Lecture 12

Linux System Administration
Booting

- PROM (BIOS) — perform basic self-test and access parameters from *nvram* (CMOS)
- OS Loader — locate and run kernel on disk
  - Located in the MBR (first sector of boot device)
  - May call secondary loader on some partition
  - LILO, GRUB
- Kernel — initializes devices, mounts root filesystem, starts first user process (init)
init

- init — reads /etc/inittab to determine what to start according to the run-level (initdefault)

<table>
<thead>
<tr>
<th>run-level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Halt</td>
</tr>
<tr>
<td>1</td>
<td>Single user mode</td>
</tr>
<tr>
<td>2</td>
<td>Multiuser, w/o NFS</td>
</tr>
<tr>
<td>3</td>
<td>Full multiuser mode</td>
</tr>
<tr>
<td>4</td>
<td>unused</td>
</tr>
<tr>
<td>5</td>
<td>X11</td>
</tr>
<tr>
<td>6</td>
<td>reboot</td>
</tr>
</tbody>
</table>
Boot Scripts

• /etc/init.d contains scripts for every managed service, e.g.
  /etc/init.d/sshd {start|stop}
• Links to these boot scripts are created in the sequencing directories /etc/rc[0-6].d
• Links started with S are called with start
• Links started with K are called with stop
Boot Scripts (cont.)

• Numbers in link determine the order the script are run, e.g.
  – S55sshd runs before S80sendmail but after S08iptables
• Maintain runlevel information for system services by manipulating files in /etc/rc[0–6].d or use chkconfig
Internet Services Daemon

- `xinetd` — listens to service ports and starts server when a request arrives
  - No need to start all the daemons at boot time
  - “Super-server”
- Services are configured in `/etc/xinetd.conf` or in individual files under `/etc/xinetd.d`
Shutting Down

- `shutdown` brings the system down safely:
  
  `/sbin/shutdown -t 600 -r "... be right back"`

- Processes are sent SIGTERM and then SIGKILL

- `halt` same as `shutdown -h`

- `reboot` same as `shutdown -r`

- `poweroff` turns off the power after halting (same as `halt -p`
User Account Management

• Local user info stored in /etc/passwd

• To create a new local user:
  1. Add new entry to /etc/passwd and /etc/shadow (and /etc/group is necessary)
  2. Create home directory for the new user with some default startup files

• Do these manually or use useradd:

  useradd -c "Bill Gates" -u 1001 -g msoft -d /home/billg -m -k /etc/skel -s /bin/bash billg
User Acct. Management (cont.)

- To delete an account:
  userdel -r billg

- To create a group:
  groupadd -g 550 web

- To delete a group:
  groupdel web
/etc/passwd

• Format of a **passwd** entry:

```
```

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
nobody:x:99:99:Nobody:/sbin/nologin
billg:x:1001:501:Bill Gates:/home/billg:/bin/bash
```
Format of a shadow entry:
username:password:lstchg:min:max:warn:inact:exp:

root:j3dghRBqe$2fjvGJ8js:12650:0:99999:7:::
bin:*:12650:0:99999:7:::
...

* does not match any password
!! account is locked
The shadow file should only be readable by root
Groups

• Format of a group entry in /etc/group
  
  groupname:password:gid:user_list

  root::0:root
  bin::1:root,bin,daemon
  senate::990:chuck,hillary

• Group passwords can be stored in /etc/gshadow

• If you belong to more than 1 groups, you can change your group with:
  
  newgrp [ group ]
Become Another User

- **su** - run shell as another user
  - Need password of the user you are su’ing to
  - No username specified means root
- **sudo** - execute command as another user
  - Authenticate with your own password
  - Run command as root by default
  - sudo privileges are defined in /etc/sudoers
Installation

- Install from CD/DVDs interactively
- Network automated installation
  - Kickstart (Red Hat)
  - Jumpstart (Solaris)
- Packages and machine configuration files located on install server
- Install a machine with a single command
  \[
  \text{linux } \text{ks=nfs:server:/path (RH Linux)}
  \]
  \[
  \text{boot net - install (Solaris)}
  \]
Disk Partition

- A *partition* is a logical section of a disk, normally with its own filesystem.
- The *partition table* contains the partition information (starting block, size, type).
- A disk can be partitioned during OS installation or (for non-system disks) afterwards using *fdisk* or *parted*. 
A Partition Table

(parted) print

Disk geometry for /dev/hda: 0.000-38146.972 megabytes

Disk label type: msdos

<table>
<thead>
<tr>
<th>Minor</th>
<th>Start</th>
<th>End</th>
<th>Type</th>
<th>Filesystem</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.031</td>
<td>25603.593</td>
<td>primary</td>
<td>ntfs</td>
<td>boot</td>
</tr>
<tr>
<td>2</td>
<td>25603.594</td>
<td>25705.568</td>
<td>primary</td>
<td>ext3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>25705.569</td>
<td>26733.164</td>
<td>primary</td>
<td>linux-swap</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>26733.164</td>
<td>38146.530</td>
<td>extended</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>26733.195</td>
<td>38146.530</td>
<td>logical</td>
<td>ext3</td>
<td></td>
</tr>
</tbody>
</table>
Filesystems

• Different filesystem types organize files and directories in different ways
• Ext3 — most common filesystem on Linux
• Ext3 is a journaling filesystem
  – Sequence of changes to filesystem treated as single transaction
• After unclean system shutdown
  – Replay journal to make filesystem consistent
  – No need to fsck
Mounting Filesystems

/etc/fstab:

```
LABEL=/       /     ext3    defaults 1 1
LABEL=/boot   /boot  ext3    defaults 1 2
none           /proc  proc    defaults 0 0
/dev/sda2      swap    swap    defaults 0 0
```

- `mount -a` causes all fs in `fstab` to be mounted
- To manually mount a filesystem not in `fstab`
  `mount -t ext3 -o ro,acl /dev/sda5 /a`
- To check filesystem usage, use `df`, e.g.
  `df /usr`
Access Control Lists (ACL)

• Traditionally, file permissions can only be set for user, group, and everyone
  – Different perms cannot be used for different users
• ACL provides finer access control
• Filesystems need to be mounted with the acl option
Setting ACL

- To give Prof. Korn `rw` access to your file that has permission 600:
  
  `setfacl -m u:kornj:rw somefile`

- To remove all permission for Prof. Korn:
  
  `setfacl -x u:kornj somefile`

- To list the ACL for a file/directory:
  
  `getfacl somefile`
Quota

- Prevent one user from using up the whole disk
- Disk quota can be configured for individual users as well as groups
- To enable quota on a filesystem, mount with `usrquota` and/or `grpquota` options
Setting Disk Quota

• To list quota for user or group:
  
  ```
  quota user or quota -g group
  ```

  Disk quotas for user foo (uid: 501):
  
  Filesystem blocks soft hard inodes soft hard
  /dev/sdb2  223652 512000 600000 23456 0 0

• To configure quota for user:
  
  ```
  edquota user
  ```

• User can exceed soft limit for a grace period

• To configure quota for group:
  
  ```
  edquota -g group
  ```
Swap

• Swap space — area on disk for transferring pages to/from physical memory (RAM)
• When RAM is (almost) full, RAM pages are saved to swap by the *page daemon*
• Can be a dedicated partition or a swap file
• Usually twice the size of RAM
  – e.g. 2048 MB swap for 1024 MB RAM
**RAID**

- **Redundant Array of Independent Disks**
  - Combine multiple smaller physical disks into one big logical disk: OS sees one big drive
  - Improve I/O performance and provide redundancy

- **Most common RAID levels**
  - Linear: concatenation
  - RAID 0: striping - no redundancy
  - RAID 1: mirroring
  - RAID 5: striping with distributed-parity (XOR)
  - RAID 6: P + Q redundancy - up to 2 disk failure
**RAID Level 5**

<table>
<thead>
<tr>
<th>Disk 1</th>
<th>Disk 2</th>
<th>Disk 3</th>
<th>Disk 4</th>
<th>Disk 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>P</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>P</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>P</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>15</td>
<td>P</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>P</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>P</td>
</tr>
</tbody>
</table>

Left-symmetric
Hardware vs. Software RAID

- **Hardware RAID**
  - RAID controller handles everything
  - Host sees one big drive
- **Software RAID**
  - Kernel handles all RAID issues (MD driver)
  - Cheaper but lower performance
  - See `md(4), mdadm(8)`
Network Configuration

• Ethernet devices are named eth0, eth1, etc.
• To statically configure a network interface:
  – IP address (128.122.20.123)
  – Netmask (defines subnet) (255.255.255.0)
  – Router (gateway) address (128.122.20.1)
• `ifconfig` is used at boot time to configure network interfaces
  – List configuration if no argument is given
DHCP

- Dynamic Host Configuration Protocol
- Dynamically allocate IP addresses to clients
- Addresses are *leased* for a certain period
- Some older clients use BOOTP
Network File System (NFS)

- Developed by Sun Microsystems
- Allowed remote filesystems to be mounted locally
  - e.g. home directory mounted on machines
- To mount a filesystem from a NFS server
  
mount -t nfs -o nosuid,intr serv1:/export/local /usr/local
NFS (cont.)

NFS client

/  

usr  

local

NFS server (serv1)

/  

export  

local

bin
lib
share

mount point for serv1:/export/local
Naming and Directory Services

- Original UNIX naming system stores info in /etc
  - Does not scale well for large network
- Network naming services
  - Information stored centrally (client-server model)
  - Usernames, passwords, hostnames/IP addr, etc.
  - *Binds* names to objects
  - *Resolves* names to objects
    - e.g. www.cs.nyu.edu is 128.122.80.245
  - DNS, NIS, LDAP
Domain Name System

• Distributed, replicated service for translating hostnames to IP addresses
• Namespace divided into hierarchy of domains
• Each DNS domain supported by 2 or more name servers
DNS Client

• The *resolver* (e.g. `gethostbyname()`) on the client queries the name server
• DNS servers in `/etc/resolv.conf`, e.g. `nameserver 128.122.128.2`
• Query DNS server interactively with `nslookup` or `dig`
Network Information Service

• Developed by Sun Microsystems - originally Yellow Pages (yp)
• Stores network, hostnames-addresses, users, and network services info in NIS maps
  – e.g. passwd.byname, passwd.byuid, hosts.byname, ethers.byaddr, netgroup, etc.
• Client-server model
• Servers are replicated (master/slave)
• NIS+ — similar to NIS, but more features and more secure
LDAP

- Lightweight Directory Access Protocol
- Specialized database optimized for reading and searching
- What can be stored in LDAP?
  - Passwords, phone numbers, date-of-birth, jpeg photos,
- Client-server model (again)
- LDAP directory service is global
- OpenLDAP is an open source implementation
LDAP Information Model

- A LDAP *entry* is a collection of *attributes* with a unique *Distinguished Name* (DN)
  
  uid=jane,ou=People,dc=cims,dc=nyu,dc=edu

- Each attribute has a *type* and one or more *values*
  
  telephoneNumber: 212-995-1234

- The values of the *objectClass* attributes decide what attributes are required/allowed
  
  objectClass: posixAccount

- objectClasses are defined in *schema*
Directory Information Tree

- Entries are arranged in a hierarchical structure
Access to information is controlled by an access control list, e.g. password hashes are not available through anonymous bind.
Name Service Switch

- Controls how a machine obtains network information, such as passwd, group, aliases, hosts, netmasks, etc.
- Config file: /etc/nsswitch.conf
- Sample entries:
  
  `passwd:    files  ldap
  hosts:      files  ldap  dns
  netmasks:   files`
Controlling Access to Services

- Firewall
  - Packet filtering
  - Software vs. hardware
- TCP Wrapper (IP address)
- Application
  - Host-based (IP address, certificates)
  - User-based (Password)
- Don’t start the daemons
Software Firewall (iptables)

• Configure tables of packet-filter rules in Linux kernel
• Each table has a number of chains
• Each chain consists of a list of rules
• Each rule specifies what to do with a matching packet
• The default table (filter) has 3 built-in chains:
  – INPUT        incoming packets
  – FORWARD      routed packets
  – OUTPUT       outgoing packets
• Rules activated at boot time is defined in /etc/sysconfig/iptables

• Sample iptables entry:

  -A INPUT -m state --state NEW -m tcp -p tcp -s 192.168.1.0/24 --d port 137 -j ACCEPT

→ Allows new TCP connections from hosts in the 192.168.1.0/24 network to port 137
TCP Wrapper

- `tcpd` logs and controls incoming requests for services such as telnet, finger, rsh, etc.
- `inetd` runs `tcpd` instead
- `tcpd` logs connection and checks if connection is allowed based on `hosts.allow` and `hosts.deny`
- `/etc/hosts.allow:`
  - `in.telnetd: .cs.nyu.edu,.cs.cuny.edu`
- `/etc/hosts.deny:`
  - `ALL: ALL`
PAM

- **Pluggable Authentication Module**
- Centralized authentication mechanism
- “Plug in” different authentication methods
- Different services can have different authentication policies
- Highly secure systems can require multiple passwords to authenticate
PAM Framework

Applications
- ftp
- login
- ssh

PAM Library
- conf
- libpam

Modules
- pam_unix
- pam_ldap
- pam_securetty
PAM Stack

- Modules are *stacked* (order is important)
- Sample PAM configuration in `/etc/pam.d`:

  
<table>
<thead>
<tr>
<th>interface</th>
<th>control flag</th>
<th>module name</th>
</tr>
</thead>
<tbody>
<tr>
<td>auth</td>
<td>required</td>
<td>pam_nologin.so</td>
</tr>
<tr>
<td>auth</td>
<td>required</td>
<td>pam_securetty.so</td>
</tr>
<tr>
<td>auth</td>
<td>sufficient</td>
<td>pam_unix.so</td>
</tr>
<tr>
<td>auth</td>
<td>required</td>
<td>pam_ldap.so</td>
</tr>
</tbody>
</table>
Date, Time, and NTP

• Date sets the system date and time:
  date MMDDhhmm[[CC]YY][.ss]

• Some applications can fail if clocks are not synchronized among machines, e.g. make

• Use Network Time Protocol (NTP)
  – A stratum 1 server is connected to a reference clock
  – Stratum 2 servers synchronize with stratum1 servers
  – Your machine synchronized with stratum 2+ servers

• Daemon: ntpd  Config file: /etc/ntp.conf
Mail

- Mail Transfer Agent (MTA)
  - Sendmail
  - Postfix
  - Qmail

- Incoming mail are deposited into `/var/mail` or forwarded to another address according to the aliases (`/etc/aliases`) or user’s `.forward`
Spam Control

- Spam filters in MTA or MUA
- Authentication
  - Microsoft’s Sender-ID
    - Outgoing mail servers for each domain published in DNS
    - Incoming mail checked against the list
  - Yahoo’s DomainKeys
    - Header contains signature of message
    - Recipient looks up sender’s published validation key in DNS and checks signature
- Legislation
Spam Filters

• Rule-based
  – Rules (mostly regex) for matching message
  – A match increases/decreases the score
  – Total score exceeding threshold ➔ SPAM!
  – *SpamAssassin*

• Whitelist

• Realtime blacklist

• Bayesian filters (statistical model)
System Logging

• **syslogd** - system logging daemon
• System log messages are normally written to files in `/var/log`
• Rules for logging are specified in `/etc/syslog.conf` in the form of
  ```
  facility.priority  action
  ```
  – **Facility**: auth, daemon, kern, mail, etc.
  – **Priority**: info, warning, crit, emerg, etc.
  – **Action**: usually a file, "*" (everyone logged in)
Scheduling Tasks

• Use `crontab` and `at` to schedule tasks to be executed automatically (`crond`, `atd`)
• *Cron* jobs are repeated at specific intervals
  – e.g. everyday at 3:15pm
• *At* jobs are executed once
  – e.g. tomorrow at midnight
crontab

• Edit the `crontab` file with `crontab -e`
  – Uses editor in the `EDITOR` environment variable
• Each line consists of the schedule and the command to execute
  – Empty lines and lines starting with `#` are ignored
    `min hr day-of-month month day-of-week`
    `5 13,19 * * 1-5 mail -s “Time to eat” me@cs < /dev/null`
• List your cron jobs with `crontab -l`
at

# at 0830 Dec 20
ps -ef > proc.list
<EOT>

• Flexible time and operand presentation
  at 12pm + 1 week
  at noon next week

• **atq**: displays scheduled jobs
• **atrm job#**: removes job from queue
## Package Management

<table>
<thead>
<tr>
<th>Package Manager</th>
<th>Red Hat</th>
<th>Debian</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package file suffix</td>
<td>.rpm</td>
<td>.deb</td>
</tr>
<tr>
<td>Primary tool</td>
<td>rpm</td>
<td>dpkg</td>
</tr>
<tr>
<td>Other tools</td>
<td></td>
<td>dselect app-get</td>
</tr>
</tbody>
</table>
rpm/dpkg Examples

- List all packages:
  rpm -qa
dpkg --list

- Install a new package:
  rpm -ivh
dpkg --install

- Remove a package:
  rpm -e
dpkg --remove
Backup

• Protect data against hardware failure and human errors
  – Disk crash
  – Accidentally deleted a file
• Can use tar to backup important files
  tar czf /dev/rmt0 /proj/src
• “untar” to recover the files
  tar xf /dev/rmt0
Backup (cont.)

• Use **dump** to backup entire filesystems
  
  `dump -0u -f /dev/st0 /usr`

• Dump levels
  
  – 0: full dump - entire filesystem is copied
  – 1–9: incremental - copy all files modified since last lower level dump

• **/etc/dumpdates** has time of each dump

• Use **restore** to restore files from backup of increasing dump levels
  
  `restore -rf /dev/st0`
dd

- Convert and copy a file
- Can be used to copy from/to block devices

```bash
dd bs=4k skip=1 if=/dev/sda3 of=/dev/st0
```
Linux Distributions

- RedHat  http://www.redhat.com
- Debian  http://www.debian.org
- SuSE    http://www.novell.com/linux/suse
- Slackware http://www.slackware.com
- Knoppix http://www.knoppix.net