**White test N°6 - RHCSA 8**

**Initialization : Systems set up**

nmtui

edit connection

**System1 :**

Hostname – system1.example.com

IP – 192.168.55.150/24

Dns – 192.168.5.1

GW – 192.168.5.1

**System2 :**

Hostname – system2.example.com

IP – 192.168.55.151/24

Dns – 192.168.5.1

GW – 192.168.5.1

**Questions**

**System1 :**

1. reset the root password of the system1 server to « password123 »

**Click e**

**add “rd.break”**

**CTRL-X**

**mount -o remount, rw /sysroot**

**passwd password**

**touch /.autorelabel**

**exit**

**exit**

1. create a backup file named « /root/backup.tar.bz2 ». the backup file should contain the content of « /usr/local » and should be zipped with bzip2 compression format.

**tar -cvfj /root/backup.tar.bz2 /usr/local**

1. Ensure selinux is in enforcing mode. If it is not, change selinux to enforcing mode.

**vim /etc/selinux/config**

**SELINUX=enforcing**

1. Configure a task with user ahmed : write the message hello in file test every monday at 17h.30

**crontab -e**

**30 17 \* \* 1 ahmed echo “hello”>>test**

1. Find the files owned by the user student, and copy it to catalog : /opt/dir

**mkdir -p /opt/dir**

**find / -user student -exec cp -a {} /opt/dir \;**

1. Write a shell script that finds files located in /etc and whose size is between 2bytes and 10M

**echo “find /etc -size +2c -a -size -10M” >> script.sh**

**chmod +x script.sh**

1. Find the rows that contain /bin/bash from file /etc/passwd, and write it to the file /tmp/testfile.

**touch /tmp/testfile**

**grep /bin/bash /etc/passwd >> /tmp/testfile**

1. On system2, configure Europe/london as the time zone

**timedatectl list-timezones**

**timedatectl set-timezone Europe/london**

**timedatectl status**

1. Configure system1 to be a ntp client for system2

**vim /etc/chrony.conf**

**server system2 iburst**

**systemctl restart chronyd**

1. Add user1, user2 and user3. The additional group of the two users : user2, user3 is the admin group. Password : redhat

**useradd user1**

**useradd user2**

**useradd user3**

**usermod -aG admin user2 / groupmod -aU user2 admin**

**usermod -aG admin user3 / groupmod -aU user3 admin**

**passwd user1**

**passwd user2**

**passwd user3**

1. Add admin group and set gid=6000

**groupadd -g 6000 admin**

1. Create a shared directory /home/admins, make it has the following characteristics :

/home/admins belongs to group admin. This directory can be read and written by members of group admin. Any files that will be created in /home/admins must have admin as owner group.

**mkdir /home/admins**

**chown :admin /home/admins // chgrp admin /home/admins**

**chmod g=rwx /home/admins**

**chmod g+s /home/admins**

copy /etc/passwd to /var/tmp/pass. Owner of the file /var/tmp/pass is root, belongs to group root, and this file can not be executed by any user

**mkdir /var/tmp**

**touch /var/tmp/pass**

**cp /etc/passwd /var/tmp/pass**

**#chown root:root /var/tmp/pass#**

**chmod o=rw- /var/tmp/pass**

1. Add the service nfs to your firewall configuration

**firewall-cmd - -add-service=nfs - -permanent**

**firewall-cmd - -reload**

1. when he connects , formateur1 will have his home folder mounted on “/home/formateur1” from server2:/tekup/tic/formateur1

**server1:**

**useraadd -u 2000 formateur1**

**echo “/home/formateur1 \*(rw,no\_root\_squash)” >> /etc/exports**

**server2:**

**useradd -M -u 2000 -d /tekup/tic/formateur1 formateur1**

**echo “/tekup/tic /etc/auto.formateur1” >> /etc/auto.master**

**echo “\* -rw server2:/home/formateur1” >> /etc/auto.formateur1**

**sustemctl restart autofs**

**su - formateur1**

1. Make balanced profile as your default profile

**tuned-adm profile balanced**

**systemctl restart tuned.service**

1. Create a container ​logserver​ from an image ​rsyslog​

●Configure the container with systemd services by an existing user “student »

,●Service name should be ​container-logserver​, and configure it to start automatically across reboot

Configure automount​ /var/log/journal​ from ​logserver​ (container) to/home/student/container\_logserver​ when container starts.

**loginctl enable-linger student**

**ssh student@localhost**

**podman login registry**

**username**

**password**

**wget link\_containerfile**

**podman build -t rsyslog .**

**podman run -d - -name logserver -v /var/log/journal:/home/student/container\_logserver:Z**

**mkdir .config/systemd/user**

**cd .config/systemd/user**

**podman generate systemd - -​container-logserver​ - -file - -new**

**vim container-logserver.service**

**[service]**

**StartLimitBurst=300000**

**Restart=always**

**RestartSec=2s**

**[install]**

**WantedBy=default.target multi-user.target**

**systemctl - -user daemon-reload**

**systemctl - -user enable - -now container-logserver.service**

1. Configure your host journal to store all journal across reboot

**vim /etc/systemd/journal.conf**

**#Storage=auto ⇒ Storage=persistent**

**reboot**

●Copy all \*.journal from /var/log/journal and all subdirectories to/home/student/container\_logserver

cp -a /var/log/journal/\*.journal /home/student/container\_logserver

**System2 :**

1. Create a 512M partition (/dev/sdb of 2GB), make it as ext4 file system, mounted automatically under /mnt/data1 and which take effect automatically at boot-start

**fdisk /dev/sdb**

**n,p,defaults,defaults,+512M, w**

**mkfs.ext4 /dev/sdb1**

**mkdir -p /mnt/data1**

**blkid /dev/sdb1**

**echo “uuid=id\_part /mnt/data1 ext4 defaults 0 0”>> /etc/fstab**

**mount -a**

**lsblk**

1. Make a swap partition having 100MB (/dev/sdb of 2GB). Make automatically usable at system boot time.

**fdisk /dev/sdb**

**n,p,defaults,defaults,+100M, w**

**mkswap /dev/sdb2**

**swapon /dev/sdb2**

**echo “uuid=id\_part none swap defaults 0 0”>> /etc/fstab**

1. Create a volume group vg (5GB), and set PE=32M.Create two logical volumes as followes :

* Lvol1 with size 1256M, configure it with ext4 file system, and mount it automatically under /mnt/lv1
* Lvol2 with 10 LE . configure it with xfs and mount it permanently to /mnt/lvo2.

**fdisk /dev/sdb**

**n,p,default,default,+5G,w**

**pvcreate /dev/sdb3**

**vgcreate vg -s 32M /dev/sdb3**

**lvcreate -L 1256M -n Lvol1 vg**

**lvcreate -l 10 -n Lvol2 vg**

**mkfs.ext4 /dev/vg/Lvol1**

**mkdir /mnt/lv1**

**mkfs.xfs /dev/vg/Lvol2**

**mkdir /mnt/lvo2**

**blkid /dev/vg/Lvol1**

**blkid /dev/vg/Lvol2**

**echo “uuid=id\_part /mnt/lv1 ext4 defaults 0 0”>> /etc/fstab**

**echo “uuid=id\_part /mnt/lvo2 xfs defaults 0 0”>> /etc/fstab**

1. Reduce the size of Lvol1 to 500M

**lvreduce -r -L 500M /dev/vg/Lvol1**

1. Extend the size of lvol2 to 620M

**lvextend -r -L 620M /dev/vg/Lvol2**