**Linux Notes, RHCSA Certification**

VNC server

**vncserver** - *starts vncserver*

**vncpasswd** - *sets VNC password*

Basic commands

**cd** - *change directory*

**ls** - *list contents*

**man** - *manual page for command*

**cat** - *concatenate contents*

**echo** - *echo contents*

**>** - *redirect (standard output)*

**>>** - *append redirect*

**2>** - *redirect (standard error)*

**&>** - *both redirects (2>&1)*

**|** - *execute commands after*

**less** - *basic viewer*

**head/tail** - *cat beginning or end*

**grep** - *finds contents within a file*

**rm** - *remove (-r recursive, -f don’t ask, -i ask)*

**locate** *- locate file based on cache,* **updatedb** *to update cache*

**df -h** *- shows available disk space*

**sudo du -sh /\* | grep G** *- shows disk usage*

Transfer commands

**scp [file] user@1.1.1.1:~/** - *transfer [file] to user at 1.1.1.1 in home directory*

**sftp user@1.1.1.1** - *connect as user; standard linux commands work*

**get [file]** - *download from user*

**put [file]** - *upload to user*

**bye** *- exits sftp*

User commands*[[1]](#footnote-0)*

**su -** - *switch user to root*

**su [user]** - *switch user to user*

File Compression/Packing

**gzip/gunzip [file]** - *.gz and un-.gz file*

**tar -c filename.tar directory/ file1** - *tar compression creation for entire directory plus file in pwd*

**tar -f -x** - *specify filename, extract*

**tar -z -j -d** *- use gzip, use bzip, show differences versus destination*

Star

**star -c** *- create*

**star -f=filename.tar** *- specify file*

**star -x** *- extract tar file (WILL NOT overwrite newer)*

**star -x -f=filename.tar filename** *- extract specific filename*

Rights

**chmod u-r filename** - *u user, g group, o others, - remove, +add, r read, w write, x execute*

**groupadd name** *- add user group*

**getent group** - *view all groups*

**chown name:groupname** *- change owner, null for null*

**usermod -G groupname username** *- change user to group*

**chmod ug-x -R finance**

**chmod ug+X -R finance** - *directories retain rights, files lose them*

**chmod 147 filename** - *u,g,o, 1ex,2write,4read; UID - first digit when 4 digits is 2group,4user*

**chmod +t +s** *- sticky bit, set uid (1 more for octal notation)*

Find

**find filepath -type parameter** *- searches filepath for the type (user, name, modify, etc) based on the given parameter*

*Commands can be executed based on find results*

System Power

**shutdown** - *-r reboot, +3 add 3 minutes, 00:00 at midnight, -P poweroff, -h halt*

**reboot  
init 0** - *both reboot*

Targets

**systemctl list-units --type=target** - lists all targets loaded

**systemctl -t help** - *list all types*

Process for interrupting boot to reset root

**e on selected boot system**

**Append .break to the rd linux16 kernel line**

**Ctrl+X**

**mount -oremount,rw /sysroot**

**chroot /sysroot**

**passwd root**

**(enter password twice)**

**exit**

**reboot**

Process for interrupting boot to change boot target

**e on selected boot system**

**Append linux16 kernel line to add:**

**systemd.unit=rescue.target**

**Ctrl+X**

Pgrep/Pkill

**ps** - *list running processes*

**pgrep** - *ps and grep in one utility*

**pgrep -u username -l [tag]** - *show all processes names containing [tag] for a specific user*

**pkill httpd** - *kill a process based on a name*

**kill -l** *- shows all kill signals*

**kill -SIGHUP** *- kill all terminals*

**kill -SIGINT** *- kill keyboard*

**kill -SIGCONT** *- kill and restart process*

**kill -SIGSTOP** *- stop process for starting again*

**kill -SIGTSTP** *- can be ignored*

Processes

**(while true; do echo -n “infinite program” >> /dev/null; done;) &** *- infinite process*

**jobs** *- see what is running in the background*

**kill -SIGSTOP %1** *- stop job 1*

**kill -SIGCONT %1** *- restart job 1*

**kill [process id]** *- kills process based on ID number*

**nice -n 0 processname** *- change process nice level (requires restart)*

**renice -n 0 processid** *- change process nice level without requiring restart*

**renice -n 0 $(pgrep name)** *- change process nice level for all process named same*

**top** *- m toggles memory, t toggles tasks, l toggles uptime, b toggles bold, r for renice, k for kill, shift+key for sort-by*

Logs

**/var/log** *- logs location*

**journalctl**

**journalctl \_SYSTEM\_UNIT=servicename.service** *- displays journal for that system*

**rsyslog.conf (/etc/)** - *logging service used to create generic log files*

Mounting

**fdisk/gdisk file** - *partitions file for mounting (MBR/GPT)*

**mkfs -t xfs file** *- makes filesystem out of file, formatted as xfs*

**mount /location/ /destination/** *- mounts partitioned file to destination*

**umount /location/** *- unmounts a mounted partition*

**blkid** *- lists drives, mounted and physical*

VGM

**lvcreate -n name -L 10G size group** - create logical volume  
**lvdisplay**

**vgcreate groupname location location2 location 3** *- create a volume group for logical volumes*

Configure Mounted

**xfs\_admin -L location name** *- xfs label structure (lowercase L to check label)*

**tune2fs -L name location** *- ext4 label structure (lowercase L to check label)*

**/etc/fstab** *- mounting file properties for persistent mounts*

**mount -a** *- mount all in fstab*

**/etc/fstab** *- making a mount persistent:*

***Standard partitions*** *- location mountlocation filesystem defaults 0 0*

***CIFS*** *- //ip.address/public /mnt/sambashare cifs username=username,password=password 0 0*

***NFS*** *- ipaddress:/nfsshare /mnt/nfsshare nfs defaults 0 0*

*Second digit should be 0 for no check, 2 for check after root*

CIFS/NFS Mounting

**smbclient -L ip.address** - Samba client login

**mount -t cifs -o username=username //ip.address/public /mnt/sambashare**

**mount -t nfs ip.address:/nfsshare /mnt/nfsshare**

SGID Bit

**chmod [u,g,o]+s** *- applies ownership to child directories and files of the directory the SGID bit is set to*

ACL

**getfacl filename**

**setfacl -m [u,g,o]:name:[rwx] filename**

**setfacl -m m:[rwx] filename** - *set mask rights*

**setfacl -d -m [u,g,o]:name:[rwx] directoryname** - *set defaults for all files created within directory*

**setfacl --remove-default filename** *- remove default permissions*

**setfacl -x [u,g,o]:name filename**

Network Commands

**ip addr show eth0** *- show network adapter information*

**ping -c# ip.address -** *ping a specific IP a certain number of times (#=number)*

**traceroute address** *- show full route between your device and the address*

**tracepath address** *- same, but not all systems support*

**ss** *- shows all listening ports*

**nmcli con add con-name “name” autoconnect yes type ethernet ifname eth1**

**nmcli con show**

**nmcli con show --active**

**nmcli con mod “mycon”**

**hostname** *- shows hostname*

**hostnamectl set-hostname name** *- sets hostname to “name”*

**hostnamectl status** *- shows hostname information*

**etc/resolv.conf** *-*

**search name.com** *will automatically add suffxed search parameters when trying to use commands that include a hostname*

**nmcli con mod “System eth0” ipv4.dns 8.8.8.8** *- persistent, will add after a reboot. + or - before the ipv4 will choose whether to add or subtract from resolv*

**getent hosts google.com** *- returns IP addresses for host. system can be configured to look in hosts file before going externally*

Time-based commands

**at now +3 minutes**

**at 12:00am**

**at q** *- show jobs*

**at rm 1** *- remove job number*

**at.deny** *- not allowed if in file*

**at.allow** *- only allowed if in file*

Cron file

**minute, hour, day, month, day of week**

**All - \***

**Every 5 mintues - \*/5**

**5 0 - minute 5, hour 0**

**x-y is a range**

**For /steps, it starts on 1**

**Day of week is listed as spelled out day**

Services

**systemctl get-default** *- default target for system boot*

**systemctl status httpd** *- show loaded service status*

**systemctl list-dependencies targetname.target | grep service**

**systemctl is-enabled servicename** *- determines whether service is enabled or disabled*

**systemctl set-default** *- set target for system boot*

Kickstart

**PXE server required**

**system-config-kickstart** *- kickstart creator (GUI)*

**pykickstart** *- package name*

**Packages:**

**@ *-*** *group package*

**#** - not to install

Virtual Guests on Physical Machine

**Required packages:** *virt-manager, qemu-kvm, qemu-img; libvirt, libvirt-python, python-virtinst, libvirt-client*

**systemctl enable libvirtd** *- required daemon for libvirt*

**systemctl start libvirtd**

**Boot VM when physical machine boots:** *virsh, list --all, autostart vm-name*

**Network services start automatically at boot***: cd /etc/sysconfig/network-scripts, nmcli con show, nmcli con mod “ens3” connection.autoconnect yes*

Time

**timedatectl** *- display time information*

**tzselect** *- walkthrough for setting time when unknown timezone*

**timedatectl set-timezone timezonename** *- sets timezone (use tzselect info)*

**date** *- displays current date*

**systemctl status chronyd** *- checks system clock and adjusts*

**chronyc sources -v** *- displays sources*

**systemctl restart chronyd && chronyc sources -v**

**chronyc tracking** *- references for time sources*

YUM - Yellowdog Update Manager

**yum check-update** *- update checker for yum*

**yum update**

**yum search packagename**

**yum list installed**

**yum list installed | grep packagename (or servicename)**

User Accounts

**/etc/passwd:**

**/sbin/nologin** *- no access to terminal shell*

**/bin/bash** *- default login*

**/etc/shadow** *- encrypted passwords*

**/etc/skel** *- all files copied to new users*

**user, useradd, userdel**

**useradd -c “Name Name Name” username**

**/etc/login.defs -** *password parameters*

**usermod -s /sbin/nologin username**

**chage -E date/amount of time** *- account expires*

**usermod -G groupname username**

Firewall

**firewalld**

**firewall-cmd --get-zones** *-*

**firewall-cmd --get-default-zone** *-*

**firewall-cmd --zone=home --permanent --add-source=ip.address/24**

**firewall-config**

**firewall-cmd --panic-on** *- shuts down all ports*

SELinux

**getenforce** *- show enforcement*

**setenforce 0** *- Permissive*

**setenforce 1** *- Enforcing*

**/etc/selinux/config for global enforcement**

**semanage fcontext** *- modify context of files*

Misc

**sed -i ‘s/a/b’ location** *- replace a with b in location.*

1. .bash\_profile is executed during any login

   .bash\_logout is executed during any logout

   .bashrc is executed in any shell

   /etc/profile is executed during any login

   /user/.bash\_profile is per specific user login [↑](#footnote-ref-0)