Review Topics for Exam #2, CS 406/591 Summer 2015
-------------------------------------------------
-------------------------------------------------
The exam will focus on the material since Exam #1, it will NOT be comprehensive (though you must of course understand some basic Linux material to do certain things). This exam will focus on the topics listed below. Questions will be similar to those on Homework #2. See the course web page under Readings for the relevant material. Questions may be asked about any material that was covered in class or is contained in the relevant sources, though less emphasis will be placed on what was not done in class.

1) Software Installation
--- Packaging formats: installer programs (.bin), gzipped-tar (.tgz), RPMs (.rpm), Debian packages (.deb)
--- Installation of source files:
   -- Steps: un-tar, configure, make, make install
   -- Issues: library dependencies, library versions, development packages
--- Package managers:
   -- Handle one particular package format or multiple
   -- Basic/low level vs. advanced/high level:
     files vs. packages, local files vs. repositories, detect dependency issues vs. automatically satisfy dependencies
   -- Command line vs. GUI
   -- Automate updating of installed packages
--- RPM package managers:
   rpm, yum, yumex, urpmi, rpmdrake
--- Debian package managers:
   dpkg, apt-get, aptitude, synaptic
--- Security (confirm package not corrupted, confirm trusted source):
   md5sum, sha1sum, gpg

2) Networking Basics
--- Network connection models: connection-oriented vs. connectionless, reliable vs. unreliable, packets, flow control, etc.
--- Internet network protocols: IP, TCP, UDP, ICMP.
--- IP addresses (IPv4):
   -- "dotted quad" or "dotted decimal" address vs. true IP address.
   -- network vs. host portion, CIDR, netmask
   -- routable vs. private IP addresses
--- IPv6: IPv6 format, new capabilities vs. IPv4
--- Loopback address: 127.0.0.1, meaning/use.
--- IP Ports: what is a port and how used, classes of ports
   (e.g., "well known" ports), relation to processes and sockets.
--- Ports and sockets: be able to define and distinguish
--- Domain Name Service: hostname vs. IP address, FQDN, basic working of DNS
--- Other name mappings: /etc/hosts, NIS
--- Network Address Translation (NAT): basic working and purposes

3) Network Security Basics
--- Terminology: attack, vulnerability, exploit, countermeasure, etc.
--- Attack vectors: physical attack, local, remote, social engineering, etc.
--- Examples of attack techniques for each vector: e.g., bootable disc for physical attack, fork bomb for local attack, DoS for remote attack, etc.
--- Examples of countermeasure for each vector: e.g., lock case for physical attack, limit user processes for local attack, firewall for remote attacks, etc.
--- Network Penetration attacks:
   -- Stages: reconnaissance/footprinting, scanning, vulnerability assessment, etc.
--- Reconnaissance techniques, tools, and countermeasures
--- Scanning techniques, tools, and countermeasures
--- Vulnerability assessment techniques, tools, and countermeasures
--- Remote penetration attacks:
   -- Classes of attack: vulnerability exploits and brute force attacks
   -- Techniques, tools, and countermeasures
--- Denial of Service (DoS) attacks: techniques, DDoS, countermeasures
--- Software vulnerabilities: buffer overflows, permissions issues, etc.
4) Firewall Basics
   -- Terminology: firewall, packet-filtering, application, border, etc.
   -- Packet-filtering firewalls:
      -- How work, stateless vs. stateful, state table, etc.
      -- Basic structure: chains of if-then rules, conditions, actions, policies
      -- Standard chains: input/ingress and output/egress
      -- Conditions: source/destination IP addresses, source/destination ports, etc.
      -- Actions: accept, drop, reject, log, etc.
   -- Netfilter (Linux kernel firewall)
      -- Linux kernel firewall in 2.6 kernels, modular organization
      -- Capabilities: stateful packet filtering, NAT, FTP, rate throttling, etc,
      -- Built-in chains (INPUT, OUTPUT, FORWARD, etc.) plus user-defined chains.
      -- Built-in actions: DROP, ACCEPT, REJECT, LOG, etc.
      -- iptables command: to manipulate Netfilter, basic syntax and capabilities

5) Kernel Configuration and Compilation
   -- Kernel boot parameters
   -- Compiled in kernel components and drivers vs. loadable kernel modules (LKMs)
   -- Change installed kernel modules/drivers:
      1smod, modprobe, insmod, rmmod, /etc/modprobe.conf, etc.
   -- Change kernel parameters:
      /proc/sys, using echo, sysctl, /etc/sysctl.conf
   -- Reconfigure and recompile kernel:
      -- Linux source tree at /usr/src/linux, configuration file (.config), patching
      -- Configuration tools: make menuconfig, make xconfig, make gconfig, compilation
      -- Compile steps: e.g., make, make modules, make modules_install, make install
      -- Files to /boot: vmlinuz, System.map, .config
      -- Initial RAM disk to load drivers: mkinitrd