Do this in Server-a:

#Q1. Configure network and set the static hostname.  
IP ADDRESS = 172.25.250.10 NETMASK = 255.255.255.0 GATEWAY = 172.25.250.254 DNS = 172.25.250.254 Domain name = lab.example.com hostname = servera.lab.example.com

- vi /etc/sysconfig/network-scripts/ifcfg-eth0

DEVICE="eth0"  
BOOTPROTO="static"  
DNS="172.25.250.254"  
GATEWAY="172.25.250.254"  
HOSTNAME="servera.lab.example.com"  
HWADDR="00:19:99:A4:46:AB"  
IPADDR="172.25.250.10"  
NETMASK="255.255.255.0"  
NM\_CONTROLLED="yes"  
ONBOOT="yes"  
TYPE="Ethernet"  
UUID="8105c095-799b-4f5a-a445-c6d7c3681f07"

> modifie l’adresse ip donnée et enregistre et quitte ( :wq)

hostnamectl set-hostname servera.lab.example.com  
# nmcli con up System \eth0  
# systemctl restart NetworkManager

#Q2. Configure YUM repos with the given link ( 2 repos: 1st is Base and 2nd is AppStream )  
● Base\_url= http://content.example.com/rhel8.0/x86\_64/dvd/BaseOS   
● AppSterm\_url= http://content.example.com/rhel8.0/x86\_64/dvd/AppStream

- vim /etc/yum.repos.d/local.repo

[BaseOS]  
name=yum repository local  
baseurl=http://content.example.com/rhel8.0/x86\_64/dvd/BaseOS  
gpgcheck=0  
enabled=1

[AppStream]  
name=yum repository local  
baseurl=http://content.example.com/rhel8.0/x86\_64/dvd/AppStream  
gpgcheck=0  
enabled=1

#yum repolist   
#yum update

#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 ( NOTE: Do not make any changes to these files )   
● Web service should automatically start at boot time.

# semanage fcontext -a -t httpd\_sys\_content\_t "/var/www/html(/.\*)?"  
# restorecon -Rv /var/www/html  
# semanage port -l | grep http  
# semanage port -a -t http\_port\_t -p tcp 82  
#firewall-cmd --permanent --add-port=82/tcp  
#firewall-cmd --reload

#Q4. Create User accounts with supplementary group.   
● Create the group a named "sysadms".   
● Create users as named "natasha" and "harry", will be the supplementary group "sysadms".   
● Create 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 sysadms  
#groups sysadms  
#useradd -G sysadms harry  
#useradd -G sysadms natasha  
#id harry  
#id nathasa  
#useradd -s /sbin/nologin sarah  
#id sarah  
#passwd sarah  
#passwd harry  
#passwd natasha  
#cat /etc/passwd

#Q5. Configure a task: plan to run echo "file" command at 14:23 every day.

# su - natasha $ crontab -e   
23 14 \* \* \* /bin/echo "file"  
# su - natasha $ crontab -l

#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/manager  
#chown :sysadms /home/manager  
#chmod 2770 /home/manager  
#ls -ld /home/manager

#Q7. Configure NTP :  
● Synchronize time of your system with the server classroom.example.com.

#yum install -y chrony  
#vi /etc/chrony.conf : server classroom.example.com iburst  
# timedatectl set-ntp true  
# systemctl restart chronyd  
#chronyc sources -v

#Q8. Configure AutoFS  
● All Ldapuser2 home directory is exported via NFS, which is available on classroom.example.com (172.25.254.254) and your NFS-exports directory is /home/guests for Ldapuser2,   
● Ldapuser2's home directory is classroom.example.com:/home/guests/ldapuse2   
● Ldapuser2's home directory should be automount autofs service.   
● Home directories must be writable by their users.   
● while you are able to log in as any of the user ldapuser1 through ldapuser20, the only home directory that is accessible from your system is ldapsuser2

# yum install -y autofs  
# vi /etc/auto.master.d

(/home/guests /etc/auto.home)  
# vi /etc/auto.home (\* -rw,sync,fstype=nfs4 classroom.example.com:/home/guests/&)  
# systemctl enable autofs.service

# systemctl start autofs.service

#ssh ldapuser5@localhost

#cd

#pwd # it should be /home/guests/ldapuser2

#Q9. ACL.   
● Copy the file /etc/fstab to /var/tmp/ and configure the "ACL" as mentioned following.   
● The file /var/tmp/fstab should be owned by the "root".   
● The file /var/tmp/fstab should belong to the group "root".   
● The file /var/tmp/fstab should not be executable by any one.   
● The user "sarah" should be able to read and write to the file.   
● The user "harry" can neither read nor write to the file.   
● Other users (future and current) should be able to read /var/tmp/fstab

#cp -rv /etc/fstab /var/tmp/

#cd /var/tmp/

#ls -al /var/tmp/fstab  
#setfacl -m u:sarah:rw- /var/tmp/fstab

#setfacl -m u:harry:--- /var/tmp/fstab

#setfacl -m o:r-- /var/tmp/fstab

# in order to check if everything is ok

getfacl /var/tmp/fstab

#Q10. Create user 'bob' with 2112 uid and set the password 'trootent'

#useradd -u 2112 bob  
#passwd bob (trootent)  
#id bob

#Q11. Locate all files owned by user "harry" and copy it under /root/harry-files

#find / -user harry -exec cp -rvpf {} /root/harry-files \; 2>/dev/null

#Q12. Find a string 'ich' from "/usr/share/dict/words" and put it into /root/lines file

#grep “ich” /usr/share/dict/words > /root/lines  
#cat /root/line\_file

#Q13. create an archivie '/root/backup.tar.bz2' of /usr/local directory and compress it with

gzip.

# tar -cvzf /root/backup.tar.bz2 /usr/local

Server-2:

#Q14. Reset root user password and make it 'trootent'

* press e for starting system
* put in last last of linux16 : rd.break
* press ctrl + x

# mount -o remount,rw /sysroot  
#chroot /sysroot  
#passwd root  
#touch /.autorelabel  
#exit   
#exit

#Q15. Configure YUM Repos   
● Base\_url= "http://content.example.com/rhel8.0/x86\_64/dvd/BaseOS"   
● AppStrem\_url= "http://content.example.com/rhel8.0/x86\_64/dvd/AppStream"

# scp -r /etc/yum.repos.d/local.repo [root@172.25.250.10:/etc/yum.repos.d/](mailto:root@172.25.250.10:/etc/yum.repos.d/)  
# cat /etc/yum.repos.d/local.repo  
# yum repolist enabled  
# yum update  
#yum install -y vdo

#Q16. Resize a logical Volume : - Resize the logical volume "mylv" so that after reboot the size should be in between 200MB to 300MB

#df -h  
#vgdisplay  
#lvextend -L 300M /dev/mapper/myvg-mylv  
#lvdisplay /dev/mapper/myvg-mylv  
#resize2fs /dev/mapper/myvg-mylv

#Q17. Add a swap partition of 956MB and mount it permanently

# fdisk /dev/vdb   
n (create new partition:)  
p (check Partition table)  
Enter   
+965M  
 t   
82   
w  
mkswap /dev/vdb2  
Copy UUID   
vim /etc/fstab UUID=XXXXX swap swap defaults 0 0  
 systemctl daemon-reload  
swapon -a  
(swapon -s)

#Q18. Create a logical Volume and mount it permanently  
● Create the logical volume with the name "wshare" by using 20PE's from the volume group "wgroup". ● Consider each PE size of the volume group as "32 MB".   
● Mount it on /mnt/wshare with file system ext3.

# fdisk /dev/vdb   
n (create new partition:)  
p (check Partition table)  
3  
Enter   
+640M  
w  
partprobe  
pvcreate /dev/vdb3  
vgcreate -s 32M wgroup /dev/vdb3  
lvcreate -n wshare -l 20 wgroup  
mkfs.ext3 /dev/wgroup/wshare  
mkdir /mnt/wshare  
vi /etc/fstab (/dev/wgroup/wshare /mnt/wshare ext3 defaults 0 0)  
mount -a

#Q19. Create a new VDO partition using to following requirements:   
● Use the unpartitioned disk   
● Vdo name "Vdo1" and logical size should be 50GB   
● Mount it on /vdomount permanently with file system xfs.

#yum -y install vdo kmod-kvdo

#systemctl enable vdo.service

#systemctl start vdo.service

#lsblk  
#vdo create –name=Vdo1 --device=/dev/vdd --vdoLogicalSize=50G  
#mkfs.xfs -K /dev/mapper/Vdo1  
#lsblk ---output=UUID /dev/mapper/Vdo1  
#mkdir /vdomount  
#vi /etc/fstab (UUID=………………….. /vdomount xfs defaults, x-systemd.requires=vdo.service 0 0)  
#systemctl daemon-reload

#Q20. Configure System Tuning:   
● Choose the recommended 'tuned' profile for your system and set it as the default.

#tuned-adm active  
#tuned-adm recommend (virtual-guest)  
#tuned-adm profile virtual-guest

#Q21.   
● Create a container logserver from an image rsyslog in node1 From registry.lab.example.com   
● Configure the container with systemd services by an existing user “Walhalla”,   
● Service name should be container-logserver, and configure it to start automatically across reboot.

# useradd user1   
# passwd user1   
# yum module install container\* -y

# ll /var/log/

# vim /etc/systemd/journald.conf [Journal] Storage=persistent

:wq!

#mkdir /var/log/journal

#mkdir /home/wallah/container-logserver

#systemctl restart systemd-journald

#cp /var/log/journal/\*/\* /home/wallah/container-logserver

#chown -R wallah:wallah /home/wallah/container-logserver  
# systemctl restart systemd-journald   
# ll /run/log   
# ll /var/log/   
# su - wallah  
# mkdir /var/log/journal  
mkdir /home/wallah/container-logserver

#systemctl restart systemd-journald

# reboot  
# ssh user1@servera.lab.example.com

#22   
● Configure your host journal to store all journal across reboot   
● Copy all \*.journal from /var/log/journal and all subdirectories to /home/Walhalla/container\_logserver ● Configure automount /var/log/journal from logserver (container) to /home/walhalla/container\_logserver when container starts

# podman login regisrty.redhat.io   
# username:   
# password:   
# podman search rsyslog   
# podman pull registry.redhat.io/rhel8/rsyslog   
# podman image list   
# podman run -d --name logserver -v /home/user1/container-logserver:/var/log/journal:Z registry.redhat.io/rhel8/rsyslog   
# podman container list   
# podman ps   
# mkdir -p ~/.config/systemd/wallah  
# cd .config/systemd/wallah/   
# loginctl enable-linger   
# loginctl show-user user1   
# podman generate systemd --name logserver -f -n   
# systemctl --user daemon-reload  
# systemctl --user enable --now container-logserver.service   
# systemctl --user start --now container-logserver.service   
# systemctl --user status --now container-logserver.service   
# podman exec -it logserver /bin/bash # ls /var/log/ # exit