Contents

[RHCSA Cheat sheet 3](#_Toc57120241)

[Archives 3](#_Toc57120242)

[Permissions 3](#_Toc57120243)

[Reset root password 3](#_Toc57120244)

[Virtual Machines 4](#_Toc57120245)

[LVM Basics 4](#_Toc57120246)

[Manipulating partitions 4](#_Toc57120247)

[Fdisk 4](#_Toc57120248)

[Gdisk 4](#_Toc57120249)

[Managing Disks 4](#_Toc57120250)

[Setting up NFS 5](#_Toc57120251)

[Setting up samba 5](#_Toc57120252)

[ACL File Permissions 5](#_Toc57120253)

[Network Configuration 6](#_Toc57120254)

[NTP 6](#_Toc57120255)

[Scheduling Tasks 6](#_Toc57120256)

[Modifying bootloader 6](#_Toc57120257)

[Yum Repos 6](#_Toc57120258)

[Using LDAP for SSO 7](#_Toc57120259)

[Configure AD using realmd 7](#_Toc57120260)

[Manipulating Users 7](#_Toc57120261)

[Password management 7](#_Toc57120262)

[Group management 8](#_Toc57120263)

[Configure firewall 8](#_Toc57120264)

[SELinux 8](#_Toc57120265)

[Make journald persistent 8](#_Toc57120266)

[Adjust process priority via nice 8](#_Toc57120267)

[Disk compression (VDO) 9](#_Toc57120268)

[Yum modules 9](#_Toc57120269)

[Linux Schedulers 9](#_Toc57120270)

[Extending LVM Disks 9](#_Toc57120271)

[Extending VDO Disks 9](#_Toc57120272)

[Sample exam questions 9](#_Toc57120273)

[systemctl cheatsheet 11](#_Toc57120274)

# RHCSA Cheat sheet

## Archives

# Compress

tar cvf filename.tar /var/log

# Extract

tar xvf filename.tar

# Compress gzip

gzip file

# Extract gzip

gzip -d file.gz

# Compress bzip2

bzip2 file

# Extract bzip2

bzip2 -d file.bz2

tar czvf filename.tar.gz /var/log

tar xzvf filename.tar.gz /var/log

tar cjvf filename.tar.bz2 /var/log

tar xjvf filename.tar.bz2 /var/log

zip -r filename.zip /var/log

unzip filename.zip

## Permissions

chmod a+w file # Gives everyone write permissions

umask 0222 # Subtracts from actual permissions

# Persist umask

vi /etc/profile

vi /etc/bashrc

## Reset root password

* Press e on boot menu
* Add rd.break at the end of linux16 line
* Press ctrl-x to boot

mount -o rw,remount /sysroot

chroot /sysroot

passwd root

touch /.autorelabel

exit

exit

## Virtual Machines

yum install qemu-kvm python-virtinst libvirt libvirt-python virt-manager libguestfs-tools virt-install qemu-img libvirt-client

virt-manager # GUI

virt-install --name centos7 --ram 1024 --vpus 2 --disk path=/var/lib/libvirt/images/image.qcow2 --os-type linux --network bridge=virbr0 --location=/tmp/centos7.iso --extra-args 'console=ttyS0'

virsh list --all

virsh start <name>

virsh edit <name>

## LVM Basics

pvcreate /dev/sdb1

vgcreate <VGNAME> /dev/sdb1 -s <extent-size>

lvcreate -L 100M <VGNAME> -n <LVNAME>

lvcreate -n <LVNAME> -l <extents> <VGNAME>

## Manipulating partitions

## Fdisk

fdisk -cul <disk>

fdisk <disk>

# n -> new partition

# d -> delete partition

# p -> print current partition

# w -> write partition table

# 82 is Swap

# 83 is Linux

# 8e is Linux LVM

partprobe

## Gdisk

gdisk -l <disk>

gdisk <disk>

# n -> new partition

# d -> delete partition

# p -> print current partition

# w -> write partition table

# 8200 is Swap

# 8300 is Linux

# 8e00 is Linux LVM

partprobe

mkfs.xfs /dev/device

partprobe

mkdir /mnt/mountpoint

mount /dev/device /mnt/mountpoint

lsblk

blkid

echo "UUID=ksjfslkfjsdaf /mnt/mountpoint xfs defaults 0 0" >> /etc/fstab

mount -a

## Managing Disks

xfs\_admin -L "LADisk" /dev/device

xfs\_admin -l /dev/device

# Reset UUID

xfs\_admin -U nil /dev/device

xfs\_admin -U restore /dev/device

xfs\_Admin -U generate /dev/device

tune2fs -L "ext4Disk" /dev/device

tune2fs -l /dev/device

## Setting up NFS

yum install nfs-utils

mkdir /nfs

chmod 777 /nfs

echo "/nfs \*(rw)" > /etc/exports

systemctl start {rpcbind,nfs-server,rpc-statd,nfs-idmapd}

showmount -e localhost

# On client

systemctl start rpcbind

showmount -e <ip-of-server>

mount -t nfs <ip>:/nfs /mnt/nfs

## Setting up samba

# On server

yum install samba

mkdir /smb

chmod 777 /smb

echo "[share]" >> /etc/samba/smb.conf

echo "browseable = yes" >> /etc/samba/smb.conf

echo "path = /smb" >> /etc/samba/smb.conf

echo "writable = yes" >> /etc/samba/smb.conf

useradd sambauser

smbpasswd -a sambauser

setenforce 0

# On client

mkdir /mnt/smb

yum install cifs-utils samba-client

mount -t cifs //<ip-of-server>/share /mnt/smb -ousername=sambauser,password=12345

## ACL File Permissions

getfacl file

# Set permissions

setfacl -m g:groupname:rwx file

# Remove permissions

setfacl -x u:username:rwx file

# Recursive permissions

setfacl -R -m g:groupname:rw folder/

# Default permissions

setfacl -d -m g:groupname:rwx folder/

# Remove default permissions

setfacl --remove-default folder/

# Copy permissions

getfacl file1 | setfacl --set-file=- file2

## Network Configuration

nmcli conn show

nmcli conn add con-name <name> autoconnect yes type ethernet ifname eth0 ip4 <ip> gw4 <ip>

nmcli conn up <name>

nmcli conn modify <name>

nmcil conn modify <name> +ipv4.dns 8.8.8.8

## Check dns first

vi /etc/nsswitch.conf

## NTP

timedatectl status

yum install chrony

chronyc tracking

chronyc sources -v

# Edit NTP servers

vi /etc/chrony.conf

systemctl restart chrony

## Scheduling Tasks

at now +1 minute

at midnight

atq # List pending jobs

atrm # Delete job by number

echo "username" >> /etc/at.deny # For denying users

crontab -e # Current User

crontab -e -u <username> # For username

echo "\*/15 \* \* \* \* script.sh" >> /etc/cron/user # Run every 15 minutes

echo "username" >> /etc/cron.deny # Deny specific user

## Modifying bootloader

grubby --info=ALL

grub2-set-default <index>

grubby --set-default-index <index>

## Yum Repos

echo "[base]" >> /etc/yum.repos.d/base.repo

echo "name = Base" >> /etc/yum.repos.d/base.repo

echo "enabled = 1" >> /etc/yum.repos.d/base.repo

echo "baseurl = http://baseurl.com" >> /etc/yum.repos.d/base.repo

echo "gpgcheck = 0" >> /etc/yum.repos.d/base.repo

yum repolist

## Using LDAP for SSO

yum install authconfig-gtk nss-pam-ldapd pam\_krb5 autofs nfs-utils openldap-clients

authconfig --enableldap --enableldapauth --enablemkhomedir --enableldaptls --ldaploadcacert=http://server.com/path/to/ca.pem --ldapserver=ldap.server.com --ldapbasedn="dc=example,dc=com" --update

# For GUI

authconfig-gtk

su - ldapuser1

## Configure AD using realmd

yum install realmd

realm discover <AD Server>

realm join <AD Server>

ssh -l test@example.com <IP-address>

## Manipulating Users

id <user>

getent shadow <user>

# Change display name

usermod -c "Name" <user>

usermod -aG <group> <user>

# Lock user

usermod -L <user>

# Skeleton directory

ll /etc/skel/

# useradd defaults

cat /etc/default/useradd

cat /etc/login.defs

## Password management

chage -l <user>

# Password validity

chage -M 30 <user>

# Inactivity after expiry

chage -I 1 <user>

# Set lockout date

chage -E 2020-11-11 <user>

# Days before expiry when user starts to receive warning

chage -W 10 <user>

## Group management

groupadd <group>

# Change primary group

usermod -g <group> <user>

# Change secondary group

usermod -G <group> <user>

# Add additional group

usermod -aG <group> <user>

groupmod -g <new-id> <group>

## Configure firewall

yum install firewalld firewall-config

firewall-cmd --get-zones

firewall-cmd --get-default-zone

firewall-cmd --zone=home list-all

firewall-cmd add-service=http

firewall-cmd add-service=http --permanent

firewall-cmd add-port=80/tcp

firewall-cmd --reload

## SELinux

getenforce

setenforce 1

getsebool httpd\_enable\_cgi

setsebool httpd\_enable\_cgi on

getsebool -a | grep httpd

semanage fcontext -l

ls -lZ /var/www/html

ps auxZ | grep httpd

semanage fcontext -a -t httpd\_sys\_content\_t '/content(/.\*)?'

restorecon -R /content

# Troubleshooting

yum install setroubleshoot-server

sealert -a /var/log/audit/audit.log

## Make journald persistent

mkdir -p /var/log/journal

systemctl restart systemd-journald

## Adjust process priority via nice

Highest priority = -20 Lowest priority = 19

nice -n 5 script.sh

renice -n <priority> -p <pid>

## Disk compression (VDO)

yum install vdo kmod-kvdo

systemctl start vdo.service

vdo create --name=MyVDO --device=/dev/nvme1n1 --vdoLogicalSize=60G --deduplication=disabled

mkfs.xfs -K /dev/mapper/MyVDO

udevadm settle

mount /dev/mapper/MYVDO /mnt/vdo

## Yum modules

yum module enable <module-name>:<stream>

yum module enable postgresql:9.6

yum install postgresql

yum module list

## Linux Schedulers

chrt -p <pid>

# Use FIFO

chrt -f -p <PRIO> <pid>

chrt -f <PRIO#> /path/to/command

chrt -d --sched-runtime 5000000 --sched-deadline 1000000 --sched-period 1666666 0 /your/command/here

## Extending LVM Disks

pvcreate /dev/sdc1

vgextend <VGNAME> /dev/sdc1

lvextend -r -L +100%FREE /dev/<VGNAME>/<LVNAME>

## Extending VDO Disks

vdo growPhysical --name=my\_vdo

vdo growLogical --name=my\_vdo -vdoLogicalSize=new\_logical\_size

vdostats --human-readable

## Sample exam questions

* Create users and groups etc

useradd tom

useradd kenny

useradd derek

groupadd instructors

usermod -aG instructors tom

usermod -aG instructors kenny

usermod -aG instructors derek

usermod -s /sbin/nologin tom

chage -E $(date -d "+10 days" %y-%m-%d) tom

* Configure apache

nmcli conn show

nmcli conn up eth0

yum repolist

yum install httpd

vi /etc/httpd/conf/httpd.conf # Change DocumentRoot to desired root

mkdir /var/web

echo "Hello world!" > /var/web/index.html

systemctl restart httpd

firewall-cmd list-services

firewall-cmd --add-service=http --permanent

firewall-cmd --reload

ls -lZ /var/web/

yum whatprovides semanage

yum install policycoreutils-python

semanage fcontext -a -t httpd\_sys\_content\_t '/var/web(/.\*)?'

restorecon -R /var/web

systemctl enable httpd

* Modify umask for all users to 006

vi /etc/profile

vi /etc/bashrc

# Change umask value to desired

* Find files that are 720 days old

find /etc -maxdepth 1 -mtime +720 > /root/oldfiles

* Save logs containing ACPI to another file

cat /var/log/messages | grep ACPI > /root/logs

tar -czf /tmp/log\_archive.tgz /var/log

* Reduce boot timout for grub to 2 seconds

vi /etc/default/grub # Change GRUB\_TIMOUT to 2

grub2-mkconfig

* Schedule for user

crontab -e derek

# 27 16 \* \* \* cat /etc/redhat-release >> /home/derek/release

* Change NTP server

vi /etc/chrony.conf

# Remove all other servers and add required one

systemctl restart chronyd

* Create swap partition

fdisk -l

# Create new partition with 82 as type for swap

mkswap /dev/vdb1

blkid

echo "UUID=<UUID> swap swap defaults 0 0" >> /etc/fstab

mount -a

free

swapon /dev/vdb1

partprobe

free

* Create new volume using LVM

yum install lvm2

pvcreate /dev/vdc

vgcreate myVG /dev/vdc -s 32m

lvcreate -n myLV -l 30 myVG

mkfs.xfs /dev/myVG/myLV

blkid

echo "/dev/mapper/myVG-myLV /mnt xfs defaults 0 0" >> /etc/fstab

partprobe

mount -a

df -h

* Change hostname

hostnamectl set-hostname <hostname>

* Configure LDAP properly

yum install -y authconfig-gtk nss-pam-ldapd pam\_krb5 autofs nfs-utils openldap-clients

authconfig-gtk

echo "/home/guests /etc/auto.ldap" >> /etc/auto.master.d/ldap.autofs

echo "\* -rw ldap.linuxacademy.com:/home/guests/&" >> /etc/auto.ldap

vi /etc/pam.d/sshd

# Add `auth sufficient pam\_ldap.so` to the start of the file

systemctl restart ssdh

su - ldapuser3

## systemctl cheatsheet

|  |  |
| --- | --- |
| VIEW systemd INFORMATION | |
| systemctl list-dependencies | Show a unit’s dependencies |
| systemctl list-sockets | List sockets and what activates |
| systemctl list-jobs | View active systemd jobs |
| systemctl list-unit-files | See unit files and their states |
| systemctl list-units | Show if units are loaded/active |
| systemctl get-default | List default target (like run level) |

|  |  |
| --- | --- |
| WORKING WITH SERVICES | |
| systemctl stop service | Stop a running service |
| systemctl start service | Start a service |
| systemctl restart service | Restart a running service |
| systemctl reload service | Reload all config files in service |
| systemctl status service | See if service is running/enabled |
| systemctl enable service | Enable a service to start on boot |
| systemctl disable service | Disable service--won’t start at boot |
| systemctl show service | Show properties of a service (or other unit) |
| systemctl -H host status network | Run any systemctl command remotely |

|  |  |
| --- | --- |
| CHANGING SYSTEM STATES | |
| systemctl reboot | Reboot the system (reboot.target) |
| systemctl poweroff | Power off the system (poweroff.target) |
| systemctl emergency | Put in emergency mode (emergency.target) |
| systemctl default | Back to default target (multi-user.target) |

|  |  |
| --- | --- |
| VIEWING LOG MESSAGES | |
| journalctl | Show all collected log messages |
| journalctl -u network.service | See network service messages |
| journalctl -f | Follow messages as they appear |
| journalctl -k | Show only kernel messages |

|  |
| --- |
| **USING UNIT FILES** |
| Besides **services**, most systemd commands can work with these unit types:  **paths**, **slices**, **snapshots**, **sockets**, **swaps**, **targets**, and **timers** |

This cheat sheet covers the basic podman command set.container is either a container name or a container ID. If tag is omitted in image:tag, the default value is latest.

ls -al /lib/systemd/system/runlevel\* (get run levels)

**systemctl isolate [target-name]**