RHCSA_CHEATSHEET_v9.0

Disclaimer:

This sheet includes most of the essential commands relevant to the RHCSA (Red Hat Certified System Administrator) exam based on version 9.0. If you are preparing for a more recent version of the exam, please consult the official Red Hat documentation or authorized training materials to ensure accuracy and completeness.

#File_System Structure:

```
/:

    /boot      -->boot files

    /dev       -->device file[use to access hardware]

    /etc       -->contains system configuration files

    /home      -->regular user directory

    /root      -->home directory for superuser

    /run       -->runtime data for processes[like RAM]

    /tmp       -->temporary files or all user can access this directory

    /lib       -->holds essential libraries and kernel modules for system boot and basic commands.

    /var       -->stores variable data like logs, caches, and spools that persist across reboots.

    /usr:

         /usr/bin        -->User Commands

         /usr/sbin       -->admin commands

         /usr/local      -->local customized software
```

#BASIC COMMANDS:

```
    whoami          -->print the current user
```

hostname	-->print hostname
--help	-->it is option for every command to take help
man	-->command for get manual of any command
ls	-->list content of current directory
cd	-->change directory
pwd	-->print working directory path
cp	-->copy file or folder
mv	-->move file or folder
touch	-->create new empty file
head	-->to read first 10 line of file
tail	-->to read last 10 line of file
cat	-->read file
less	-->read file in another tab
more	-->read file content in parts of percentage
grep	-->user to filter any string [its use only when the command return result in terminal].
echo	-->print the string content on terminal
mkdir	-->make directory
rm	-->remove file
rmdir	-->remove only empty directory
rm -rf	-->remove directory recursively and force fully also use for file
whereis	-->locate program-related files[ex. whereis ls]
find	-->locate any file or directory based on conditions

#CREATE LINK FOR FILE AND FOLDER:

```
ln -s     -->create soft link
ln        -->create physical link[to verify look the inode value of both file]
```

#TABLE FOR MATCH CHARACTER:

Pattern	Matches
*	Any string of zero or more characters
?	Any single character
[abc…]	Any one character in the enclosed class (between the square brackets)
[!abc…]	Any one character not in the enclosed class
[^abc…]	Any one character not in the enclosed class
[[:alpha:]]	Any alphabetic character
[[:lower:]]	Any lowercase character
[[:upper:]]	Any uppercase character
[[:alnum:]]	Any alphabetic character or digit
[[:punct:]]	Any printable character that is not a space or alphanumeric
[[:digit:]]	Any single digit from 0 to 9
[[:space:]]	Any single white space character, which might include tabs, newlines, carriage returns, form feeds, or spaces

NOTE:-its use with listed command-->ls, cp, mv, rm, find, grep

#VIM FILE EDITOR:

vim-->edit file

[press] esc: -->option mode

options:

- i-insert
- d-delete
- u-undo
- x-delete single character
- v-character mode
- ctrl+V-block selection(block mode)
- shft+V-enter to visual mode(line mode)

[type] : -->command mode

command:

- q!-exit
- wq-write and exit

NOTE:-if you not remember all things then use 'vimtutor' command.

#MANAGE LOCAL USERS AND GROUPS:

IMP DIRECTORIES:

/etc/passwd -->each line contain information about user except passwd

```
            -->username:user id:group id:comment:home dir:shell type
        /etc/group      -->each line contain information about groups
            -->group name:group passwd:group id:list of USERS
        /etc/sudoers    -->main config file for sudoers
            -->[%group][user]          ALL=(ALL:ALL)               ALL
                     |                        |                      |
                user/group    host = (run-as-user:run-as-group)    command
                **NOTE:-[done all configuration in sudoers.d directory]**
                ex.[ansible        ALL=(ALL)       NOPASSWD*: ALL]
                *NOPASSWD-allow a user to run commands as another user without entering their password
        /etc/shadow     -->contain password hash for all users
            -->ex.user03:$6$CSsXsd3rwghsdfarf:17933:0:99999:7:2:18113:
            -->username:hash:lastchange:minage:maxage:warndays:inactive:expiry
Commands:
        id          -->view current user id and other information like group id,primary
                       group,secondary group,context etc.
        su          -->change user[use '-' for change user with home dir]
        useradd     -->add user
        usermod     -->modify user configuration[/sbin/nologin-shell dir for nologin in shell]
        groupadd    -->add group
        groupmod    -->modify group configuration
        passwd      -->change password
        chage       -->change password policy[/etc/login.defs->dir for modify permenant password
                       policy]
```

#CONTROL ACCESS FILE:

```
    read-->4    write-->2   execute-->1
    Commands:
        chmod       -->To modify the permission for file and folder
            [chmod   Who/What/Which     file|directory]
                who     -->u-user,g-hroup,o-others,a-all
                what    -->'+'-add,'-'-remove,'='-set exactly
                which   -->r-read,w-write,x-execute,X-special execute[recursive permission change]
            ex.,chmod ugo+rwx file/dir
        chown       -->change ownership of file or folder
            [chown user:group file/dir](-R option for change ownership of entire directory tree)
    #SPECIAL PERMISSIONS
        =>PERMISSION-->EFFECT ON FILE-->EFFECT ON DIRECTORIES
            u+s --> File runs as file owner --> No effect
            g+s --> File runs as group owner --> New files inherit directory's group
            o+t --> No effect --> Only file owners can delete their files
             Symbolic : setuid = u+s; setgid = g+s; sticky = o+t
             Octal : In the added fourth preceding digit; setuid = 4; setgid = 2; sticky = 1
    #Effect of umask Utility on Permissions
        command:- umask[temporary][if want to permanent then config /etc/profile]
        file permission=>
            Symbolic-> rw-rw-rw- Numeric octal->0666
        directory permission=>
            Symbolic-> rwxrwxrwx Numeric Octal->0777
```

#MONITER AND MANAGE LINUX PROCESSES:

```
================================================
[NAME       FLAG    Kernel-defined state]
 Running    --> R       --> TASK_RUNNING
 Sleeping   --> S       --> TASK_INTERRUPTIBLE
            --> D       --> TASK_UNINTERRUPTIBLE
            --> K       --> TASK_KILLABLE
            --> I       --> TASK_REPORT_IDLE
 Stopped    --> T       --> TASK_STOPPED
            --> T       --> TASK_TRACED
 Zombie     --> Z       --> EXIT_ZOMBIE
            --> X       --> EXIT_DEAD
=================================================
Commands:
    top      -->Shows real-time system processes and resource usage (dynamic, updates live).
    ps       -->Displays a snapshot of current processes (static, one-time view).
        'aux'option displays all processes including processes without a controlling terminal
    sleep    -->create process[option '&' use for run process in background]
    jobs     -->return list of jobs
    fg       -->run process in foreground
    bg       -->run process in background[%<id> to choosee jobs id]
    kill     -->kill process[-l option for list signals]
    pstree   -->to view a process tree for the system or a single user
```

#MONITOR PROCESS ACTIVITY:

```
    Commands:
        uptime  -->display load average of CPU
        lscpu   -->list CPU related info
        w       -->Displays logged-in users and their current activities, along with system uptime and
                   load.
            ==CALCULATE SYSTEM'S LOAD==
             From lscpu, the system has four logical CPUs, so divide by 4:
                                        load average: 2.92, 4.48, 5.20
                divide by number of logical CPUs:    4     4     4
                               per-CPU load average:   0.73  1.12  1.30
```

#CONTROL SERVICES AND DAEMONS:

```
    Commands:
        systemctl   -->Manage system SERVICES
        NOTE:-Use the help option and man page for more.
```

#CONFIGURE AND SECURE SSH:

#IMP DIRECOTORIES-

```
/etc/ssh/ssh_config     -->Global configuration file
~/.ssh/config           -->Stores per-user SSH client configuration settings, like aliases, ports,
                           usernames, and key files for remote hosts.
~/.ssh/known_hosts      -->stores the host public keys of remote servers you’ve connected to via
                           SSH.
/etc/ssh/ssh_known_hosts-->Global file
```

Commands:

```
ssh username@<ip addr/hostname> -->take remote access of another user.
ssh-keygen                      -->generate key
ssh-copy-id -i user@remotehost  -->copy id [option 'i' for selecting custom key-file]
eval $(ssh-agent)               -->create agent for password[dont forget to run ssh-add command]
ssh-add                         -->give password to agent process
```

=>uses the PermitRootLogin configuration setting in the /etc/ssh/sshd_config file to allow or prohibit users to log in to the system as the root user <permitrootlogin yes>

=>uses the PasswordAuthentication parameter in the /etc/ssh/sshd_config file to control whether users can use password-based authentication to log in to the system.

#MANAGING NETWORKING:

Connection config file:

```
    /etc/NetworkManager/system-connections/
Commands:
    ====== nmcli con show <con-id> ======
    ip                       -->view interface and address
    ping                     -->check connection by sending ICMP packets
    nmcli dev status         --> Show the NetworkManager status of all network interfaces.
    nmcli con show           --> List all connections.
    nmcli con show name      --> List the current settings for the connection name.
    nmcli con add con-name name --> Add and name a new connection profile.
    nmcli con mod name       --> Modify the connection name.
    nmcli con reload         --> Reload the configuration files, after manual file editing.
    nmcli con up name        --> Activate the connection name.
    nmcli dev dis dev        --> Disconnect the interface, which also deactivates the current
                                 connection.
    nmcli con del name       --> Delete the specified connection and its configuration file.
                                 Configure Hostnames and Name Resolution:
    directory:-
        /etc/hostname-->store static hostname
        /et/resolv.conf-->Stores DNS server addresses used to resolve domain names into IP addresses.
    Commands:
        hostname    -->display hostname
        hostnamectl -->configure hostname
```

#INSTALL AND UPDATE SOFTWARE PACKEGES

```
    yum/dnf     -->package install update uninstall
    uname -r    -->shows only the kernel version and release
    uname -a    -->shows the kernel release and additional information.
    #Enable DNF Software Repositories:
        IMP DIRECTORY:
            /etc/yum.repos.d/       -->configuration repo file[source list]
                file content:[extension is '.repo']
                        "name=EPEL 9
                        baseurl=https://dl.fedoraproject.org/pub/epel/9/Everything/x86_64/
                        enabled=1
                        gpgcheck=1
                        gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-9"
        commands:
            dnf repolist all        -->lists all available repositories and their statuses.
```

#ACCESS LINUX FILE SYSTEMS:

```
    IMP DIRECTORIES:
        /etc/fstab      -->persistent mount file
    Commands:
        lsblk       -->list the details of a specified block device or of all the available devices.
        mount       -->manage mount Point to file `system
```

umount   -->unmount the file system

lsof   -->lists all open files and the processes that are accessing the file system.

#SCHEDULE FUTURE TASKS:

IMP DIRECTORIES:

/etc/crontab

/etc/cron.d/   -->custom files

/etc/cron.hourly,/etc/cron.daily,/etc/cron.monthly,/etc/cron.weekly

/var/spool/anacron/-->directory determine the daily, weekly, and monthly jobs.

/etc/anacrontab-->this configuration is make sure the crontab task should be run.

Commands:

at   -->schedule task within terminal temporary

atq   -->list the scheduled jobs

crontab   -->to manage scheduled jobs.

crontab [options] filename

remember if dont know format than look /etc/crontab file.

systemctl daemon-reload-->After you change the timer unit configuration file.
to ensure that the systemd timer unit loads the changes.

*make an any entry of that user in /etc/cron.deny.-->its restrict a user to create crontab file.

#ANALYZE AND STORE LOGS

IMP DIRECTORIES:

/run/log/journal -->in this file all the logs are stored[journalctl is used to read]

/etc/rsyslog.conf -->main config file for store log persistent.[but we edit only file in /etc/rsyslog.d/]

/etc/systemd/journald.conf -->configuration settings of the systemd-journald service.so that the journals persist across a reboot.

/etc/chrony.conf -->contain NTP server configuration[chronyd service]

*server classroom.example.com iburst

Commands:

journalctl -->to view journal log file.

timedatectl -->fetch current timezone.

tzselect -->guided way to set timezone

chronyc -->verify that the local system is seamlessly using the NTP server to synchronize the system clock

#TUNE SYSTEM PERFORMANCE

IMP DIRECTORIES:

/etc/tuned/tuned-main.conf -->main config file

/usr/lib/tuned -->stores the tuning profiles

/etc/tuned -->stores the tuning profiles

Commands:

tuned-adm -->manage profiles

nice -->for new pew process to modify nice value[-20 to 19 where -20 nice value represent priority

renice -->we can change nice value of existing process

[in this the commands of process management are use like ps and top]

#MANAGE SELINUX SECURITY:

IMP DIRECTORIES:

/etc/selinux/config -->selinux config file[after change this file reboot the server]

Commands:

getenforce -->to get selinux mode

setenforce -->set mod to enforce[1-enforce,0-permissive]

-Z -->it is option which give the context of file or folder

chcon -->use to change context

semanage fcontext -->always use this command to change context.[semanage fcontext -a \
-t httpd_sys_content_t '/lab-content(/.*)?']

restorecon -->to relabel the contents of the file system.[run it always after use semanage fcontext cmd to apply the changes]

getsebool --> Display current status (on/off) of SELinux booleans.

setsebool --> Change the current state of an SELinux boolean (temporarily or permanently).

semanage --> SELinux policy management tool for managing SELinux configuration settings (like file contexts, ports, booleans, etc.).

semanage-boolean --> Manage SELinux booleans using the semanage command-line interface.

#MANAGE BASIC STORAGE:

IMP DIRECTORIES:

/etc/fstab -->to persistent mount point then entry must be in this file[after run this command run systemctl daemon-reload]

Commands:

fdisk	-->use to make partitions
lsblk	-->list the blocks
pvdisplay	-->display physical volume
pvcreate	-->create physical volume
pvremove	-->remove physical volume
vgdisplay	-->display volume group
vgcreate	-->create volume group
vgremove	-->remove volume group
lvdisplay	-->display logical volume
lvcreate	-->create logical volume
lvremove	-->remove logical volume
lvextend	-->extend logical volume
mkfs	-->assign filesystem to new logical volume
resize2fs	-->assign file system to extended logical volume.
mkswap	-->to format the LV as a swap space

```
partprobe   --> Updates the kernel with changes made to the partition table.
xfs_growfs  --> Expands an XFS filesystem to use additional space from its underlying device or

                logical volume.
swapon      --> Enables and activates swap space on a device or file.
swapoff     --> Disables and deactivates swap space on a device or file.
```

#ACCESS NETWORK-ATTACHED STORAGE:

```
     $Manually by using the mount command.
     $Persistently at boot by configuring entries in the /etc/fstab file.
     $On demand by configuring an automounter method.
  IMP DIRECTORIES:
      /etc/fstab
          **server:/export  /mountpoint  nfs  rw  0 0**
  Commands:
      mount -t nfs -o rw,sync server:/export /mountpoint //
      umount      -->unmount
      mount.nfs   -->Mounts a Network File System (NFS) share to a local directory.
  #AUTOFS NETWORK ATTACHED STORAGE
      dnf install autofs nfs-utils
      1.make master file with .autofs extension
          mount-point     map-file name
      2.make map file start with auto. in /etc dir
```

```
    mountpoint      -rw,sync    servera:/tmp/demo
```

3.enable autofs service

```
    systemctl enable autofs.service --now
```

$ 3 TYPES OF AUTOFS CONFIG

1.DIRECT--> directory exist in / dir , 2.INDIRECT-->dir does not exist in / , 3.WILDCARD—> sharing dir contain many dir

#CONTROL THE BOOT PROCESS:

Commands:

```
systemctl get-default          -->get default target like graphical or multi-user
systemctl set-default          -->set default target like graphical or multi-user
systemctl isolate              -->switch to a different target temporary
```

#RESET ROOT PASSWORD:

Commands and steps:

1. Reboot and interrupt GRUB:

   -Reboot server (Ctrl+Alt+Del)

   -At GRUB menu, press any key (except Enter) to stop countdown.

2. Edit rescue kernel:

   -Select the rescue kernel → Press e

   -Find line starting with linux

   -Remove any console= entries //like "console=tty0"

-At end of the line, add: rd.break

-Press Ctrl + x to boot

3.At switch_root:/# prompt:

```
>>>mount -o remount,rw /sysroot
>>>chroot /sysroot
```

4.Reset password:

```
>>>passwd root
```

5.Force SELinux relabel:

```
>>>touch /.autorelabel
```

6.Exit and reboot

#MANAGE NETWORK SECURITY:

Commands:

firewall-cmd --> Command-line tool to configure and manage firewalld settings.

firewalld --> Daemon that dynamically manages firewall rules and zones.

firewalld.zone --> Man page describing the structure and options of individual firewalld zone configuration files.

firewalld.zones --> Directory containing predefined zone configuration files used by firewalld.

#RUN CONTAINERS:

Commands:

```
yum install container-tools
```

```
podman login quay.io                                  --> Login to the container registry.
podman search quay.io/httpd                           --> List all available images from the
                                                          registry.
podman pull FQIN                                      --> Pull the specified image locally.
podman images                                         --> List all locally available images.
podman ps                                             --> List currently running containers.
podman ps -a                                          --> List all containers (running and
                                                          exited).
podman run -it FQIN bash                              --> Run container interactively with a shell
                                                          (random name assigned).
podman run -it --name my-http FQIN bash               --> Run container interactively with custom
                                                          name.
podman run -d -it --name my-http1 FQIN bash           --> Run container in detached mode.
podman exec -it my-http1 bash                         --> Access running container nteractively.
podman run --rm FQIN cat /etc/passwd                  --> Run a command in a container and remove
                                                          it after exit.
cat /etc/containers/registries.conf                   --> View container registry configuration.
podman info                                           --> Display system and registry information.
podman inspect FQIN                                   --> Inspect details of the specified image.
podman start my-http1                                 --> Start a stopped container.
podman stop my-http1                                  --> Stop a running container.
podman restart my-http1                               --> Restart a container.
```

```
    podman rm my-http1                                     --> Remove a specific container.
    podman rm -a                                           --> Remove all containers.
    podman rmi FQIN                                        --> Remove the specified image.
```

1 Port Forwarding

Command:

```
podman run -d --name my-container -p 8080:80 FQIN
```

Description: Forwards traffic from host port 8080 to container port 80

Verify: curl localhost:8080

2 Persistent Storage

Command:

```
podman run -d -v /host/path:/container/path:Z FQIN
```

Description: Mounts persistent storage from host to container with SELinux context

Note: /host/path must exist

3 Container as a Service

For Root Users

```
podman generate systemd --name nextcloud > /etc/systemd/system/nextcloud-container.service
cat /etc/systemd/system/nextcloud-container.service
systemctl daemon-reload
systemctl start nextcloud-container.service
systemctl enable nextcloud-container.service
systemctl status nextcloud-container.service
```

```
podman kill nextcloud

podman ps
```

For Unprivileged Users

```
mkdir -p ~/.config/systemd/user/

cd ~/.config/systemd/user/

podman generate systemd --name myweb --files --new

systemctl --user daemon-reload

systemctl --user enable --now container-myweb.service

systemctl --user start container-myweb.service

loginctl enable-linger              --> Allow user services to run after logout
```

BEST OF LUCK😉 !

MADE_BY:-RAHUL_PRAJAPATI
CONTACT:-rahul2100007@gmail.com