1. Semester and Year	: 2012 FALL
2. Course Number	: CS-227
3. Course Title	: Linux & System Software
4. Work Number	: LA-05
5. Work Name	: Lab 05
6. Work Version	: Version1
7. Long Date	: Sunday, 30, September, 2012
8. Author(s) Name(s)	: Jake Waffle

Task 4.3: Use System Log Files

Understanding syslogd

The sysklogd package installs two daemons: syslogd and klogd. These two daemons pretty much just handle log files in a consistent manner.

Setting Logging Options

From what I got out of the book, the /etc/syslog.conf file is where the syslogd stores its logic for storing logs. And this logic is composed of a multitude of lines that basically say that this type of program with this type of message will have its log stored in this directory (facility.priority action.) This can be used to basically organize your logs into different folders/categories.

Rotating Log Files

Rotating log files is done to prevent log files from growing too big – because system logging daemons only add onto log files and don't check their size. The book tells of a common log rotation tool called logrotate (page 173.) It can be used through cron jobs to periodically rotate log files. A log rotation can be as simple as compressing an old log file under a different name and then starting with a new log file.

Reviewing Log File Contents

The book proposes several ways to look through logs (page 175.) It's first suggestions are with a text editor (vi, nano) and the less program. Although the user may not wish to view an entire log file.

Sometimes the user may want to search for keywords inside a log because they know exactly what they're looking for. In this case, grep is useful because it allows just that.

Other times, the user may just want to check the latest entries in a log after previously doing something. The tail command allows this and even allows for an option (-f) that prints lines on the terminal as they get added to the log.

And when these simple tools aren't enough, there are packages available for analyzing logs.

Task 6.1: Understand TCP/IP

Understanding Network Hardware

Modems – These are used with Point-to-Point Protocol to connect a computer with its Internet Service Provider. And somewhere I think I read that without a PPP connection or an alternative, one would not be able to get pictures on their browser.

Broadband – I have a broadband connection at my home and we have the DSL version. There's a modem that has a telephone plugged into it. And the modem has an Ethernet cord that can connect to a computer. But it also able to use Wi-Fi in order to connect to computers.

Ethernet – In terms of wired LAN connections, Ethernet is the dominating hardware. I've only really known about Ethernet cords for LAN connections. The Xbox and the Xbox 360 I know both have and still use Ethernet cords for LAN connections – it's also used to connect to Xbox live.

Wireless Devices – These allow for data transfer over radio waves instead of wires and is generally slower than the Ethernet route. But they seem to be growing ever more popular as time goes by. Tablets seem to be using Wi-Fi connections as a stand alone means of accessing the Internet now in certain models (the Microsoft Surface will be and a certain type of Ipad is like this.)

Upon entering “dmesg | less” I tried looking for network-related things, but only found some “wlan0 . . .” entries. I do know that ifconfig (ipconfig's Linux version basically) will display information regarding your computer's and modem's IP address and other network information.

Using Network Protocol Stacks

The TCP/IP protocol stack is what the Internet is based on. And it was really good that we did this, because it allows different computers to be able to communicate with each other. The TCP/IP stack consists of four layers: the Application, Transport, Internet and Link.

The Application layer is made up of network-enabled programs such as a web browser or email client.

The Transport layer is where the Transmission Control Protocol (TCP) comes into play (along with other protocols like UDP, which is more unreliable in comparison to TCP.) TCP uses a three-way handshake to ensure that packets are sent successfully to another computer.

The Internet layer is where the Internet Protocol (IP) comes into play. And it is very important because it manages the sending of datagrams from one host to another address.

The Link layer allows low-level communication with other computers. But I don't really know much about it.

Understanding Network Addressing

This section goes into the different addresses a computer may have and how to look it up. The ifconfig command will display information on all of your computer's network interfaces (eth0 is probably one.) And the information includes a MAC address, an IP address and a hostname.

Media Access Control (MAC) addresses are built into Ethernet cards and no two cards will have the same MAC address.

IP addresses can either be dynamic or static for a computer. To be dynamic they would have to be assigned by the Dynamic Host Configuration Protocol (DHCP) from the DHCP server.
I'm not really sure on hostnames, but I guess they are used as a means of finding someone's IP address (or it could be the other way around for all I know.) But next section will go into greater detail on hostnames.

Resolving Hostnames

The Domain Name System (DNS) is for looking up hostnames/domains and obtaining their designated IP addresses. There are a few commands that allow one to manually look up IP addresses.

The nslookup command allows users to look up either a IP address or a hostname given the connecting address (being either IP address or hostname.) I searched for google.com's IP address and got an assortment of them (Google is a pretty big search engine.)

Understanding Ports

There are many ports in a computer, but the ones that I have at least found to be important are the well-known ports (for things like finger, telnet, SMTP, etc) which are always the same. And server programs are linked to the well-known ports in a server computer. But clients generally don't use specific ports.

Task 6.2: Bring Up the Network

Network Hardware Configuration

Network hardware configurations are usually already set up to automatically detect network cards and load the corresponding drivers when a computer is distributed. But when network hardware isn't correctly detected, later configurations will not work (DHCP/static IP configuration.) There is however a command that allows one to load the network hardware driver for a given kernel. This command is called modprobe and it requires the name of your kernel module.

DHCP Configuration

When setting up my configuration files for an Arch Linux installation last school year, I ended up using DHCP for determining my IP address. And it was really simple to do. All I had to do was just set "interface = eth0" I think it was within /etc/rc.conf. And what this does exactly is make it so that my IP address is dynamic instead of static (the IP address will change over time.)

Static IP Address Configuration

Setting up a static IP address in Arch Linux is done in the same spot as DHCP, the user just needs to enter more information into /etc/rc.conf (ip address, netmask, gateway, etc.) Another way to change the IP address (temporarily) is to use ifconfig (only across your network) and route (goes out past your network) commands with their corresponding options. I think updating the configuration files with the appropriate values is a lot easier though (on Arch Linux anyways, but only because I know where to go now.)

Task 6.3: Monitor Network Connections

Testing Basic Connectivity

Ping seems to be the only network testing command listed in the book (page 245.) It's very simply to use, it just takes the command and a URL. And like with the Python shell, ping can be interrupted with "ctrl+c." Ping is useful in seeing whether a website is connectible to the computer in use. It does this by sending almost empty packets to the given URL and checking to see when and if the packets return. Ping will also give the IP address of the URL (which is useful in bypassing mediocre website blocking on a High School's network.)

Tracing a Route

The traceroute command is said to be a "step up" from the ping command (page 245.) It relays three test packets instead of one in-between your system and the target system. And because it sends three packets instead of one, the packets can have their response times compared. This comparison allows one to determine if their router is overloaded or the link is unreliable (highly varying response times are indicators of these problems.)

Checking Network Status

Netstat is said by the book to be the Swiss Army knife of network tools (page 246.) It allows for an assortment of different network information depending on the options given. But alone it just prints a list of internet connections and sockets. With the -p option, netstat will just print internet connections, the program that opened the connection and a port.

Task 6.10: Add a Printer Using CUPS

Obtaining CUPS Printer Definitions

This section is pretty much just about where to obtain drivers for your printer (because only a few printers are inherently supported.) The drivers pretty much just require a printer model and the rest is as easy as downloading.

Using the Web-Based CUPS Utilities

CUPS operates under the Internet Printing Protocol (IPP), which is so related to HTTP that the CUPS daemon can be accessed via a web browser. To use it in the web browser, just enter http://localhost:631. And that will access your computer at port 631 (where the CUPS daemon is.) CUPS also can be used from a different computer with the target's computer's hostname.

From here on, it's all GUI stuff and looks pretty straight forward (as GUI's explicitly list the user's options.) There is an add printer button at the bottom of the printer area (which can be found by clicking manage printers upon opening the CUPS daemon.) Then the user will be able to set up the printer s/he wishes to set up.