Overview

- Last Lecture
  - Network hardware
- This Lecture
  - Basic system/network administration
  - Reference:
    - *Linux Network Administrators Guide*, O. Kirch & T. Dawson, O-Reilly
    - http://en.tldp.org/LDP/nag2/index.html
- Next Lecture
  - Scripting technique

Security Awareness

- When a computer is connected to a network, it is under potential attack!
- Physical network/machine protection
- Attacks are from the network and through servers run by the computer
  - Remove the services if you don't need them
- Internet attacks
  - Worms
  - Viruses
  - Denial of Service (DoS)

Security Awareness (cont.)

- Computer/Internet hazards
  - SPAM/UCE (Unsolicited Commercial Email)
  - Phishing
  - Disk crashes/data loss
  - Loss of services due to outage
  - TCP/IP spoofing and sniffing (privacy)
  - Pornography
  - Ignorant users
  - Administrators of the untrained kind
  - …

Roles in a Network Community

- To be a good system admin, you should be aware of the roles in a network community and their strengths and weaknesses.
- Important roles include users, hosts, and operating systems.
  - Users - should be trained to be aware of the community
  - Host machines - should be allocated different tasks on different server machines
  - OS - have different pros and cons
    - UNIX/Linux, Windows, MAC OS, Netware

Network Administration

- Administration models
  - Reboot
  - Manual
  - Control
  - Immunology (self-maintenance)
- Network organisation
  - Homogeneity/uniformity
  - Delegation
- Principles of stable infrastructure
  - Scalability
  - Reliability
  - Redundancy
  - Homogeneity/uniformity
  - Reproducibility

Network Administration (cont.)

- Virtual machine model (nothing to do with VirtualBox or VMWare)
  - Treat a network as a single virtual machine
  - Good for resource management
    - share software by NFS, shared printers, shared home dir.
- Automation for uniformity
  - Script languages are helpful
- Revision control for configuration files
  - Revision Control (SVN, git)
- Cloud model

Network Kits

- Configuration tools
  - **ifconfig**
  - **route**
- How to find out info about your network?
  - **uname -a**
  - Find name server in **/etc/resolv.conf**
  - Various configuration files such as **/etc/services**, **/etc/inetd.conf**
- Find out info about other domains
  - **dig** or **host**
- If there is a problem from another domain
  - Send email to postmaster@domain or webmaster@domain, www@domain
  - Use **whois domain** to get info about a domain

Network Kits (cont.)

- Diagnostic/query tools
  - **Wireshark**
  - **Ping**
  - **traceroute/tracepath**
  - **netstat**
  - **lsof**
- Discover what you can do about a network
  - **nmap**: scan a network for security holes.
- Proprietary network monitoring software
  - E.g. from Cisco

User Management

- User account
  - Includes all the files, resources, and info belonging to one user. For commercial systems, it may include billing info.
- Create a new account
  - **adduser**
  - Account info: username, password, user id, group id, full name of user, home directory, login shell
  - Check after adding

User Management (cont.)

- Password
  - Very important for security
  - Should not be names of persons, books, places, your computer, nor your phone number, birthday, car registration plate, login name, words in dictionaries, keyboard sequence
  - Should be composed of letters (lower and upper cases), digits, and special characters like $,@
  - Refer to http://en.wikipedia.org/wiki/Password_strength
  - **passwd** imposes similar rules to make passwords secure.
- User id and group id
  - Users should be divided into groups for security reasons, e.g. students, staff, admin
  - Special users/groups: nobody, mail, ftp

User Management (cont.)

- Involved files
  - **/etc/passwd**, **/etc/group**, **/etc/shadow**
- User login environment
  - **.bash_profile, .bashrc, /etc/profile**
  - Place global files such as **profile** under **/etc**
    - Other scripts can be referred in it
  - Use **env/set** to check/set your environment
    - Paths and prompts
  - Keep a copy of your shell scripts (initial setups) in order to survive them from upgrade of OS/software
- For more detailed info, **man bash**

User Management (cont.)

- Remove a user: **deluser**
  - The relevant lines from **/etc/passwd**, **/etc/group**, and **/etc/shadow** will be removed.
  - It is a good idea to first disable the account before you start removing stuff
- Disable a user temporarily
  - A better way when you are not sure if a user will come back
  - Way 1: Put an * in the password field of **/etc/passwd** or **/etc/shadow**
  - Way 2: use **passwd -{l|u} username**
  - Way 3: Change the login shell to a script file

User Management (cont.)

- How to manage user accounts on different computers?
  - Share home directory using NFS
  - Share passwords using NIS (Network Information System) or LDAP (lightweight directory access protocol)
  - Allocate an Email server
  - Directory services like LDAP
- How to remember different passwords for different accounts on different computers?

User Management (cont.)

- Control user resources
  - Disk space
    - Separate disk partition for problem users
  - Use **df** command to monitor space
  - Quotas and limits
    - Better not to put them on users until necessary
    - Check **limits.conf** under **/etc/security**
  - Killing old processes
    - Don't do it unless you are absolutely sure
- Account policy
  - Who shouldn't have a user code?
  - How to deal with weak passwords?

User Management (cont.)

- User support services
  - cshelp
- User training and well-being
- How to treat the users?
  - Your enemies?
  - Your friends?
  - Your co-operators?
  - …

Least Privilege Principle

- No process or file should be given more privileges than it needs to do its job.
- Setuid programs: don't set unless necessary
- Run programs under special user id such as **www** and **nobody** if possible
- Some applications such as **httpd** can change its user id from **root** to **nobody** after opening the privileged port number 80.
- Temporary files shouldn't be in **/tmp**

Keeping Time

- Time zone
- Showing and setting time
  - **date**
  - **date -u**: showing the universal time
  - Get a time stamp: **date +%y%m%d%H%M%S**
- Hardware and software clocks
  - Use **date** to update software clock
  - Then use **hwclock -w** to set hardware clock

Keeping Time (cont.)

- Time server
  - Use some time server with accurate time
  - **netdate udp hostname** will set the time of the current machine to that of **hostname** (It seems netdate is not available now)
  - Can automatically adjust time by putting the command in cron table.
  - Can also use NTP for more accuracy
- Network Time Protocol (NTP)
  - Used to synchronize the time of a computer to another time server or reference time source.
    - **ntpdate**
  - Accuracy: 1 ms to dozens of milliseconds
  - Cryptography for security
  - How does it work? For more details, please refer to http://www.eecis.udel.edu/~ntp/ntp_spool/html/index.html

Host Management

- Shutting down a host
  - Turn off the power?
  - Should use command **shutdown**
  - **shutdown -h time** halt the system. **time** can be **now**.
  - **shutdown -r time** reboot the system
- Log files and audits
  - **syslogd**: a daemon for logging messages. Its configuration file is **/etc/syslog.conf**
  - **dmesg**: check kernel messages
  - **lastlog:** check the last login time of every user
  - **syslog** under **/var/log**: the log file of the system
  - They should be rotated regularly

Host Management (cont.)

- Making a file system (formatting)
  - Formatting floppy: **fdformat** (low level format)
  - make file systems: **mkfs/mke2fs**
    - **mkfs -t fstype filesystem**
  - Dump file system info: **dumpe2fs**
- Make a device
  - **mknod** or **/dev/MAKEDEV**
  - Make a device name in a file format so that you may be able to use the device as a file

Software Installation

- How to separate different third party software?
  - One software per directory?
- GNU software structure
  - lib, bin, sbin, etc, src
- GNU software installation
  - **./configure**
  - **make**
  - **make -n install:** before real installation.
  - **make install**
- Package management
  - **apt-get**