Computer System & Network Administration

Lecture 03. User Management & Service Management

refer to: https://nasa.cs.nctu.edu.tw/sa/2020/

Outline

- User Management
- Root User
- User related commands
- Permission
- Service Management

User Management

ID

- User ID (uid), Group ID (gid)
  - One user can only have one uid and one **default** gid
  - But one user can be in multiple groups
- **root**
  - uid: 0, default gid: 0
  - System Administrator

Some other important users

Username	Description
bin	An account inherited from Unix world, not being used in modern Linux
daemon	An unprivileged user/group for daemons to execute under in order to limit their access to the system. Also not being used in modern Linux
sys	owner of the kernel and memory images
mail	An account specified to handle mail tasks
ftp	An account specified to handle ftp tasks
nobody	An account to represent the least permission on the system
dbus messagebus	To assist D-bus, a message bus system provides inter-process communication
systemd-*	Accounts to assist systemd
uuidd	UUID generation daemon

To add a new user in Linux

- Two options
  - useradd
  - adduser
- Both of these commands can create a new user in Linux

adduser? useradd? Hmmm

Difference between useradd and adduser

- useradd
  - A native binary compiled with the system
  - Need to provide options while running the command
- adduser
  - A perl script uses useradd in backend
  - More user friendly and interactive

```
$ which useradd
/usr/sbin/useradd
$ which adduser
/usr/sbin/adduser
$ file /usr/sbin/useradd
/usr/sbin/useradd: ELF 64-bit LSB shared object,
x86-64, version 1 (SYSV), dynamically linked,
interpreter /lib64/ld-linux-x86-64.so.2,
BuildID[sha1]=65ff024ba1111b0ce86873f9b22b9c6eded
9c501, for GNU/Linux 3.2.0, stripped
$ file /usr/sbin/adduser
/usr/sbin/adduser: Perl script text executable
```

useradd reference

- `useradd -D`
  - Default Group
  - Home directory basedir
  - Inactive information
  - Expire information
  - Default shell
  - skeleton (Base of home directory)
  - Whether to create mailbox for new user

```
$ useradd -D
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/sh
SKEL=/etc/skel
CREATE_MAIL_SPOOL=no
```

Adding a User by adduser

```
$ sudo adduser testuser
Adding user `testuser' ...
Adding new group `testuser' (1003) ...
Adding new user `testuser' (1003) with group `testuser' ...
Creating home directory `/home/testuser' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for testuser
Enter the new value, or press ENTER for the default
        Full Name []: Test User
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
$
```

/etc/login.defs

- Basic information about
  - Password expiration limitation
  - UID / GID limit
  - useradd/userdel limit
  - Login limit
  - ...etc

```
$ grep PASS /etc/login.defs
#  PASS_MAX_DAYS   Maximum number of days a password may be used.
#  PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#  PASS_WARN_AGE   Number of days warning given before a password expires.
PASS_MAX_DAYS   99999
PASS_MIN_DAYS   0
PASS_WARN_AGE   7
```

To delete a user in Linux

- userdel or deluser?

```
$ file /usr/sbin/userdel
/usr/sbin/userdel: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV),
dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2,
BuildID[sha1]=035b62d87de8c55f491cde51bfe5810c6d9a8944, for GNU/Linux
3.2.0, stripped
$ file /usr/sbin/deluser
/usr/sbin/deluser: Perl script text executable
```

To disable a user from logging in

- Change user's login shell to /usr/sbin/nologin
- Put a "#" in front of the account entry
- Put a "-" in front of the account entry

```
$ grep testuser /etc/passwd
testuser:x:1003:1003:Test User,,,:/home/testuser:/bin/bash
```

/etc/passwd

- Each line represents an account
- Colon-separated file contains these information
  - Username
  - Password information
  - uid
  - gid
  - GECOS field
    - Full name, Office, Extension, Home Phone
  - Home directory
  - Login shell

```
testuser:x:1003:1003:Test User,,,:/home/testuser:/bin/bash
```

/etc/passwd (cont.)

- GECOS
  - General Electric Comprehensive Operating System
- Commonly used to record personal information
- “,” separated
- GECOS information can be fetched from command “finger”
- Use chfn to change GECOS information

/etc/passwd (cont.)

- Login Shell
  - Command interpreter
    - /bin/sh
    - /bin/csh
    - /bin/bash
    - /bin/tcsh
    - /bin/zsh
  - Use chsh to change default shell

/etc/passwd

- Wait, password information?
  - Your password is nothing more than a “x”
- Yes, in older Linux distro, the password is stored in this field
  - But `/etc/passwd` can be read by everyone, meaning everyone can access this information
  - Which they shouldn’t.
- So in modern Linux, the password information is being stored in `/etc/shadow`

/etc/shadow

- Each line also represents an account
- Colon-separated file contains these information
  - Username
  - Password
  - Last Changed Date
  - Minimum number of days required between password changes
  - Maximum number of days the password is valid
  - Warn the user days before password is to expire
  - Number of days after password expires that account is disabled
  - Days since Jan 1, 1970 that account is disabled

```
$ sudo grep testuser /etc/shadow
testuser:$6$vtIeqh8CuY.G3xWj$drz4eJD4vl3ivrB5SYhWxYabZytl0.b7QftFq0UsXAJ2611GBQ90L6syJTQsGx6Nse1RmlUtkd5.uky/sUiqz1:18695:0:99999:7:::
```

/etc/shadow

- Encrypted methods
  - des
    - Plaintext: at most 8 characters
    - Cipher: 13 characters long
    - vFj42r/HzGqXk
  - md5
    - Plaintext: arbitrary length
    - Cipher: 34 characters long started with "1"
    - 1xbFdBaRp$zXSp9e4y32ho0MB9Cu2iV0

/etc/shadow

- Encrypted methods
  - blf
    - Plaintext: arbitrary length
    - Cipher: 60 characters long started with "$2a$"
    - $2a$04$jn9vc7dDJOX7V335o3.RoujuK/uoBYDg1xZs1OcBOrIXve3d1Cbm6
  - sha512
    - Plaintext: arbitrary length
    - Cipher: 106 characters long started with "6"
    - 6o4B4Pa/ql3PpRAQo$196.cCzrTCOIpPqk.VX7EqR0YNtf0dRLdx5Hzl6S7uGaPz4EDJdoXnmsSf.A21xS2zimI1XsHAglCR2Pw7ols1

/etc/group

- Each line represents a group
- Colon-separated file contains these information
  - Group name
  - Group password
  - Group ID
  - User belong to the group
- A user can be in multiple groups

/etc/group (cont.)

/etc/passwd:
ubuntu:x:1001:1001:Ubuntu:/home/ubuntu:/bin/bash

/etc/group:
ubuntu:x:1001:

/etc/group (cont.)

- `group`
  - List what group you're in
  - The first group is default group
  - To change default group, use
    - `newgrp <GROUP_NAME>`
- To add user to group
  - `gpasswd -a <USERNAME> <GROUP>`
  - `usermod -aG  <GROUP> <USERNAME>`

Root user

Root

- Root
  - Equivalent to Administrator in Microsoft Windows
  - UID is 0, GID is 0
- UNIX/Linux permit super-user to perform any valid operation on any file or any process
  - Changing root directory of certain process with chroot
  - Setting system clock
  - Limiting user resource usage
  - Network configuration
  - Power management

Login as root

- Console login
  - Using tty / serial / console
- Remote login
  - sshd
    - /etc/ssh/sshd_config
    - #PermitRootLogin yes
  - DO NOT UN-COMMENT THIS LINE!!!!!

Becoming root

- su
  - su, su - , su <USERNAME>
    - Environment is unmodified with the exception of USER, HOME, SHELL which will be changed to target user
    - "su -" will simulate as a full login.
      (All environment variables changed)

Becoming root (cont.)

- sudo: a limited su
  - Subdivide power of superuser
    - Who can execute what command on which host as whom.
  - Each command executed through sudo will be logged (/var/log/auth.log)
  - Edit /usr/local/etc/sudoers using visudo command
    - Check mutual exclusive access of sudoers file
    - Syntax check

/etc/sudoers

- sudoers format
  - Who can execute what command on which host as whom.
    - The user to whom the line applies
    - The host on which the line should be noted
    - The commands that specified users may run
    - The users as whom they may be executed
  - Use absolute path

/etc/sudoers (cont.)

- Line format
  - <USERNAME>/%<GROUPNAME>
  - Allow on which host
  - Allow as which user
  - Allow what command
- `user ALL=(root) /usr/sbin/,/sbin/, /usr/bin/`
  - Allow user
  - to run on all hosts
  - as root
  - commands in /usr/sbin, /sbin and /usr/bin

```
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
ncku-nasa ALL=(ALL:ALL) NOPASSWD:ALL
```

/etc/sudoers (cont.)

- Alias
  - Host_Alias LINUX=linux1,linux2
  - Cmd_Alias PRINT=/usr/bin/lpc, /usr/bin/lprm
  - User_Alias student=adam, eve
- Usage
  - student1 LINUX=(root) /bin/bash
  - student ALL=(root) PRINT
  - teacher ALL=(student) ALL

sudo

- Run date as nobody
  - sudo -u nobody date
- Blacklist is not always safe
  - Use whitelist would a better choice

User related commands

w, who, last, lastlog

- w
  - Show who is logged on and what they are doing.
- who
  - Who is logged on
- last
  - Show listing of last logged in users
- lastlog
  - Reports the most recent login of all users or of a given user

w, who, last, lastlog

```
F74076310@F74076310:~$ w
 05:05:00 up 18:21,  1 user,  load average: 0.04, 0.01, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
F7407631 pts/0    140.116.252.158  05:04    1.00s  0.08s  0.00s w
```

```
F74076310@F74076310:~$ who
F74076310 pts/0        2021-03-07 05:04 (140.116.252.158)
```

w, who, last, lastlog (cont.)

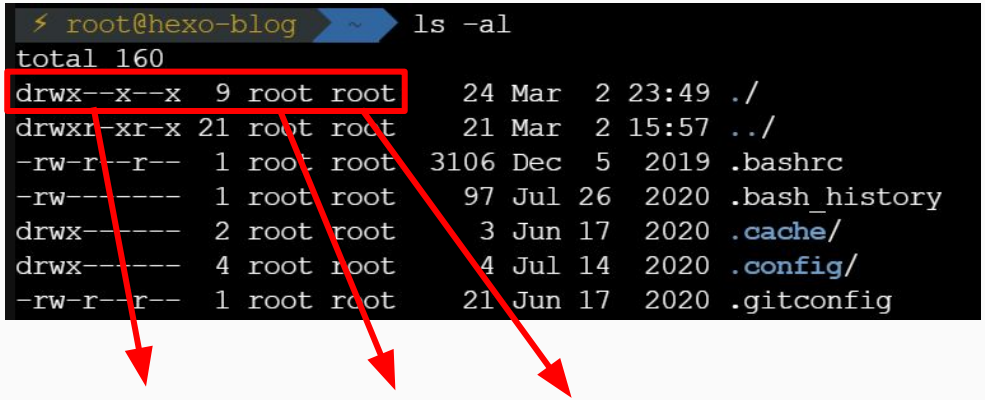
```
F74076310@F74076310:~$ last
F7407631 pts/0        140.116.252.158  Sun Mar  7 05:04   still logged in
F7407631 pts/1        140.116.252.158  Sun Mar  7 05:04 - 05:04  (00:00)
F7407631 pts/1        140.116.252.158  Sun Mar  7 04:47 - 05:04  (00:16)
F7407631 pts/1        140.116.252.158  Sun Mar  7 04:47 - 04:47  (00:00)
F7407631 pts/0        140.116.252.158  Sun Mar  7 03:00 - 05:04  (02:03)
F7407631 pts/0        111.254.5.141    Sat Mar  6 12:26 - 12:31  (00:04)
reboot   system boot  5.4.0-66-generic Sat Mar  6 10:43   still running
F7407631 ttyS0                         Sat Mar  6 10:14 - 10:14  (00:00)
reboot   system boot  5.4.0-66-generic Fri Mar  5 17:28 - 10:14  (16:46)
ncku-nas pts/0        172.26.4.252     Wed Feb 24 19:09 - 19:10  (00:01)
reboot   system boot  5.4.0-66-generic Wed Feb 24 19:08 - 19:10  (00:01)
ncku-nas pts/0        172.26.4.252     Wed Feb 24 18:52 - 18:52  (00:00)
reboot   system boot  5.4.0-66-generic Wed Feb 24 18:52 - 18:52  (00:00)
ncku-nas pts/0        172.26.4.252     Wed Feb 24 18:50 - 18:51  (00:01)
reboot   system boot  5.4.0-66-generic Wed Feb 24 18:48 - 18:51  (00:03)
ncku-nas pts/1        172.24.1.254     Wed Feb 24 18:25 - 18:29  (00:03)
reboot   system boot  5.4.0-65-generic Wed Feb 24 18:23 - 18:29  (00:05)

wtmp begins Wed Feb 24 18:23:46 2021
```

w, who, last, lastlog (cont.)

```
F74076310@F74076310:~$ lastlog
Username         Port     From             Latest
root                                       **Never logged in**
daemon                                     **Never logged in**
bin                                        **Never logged in**
sys                                        **Never logged in**
sync                                       **Never logged in**
games                                      **Never logged in**
man                                        **Never logged in**
lp                                         **Never logged in**
mail                                       **Never logged in**
news                                       **Never logged in**
uucp                                       **Never logged in**
proxy                                      **Never logged in**
www-data                                   **Never logged in**
backup                                     **Never logged in**
list                                       **Never logged in**
irc                                        **Never logged in**
gnats                                      **Never logged in**
nobody                                     **Never logged in**
systemd-network                            **Never logged in**
systemd-resolve                            **Never logged in**
systemd-timesync                           **Never logged in**
messagebus                                 **Never logged in**
syslog                                     **Never logged in**
_apt                                       **Never logged in**
tss                                        **Never logged in**
uuidd                                      **Never logged in**
tcpdump                                    **Never logged in**
sshd                                       **Never logged in**
landscape                                  **Never logged in**
pollinate                                  **Never logged in**
systemd-coredump                           **Never logged in**
ncku-nasa        pts/0    172.26.4.252     Wed Feb 24 19:09:01 +0000 2021
lxd                                        **Never logged in**
ubuntu                                     **Never logged in**
F74076310        pts/0    140.116.252.158  Sun Mar  7 05:04:41 +0000 2021
root                                       **Never logged in**
nobody                                     **Never logged in**
F74076310@F74076310:~$
```

Permission

Unix-like file permission

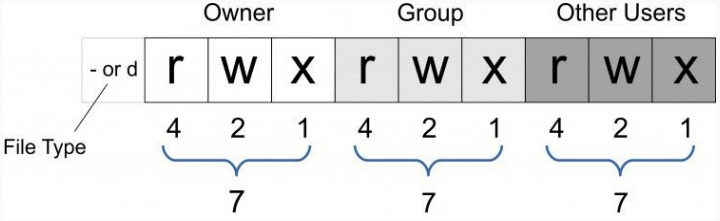


If you don’t have permission

(.bashrc don’t have execute(x) permission)

How to interpret permission

1. r: Who can read this file or list files in the directory
2. w: Who can write this file or add & remove something in the directory
3. x: Who can execute this file or enter this directory & sub directory



chmod & chown

- chmod: change mode
  - chmod 755 filename.txt
  - chmod a+x filename.txt
  - chmod -R 755 directoryname
- chown: change owner
  - chown user filename.txt
  - chown user:group filename.txt
  - chown -R user:group directoryname

Sometimes it won’t work

- chown www-data filename.txt
  - Permission denied
  - Normal user don’t have permission to change owner

- You need more permission to do that, but how?

Service Management

Service?

- There are lots of programs running in the background
  - These are “services”
- HTTP server / SQL server / DNS server...etc
- There’s a program in the system monitoring these services
  - Including boot process

How does your OS boot up?

- Load BIOS, select first bootable device
- Load MBR/EFI, load into boot loader
- Load initramfs and kernel, decompress initramfs, let kernel load drivers
- Then kernel would call “systemd” to start system initialization

- What’s systemd?

systemd

systemd is a suite of basic building blocks for a Linux system. It provides a system and service manager that runs as PID 1 and starts the rest of the system.

systemd provides aggressive parallelization capabilities, uses socket and D-Bus activation for starting services, offers on-demand starting of daemons, keeps track of processes using Linux control groups, maintains mount and automount points, and implements an elaborate transactional dependency-based service control logic. systemd supports SysV and LSB init scripts and works as a replacement for sysvinit.

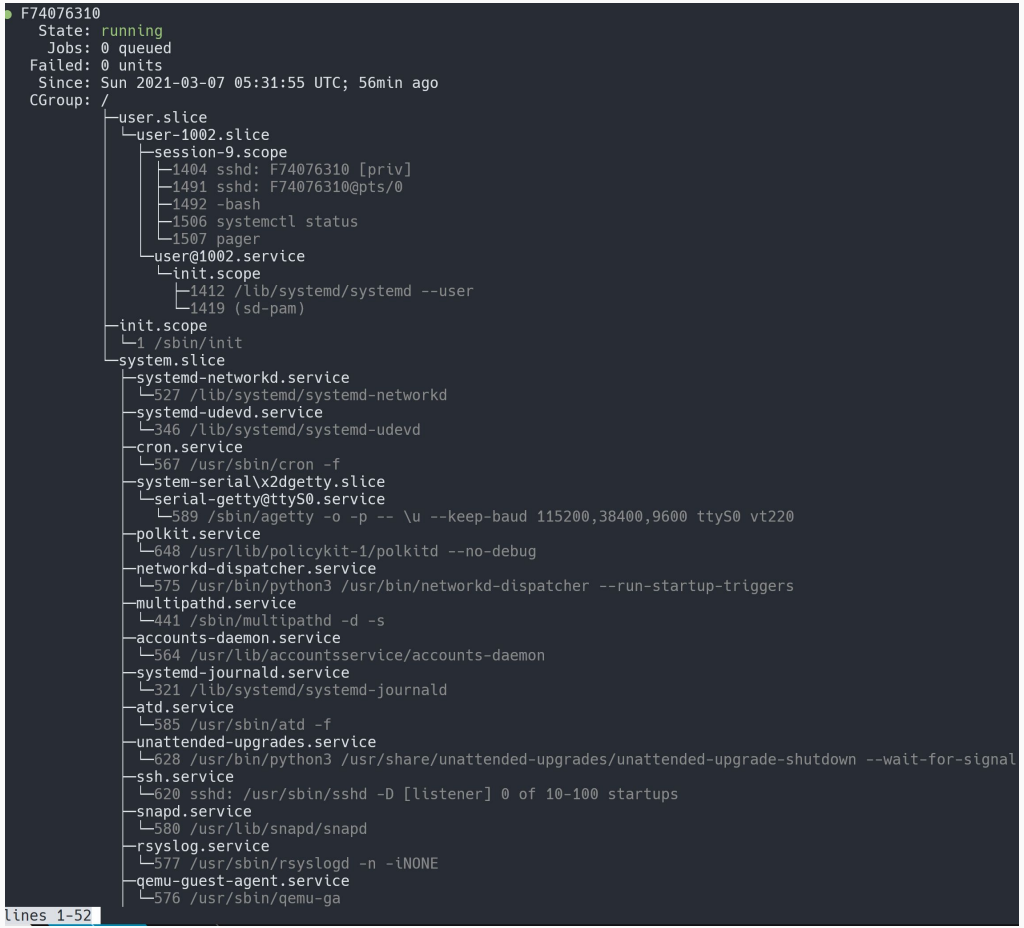
Other parts include a logging daemon, utilities to control basic system configuration like the hostname, date, locale, maintain a list of logged-in users and running containers and virtual machines, system accounts, runtime directories and settings, and daemons to manage simple network configuration, network time synchronization, log forwarding, and name resolution.

systemd initialization sequence

- systemd boots up the system according to default.target
- systemd calls sysinit.target & basic.target to initialize system
- systemd starts service under multi-user.target
- systemd run file /etc/rc.d/rc.local under multi-user.target
- systemd starts getty.target and login service under multi-user.target
- (systemd starts graphical services under multi-user.target)

Comparison of SysV Runlevels with systemd Targets

Runlevel	Target Units	Description
0	runlevel0.target poweroff.target	Shut down and power off the system
1	runlevel1.target, rescue.target	Set up a rescue shell
2	runlevel2.target multi-user.target	Set up a non-graphical multi-user system
3	runlevel3.target multi-user.target	Set up a non-graphical multi-user system
4	runlevel4.target multi-user.target	Set up a non-graphical multi-user system
5	runlevel5.target graphical.target	Set up a graphical multi-user system
6	runlevel6.target reboot.target	Shut down and reboot the system


F74076310
State: running
Jobs: 0 queued
Failed: 0 units
Since: Sun 2021-03-07 05:31:55 UTC; 56min ago
CGroup: /
├─user.slice
│ └─user-1002.slice
│ ├─session-9.scope
│ │ ├─1404 sshd: F74076310 [priv]
│ │ ├─1491 sshd: F74076310@pts/0
│ │ ├─1492 -bash
│ │ ├─1506 systemctl status
│ │ └─1507 pager
│ └─user@1002.service
│ └─init.scope
│ ├─1412 /lib/systemd/systemd --user
│ └─1419 (sd-pam)
├─init.scope
│ └─1 /sbin/init
└─system.slice
├─systemd-networkd.service
│ └─527 /lib/systemd/systemd-networkd
├─systemd-udevd.service
│ └─346 /lib/systemd/systemd-udevd
├─cron.service
│ └─567 /usr/sbin/cron -f
├─system-serial\x2dgetty.slice
│ └─serial-getty@ttyS0.service
│ └─589 /sbin/agetty -o -p -- \u --keep-baud 115200,38400,9600 ttyS0 vt220
├─polkit.service
│ └─648 /usr/lib/policykit-1/polkitd --no-debug
├─networkd-dispatcher.service
│ └─575 /usr/bin/python3 /usr/bin/networkd-dispatcher --run-startup-triggers
├─multipathd.service
│ └─441 /sbin/multipathd -d -s
├─accounts-daemon.service
│ └─564 /usr/lib/accountsservice/accounts-daemon
├─systemd-journald.service
│ └─321 /lib/systemd/systemd-journald
├─atd.service
│ └─585 /usr/sbin/atd -f
├─unattended-upgrades.service
│ └─628 /usr/bin/python3 /usr/share/unattended-upgrades/unattended-upgrade-shutdown --wait-for-signal
├─ssh.service
│ └─620 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
├─snapd.service
│ └─580 /usr/lib/snapd/snapd
├─rsyslog.service
│ └─577 /usr/sbin/rsyslogd -n -iNONE
├─qemu-guest-agent.service
│ └─576 /usr/sbin/qemu-ga
lines 1-52

Available systemd Unit Types

Unit Type	File Extension	Description
Service Unit	.service	A system service
Target Unit	.target	A group of systemd units
Automount Unit	.automount	A file system automount point
Device Unit	.device	A device file recognized by the kernel
Mount Unit	.mount	A file system mount point
Path Unit	.path	A file / directory in a file system
Scope Unit	.scope	An externally created process

Available systemd Unit Types

Unit Type	File Extension	Description
Slice Unit	.slice	A group of hierarchically organized units that manage system process
Snapshot Unit	.snapshot	A saved state of the systemd manager
Socket Unit	.socket	An IPC socket
Swap Unit	.swap	A swap device or a swap file
Timer Unit	.timer	A systemd timer

Systemd Unit Files Locations

Directory	Description
/usr/lib/systemd/system/	Systemd unit files distributed with installed DEB/RPM packages.
/run/systemd/system/	Systemd unit files created at run time. This directory takes precedence over the directory with installed service unit files.
/etc/systemd/system/	Systemd unit files created by systemctl enable as well as unit files added for extending a service. This directory takes precedence over the directory with runtime unit files.

systemd basic commands

- systemctl status
  - Show system status
- systemctl list-units
  - List running units
- systemctl --failed
  - List failed units
- systemctl list-unit-files
  - List installed unit files

systemd basic commands

- systemctl status *unit*
  - Check the status of unit
- systemctl is-enabled *unit*
  - Check whether a unit is enabled (Start on boot)
- systemctl start *unit*
- systemctl stop *unit*
- systemctl restart *unit*
- systemctl reload *unit*
- systemctl daemon-reload
  - Reload systemd manager configuration, scanning for new or changed units

systemd basic commands

- systemctl enable *unit*
  - Enable an unit (Start on boot)
- systemctl enable --now unit
  - Enable an unit and start the unit
- systemctl disable *unit*
- systemctl reenable *unit*
- systemctl mask *unit*
  - Mask a unit to make it impossible to start
- systemctl unmask *unit*

systemd basic commands

- systemctl halt
- systemctl poweroff
- systemctl reboot
- systemctl suspend
- systemctl hibernate
- systemctl hybrid-sleep

```
F74076310@F74076310:~$ cat /etc/systemd/system/test.service
[Unit]
Description=Test Service

[Service]
ExecStart=/bin/bash /usr/local/bin/run.sh

[Install]
WantedBy=multi-user.target
```

A basic systemd unit file

systemd unit file

- [Unit]
  - Description, Dependency...etc
- [Service], [Socket], [Timer], [Mount], [Path]..
  - Different kinds of unit uses different kinds of block here
- [Install]
  - Which target should this unit file be installed into

[Unit]

Option	Description
Description	A meaningful description of the unit. This text is displayed for example in the output of the systemctl status command.
Documentation	Provides a list of URIs referencing documentation for the unit.
After	Defines the order in which units are started. The unit starts only after the units specified in After are active. Unlike Requires, After does not explicitly activate the specified units. The Before option has the opposite functionality to After.
Requires	Configures dependencies on other units. The units listed in Requires are activated together with the unit. If any of the required units fail to start, the unit is not activated.
Wants	Configures weaker dependencies than Requires. If any of the listed units does not start successfully, it has no impact on the unit activation. This is the recommended way to establish custom unit dependencies.
Conflicts	Configures negative dependencies, an opposite to Requires.

[Service]

Option	Description
Type	• simple • forking • oneshot • dbus • notify • idle
ExecStart	Specifies commands or scripts to be executed when the unit is started
ExecStop	Specifies commands or scripts to be executed when the unit is stopped
ExecReload	Specifies commands or scripts to be executed when the unit is reloaded
Restart	With this option enabled, the service is restarted after its process exits, with the exception of a clean stop by the systemctl command.
RemainAfterExit	If set to True, the service is considered active even when all its processes exited. Default value is False. This option is especially useful if Type=oneshot is configured.

[Install]

Option	Description
Alias	Provides a space-separated list of additional names for the unit. Most systemctl commands, excluding systemctl enable, can use aliases instead of the actual unit name.
ReqiuredBy	A list of units that depend on the unit. When this unit is enabled, the units listed in RequiredBy gain a Require dependency on the unit.
WantedBy	A list of units that weakly depend on the unit. When this unit is enabled, the units listed in WantedBy gain a Want dependency on the unit.
Also	Specifies a list of units to be installed or uninstalled along with the unit.
DefaultInstance	Limited to instantiated units, this option specifies the default instance for which the unit is enabled

```
F74076310@F74076310:~$ cat run.sh
#!/bin/bash
echo The script ran at $(date) > /tmp/result
F74076310@F74076310:~$ cat test.service
[Unit]
Description=Test Service

[Service]
ExecStart=/bin/bash /usr/local/bin/run.sh

[Install]
WantedBy=multi-user.target
F74076310@F74076310:~$ sudo mv test.service /etc/systemd/system
system/      system.conf
F74076310@F74076310:~$ sudo mv test.service /etc/systemd/system
system/      system.conf
F74076310@F74076310:~$ sudo mv test.service /etc/systemd/system/
cloud-final.service.wants/              graphical.target.wants/                 rescue.target.wants/                    sockets.target.wants/
cloud-init.target.wants/                iscsi.service                           snap-core18-1988.mount                  sshd.service
dbus-org.freedesktop.resolve1.service   mdmonitor.service.wants/                snap-lxd-19188.mount                    sysinit.target.wants/
dbus-org.freedesktop.timesync1.service  multi-user.target.wants/                snap-snapd-11036.mount                  syslog.service
default.target.wants/                   multipath-tools.service                 snap-snapd-11107.mount                  timers.target.wants/
emergency.target.wants/                 network-online.target.wants/            snap.lxd.activate.service               vmtoolsd.service
final.target.wants/                     open-vm-tools.service.requires/         snap.lxd.daemon.service
getty.target.wants/                     paths.target.wants/                     snap.lxd.daemon.unix.socket
F74076310@F74076310:~$ sudo mv test.service /etc/systemd/system/
cloud-final.service.wants/              graphical.target.wants/                 rescue.target.wants/                    sockets.target.wants/
cloud-init.target.wants/                iscsi.service                           snap-core18-1988.mount                  sshd.service
dbus-org.freedesktop.resolve1.service   mdmonitor.service.wants/                snap-lxd-19188.mount                    sysinit.target.wants/
dbus-org.freedesktop.timesync1.service  multi-user.target.wants/                snap-snapd-11036.mount                  syslog.service
default.target.wants/                   multipath-tools.service                 snap-snapd-11107.mount                  timers.target.wants/
emergency.target.wants/                 network-online.target.wants/            snap.lxd.activate.service               vmtoolsd.service
final.target.wants/                     open-vm-tools.service.requires/         snap.lxd.daemon.service
getty.target.wants/                     paths.target.wants/                     snap.lxd.daemon.unix.socket
F74076310@F74076310:~$ sudo mv test.service /etc/systemd/system/test.service
F74076310@F74076310:~$ sudo chown root:root /etc/systemd/system/test.service
F74076310@F74076310:~$ sudo systemctl daemon-reload
sudo F74076310@F74076310:~$ sudo mv
```