

Linux Administration

The Root Account

- The administrator account on UNIX.
- Ubuntu Linux disables the root account by default.
 - You may enable root account by setting password for root.
 - You don't need to enable root account to be root.
- Initially, the account created during the set up can be elevated to root:

```
$ sudo -i
```

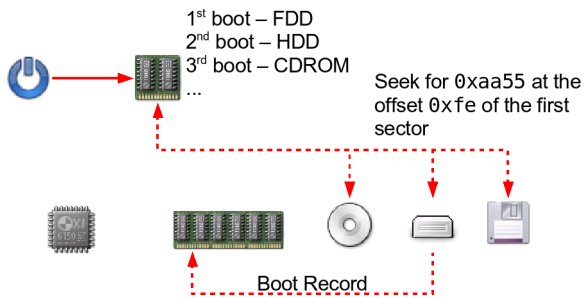
- Only accounts in `/etc/sudoers` can use `sudo`.

If you want to use the root account ...

- The root's password should be at least 8-character long.
 - With a good combination of uppercase, lowercase, numbers, symbols, etc.
- Do **NOT** login root remotely.

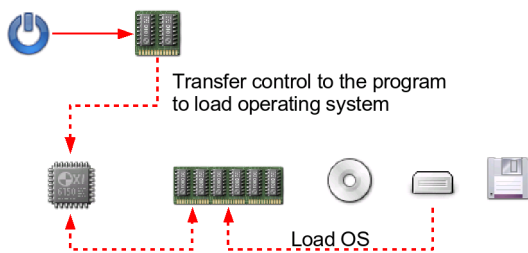
Boot Loader

- For a computer to start, it needs to have an initial program to run – a *bootstrap*.



Load an operating system

- The boot record knows how to load an operating system.
- The operating system starts the first process, i.e., *init*.



Grub

- A boot loader for Linux.
- Grub allows to change boot options temporarily at the grub menu.
 - Move the cursor to the menu you want, press e.
 - Move the cursor to the line to edit, press e.
 - Edit the options, press enter to accept, or ESC to cancel.
 - Press b to boot.
- To boot to any *runlevel*, append the runlevel number at the end of the line kernel . . .
- For permanent changes, edit `/boot/grub/menu.lst`

Logs

- dmesg – daily message.
- Log files:
 - /var/log/messages – main log
 - /var/log/kern.log – kernel log
 - /var/log/daemon.log – server software log
 - /var/log/mail.log – server software log
 - /var/log/syslog – system log

Kernel

- The core operating system.
- A program to manage system resources.
 - CPU time
 - Memory
 - File
 - Disk
 - I/O
 - etc. etc. etc.
- A kernel might only be the program that must be run all the time (as long as the system is on).
- All the drivers are implemented in the kernel.
 - They can be compiled to be a kernel modules.

(cont.)

- Kernel modules can literally load as a part of the kernel at anytime, allowing to create a small-footprint kernel.
 - They can also be unload from the kernel if nobody no longer use them.
- Kernel modules are in /lib/modules/<kernel>/.
- For Ubuntu, kernel modules can be configured in:
 - /etc/modules
 - /etc/modprobe.d/*

(cont.)

- To list the currently-using kernel modules
lsmod
- To load the kernel module immediately
modprobe <kernel-module>
- To unload the kernel module
modprobe -r <kernel-module>
- To show the info
modinfo /path/to/kernel-module-file

Services

- There are 7 runlevels for Linux/UNIX.
 - 0 – halt
 - 1 – single-user
 - 2 – multi-user, no network – default for Ubuntu
 - 3 – multi-user, network
 - 4 – undefined
 - 5 – X-Window
 - 6 – reboot
- To check runlevel
\$ runlevel
- To change runlevel immediately
telinit <level>

(cont.)

- Each runlevel can be configured to run different services.
- Startup scripts
 - Originals are in /etc/init.d/
 - Each runlevel has its own startup scripts in /etc/rc[0-6].d
 - Scripts to run in every runlevel are in /etc/rcS.d
- To configure services for a runlevel manipulates a symlink in /etc/rc[0-6].d
 - S??service = on
 - K??service = off

(cont.)

- To control the service directly
 - # /etc/init.d/<service> start
 - # /etc/init.d/<service> stop
 - # /etc/init.d/<service> restart
- Use /etc/rc.local to run program on all runlevels.

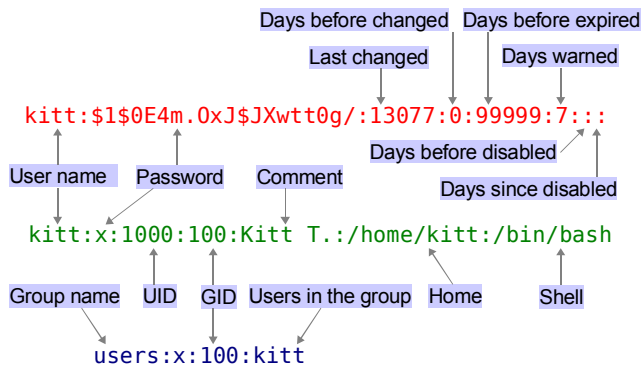
Halt/Reboot

- There are several ways to reboot
 - # reboot
 - # shutdown -r now
- And, there are several ways to halt
 - # halt
 - # shutdown -h now
- Also, a tricky way
 - # kill -9 <pid of init>

Users

- Mainly, account information is in two files:
 - /etc/passwd – general information
 - /etc/shadow – password information
- The account with the password of * or ! will not be able to login.
- All time recorded in /etc/shadow is expressed in the unit of days starting from the UNIX epoch.
 - 1 January 1970 00.00 GMT
- Group information is in
 - /etc/group
 - /etc/gshadow

(cont.)



(cont.)

- Add a new user
useradd [-c "comment"] [-d home] [-g group] [-s shell] <username>
- Modify a user
usermod [-c "comment"] [-d home] [-g group] [-s shell] <username>
- Delete a user
userdel [-r] <username>
- There are more options, see the manpages.

(cont.)

- Add a new group
groupadd [-g gid] <group-name>
- Modify a group
groupmod [-g gid] <group-name>
- Delete a group
groupdel <group-name>

(cont.)

- To prevent a user to login
 - Change password in `/etc/shadow` to `*` or `!`.
 - Change shell to `/sbin/nologin`.

File Systems

- A file system defines methods to manage and store files within a partition.
 - A disk can be divided to multiple partitions, or
 - A partition may be a set of disks.
- Linux provides several types of file systems.
 - `ext2`, `ext3`, `ext4`, `jfs`, `xf`s, `ntfs`, `fat/vfat`, `iso9660`, `hpfs`, `ufs`, `reiserfs`, ...
- Most Linux distributions use `ext3` by default.
 - It is slow, but very stable.

Disk management

- Disk devices
 - SCSI/SATA: `/dev/sd?`
 - IDE/ATA
 - `/dev/hda` – primary master
 - `/dev/hdb` – primary slave
 - `/dev/hdc` – secondary master
 - `/dev/hdd` – secondary slave
- Partition devices
 - A disk device + partition number

(cont.)

- A disk may have at most 4 partitions
 - The partition number for primary or extended = 1 – 4
- An extended partition can be divided to one or more logical partition
 - The partition number for logical partitions = 5+

(cont.)

- To partition a disk
 - # fdisk <disk-device>
- Main menu
 - m – help
 - q – quit
 - p – print the partition table
 - w – write the partition table
 - n – create a new partition
 - d – delete a partition
 - t – change the file system of a partition
 - a – toggle the bootable flag

(cont.)

- Create a new partition
 - Press n
 - Choose partition type
 - p – primary
 - e – extended
 - l – logical
 - Specify starting point of the partition
 - Specify ending point or size of the partition
- Delete a partition
 - Press d
 - Specify the partition to delete

(cont.)

- Change the file system of a partition
 - Press t
 - Choose the partition to be changed
 - Specify file system
 - Press L to see the list of file systems
- Change the bootable flag
 - Press a
 - Choose the partition to be changed

(cont.)

- Format a partition
 - The 2nd extended file system (ext2)
mke2fs <partition-device>
 - The 3rd extended file system (ext3)
mke2fs -j <partition-device>
 - Swap file system
mkswap <partition-device>
 - JFS
jfs_mkfs <partition-device>

(cont.)

- Mount a partition
mount <partition-device> <directory>
- Unmount a partition
umount <partition-device>
- Modify /etc/fstab to mount a partition automatically at boot time

/dev/hda3 /usr2 ext3 defaults 0 2

(cont.)

- Checking the file system
 - Unmount the file system before check
 - # fsck <partition-device>
 - # fsck -v <partition-device>
 - # fsck -cv <partition-device>
 - # fsck -ccv <partition-device>
- Check disk space
 - # df
- Check file/directory space
 - # du [directory]
 - # du -hs [directory]

Process Management

- Process information includes
 - Process ID (pid)
 - Parent PID (ppid)
 - UID
 - Nice
 - -20 to 19
 - Default = 0
 - Less nice, more CPU time
 - Status
 - Runnable
 - Sleeping
 - Zombie
 - Stopped

(cont.)

- uptime – system uptime and workload
- w – show users and running process
- ps – show process information
 - \$ ps -aux
 - \$ ps -lax
- top – monitor processes
 - \$ ps -aux
 - \$ ps -lax

(cont.)

- nice – run a program at specified nice
 - User can run at nice of 0 to 19
 - Root can run at nice of -20 to 19
 - # nice -n 10 <program>
- renice – change process's nice
 - Process owner can only increase the nice
 - # renice 5 <pid>

(cont.)

- kill – send killing signal to a process
 - # kill [-signal] <pid>

 - HUP (1) hangup / reset the process
 - INT (2) interrupt
 - QUIT (3) terminate + coredump
 - KILL (9) kill
 - TERM (15) terminate (default)
- killall – kill all the processes of a program
 - # killall [-signal] <program>

Networking

- Network interface configuration
/etc/network/interfaces
- Static IP address

```
auto eth0
iface eth0 inet static
address <IP address>
netmask <netmask>
gateway <default gateway>
```
- DHCP

```
auto eth0
iface eth0 inet dhcp
```

(cont.)

- ifup – start an interface
 - # ifup <network-interface>
- ifdown – stop an interface
 - # ifdown <network-interface>
- ifconfig – show/configure an interface
 - # ifconfig [network-interface [ip-address] [options]]
 - netmask <netmask>
 - broadcast <broadcast address>
 - up/down

(cont.)

- route – show/set routing table
 - # route
 - # route add default gw <gateway address>
 - # route add -net <network address> netmask <netmask> gw <gateway address>
 - # route add -host <host address> gw <gateway address>
 - # route del -net <network address> netmask <netmask>
 - # route del -host <host address>

(cont.)

- DNS servers
 - /etc/resolv.conf
 - search kitty.in.th
 - nameserver 202.28.48.130
 - nameserver 202.28.93.1
 - nameserver 202.28.93.13

(cont.)

- dig – resolve FQDN/IP address
 - \$ dig <domain-name>
 - \$ dig <FQDN>
 - \$ dig -x <ip-address>
- netstat – show network status/statistic
 - \$ netstat -rn
 - \$ netstat -ant
 - \$ netstat -anu
 - \$ netstat -i
- arp – show ARP table
 - \$ arp -a

(cont.)

- ping – a simple ICMP echo request/reply test
 - \$ ping <host or IP address>
- traceroute – trace path to a host
 - \$ traceroute <host or IP address>
- mtr – a better trace tool
 - \$ mtr <host or IP address>
- tcpdump – dump the traffic
 - # tcpdump -nvvi eth0
 - # tcpdump host <host or IP address>
- iptraf – IP traffic monitoring
 - # iptraf

(cont.)

- whois – search whois database
 - \$ whois <IP-address>

Backup/Archive

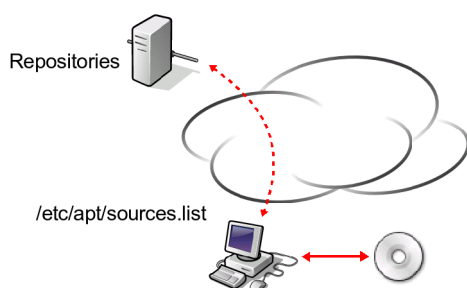
- Create a disk image
dd if=<disk device> of=<file>
- Restore from a disk image
dd if=<file> of=<disk device>
- Create a partition image
dd if=<partition device> of=<file>
- Restore from a partition image
dd if=<file> of=<partition device>

(cont.)

- Create an archive
\$ tar cvzf <file.tar.gz> <directories>
\$ tar cvjf <file.tar.bz2> <directories>
- Restore from an archive
\$ tar xvzf <file.tar.gz>
\$ tar xvjf <file.tar.bz2>

Advanced Package Tool (APT)

- The best package management system.
- Automatically install dependencies.



(cont.)

- Search package
aptitude search <keyword>
- Show package information
aptitude show <package>
- Install package
aptitude install <package>
- Remove package
aptitude purge <package>
aptitude -purge-unused purge <package>

(cont.)

- Update package information
aptitude update
- Upgrade installed packages
aptitude upgrade
- Upgrade the distribution
aptitude dist-upgrade
- Clean package files
aptitude clean
- Clean unused package files
aptitude autoclean

Debian Package Manager

- Install package
dpkg -i <package file .deb>
- Show information of the .deb
\$ dpkg -I <package file .deb>
- List packages
\$ dpkg -l
- Package status
\$ dpkg -s <package>
- List installed file
\$ dpkg -L <package>

Build and install software from source

- Most pieces of software are developed under the framework of GNU autotools

```
$ tar xzf source.tar.gz
$ cd source/
$ ./configure
$ make
$ sudo make install
```
- If it does not compile, see the documents
 - README, INSTALL, NEWS, ChangeLog
 - ./configure --help

(cont.)

- Patches might be required

```
$ cat <patch-file> | patch -p1
$ zcat <patch-file.gz> | patch -p1
$ bzcat <patch-file.bz2> | patch -p1
```

Build and Install a New Kernel

- The latest kernel is available at <http://kernel.org/>
- To build a kernel

```
# make menuconfig
# make
# make modules_install
# make install
# update-grub
```