[root@node2.com]# *yum repolist*

[root@node2.com]# *yum clean all*

[root@node2.com]# *yum update*

Q19: Resize the logical volume name ***lv*** to ***300 MB***. Make sure in ***lv*** volume have some data, data should not be affected by resizing. Do not remove or modify /etc/fstab

Answer:

[root@node2.com]# *lvs*

LV VG Attr LSize Pool Origin Data% Meta% Move Log Cpy%Sync Convert

lv vg -wi-ao---- 76.00m

[root@node2.com]# *vgs*

VG #PV #LV #SN Attr VSize VFree

vg 1 1 0 wz--n- 1020.00m 944.00m

[root@node2.com]# *lvdisplay*

--- Logical volume ---

LV Path /dev/vg/lv

[root@node2.com]# *lvextend -L +224M /dev/vg/lv (just increase the difference and it might be adjusted a little bit automatically depending on the PE size but it is acceptable)*

[root@node2.com]# *blkid*

/dev/mapper/vg-lv: UUID="aee30843-e09c-4035-8214-a1be08be4b87" TYPE="ext4"

[root@node2.com]# *resize2fs /dev/vg/lv*

Verification:

[root@node2.com]# *df -h*

[root@node2.com]# *lvs*

[root@node2.com]# *lvdisplay*

Q20: Add a ***swap*** partition of ***715MB*** and mount it permanently.

Answer:

[root@node2.com]# *fdisk -l*

[root@node2.com]# *fdisk /dev/vdb*

p

m

n

enter

enter

enter

+715M

t

2

l

82

p

w

[root@node2.com]# *free -m*

[root@node2.com]# *swapon -s*

[root@node2.com]# *mkswap /dev/vdb2*

[root@node2.com]# *vi /etc/fstab*

/dev/vdb2 swap swap defaults 0 0

:wq!

[root@node2.com]# *mount -a*

[root@node2.com]# *swapon -a*

Verification:

[root@node2.com]# *free -m*

[root@node2.com]# *swapon -s*

[root@node2.com]# *init 6*

[root@node2.com]# *free -m*

[root@node2.com]# *swapon -s*

Q21: Create a Logical Volume

- Create a logical volume "**database**" by using **50** PEs from the volume group **"datastore"**.
- Consider that each ***PE size*** of volume group as "***16MB***".
- Format with **ext3** filesystem and mount permanently under ***/mnt/database***

Answer:

Volume group size should be greater than 50 * 16 (800 MBs)

[root@node2.com]# *fdisk -l*

[root@node2.com]# *fdisk /dev/vdb*

m

n

enter

enter

enter

+1024M

p

t

3

l

8E

w

[root@node2.com]# *pvcreate /dev/sda3*

[root@node2.com]# *vgcreate -s 16M datastore /dev/sda3*

[root@node2.com]# *lvcreate -l 50 -n database datastore*

[root@node2.com]# *mkdir /mnt/database2*

[root@node2.com]# *mkfs.ext3 /dev/datastore/database*

[root@node2.com]# *vi /etc/fstab*

/dev/datastore/database /mnt/database ext3 defaults 0 0

:wq

[root@node2.com]# *mount -a*

Verification:

[root@node2.com]# *df -h*

[root@node2.com]# *init 6*

[root@node2.com]# *df -h*

Q22: Configure recommended tuned profile.

Answer:

[root@node2.com]# *rpm -qa | grep tuned*

[root@node2.com]# *systemctl status tuned.service*

[root@node2.com]# *systemctl start tuned.service*

[root@node2.com]# *systemctl enable tuned.service*

[root@node2.com]# *tuned-adm list*

[root@node2.com]# *tuned-adm active*

[root@node2.com]# *tuned-adm recommend*

[root@node2.com]# *tuned-adm profile virtual-host*

Verification:

[root@node2.com]# *tuned-adm active*