CSI3660 – System Administration

Prof. Fredericks

Troubleshooting and Ethics

Objectives

- Describe and outline good troubleshooting practices
- Effectively troubleshoot common hardware- and software-related problems
- Monitor system performance
- Identify and fix common performance problems

Troubleshooting Methodology

- After installing, configuring, and documenting your Linux system
  - You must maintain the system's integrity over time
- This includes:
  - Monitoring
  - Proactive maintenance
  - Reactive maintenance

Troubleshooting Methodology

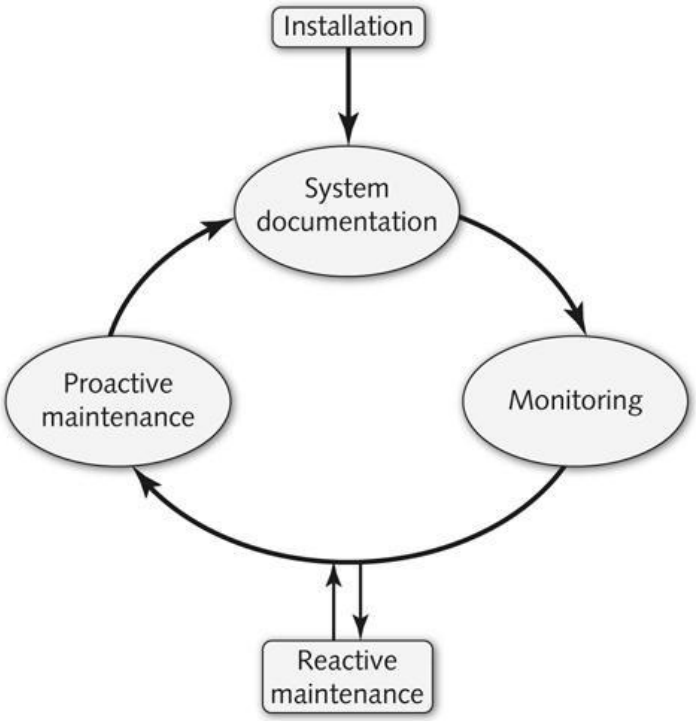


Figure 14-1: The maintenance cycle

Troubleshooting Methodology

- Monitoring
  - Examining log files
  - Running performance utilities periodically
  - identify problems and their causes
- Proactive maintenance
  - Minimizing chance of future problems
    - e.g., perform regular system backups

Troubleshooting Methodology

- Reactive maintenance
  - Correcting problems when they arise
  - Documenting solutions
  - Developing better proactive maintenance methods
- Documentation
  - System information stored in a log book for future references
  - All maintenance actions should be documented
- Troubleshooting procedures
  - Tasks performed when solving system problems

Troubleshooting Methodology

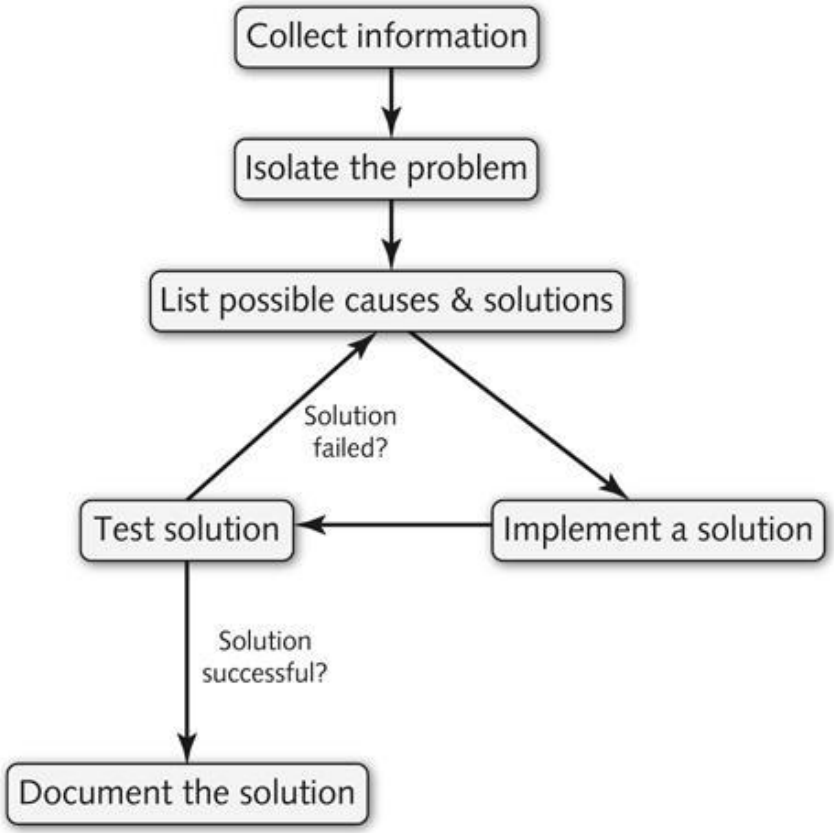


Figure 14-2: Common troubleshooting procedures

Troubleshooting Methodology

- Two troubleshooting golden rules
  - Prioritize problems according to severity
    - Spend reasonable amount of time on each problem given its priority
    - Ask for help if you can't solve the problem
  - Try to solve the root of the problem
    - Avoid missing underlying cause
    - Justify why a certain solution is successful

Resolving Common System Problems

- Three categories of problems:
  - Hardware-related
  - Software-related
  - User interface-related

Hardware-Related Problems

- Often involve improper hardware or software configuration
  - Peripheral communications
  - Video card and monitor configuration
  - POST test alerts
    - Loose hardware connections

  - Problems specific to the type of hardware
    - View output of `dmesg` or `journalctl -b` command
    - View content of:
      - /var/log/boot.log
      - /var/log/messages

Hardware-Related Problems

- Absence of device drivers prevent OS from using associated devices
  - `dmesg` or `lshw` command
    - Displays the hardware that is detected by the Linux kernel
  - `lsusb` command
    - Displays a list of USB devices detected by the Linux kernel
  - `lspci` command
    - Displays a list of PCI devices detected by the Linux kernel
  - Compare outputs of commands to output of `lsmod` to determine if driver module is missing from kernel

Hardware-Related Problems

- Hardware failure can render a device unusable
  - HDDs most common hardware components to fail
  - If HDD containing partitions mounted on noncritical directories fails:
    - Power down computer and replace failed HDD
    - Boot Linux system
    - Create partitions on replaced HDD
    - Use `mkfs` to create filesystems
    - Restore original data
    - Ensure /etc/fstab has appropriate entries to mount filesystems

Hardware-Related Problems

- If HDD containing / filesystem fails:
  - Power down computer and replace failed HDD
  - Reinstall Linux on new HDD
  - Restore original configuration and data files using a backup utility

Software-Related Problems

- Software-related problems are typically more difficult to identify and resolve than hardware-related problems
- Identify whether the software-related problem is related to application software or operating system software

Application-Related Problems

- Reason applications can fail:
  - Missing program libraries/files
  - Process restrictions
  - Conflicting applications
- Dependencies
  - Prerequisite shared libraries or packages required for program execution
  - Programs usually check at installation
  - Package files may be removed accidentally

Application-Related Problems

- `rpm -V <package>` command
  - Identify missing files in a package or package dependency
- `ldd` command
  - Display shared libraries used by a program
- `ldconfig` command
  - Updates list of shared library directories (/etc/ld.so.conf) and list of shared libraries (/etc/ld.so.cache)

Application-Related Problems

- Too many running processes can use all available PIDs in the process table
  - Solve by killing the parent process of zombie processes
- File handles
  - Connections programs make to files
- `ulimit` command
  - Used to modify process limit parameters in current shell
  - Can also modify max number of file handles

Application-Related Problems

- /var/log directory
  - Contains most system log files
- If applications stop functioning due to difficulty gaining resources, restart (application) using SIGHUP
  - To determine if another process is trying to access the same resources attempt to start application in Single User Mode
    - If resource conflict is the cause of the problem, download newer version of application or patch
  - kill –SIGHUP <pid> : reload process configuration
    - Generally will restart daemon
    - Sometimes will kill process and you have to manually restart

Operating System-Related Problems

- Most software-related problems are related to OS
  - X windows, bootloader, and filesystem problems
- Problem detecting video card or monitors by the kernel
  - An updated driver may be required
- To isolate a problem starting X Windows or gdm:
  - View the Xorg or XFree86 configuration file
  - Execute the `xwininfo` command

Operating System-Related Problems

- For LILO boot loader problems
  - Place “linear” and remove “compact” from the /etc/lilo.conf file
- GRUB and GRUB2 boot loader errors are typically a result of a missing file in /boot directory

Operating System-Related Problems

- If the filesystem on a partition mounted to a noncritical directory, such as /home or /var, becomes corrupted:
  - Unmount the filesystem
    - `umount`
  - Run `fsck` command with `-f (full)` option
  - If `fsck` command cannot repair filesystem, use `mkfs` command to re-create the filesystem
  - Restore filesystem's original data

Operating System-Related Problems

- If the / filesystem is corrupted:
  - Boot from installation or live media and enter System Rescue
  - At the BASH shell prompt within System Rescue:
    - Use `mkfs` to recreate the filesystem
    - Use backup utility to restore original data to the re-created / filesystem
  - Exit System Rescue and reboot system
- Knoppix Linux
  - Bootable Linux distributions with many filesystem repair utilities

Performance Monitoring

- Hardware that is improperly configured might still work, but at a slower speed
- Jabbering
  - Failing hardware components send large amounts of information to CPU
- Other causes of poor performance:
  - Software monopolizes system resources
  - Too many processes
  - Too many read/write requests to HDD
  - Rogue / zombie processes

Performance Monitoring

- To solve software performance issues
  - Remove software from the system
  - Move software to another Linux system
  - Add CPU or otherwise alter hardware
- Bus mastering
  - Peripheral components perform tasks normally executed by CPU
  - Reduces the amount of processing the CPU must perform and increases system speed
    - "Skip CPU"
    - PCI supports this

Performance Monitoring

- To increase performance:
  - Add RAM
  - Upgrade to faster HDDs (SSD hard disks)
  - Use disk striping RAID
  - Keep CD/DVD drives on a separate HDD controller
- Run performance utilities on a regular basis
  - Record results in a system log book
  - Eases identification of performance problems
- Baseline
  - Measure of normal system activity

Monitoring Performance with sysstat Utilities

- System Statistics (`sysstat`) package
  - Contains wide range of system monitoring utilities
  - Already installed on Scientific!
- `mpstat` (multiple processor statistics) command:
  - Displays CPU statistics
  - Used to monitor CPU performance
  - Can specify interval and number of measurements rather than displaying average values
  - `%sys` should be smaller than `%usr` and `%nice` combined
    - Sys: % cpu utilization at kernel (system) level
    - Usr: % cpu utilization at user level
    - Nice: % cpu utilization at user level with nice priority

Monitoring Performance with sysstat Utilities

- `iostat` (Input/Output Statistics) command: measures flow of information to and from disk devices
  - Displays CPU statistics similar to `mpstat`
  - Displays statistics for each disk device on the system
  - Output includes:
    - Transfers per second
    - Number of kilobytes read and written per second
    - Total number of kilobytes read and written for the device

Monitoring Performance with sysstat Utilities

- `sar` (system activity reporter) command:
  - Displays various system statistics taken in the last day
  - Provides more information than `mpstat` and `iostat`
  - By default scheduled to run every 10 minutes
    - Output logged to a file in /var/log/sa directory
  - `-f` option: View statistics from a specific file
  - Can be used to take current system measurements
    - Take four CPU statistics every two seconds with this command: `sar 2 4`

Monitoring Performance with sysstat Utilities

- Additional `sar` options:
  - `-q` option: Displays processor queue statistics
    - `runq-sz` value: Number of processes waiting for execution on processor run queue
    - `plist-sz` value: Indicates number of processes currently running
    - `ldavg` values: Represent average CPU load
  - `-W` option:
    - Displays number of pages sent to and taken from swap partition
    - Large number causes slower performance
    - Add RAM to resolve

Monitoring Performance with sysstat Utilities

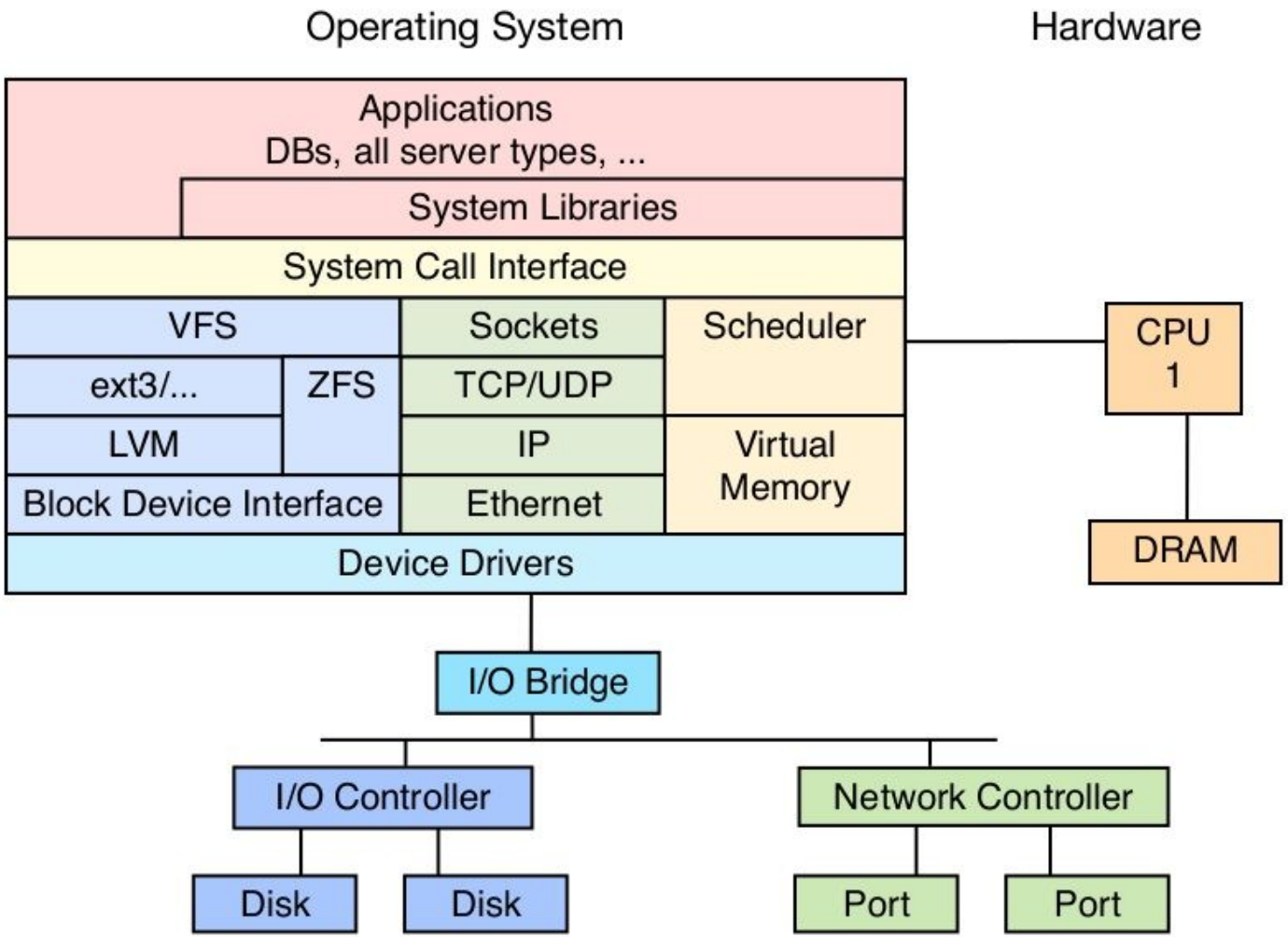
Option	Description
`-A`	Displays the most information; this option is equivalent to all options
`-b`	Displays I/O statistics
`-B`	Displays swap statistics
`-d`	Displays Input/Output statistics for each block device on the system
`-f` *`file_name`*	Displays information from the specified file; these files typically reside in the /var/log/sa directory
`-n ALL`	Reports all network statistics
`-o` *`file_name`*	Saves the output to a file in binary format
`-P` *`CPU#`*	Specifies statistics for a single CPU (the first CPU is 0, the second CPU is 1, and so on)
`-q`	Displays statistics for the processor queue
`-r`	Displays memory and swap statistics
`-R`	Displays memory statistics
`-u`	Displays CPU statistics; this is the default action when no options are specified
`-v`	Displays kernel-related filesystem statistics
`-W`	Displays swapping statistics

Table 14-1: Common options to the `sar` command

Other Performance Monitoring Utilities

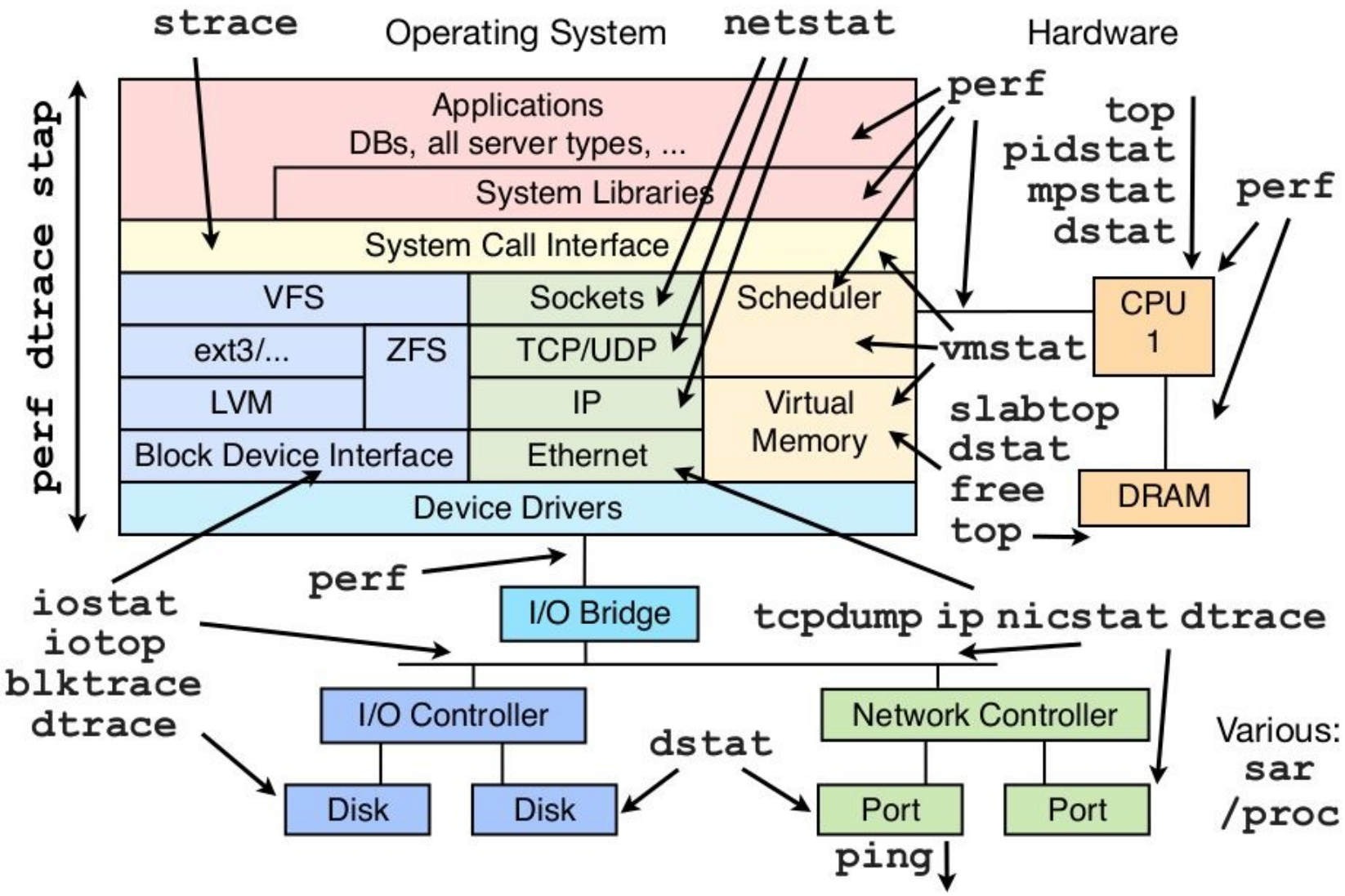
- `top` command:
  - Displays CPU statistics, memory usage, swap usage, and average CPU load
- `free` command:
  - Displays total amounts of physical and swap memory and their utilizations
  - Can be used to indicate whether more physical memory is required
- `vmstat` command:
  - Displays memory, CPU, and swap statistics
  - Can be used to indicate whether more physical memory is required

Analysis and Tools



http://www.slideshare.net/brendangregg/linux-performance-analysis-and-tools/15

Analysis and Tools



http://www.slideshare.net/brendangregg/linux-performance-analysis-and-tools/16

https://www.youtube.com/watch?v=xClSDpKualE

Ethics

- Ethics
  - The principles of conduct that govern a group of people
- Morals
  - Proclamation of what is right and good

- We are discussing **Ethics**
  - Policies concerning computer use are generally either for users or admins

System Administrator's Guild

- Example code from SAGE: System Administrators' Guild
- Not:
  - a set of enforceable laws
  - an enumeration of procedures
  - all-encompassing
  - an enumeration of sanctions and punishments
- Reinforces need for sysadmins to maintain a high standard of professionalism

SAGE Code of Ethics

- **The integrity of a system administrator must be beyond reproach.**
  - SAs come in contact with privileged information regularly
  - Need to protect integrity and privacy of data
  - Must uphold law and policies as established for their systems

- **A system administrator shall not unnecessarily infringe upon the rights of users.**
  - No tolerance for discrimination except when required for job
  - Must not exercise special powers to access information except when necessary

SAGE Code of Ethics

- **Communications of system administrators with all whom they may come in contact shall be kept to the highest standards of professional behavior.**
  - Must keep users informed of computing matters that might affect them
  - Must give impartial advice, and disclose any potential conflicts of interest

- **The continuance of professional education is critical to maintaining currency as a system administrator.**
  - Reading, study, training, and sharing knowledge and experiences are requirements

SAGE Code of Ethics

- **A system administrator must maintain an exemplary work ethic.**
  - A sysadmin can have a significant impact on an organization
    - A high level of trust is maintained by exemplary behavior

- **At all times system administrators must display professionalism in the performance of their duties.**
  - Need to be professional, even when dealing with management, vendors, users, or other sysadmins

Acceptable Use Policy

- **Acceptable Use Policy** required for users
  - When is personal use of equipment permitted?
  - What types of personal use are forbidden?
    - Can you start a business?
    - Can you surf adult sites?
  - What if the equipment is at your home?
- Might combine with a monitoring/privacy policy
  - Explain that monitoring might happen as part of running the network/server
- Look for archived/existing policies for starting point

Privileged Access

- Many users need privileged access
  - Sysadmins, programmers of device drivers, software installers, etc.
  - Such people need a special code of conduct
    - Since privileges can be abused
- Such users should sign a statement of having read the policy, and be given a copy
- Sysadmins should track those who have privileges on which systems
- Such access should expire unless renewed by signing again

Privileged Access

- Privileged access comes with responsibility to use it properly
- Access to be used only when necessary
  - Management will describe such uses
- Acknowledge that mistakes happen, and encourage procedures (such as backups) to minimize damage

Privileged Access

- Procedures to deal with situation in which sysadmin gets information that would not otherwise be public
  - E.g., learn about illegal or prohibited activities, or privileged info (pending sale of business)
- Warning about possible penalties for violations, including termination
- Legal requirements may also apply (e.g., SEC, FCC rules)

Copyrighted Information

- Acceptable use policy ***should*** require members to abide by current copyright laws
  - "Borrowing" non-freely redistributable software or content is usually illegal
  - Companies caught using pirated software have significant legal and financial liabilities
  - Pirated software can also be a source of viruses

- Sysadmins are often blamed for copyright violations found on their networks (permitted or installed)
  - Best approach is to make it easy for users
    - Use open source, or get broad site licenses for other packages

Law Enforcement

- Sysadmins are often contacted to help with investigations into computer-related crime
  - Also harassment issues, or need for records
- Need a procedure (prevent panic, significant mistakes)
  - Often work with a manager or legal department
  - Keep records of all communication and work performed
    - (e.g., commands typed)
  - Must verify identity of investigator before anything else
    - Social engineering is often a successful attack method
- Working with law enforcement can take a lot of time
  - Might need to make policies to reduce likelihood of need

Ethical Dilemmas

- What do you do when
  - You overhear (or read) about
    - A co-worker dealing drugs from the office?
    - Plans for sabotaging the company?
    - Stealing equipment and reselling via online auction?
    - Having an affair with the boss?
  - You are asked to read someone else's email?
    - By a non-admin colleague
    - By your manager

Protecting Yourself

- Have organizational policies that you can point to, and get guidance from
- Verify the (unreasonable) request
  - Perhaps you misheard it?
  - Get request in writing
- Verify with your manager (get permission)
- Make logs of all requests and communication
- Have a witness
  - Someone to watch what you are doing and agrees with your actions
  - Or at least verify your actions
- Contact organizational ombudsman, security, police if appropriate