#Users

1. Create a group called admin with a GID of 10015

groupadd -g 10015 admin

2.Create 3 users named: Andrew, Dan and Natalie
2.1: Andrew should have an UID of 1046

useradd -u 1046 Andrew

2.2: Natalie should have a non interactive login shell

useradd -s /sbin/nologin Natalie

2.3:Andrew and Dan should be have the group admin as a supplementary group

useradd Dan
usermod -aG admin Andrew
usermod -aG admin Dan

c

#Privileges

1. All members of the admin group should have sudo privileges for all commands

vim /etc/sudoers.d/admin
#Add the following line within that new file
%admin ALL=(ALL) ALL

2. Dan should have sudo privileges for all commands

vim /etc/sudoers.d/Dan
#Add the following line within that new file
Dan ALL=(ALL) ALL

#User-defaults

1.All newly created users should have to change their passwords every 30 days

#Within /etc/login.defs
PASS_MAX_DAYS 30

2.All newly created users should receive a warning 5 days before expiry

#Within /etc/login.defs
PASS_WARN_AGE 5

3.All newly created users should have a minimum password age of 10 days

#Within /etc/login.defs
PASS_MIN_DAYS 10

4.All newly created users should have a file called "All-users" with the message: "Created for all homes" within their home directory.'

echo "Created for all homes" >> All-users

#User-passwords

1.Create 3 users named Pass1,Pass2 and Pass3 with a password of Redhat

useradd Pass1
passwd Pass1
useradd Pass2
passwd Pass2
useradd Pass3
passwd Pass3

1.1: The user Pass1's password should expire every 10 days

chage -M 10 Pass1

1.2 The users Pass2's account should expire in 30 days from the current date

date -d "+30 days" +%F
chage -E #replace with date Pass2
||
chage -E $(date -d "+30 days" +%F) Pass2

1.3 The user Pass3 should change their password upon first login

chage -d 0 Pass3

#Permissions

1. Create a directory called /home/perms

mkdir /home/perms

2.Ensure the directory is owned by the user Andrew and the group admin

chown Andrew:admin /home/perms

3.Ensure that all newly created content within the directory will inherit the group ownership of the directory
4.Ensure only the owner of the directory or owner of the files within that directory can delete files within that directory
5.Ensure the following privileges are set on the directory: Owner: Read-Write-Execute, Group-Read-Write-Execute, Others-Execute

chmod 3771 /home/perms

#Recuring jobs Cron

1.Create a recuring job called sysjob which will run the script called /home/myscript.sh every Tuesday at 5 PM at minute 0 as root

vim /etc/cron.d/sysjob

#Within the file add:

0 17 * * 2 root /home/myscript.sh

2.Create a recuring job for the user Andrew which will run the: "echo test" command at minute 10 every hour and every day in January

crontab -e -u Andrew # As root
crontab -e # As the user Andrew

#Within Crontab add

10 * * 1 * echo test

#Systemd-Targets

1.Set the default target to multi-user.target

systemctl set-default multi-user.target
systemctl get-default

#Time and Date

1.Set the local timezone to Bucharest

timedatectl list-timezone

timedatectl set-timezone Europe/Bucharest

2.Add an NTP server with the following address: ntp.server.com

vim /etc/chrony.conf

#Within chrony.conf add:
server ntp.server.com iburst

3.Enable NTP

timedatectl set-ntp true

#Repositories

1.Set up a repository file called BaseOS.Repo with the following details :
name: BaseOS
baseurl: http://myrepo.com
enabled: true
gpgcheck: false
id: BaseOS.Dvd

vim /etc/yum.repos.d/BaseOS.repo

#Within /etc/yum.repos.d/BaseOS.repo file add:

[BaseOS.Dvd]
name=BaseOS
baseurl=http://myrepo.com
enabled=true
gpgcheck=false

#Networking

#You can do all except number 2 graphically using the command nmtui

1.Set the hostname to Mytesthost.com

hostname Mytesthost.com

2.Reference the IP 10.1.1.1 to the name "private" on the localhost file

echo "10.1.1.1 private" >> /etc/hosts

3.Create an ethernet network connection with the following details:
Name: myconnection
Interface: eth0 (If exists)
IPv4: 192.168.1.1/24 and 192.168.1.2/24
Gateway: 192.168.1.254
Dns: 192.168.1.254
Autoconnect: True

nmcli con add con-name myconnection type ethernet ifname eth0 ipv4.addresses 192.168.1.1/24 ipv4.gateway 192.168.1.254 ipv4.dns 192.168.1.254 autoconnect yes

nmcli con mod myconnection +ipv4.addresses 192.168.1.2/24

#Firewalld

1.Set the default zone to public

Firewall-cmd --set-default-zone=public

#SElLinux Modes

1.Set runtime enforcement to permissive

setenforce 0

3.Configure SELinux to start in permissive mode on boot:

vim /etc/selinux/config

#Within /etc/selinux/config change:

SELINUX=permissive

#Troubleshoot Booting

1.Reset Root Password

Interupt the boot process and press “e” to edit the kernel options
At the end of the linux line write “rd.break”
Press CTRL+X to continue

#Now within the terminal

#Make /sysroot read-write
mount -o remount,rw /sysroot

#Treat it as root filesystem
chroot /sysroot

#Change root password
passwd root

#Relabel SELinux filesystem
touch /.autorelabel

#Exit by running the exit command twice
exit
exit

#Troubleshoot Booting

1.Fix Boot Issues

Interupt the boot process and press “e” to edit the kernel options
At the end of the linux line write systemd.unit=emergency.target

#Now within the terminal

#Make / read-write
mount -o remount,rw /

#Look at /etc/fstab
vi /etc/fstab

#Fix the problem
#Reboot

#Advanced Storage

1.Create a volume group called vg1 using one of the previously created partitions

vgcreate vg1 /dev/vdb1

2.Create a logical volume called lv01 with a size of 300M

lvcreate -n lv01 -L 300M vg1

3.Format the logical volume lv01 with an xfs filesystem

mkfs.xfs /dev/vg1/lv01

4.Create a directory called /mountlvm

mkdir /mountlvm

5.Mount the logical volume lv01 on /mountlvm

mount /dev/vg1/lv01 /mountlvm

6.Ensure persistent mounting on boot of the logical volume lv01 on /mountlvm by UUID

#Obtain UUID of filesystem
lsblk --fs

#Enter fstab
vim /etc/fstab

#Within fstab add the following line:
UUID=<ID> /mountlvm xfs defaults 0 0

#Advanced Storage 2

1.Extend the volume group vg1 with the second partition

vgextend vg1 /dev/vdb2

2.Extend the logical volume lv01 by 200M

lvextend -L +200M /dev/vg1/lv1

3.Grow the filesystem of the logical volume lv01

xfw_growfs /mountlvm

4.Create a logical volume named lv-swap1 with the size of 300M

lvcreate -n lv-swap1 -L 300M vg1

5.Format lv-swap1 as swap space

mkswap /dev/vg1/lv-swap1

6.Mount the logical volume lv-swap1 persistently on boot using it's UUID

#Get the UUID
lsblk

#Enter fstab
vim /etc/fstab

#Within fstab add the following line:
UUID=<ID> swap swap defaults 0 0

#Storage Stack

1.Install the stratisd service and cli

dnf install stratisd
dnf install stratis-cli

2.Start the stratisd service

systemctl start stratisd

3.Enable the stratisd service

Systemctl enable stratisd

4.Create a stratis pool called pool1

stratis pool create pool1 /dev/vdb

5.Create a stratis filesystem within pool1 called fs1

stratis filesystem create pool1 fs1

6.Create a directory called /mountstratis

mkdir /mountstratis

7.Persistently mount the stratis filesystem fs1 on /mountstratis using it's UUID

vim /etc/fstab

 # Get the UUID of the filesystem
stratis filesystem list

#Within fstab add the following line:
UUID=<UUID> /mountstratis xfs defaults,x-systemd.requires=stratisd.service 0 0

#Podman Services

0.As the user student directly
1.Create the following path: /home/student/.config/systemd/user

mkdir -p /home/student/.config/systemd/user

2.Create a service file based on the previously created webserver container

cd /home/student/.config/systemd/user
podman generate systemd --name webserver --files –new

#Delete the previous container
podman rm -f webserver

systemctl --user daemon-reload

3.Enable the newly created service and ensure it starts whenever the system boots

systemctl --user enable container-webserver.service
loginctl enable linger

#Tuned

1.Enable the tuned service to start at boot-time

systemctl enable tuned

2.Start the tuned service

systemctl start tuned

3.Check the recommended profile by tuned

tuned-adm recommend

4.Set the recommended profile by tuned

tuned-adm profile <Recommended profile>