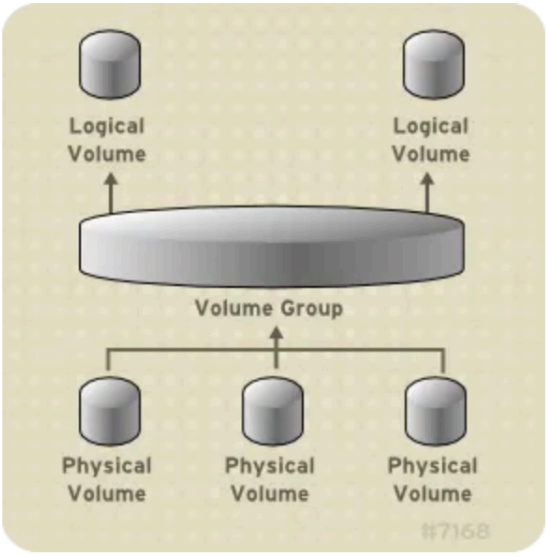



```
subnet 192.168.80.0  netmask 255.255.255.0 {
  range 192.168.80.15 192.168.80.20;
  option routers rtr-239-0-1.example.org, rtr-239-0-2.example.org;
}
```

```
OBJECTIVE: SCORE
Manage basic networking: 100%
Understand and use essential tools: 10%
Operate running systems: 0%
Configure local storage: 25%
Create and configure file systems: 25%
Deploy, configure and maintain systems: 29%
Manage users and groups: 0%
Manage security: 0%
Manage containers: 0%
Create simple shell scripts: 0%
```

**RHCSA**

RHEL9 virtual instance:
root
redhat

groupadd -o -g 5000 dba
groupadd -o -g linuxadm
useradd user1000
usermod -aG dba user1000
cat /etc/group
cat /etc/passwd
groupmod -n sysadm linuxadm
groupmod -g 6000 sysadm
groupdel sysadm
cat /etc/group
visudo
groupadd -g 6000 lnxgroup
useradd -u 5000 -g 6000 user5000
passwd user5000
chage -m 4 -M 30 user5000

groupmod -g 4000 lnxgroup
`sudo groupmod -g 7000 lnxgrp`

When two groups have an identical GID, members of both groups get identical rights on each other's files.

Job scheduling and execution is taken care of by two service daemons: atd and crond.

While atd manages the jobs scheduled to run one time in the future, crond is responsible for running jobs repetitively at pre-specified times.

/var/spool/cron and /etc/cron.d directories

/etc directory for either service. These files are named at.allow and at.deny for the at service, and cron.allow and cron.deny for the cron service.

| Variable | Description |
|---|---|
| DISPLAY | Stores the hostname or IP address for graphical terminal sessions |
| HISTFILE | Defines the file for storing the history of executed commands |
| HISTSIZE | Defines the maximum size for the HISTFILE |
| HOME | Sets the home directory path |
| LOGNAME | Retains the login name |
| MAIL | Contains the path to the user mail directory |
| PATH | Defines a colon-separated list of directories to be searched when executing a command. A correct setting of this variable eliminates the need to specify the absolute path of a command to run it. |
| PPID | Holds the identifier number for the parent program |
| PS1 | Defines the primary command prompt |
| PS2 | Defines the secondary command prompt |
| PWD | Stores the current directory location |
| SHELL | Holds the absolute path to the primary shell file |
| TERM | Holds the terminal type value |
| UID | Holds the logged-in user's UID |
| USER | Retains the name of the logged-in user |

export and unset command. env/printenv will print env variables. command subs: \u@\h

1. Where does GRUB2 read its configuration from on a BIOS system?/boot/grub2/grub.cfg
2. How can the redhat-support-tool be used to search and display the same Knowledgebase content as on the Red Hat Customer Portal? Using the search command followed by keywords or error codes
3.How does file system metadata alignment impact the performance of striped arrays (RAID 0, RAID 4, RAID 5, RAID 6)? If the write request is wider than the strip size, I/O requests could

require two writes per disk instead of one or having all metadata ends up on one disk, causing that disk to become a hot spot.
4. How does the spare area of SSDs impact their performance on random writes?
improvement
5. How does data striping in RAID increase throughput?
By dividing data into stripes and distributing them among several disks in the RAID array
6. What are the steps in a typical change management procedure for performance tuning changes? Set a baseline by running the test workload and gathering metrics. 2. Perform changes one at a time, measuring the effect after each change. 3. Verify the effectiveness of the change by rerunning the test workload. 4. Reverse the change and compare with the baseline.
5. Apply and document the definitive change.
7. What are the main ideas behind the USE Method in performance tuning?
Checking Utilization, Saturation, and Errors for user interactions
What is a drop-in file in systemd and how is it used to configure unit settings?

file in systemd that overrides or adds specific options for a unit, created by making a directory under /etc/systemd/system/ named after the unit with .d appended, and then creating .conf files in this directory. For example, enabling memory accounting for sshd.service can be done by creating a 20-accounting.conf file in the directory /etc/systemd/system/sshd.service.d/.

While the sticky bit is most commonly used with directories, it can also be set on files. On files, the sticky bit has an outdated and limited use. It was historically used to keep a file in memory after execution.

How can you enable CPU, memory, and block I/O accounting for a service or a slice in systemd?

Create a drop-in file under /etc/systemd/system/ with the desired unit or slice name and .d appended and include the CPUAccounting, MemoryAccounting, and BlockIOAccounting settings.

What are the advantages of using custom slices in systemd?

System resource granularity and distribution equality

chmod +t (add sticky bit) drwxrwxrwxt (t) at the end refers to sticky
chmod g-s /usr/bin/write -v (gid)

ncdu
sudo find / -type f -exec du -h {} + 2>/dev/null | sort -rh | head -20
sudo du -ah / 2>/dev/null | sort -rh | head -20

find /var/log -min -100 -exec file {} \;
find /usr -maxdepth -type d -name src
find /tmp -perm -u=r
find /tmp -type f -exec ls -ld {} \;
find /tmp -name *.txt -ok cp {} \;

locate .sh -n2
locate -S
setfacl -m u:user1:r a.txt
setfacl -dm u:user100:7,user200:rwx /tmp/prj

find /dev -type c -perm 660
useradd user1000
usermod -aG sgroup user1000

ps -efl
pidof rsyslogd
pgrep rsyslogd
ps -U user1
ps -G root
nice -n -10
renice 5 5572
can also renice from top command(type r and give pID)
cat /var/log/cron

*/1 * 1-10 3 * : Any day, in March, from 1 to 10th, every hour, every one minute.

| Column | Description |
|---|---|
| UID | User ID or name of the process owner |
| PID | Process ID of the process |
| PPID | Process ID of the parent process |
| C | CPU utilization for the process |
| STIME | Process start date or time |
| TTY | The controlling terminal the process was started on. “Console” represents the system console and “?” represents a daemon process. |
| TIME | Aggregated execution time for a process |
| CMD | The command or program name |

useradd user100
useradd user200
usermod -aG sgroup user100
usermod -aG sgroup user200
mkdir /sdir
chmod g+s /sdir
chmod o-t /tmp

tree -hapf
uname -snovpr
wc -l , -w, -c
rpm -qa (list all packages)
rpm -q perl (list perl packages)
**rpm -qf /etc/passwd** (see which package owns the file)
**rpm -qf /etc/group**

**dnf install policycoreutils**
**dnf info policycoreutils**
**dnf deflist policycoreutils**
rpm -qi setup
rpm sushi -ve (remove package)
rpm -qf /etc/chrony.conf ( see where the package is coming from)
cd /tmp
rmp2cpio /mnt/baseos/package/chrony-3.3.e18.x86_64.rpm | cpio -imd
rpm2cpio
find . -name chrony.conf 9
**rpm -K /tmp/chrony.. –no-signature** ( Use the MD5 checksum for verifying its integrity and the GNU Privacy Guard (GnuPG or GPG) public key signature for ensuring the credibility of its developer or publisher.
cp -r /tmp/chrony.. /etc/chrony
rpmkeys –import /etc/pki/rpm-gpg/rpm/gpg-key-redhat-release
rpmkeys -K /mnt/BaseOS/packages/zsh-3.3 (answer should be digests signatures ok)
rpm -q gpg-pubkey
rpm -qi <key> (view specific details)
rpm -Vf /etc/sysconfig
rpm -qi zlib
rpm -qa | sort

environment groups and package groups:

The environment groups available in RHEL 8 are server, server with GUI, minimal install, workstation, virtualization host, and custom operating system. These are listed on the software selection window during RHEL 8 installation. The package groups include container management, smart card support, security tools, system tools, network servers, etc.

main configuration file for dnf is /etc/dnf/dnf.conf preferred configuration location /etc/yum.repos.d . dnf runs rpm in the background.

RHEL 8 is shipped with two core repositories called BaseOS and Application Stream (AppStream).

```
[BaseOS_RHEL_8.0]
name= RHEL 8.0 base operating system components
baseurl=file:///mnt/BaseOS
enabled=1
gpgcheck=0
```

**EXAM TIP:** Knowing how to configure a dnf/yum repository using a URL plays an important role in completing some of the RHCSA exam tasks successfully. Use two forward slash characters (//) with the baseurl directive for an FTP, HTTP, or HTTPS source.

dnf module list (node.js, mariadb etc.)
**dnf group list (security, monitoring vs)**
**dnf group info "system tools" (show content of group)**
**dnf group install "system tools"**
dnf list installed
dnf repoquery cifs-utils
dnf list installed | grep cifs-utils
dnf check-update
**creating a directory inside `/dev` is not advisable**, because `/dev` is dynamically managed by the system and device nodes are created automatically.
dnf repolist

Use the MD5 checksum for verifying its integrity and the *GNU Privacy Guard* (GnuPG or GPG) public key signature for ensuring the credibility of its developer or publisher.

rpm -K /mnt/package/baseOS/zsh.3.3 —no-signature (checksum)

rpm -q gpg-pubkey (viewing keys)

rpm -qi <key> (view specific details)

rpm -Vf /etc/sysconfig (show package modification details)

chmod -v 644 /etc/sysconfig/atd (back to original state)

ls-lR
ls -lai | grep dir1
set -o noclobber
!! (repeat the last command)
!?grep? (repeat last command contains ls)
alias
**ls -ld /etc/??? (prints all three letters folders)**
**ls /usr/bin/[g]* (all folders starting with g)**
**ls /usr/bin[a-c]* (folders between a and c]**

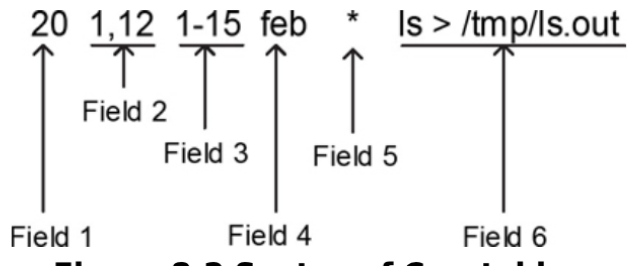
ls /etc | less
grep operator /etc/passwd
grep -v nologin /etc/passwd (print lines that don’t have nologin)
grep -n pattern /etc/passwd (print findings with number)
grep -w acce.. /etc/lvm/lvm.conf (prints lines including word acce..(i.e accept, access)

VIM <x> will delete current cursor, :<4,6d> ill delete lines
:%s/word/word1 -> replace word with word1
sed -i "s/old/new/g" replace old with new inplace
touch will change timestamp of the file
grep -i path ~/.bashrc (in-case-sensitive search)

file system check
df -h (see mounted disks)
(disk should be not mounted )
umount /dev/sdb
vi test & (run in the background)
jobs (check jobs)
fg %1 (bring job to fg)
kill <pid> (kill job)
ls /cdr /usr > output 2>&1
top then r (renice)

at 12:13pm 3/7/25
> date &> date.out
>ctrl+d
at -l (list jobs)

crontables:




The system runs the /etc/profile file first, followed by .bash_profile, .bashrc, and finally the /etc/bashrc file.

mandb
man -k xfs
In addition to the manual pages, apropos, whatis, info, and pinfo commands as well as documentation located in the /usr/share/doc directory are also available on the system.
COMMANDS:

mkdir -p A/B/C
touch (will change mod time)
stat (will print brith, mod, access time)
find / -name passwd
find / -maxdepth 3
find . -user root
ls -l | head -20 >> files
chmod ugo+rwx file / cmod ugo-rwx file
echo "umask 027" >> ~/.bashrc (newly created files: 640, directories: 750)
cat > b (write your input)
tar cvf b.tar b a (compress)
tar tf (view)
tar xf (extract)
tar cvfz /root/com.gz /etc
grep -nr "pass" /etc/passwd > /mnt/pass
ls /etc/yum.repos.d
vi /etc/yum.repos.d/local.repo

---
ROOT PASSWORD
1. "e" on boot screen
2. after UUID section, "rd.break" , befor initrd. and type ctrl+x.
3. mount -o remount,rw /sysroot
4. chroot /sysroot
5. passwd (will be given in exam)
6. touch /.autorelabel
7. exit
8. reboot

RAID(redundant array of independent disks)
stripe, mirror

IPTABLES
rule based firewall scans until it finds matches
allow block specific IP addresses or ports
powerful firewall built in linux
-

VIRTUALISATION
```
yum install qemu-kvm libvirt virt-install virt-manager
```

REPOSITORY CONFIGURATION
0. dnf install <copy given epel release url>
1. dnf repolist
2.

CONTAINER
0. container file will be given
1. dnf install podman
2. podman build -t demo .
3. podman images
4. podman -d run -p 8080:80 <image>
5.

cal 2 2025
tar cvf , tar xvf
touch command can be used with -d and -t to add sepcific date and time, a and m will enable you to change access and modification time. touch -d 2020-09-20 sec.txt .
touch -m command will reverse it to original time.
soft link can link directories hard linnk can not link directories .

Boot: BIOS-> Master Boot Record(MBR)->partition Table(PT)->Boot loader(Grub)->Kernel-> Mounting / /usr->/etc/inittab(default run level)->/etc/fstab

run levels:
run level 0 power off
run level 1: singleuser mode text mode
run level 2 multiusr text mode except NFS, NIS
run level 3: support all services including NFS and NIS(network information service) (default)
run level 4: unused
run level 5: multiuser graphical mode

User account information for local users is stored in four files that are located in the /etc directory. These files—passwd, shadow, group, and gshadow.
UIDs between 1 and 200 are used by Red Hat to statically assign them to core service accounts. UIDs between 201 and 999 are reserved for non-core service accounts, and UIDs 1000 and beyond are employed for normal user accounts.
cat /etc/login.defs
head -3 /etc/passwd
tail -3 /etc/passwd

usermod -l user2new -u 2000 -d /home/user2new -m -s /sbin/nologin user2000
useradd user4 -s /sbin/nologin
echo redhat | passwd --stdin user4
vi /et

(nologin is when user does not need login access)
userdel user3new

grep user2new /etc/passwd

SUBSCRIPTION
subscription-manager register --user stanermetin --password staN#123123123 --auto-attach

tuned-adm list
tuned-adm active (check which profile is active)
tuned-adm recommend
tuned-adm profile virtual-guest

docker run --cap-add=SYS_TIME -it bb4496e662fb /bin/bash

dnf install chrony (NTP)
dnf install -y procps

docker run --privileged -d --name systemd_container -v /sys/fs/cgroup:/sys/fs/cgroup:ro 5c79fba2bcae
What is the default number of days files in /tmp are kept before they are automatically deleted if not accessed or modified?
uname -v or cat /proc/version10 days (apropos -a list directory
apropos " list directory" / man -k ext4
The udev service (part of systemd-udevd) is responsible for creating device nodes dynamically at system startup.
Wayland has replaced the X Window System as the default display protocol in RHEL 8.
RHEL supports seven types of files: regular, directory, block special device, character special device, symbolic link, named pipe, and socket.

ln original.txt hardlink.txt # Create a hard link
ls -li original.txt hardlink.txt # Check inode numbers(same in hardlinnk)

local.repo
name=baseOS
baseurl=https://xyz.server.com/baseOS
gpgcheck=0
enabled=1

name=appStream
baseurl=https://xyz.server.com/appStream
gpgcheck=0
enabled=1

These permission bits are set user identifier bit (commonly referred to as setuid or suid), set group identifier bit (a.k.a. setgid or sgid), and sticky bit.
The setuid and setgid bits may be defined on binary executable files to provide non-owners and non-group members the ability to run them with the privileges of the owner or the owning group, respectively. The setgid bit may also be set on shared directories for group collaboration. The

sticky bit may be set on public directories for inhibiting file erasures by non-owners.

A common example is the su command that is owned by the root user. T

timedatectl set-ntp true

groupadd newgroup (should be listed in /etc/group)
useradd harsh -G newgroup
useradd nolog -s /sbin/nologin
passwd nolog (set password to redhat)
setfacl -m u:natasha:rw /var/fstab (m for modify)
getfacl /var/fstab
setfacl -m g:Mac:--- /var/fstab
4: read , 2: write 1 : execute
chmod u-x testfile -v (verbose)
chmod go+w testfile -v

chown :Mac /linux
groupadd blue
chgrp blue
chmod g+s . (now all the files will belong to group blue)
chmod +t (Sticky Bit: ensure only linux group can delete files)
chmod g-w, u+r testfile -v (from group remove writing, to user add read)
chmod -v u+s /usr/bin/su (adding user id to /usr/bin/su)

systemctl not default on container env. should enable cgroups while running:
(i. e. : docker run --privileged -d --name systemd_container -v /sys/fs/cgroup:/sys/fs/cgroup:ro
centos:8 /usr/sbin/init)

DISK PARTITION:
lsblk (check devices)
fdisk /dev/sda2 (+1G for 1 GB partition, then n, p, w for write)
partprobe /dev/sda2
lsblk(check the newly created partition)
mkdir newdisk
mkfs.xfs /dev/sda2p1
mount /dev/sda2p1 /newdisk (not persistent yet)
vi /etc/fstab (write in the file: /dev/sdap1 /newdisk defaults 0 0)
mount -a

SWAP MANAGEMENT:
free -m
fdisk nvmep2 (from the input menu type:
0.( type letters, m for help)
1. n(new), p(primary), w(write)
2. partition number default

3. first sector: default
4. last sector: +750M
5. type: t
6. partition number: 3
7. L (get hex codes, 82 for linux swap) (change partition type to linux swap)
8. partprobe nvmep2 (will make changes permanent)
9. mkswap /dev/nvmep2
10. vi /etc/fstab (/dev/nvmep2 swap swap defaults 0 0)
11. swapon -a (if no error, check with free -m, see swap has has increased by 750Mb)

LVM: (create logical volume, give size, extend existing logical volumem)
1. Create Physical Volume (pv)
2. Create Volume Group (vg)
3. Create Logical Volume (lv)

pvcreate nvme0v3
pvcreate nvme0v4
pvcreate nvme0v5
pvs (show)
vgcreate vgtest /dev/nvme0v3 nvme0v4 nvme0v5 (create volume groups)
vgs (show)

(linear, striped, mirrored volumes)
lvcreate -L 8Gb -n lvtest vgtest
lvs
vi /etc/fstab (/dev/vgtest/lg1/ /lv xfs defaults 0 0)
mkdir /lv
mkfs.xfs /dev/vgtest/lvtest
mount -a

LVM Extension:
vgs
lvextend -r -L +2Gb /dev/vg1/lv1
vgextend vg1 /dev/nvme0v5
vgs
lvremove /dev/vg1/lv1 (you'll get warning filesystem is in use)
vim /etc/fstab (comment vg1/lv1)
unmount /lv
lvchange -an /dev/vg1/lv1
lvremove /dev/vg1/lv1
lvs
vgremove vg1
vgs
vgcreate -s 8M vg1 /dev/nvme0v3
lvcreate -l 10 -n lv2 /dev/vg1 (creating 80M logic volume 10 times)

STRATIS:
blockdev:minsize 1 gb

pool (combined block devices to create pool)
filesytem(no fixed size for filesystem, automatically grows)

dnf install stratisd stratis-cli
systemctl start stratisd
systemctl enable stratisd
stratis pool create pool1 /dev/nvme0n5 (create pool)
stratis pool list
stratis pool add-data pool1 /dev/nvme0n4 (extend pool)
stratis filesystem create pool1 fs1 (create filesystem)
stratis filesystem list
stratis filesystem create pool1 fs2
stratis filesystem list
mkdir /fs1
vi /etc/fstab (copy UUID from filesystem list output /fs1 xfs
defaults,x-system.requires=stratisd.service 0 0)
mount -a

VIRTUAL DATA OPTIMISER (VDO)-deprecated.
New one: lvmvdo:
compression, thin provisioning, deduplication

dnf install lvm2 kmod-kvdo vdo

vdo create --name vdo1 --device /dev/nvme0n2 --vdoLogicalSize=50G
vdo list
mkfs.xfs /dev/mapper/vdo1
mkdir /vdo1
vi /etc/fstab (/dev/mapper/vdo1 /vdo1 auto defaults,x-systemd.requires=vdo.service 0 0)
mount -a
man vdo

dnf module info postresql:10
dnf module install -y postgresql:10

you can only have one module installed at a time.

CRON:
execute command /usr/local/bin/backup at 10:00 am on Feb 4th every year.
crontab -e (0 10 4 2 * /usr/local/bin/backup)

configure cron job for a user jiu at 12:08 every Thursday execute /bash/echo hello
crontab -u jiu -e (08 12 * * THU /bash/echo hello)

GREP:
grep -i "root" /etc/group

grep -i "sbin" /etc/passwd > /tmp/pass

CH ROOT PASSWD:
press "e" boot screen
put "rd.break" after word quiet --
mount -o remount,rw /sysroot --
chroot /sysroot --
passwd
touch /.autorelabel --
exit
reboot

NETWORKING/HOSTNAME

ip addr show ens160
nmcli con add con-name "Default" type ethernet ifname ens160
nmcli con show
nmcli con add con-name "Default1" type ethernet ifname ens160 ipv4 192.168.1.1/24 gw4 192.168.1.2
nmcli con up "Default1"
nmcli con show Default1
nmcli con mod "Default1" connection.autoconnect yes
nmcli con show Default1
nmcli con mod Default1 ipv4.addresses 192.168.2.2/16
nmcli con mod Default1 ipv4.dns 172.2.2.2
nmcli con mod Default1 ipv4.addresses 192.168.3.3/24 (multiple ip addresses can be added)
nmcli con add "Net" type ethernet ifname eth0 ipv4.addresses 200.0.0.12/16 gw4 20.0.0.1
nmcli con mod Net ipv4.dns 8.8.8.8
nmcli con show Net
nmcli con up Net

nmcli con add "net2" type ethernet ifname ens160 ipv4.addresses 172.24.5.10/24 gw4 172.24.5.48

nmtui (alternative to nmcli)
/etc/hostname
hostnamectl status
hostnamectl set-hostname server

SELINUX
touch /var/www/html
ls -ld /var
mkdir /new
touch /new/index.html
ls -ls /new/index
vi /etc/httpd/conf/httpd.conf (check DocumentRoot="/var/www/html"
vi /new/index "DocumentBoot "/var/www/html"

ls -lZ /var/www/html/index.html (httpd_sys_content_t) is the content
ls -lZ /new/index (default) is the content

```
<Directory "/new">
   AllowOverride All
   #Allow open access
   Require all granted
</Directory>
```

--> add above to /etc/httpd/conf/httpd.conf

SEMANAGE:
semanage fcontext -a -t httpd_sys_content_t "/new(/.*)?"
restorecon -Rv /new

SElinux modes: disabled permissive(0) enforced(1)
getenforce
setenforce 0 | 1
/etc/sysconfig/selinux (modify file to disable, reboot is required)
getsebool -a | grep httpd_enable (policy bool)
setsebool -P httpd_enable_homedirs on (-p flag permananent change)

selinux modes, booleans, context, port

httpd is able to access home dir: (getsebool | grep httpd_enable_homedirs)
system is not able to access httpd on port 82
semanage port -a -t http_port_t -p tcp 82 (systemctl restart httpd is required)

ensure httpd is able to access files at test directory.

PODMAN:
podman login registry.redhat.io (optional)
podman search httpd
podman pull docker.io/registry/httpd
podman rmi <image>
podman run -d --name httpd -p 8080:80 <imageID>
podman ps
curl localhost:8080 (outside the httpd container, see It works!)
podman stop <container>
podman rm <container>
podman run --d it <imageID> /bin/bash
podman exec -it <containerID> /bin/bash (inside the container check find . -name index.html see: /usr/local/apache2/htdocs)
mkdir /web && touch /web/mypage.html && vi mypage.html (add some context)
podman run -d --name web1 -p 8080:80 -v /web:/usr/local/apache2/htdocs/:Z <imageid>
after mapping
semanage fcontext -a -t httpd_sys_content "/web(/.*)?"
restore con -Rv /web

curl localhost:8080/mypage.html

QUADLET
podman info --debug | grep "rootless"
systemctl --user enable podman-auto-update-timer
systemctl --user daemon-reload

/etc/containers/systemd/sleep.container
[Unit]
Description=A minimal container

[Container]
Image=centos
Exec=sleep 60

[Service]
Restart=always

systemctl daemon-reload
systemctl start sleep.service
systemctl start sleep.service (auto restart once system rebooted)
systemctl status sleep.service

Quadlet files to be stored in either:
user: /usr/share/containers/systemd/
system wide: /etc/containers/systemd/

https://dokumen.pub/qdownload/rhcsa-red-hat-enterprise-linux-8-training-and-exam-preparation-guide-ex200.html

FILESYSTEM:
fsck.ext4 /dev/sda1
xfs-repair -L /dev/sda1

GROUP QUOTA:
mount -o remount, usrquota, grpquota /dev/sdb2/ /quota

/dev/sdb2 /quota ext4 ->/etc/fstab

SCHEDULING AND PROCESS ADMIN
at 10:03am today
> command
ctrl+D
atq

CPIO: occupies less space compared to tar.

cpio -icvf -I /root/backup

FTP.

AUTOFS:
SERVER
dnf install -y nfs* nfs-utils autofs
mkdir /share
touch /share/f1 /share/f2
chmod 777 /share
/share <clientIP>(ro,sync) >> /etc/exports
exportfs -avr
firewall-cmd --add-service={nfs,mountd,rpc-bind} --permanent
firewall-cmd --reload

CLIENT -in exam responsiblity only client part-
dnf install nfs-utils autofs
showmount -r <clientIP>
/auto_mount /etc/auto_misc --timeout=60 >> /etc/auto.master
access --rw,soft,intr <serverIP>:/share >> /etc/auto.misc
systemctl enable autofs --now (after this command /auto_mount or /afs directory should be created)
cd auto_mount && cd access && ls (should see files f1 and f2)

SHELL
```
if [ $? -eq 0 ]; then
     echo "succesfull exit"
```

$0 -> script's name
$1 -> firstarg
$# -> number of args

SSH
systemctl status sshd
firewall-cmd --zone=public --permanent --add-service=ssh

A list of the users who have successfully signed on to the system with valid credentials can be printed using one of the two basic Linux tools: who and w.
last
last user1000
last root
lastb
lastlog
id
id user1000
groups user1000

Service accounts take care of their respective services, which include apache, ftp, mail, and chrony.

3 main: networking | storage | manage groups

find -mmin -300 -exec file {} \;
find / -type p -o -type s 2>/dev/null
find /usr -atime -100 -size -5M -user root
setfacl -x u:user2000 /tmp/testfile

in vi %s/tes/fes/
sed -i 's/globe/earth/g'

/etc/nologin.txt (custom no login test if -s /sbin/nologin is defined when creating useradd user5 -s /sbin/nologin)
useradd -D default login settings
useradd -D -s /bin/sh -b /custom/home: this would set the default shell to /bin/sh and home directory to /custom/home for all future users.
Name the four local user authentication files. /etc/passwd, /etc/group, /etc/shadow, /etc/gshadow
The who command in Linux consults the /var/run/utmp file to display information about currently logged-in users.
/var/log/wtmp: Keeps a history of all logins and logouts.
/var/log/btmp: Logs failed login attempts.
/etc/shadow- is the backup for /etc/shadow
/etc/nologin is a special file in Linux. When it exists, it prevents all non-root users from logging into the system.
who, w, id, groups

The lastlog command in Linux displays the most recent login information of all system users.

password aging is a secure mechanism to control user passwords in Linux
~

PACKAGES
rpm -q vsftpd
rpm -q createrepo
rpm -qf /bin/bash : Queries which installed RPM package provides the file `/bin/bash`.

NFS Server:
yum install nfs*
(remote server)
vi /etc/exports -> /remote 192.168.11.8(rw, sync)
exportfs
(on the client)

ifconfig -a (check IP can reach to remote server
show mount -2 192.168.11.7
mount -t nfs 192.168.11.7:/remote /nfs (change IP of client to 11.7 if it does not mount)
service network restart
df -h (check mount)
vi /etc/fstab -> 192.168.11.7:/remote /nfs defaults 0 0
yum list autofs
vi /etc/auto.master
vi /etc/auto.misc -> ram -fstype:nfs 192.168.11.7 (config for mount point)

SAMBA Server(device and file share across heterogenous OSes.
Ports: 137(name), 138(datagram), 139(session)
vi /etc/samba/smb.conf
cd /etc/samba && grep "log" *
service smb restart
(on the client)
smbclient -L //182.168.11.7/ -N
(exercise: secure shared shares in samba server)(disable printer sharing)

DHCP server:
PORTS: 67-bootp, 68-dhcp
yum install dhcp -y
cd /usr/share/doc/dhcp_server
cp dhcpd.conf.example /etc/dhcpd/dhcpd.conf
vi /etc/dhcpd.conf (edit subnet and range according to IP)
service dhcpd start

NETWORK INFORMATION SERVICE a.k.a YellowPages (NIS)

DNS RECORDS
CNAME. files.example.org alias hostname
A RECORD IP address of the domain. maps hostname to IPv4 address to be saved in icann.net
MX RECORD maps domain name to mailexchange server. host can have multiple MX
PTR RECORD maps ipv4 to the canonical name for the host. adds 192.168.1.10.in-addr.arpa (reverse address)
NS record maps domain name to list of DNS servers authoritative for that domain.
named-checkconf /etc/named.conf
named-checkconf /etc/rfc1912.zones
named-checkzone example.com example.for
named-checkzone 192.168.1.11.in-addr.arpa

APACHE:
port: 80
yum install http*

Additionally, certain configuration options have been deprecated or removed in recent BIND 9 releases. For instance, the `auto-dnssec` configuration statement was removed, and users are advised to use `dnssec-policy` or manual signing instead.

BIND 9 Documentation
It's advisable to consult the release notes of the specific BIND version you're using to stay informed about any changes to configuration options.

yum install system-config-kickstarter
system-config-kickstart(will bring gui)
add
root password
installation method: ftp: server 192.168.10.7, ftp: directory: pub
partition info: add 10GB mount boot filetype ext4,add  swap 2048
network config: eth0: dhcp.
authentication: keep default settings
firewall: SElinux disabled
installation:
vmlinuz initrd.initrd.img repo=ftp://192.168.10.7/

Basic sendmail (deprecated)
yum install sendmail*
vi /etc/mail/sendmail.mc (nothing to be added)
make -C /etc/mail
vi /etc/mail/sendmail.mc
service sendmail start
service sendmail status
sendmail -v -s “Test Email” user@server.example.org

POSTFIX:
dnf remove sendmail
dnf install postfix
dnf install chkconfig
chkconfig postfix off
systemctl status postfix
vi postfix.sh …
vi /etc/rc.local -> /root/postfix.sh
LVM

google ad

redirection
umask 022
chmod ugo
tar xvf gz
chmod g-s /usr/bin/write -v (gid)