

# System Administration

- Based on

- book chapter 14

# System Administration

## ■ Becoming super-user

- user id (0)
- be aware....
- login as root
- use **su** utility
- use **sudo**

# System Administration

- When you start a Linux system, the following sequence of events occurs:
  - The hardware performs diagnostic self-tests.
  - The Linux kernel is loaded from the /boot directory on the root device.
  - The kernel starts running and initializes itself.
  - The kernel starts init, the first user-mode process.

# System Administration

- Run levels are defined in `"/etc/inittab."` The commonly predefined run levels are:
  - 0 halt
  - 1 single-user (i.e., maintenance)
  - 2 multi-user without networking
  - 3 server (full multi-user)
  - 5 workstation (full multi-user, graphical login)
  - 6 reboot

# System Administration

## ■ Stopping Linux

■ Utility: `shutdown -hkr time [ message ]`

■ shutdown time must be specified in one of three ways:

■ `now`: the system is shut down immediately.

■ `+minutes`: the system is shut down in the specified number of minutes.

■ `hours:minutes`: the system is shut down at the specified time (24-hour format).

■ The specified warning message (or a default one if none is specified) is displayed periodically as the time of shutdown approaches. Logins are disabled five minutes prior to shutdown.

# System Administration

- Utility: `shutdown -hkr time [ message ]`
- If neither `-h` nor `-k` is specified, `shutdown` brings the system down to single-user mode by signaling `init` to change to run level 1.
- Using the `-h` option causes `shutdown` to change the run level to 0 (halt). The `-r` option causes `shutdown` to change the run level to 6 (reboot).
- The `-k` option is funny; it causes `shutdown` to behave as if were going to shut down the system, but when the shutdown time arrives, it does nothing.
- The "k" stands for "just kidding"!

# File System Integrity

- Utility: `fsck -p [ fileSystem ]*`
- `fsck` (file system check) scans the specified file systems and checks them for consistency. The kind of consistency errors that can exist include:
  - A block is marked as free in the bitmap but is also referenced from an inode.
  - A block is marked as used in the bitmap but is never referenced from an inode.
  - More than one inode refers to the same block.
  - An invalid block number.
  - An inode's link count is incorrect.
  - A used inode is not referenced from any directory.

# Disk Space

■ Utility: `df [ fileSystem ]*`

- `df` (disk free) displays a table of used and available disk space on the specified mounted file systems.
- If no file system is specified, all mounted file systems are described.



# Disk Space

- Utility: `du [-h] [-s] [ fileName ]*`
  - `du` displays the number of kilobytes that are allocated to each of the specified filenames.
  - If a filename refers to a directory, its files are recursively described.
  - When the `-h` option is used, the numeric values are changed to more human-readable values (i.e., 63844 blocks is displayed as 63M).
  - When used with the `-s` option, `du` displays only the grand total (summary) for each file or directory.
  - If no filenames are specified, the current directory is scanned.

# System Administration

## ■ Creating new file system

- 1. Partition and format the drive if necessary
- 2. Create a file system on the medium.
- 3. Mount the disk into the file system hierarchy.

# System Administration

- Utility: `mkfs -t type specialFile [ sectorCount ]`
  - `mkfs` creates a new file system on the specified special file.
  - A new file system consists of a superblock, an inode list, a root directory, and a "lost+found" directory.
  - The file system is built to be `sectorCount` sectors in size.
  - Only a super-user can use this command.
- As with `fsck`, `mkfs` is really a front-end to a specific file system creator for each specific file system supported (e.g., `mke2fs`), determined by the file system type specified by `type`.

# System Administration

## ■ User accounts

- Add a new entry to the password file (and shadow password file if in use).
  - what is this “shadow password business”?
- Add a new entry to the group file.
- Create a home directory for the user.
- Provide the user with some appropriate startup files.

# System Administration

## ■ User accounts

- Utility: `useradd [ -d directory ] [ -s shell ] userName`
  - `useradd` creates a new user on the system called `userName`.
  - When `-d` is specified, make the new user's home directory.
  - If `-s` is specified, make the new user's login shell.
  - The next available UID is assigned to the new user.

# System Administration

- User accounts

- Utility: `userdel [ -r ] userName`

- `userdel` removes a user account from the system.

- If `-r` is specified, the user's home directory is removed as well.

# System Administration

## ■ Password File

- located in `/etc/passwd`

- format:

- `username:password:userId:groupId:personal:homedir:startup`

# System Administration

**Figure 14-9. Fields in the Linux password file.**

Field	Meaning
username	The user's login name.
password	The encrypted version of the user's password or "x" if a shadow password file is in use.
userId	The unique integer allocated to the user.
groupId	The integer corresponding to the user's group.
personal	The description of the user that is displayed by the <b>finger</b> utility.
homedir	The home directory of the user.
startup	The program that is run for the user at login.



# System Administration

## ■ The Group File

- To add a new user, you must decide which group the user will belong to and then search the group file to find the associated group ID.
- Every group in the system has an entry in the group file ("/etc/group") in the format:
  - `groupname:groupPassword:groupId:users`

# System Administration

**Figure 14-10. Fields in the Linux group file.**

Field	Meaning
groupname	The name of the group.
groupPassword	The encrypted password for the group.
groupId	The unique integer corresponding to the group.
users	A list of the users in the group, separated by commas.

# System Administration

- Installing new software

- Utility: `rpm -aiqU packageName`

- `rpm` installs or updates an RPM package in the file `packageName` depending on whether `-i` or `-U` is specified.
    - If the `-q` option is specified, the system is queried to see if `packageName` is currently installed.
    - If `-q` is used with `-a`, all packages installed on the system are listed.

# System Administration

## ■ Peripheral devices

- The basic steps of device installation are as follows:
- 1. Install the device driver if it isn't currently in the kernel and loadable device drivers are not used.
- 2. Determine the device's major and minor numbers.
- 3. Use `mknod` to associate a filename in `"/dev"` with the new device.

# System Administration

- Utility: **mknod** fileName [ c ] [ b ] majorNumber  
minorNumber
  - `mknod fileName p`
  - `mknod` creates the special file `fileName` in the file system.
  - The first form of `mknod` allows a super-user to create either a character-oriented or block-oriented special file with the specified major and minor numbers.
  - The **major number** identifies the class of the device, and the **minor number** identifies the instance of the device.
  - The second form of `mknod` creates a named pipe, and may be used by anyone (or the `mkfifo` command may also be used).

# Network Interface

- Information about the network interface is given by the **ifconfig** command
- ifconfig can be used to manage network, e.g.
  - `ifconfig eth0 194.27.1.14 up`