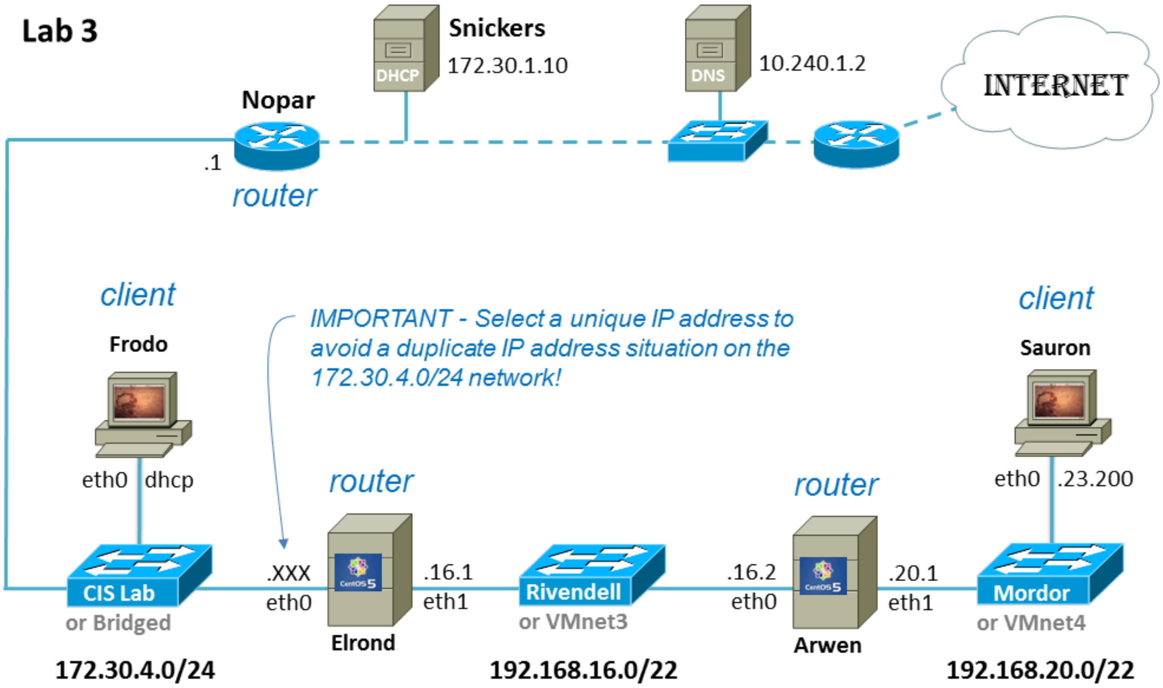
Cabrillo College

CIS 192 Linux Lab Exercise

Lab 3: Routing
Fall 2011

Lab 3: Configuring a Network Router
The purpose of this lab is to join three network segments (CIS Lab, Rivendell and Mordor) by configuring two routers between them. The three networks are shown below. You will be using four different systems: Frodo will be a client in CIS Lab. Elrond will route between the CIS Lab and Rivendell networks. Arwen will route between Rivendell and Mordor networks. Sauron is a client on the Mordor network. The goal is to obtain connectivity between Frodo in the CIS Lab and Sauron in Mordor with by utilizing layer 3 routing. You will verify success by successfully pinging from one side of the network to the other.



IMPORTANT - Select a unique IP address to avoid a duplicate IP address situation on the 172.30.4.0/24 network!

Supplies

- Virtualization: VMware ESXi/vSphere (for VLab) or Workstation (for CIS Lab PCs)
- Centos VMs: Elrond and Arwen
- Ubuntu VMs: Frodo and Sauron
- Virtual networks: Rivendell/VMnet3 and Mordor/VMnet4

Forum

Use the forum to ask questions, collaborate, post tips and any lessons learned when you have finished. Forum is at: http://opus.cabrillo.edu/forum/viewforum.php?f=39

Preparation

- □ Revert to the "Pristine" snapshot on all four VMs.
- □ On Opus, make a copy of the lab03 report template file in /home/cis192/depot in your home directory. Edit the header of this file with your own information and record all the information requested.

Make a plan

- □ Make your own network diagram/crib sheet that you can email to me. You can use your tool of choice (e.g. Visio, PowerPoint, etc.) or photograph/scan something you have done by hand (drawing/whiteboard). This diagram should be no more than one page but should have everything you need to complete this lab (now or five years from now). Be as creative as you want as there is no right answer for how it should look.
- □ Your diagram/crib sheet should include a graphical map showing labeled devices and interfaces and how they are connected. Include notation of default and static routes, network specification and all IP addresses used.
- □ Think through key ifconfig and route commands/files you will need to do this lab and add them to the crib sheet portion of your diagram. Include commands you intend to use for the firewall and packet forwarding. Be creative, there is no right answer for how to do this.
- □ Make sure the IP address you select for Elrond's eth0 interface is unique. Use the Static IP tables to avoid trouble!
- □ Use the ipcalc command on Frodo to verify your understanding of the sizes and aspects of all three networks being used in this lab: 172.30.4.0/24, 192.168.16.0/22 and 192.168.20.0/22.

❖ Record the Mordor network information in your lab report.

Step I - Interfaces

- □ Cable the VMs to match your diagram.
- □ Power on the VMs.
- □ Tip: layout the consoles on your screen so they align with your diagram/crib sheet.
- □ Use the **ifconfig** command to configure the interfaces on Elrond, Arwen and Sauron with IP addresses and subnet masks. Frodo uses DHCP so it should already be configured. For example, to configure Arwen's eth1 interface, use: **ifconfig eth1 192.168.20.1/22**
- □ Use the **ifconfig** command with no arguments to verify interface settings.
- □ Check that each system can ping its neighbor. For example, Arwen should be able to use **ping 192.168.23.200** to reach Sauron.

- □ Can a system ping an IP address on a network it's not connected to? For example, can Sauron ping Arwen's eth0 interface? This should not succeed. In fact you should get "Network is unreachable" messages.

Step 2 - Packet forwarding
Packet forwarding is disabled by default. For a Linux system to be a router, packet forwarding must be enabled.

- □ On both Elrond and Arwen, enable packet forwarding:

  ```
  echo 1 > /proc/sys/net/ipv4/ip_forward
  ```

- □ Do we need to configure packet forwarding on Frodo and Sauron? No, they are clients and not being used as routers.

Step 3 - Firewall
The default firewall on the CentOS systems does not allow forwarded traffic.

- □ On both Elrond and Arwen, use the following command to stop the firewall from blocking forwarded traffic:

  ```
  iptables -D FORWARD 1
  ```

- □ Do we need to do this on Frodo and Sauron? No, they are clients and not being used as routers.

Step 4 - Default gateways
In this step you will configure default gateway on Sauron, Arwen, and Elrond. Frodo uses DHCP and should already have a default route. Use default gateways to guide network traffic out toward the Internet.

Note: The default gateway selected should be the closest interface of the "next hop" gateway router on the way towards the Internet. Only specify an interface you can ping!

- □ To set the default gateway on Arwen, use:

  ```
  route add default gw 192.168.16.1
  ```

  Tip: A simple typo on the route command can result a most unhelpful error message. For example, typing 192.169.16.1 instead of 192.168.16.1 above would get the lovely message:
  "SIOCADDRT: No such process"
  Just translate it to:
  "That gateway is not reachable via a directly connected network -- try again"

- □ Now set default gateways on Elrond and Sauron using the same way of thinking (direct traffic toward the Internet via the closest interface on the "next hop" router)
- □ Use **route -n** to verify the default gateways on each system.
- □ Does your default gateway on Elrond match the default gateway on Frodo which was automatically configured by the DHCP? It should.

Step 5 - Static routes

Now the real fun begins. The default routes allow us to send traffic out towards the Internet. However at this point there are no routes to guide packets toward the Rivendell and Mordor networks.

If Frodo tries to ping Sauron without static routes, the pings will get sent to the Internet (where they will be dropped). Same thing if Elrond tries to ping Sauron, each ping gets directed out to the Internet (because of the default route) and will never return.

If Sauron sends a ping to Frodo without static routes, the ping request will reach Frodo but the reply will never get back to Sauron!

Let get those static routes configured:

- □ Configure a static route on Frodo that will send Mordor destination traffic in the right direction. This would be done with:

  route add -net 192.168.20.0/22 gw 172.30.4.xxx

  (where xxx is based on the unique IP address on Elrond's eth0 interface)
- □ Next configure a second static route on Frodo that will direct Rivendell destination traffic in the right direction.
- □ Is it possible to replace these two static routes with a single route? Try it and if it works, give yourself a ***super*** pat on the back!
- □ Configure a static route on Elrond so that packets intended for the Mordor network will be directed in the right direction. Optionally use the forum to collaborate with others on how to do this.
- □ Do we need a static route on Arwen to get to the Mordor network? No, it's directly connected.
- □ Do we need a static route on Sauron or Arwen to get to the CIS Lab network? No, that's already handled by the default gateway.
- □ You haven't finished this step till Frodo can ping Sauron. Don't worry about Arwen and Sauron lacking Internet access -- we will solve that in a later lab.

Tip: Test your way one system at a time till everything works. From Frodo, first ping Elrond's eth0 interface. If that works ping Elrond's eth1 interface. If that works ping Arwen's eth0 interface and so on. Make use of ifconfig, route -n, pings, and even tcpdump to verify and understand what is going on. Ping error messages are particularly useful in pinpointing an issue.

Tip: Use the forum to collaborate. Include route -n, ifconfig and ping output in your posts so others can offer help.

- ❖ On Frodo, record output from **ifconfig**, **route -n** and **ping -c3 192.168.23.200** commands in your lab report.
- ❖ On Elrond and Arwen, record output from **ifconfig**, **route -n** , **iptables -L** and **cat /proc/sys/net/ipv4/ip_forward** commands in your lab report.
- ❖ On Sauron, record output from **ifconfig** and **route -n** commands in your lab report.

To turn in

Check your work for completeness then submit as many times as you wish up until the due date deadline. Remember, **late work is not accepted**, so start early, plan ahead for things to go wrong and use the forum to ask questions.

- □ Email an electronic version of your network diagram/crib sheet to **risimms@cabrillo.edu**
- □ Submit your lab report on Opus using:

```
cp lab03 /home/rsimms/turnin/lab03.$LOGNAME
```

Grading rubric (30 points)

Lab report
1 point for a correct submittal into the turnin directory
2 points for a professional quality lab report than can be read using vi
2 points for a complete header including (name, date, station/pod ID, time spent)

Network configuration
1 points for correct static IP address on Elrond's eth0 interface (based on your station/pod ID)
1 point for correct Mordor network description (size, broadcast address, etc.)
2 points for correct configuration of Frodo
2 points for correct configuration of Elrond
2 points for correct configuration of Arwen
2 points for correct configuration of Sauron
5 points for successful ping from Frodo to Sauron

Network diagram/crib sheet
1 point for correctly emailing it to risimms@cabrillo.edu
3 points for a correct network map (showing four VMs, lab router, and switches)
1 point for labeling all networks (network address with prefix/mask)
1 point for labeling all interfaces with their names and IP addresses
1 point for labeling all VMs with hostnames
1 point for including all default gateways
1 point for including all static routes
1 point for including useful commands/files

Extra Credit (5 points)

- (1 point) Update /etc/hosts on Frodo so you can ping Elrond, Arwen and Sauron by name rather than just by IP address. Include the modified /etc/hosts files in your lab report.
- (1 point) From Frodo, ping Elrond, Arwen, and Sauron and explain why the TTL value differs depending on which system is pinged. Copy the ping outputs to your lab report.
- (1 point) Configure Elrond with permanent settings for IP addresses, subnet masks, gateway, and static routes. Include contents of the following files in your lab report:
  - /etc/sysconfig/network
  - /etc/sysconfig/network-scripts/ifcfg-eth0

- /etc/sysconfig/network-scripts/ifcfg-eth1
- /etc/sysconfig/network-scripts/route-eth1

Be sure to test your settings by restarting Elrond.

(1 point) Modify the firewall permanently on Elrond to allow all packets to be forwarded. Include your /etc/sysconfig/iptables file in your lab report. Be sure to test your settings by restarting Elrond.

(1 point) Enable packet forwarding permanently on Elrond. Include your /etc/sysctl.conf file in your lab report. Be sure to test your settings by restarting Elrond.

Command Appendix

IP Addressing

ipcalc - utility for calculating addresses and size of IP networks

Example on Ubuntu:

ipcalc 192.168.16.0/22

```
Address:   192.168.16.0          11000000.10101000.000100 00.00000000
Netmask:   255.255.252.0 = 22    11111111.11111111.111111 00.00000000
Wildcard:  0.0.3.255             00000000.00000000.000000 11.11111111
=>
Network:   192.168.16.0/22       11000000.10101000.000100 00.00000000
HostMin:   192.168.16.1          11000000.10101000.000100 00.00000001
HostMax:   192.168.19.254        11000000.10101000.000100 11.11111110
Broadcast: 192.168.19.255        11000000.10101000.000100 11.11111111
Hosts/Net: 1022                   Class C, Private Internet
```

Example on Red Hat family:

ipcalc -npmb 192.168.16.0/22

```
NETMASK=255.255.252.0
PREFIX=22
BROADCAST=192.168.19.255
NETWORK=192.168.16.0
```

top

Interfaces	
ifconfig or **/sbin/ifconfig**	Show the interface configurations. The full absolute pathname may be required if user is not logged in as root and /sbin is not in the user's path.
ifconfig eth*n* (where *n* is the interface number)	Show settings for selected interface. Example: **ifconfig eth1** will show information on the eth1 interface.
ifconfig eth*n* down (where *n* is the interface number)	Bring an interface down Example: **ifconfig eth1 down** will disable the eth1 interface.
ifconfig eth*n* up (where *n* is the interface number)	Bring an interface down Example: **ifconfig eth1 up** will enable the eth1 interface.

top

Interfaces - obtain dynamic IP address (temporary)	
dhclient eth0	Obtain an IP address for the eth0 interface from a DHCP server.
dhclient -r	Release the IP address previously obtained.
Interfaces - configure static IP configuration (temporary)	

ifconfig eth_n_ _xxx.xxx.xxx.xxx_**/**_pp_
 	
_n_ = interface number
xxx.xxx.xxx.xxx = IP address
_pp_ = the slash network prefix
	Same as previous command but the subnet mask is specified instead using a CIDR network prefix.
	
	Example: **ifconfig eth0 172.30.4.149/24**

ifconfig eth_n_ _xxx.xxx.xxx.xxx_ **netmask** _nnn.nnn.nnn.nnn_
 	
_n_ = interface number
xxx.xxx.xxx.xxx = IP address
nnn.nnn.nnn.nnn = subnet mask
	Configure an interface with an IP address and subnet mask.
	
	Example:
	ifconfig eth0 172.30.4.149 netmask 255.255.255.0
	(all on one line)
	Would configure eth0 with that IP address and mask.

ifconfig eth_n_ _xxx.xxx.xxx.xxx_ **netmask** _nnn.nnn.nnn.nnn_ **broadcast** _bbb.bbb.bbb.bbb_
(all on one line)
 	
_n_ = interface number
xxx.xxx.xxx.xxx = IP address
nnn.nnn.nnn.nnn = subnet mask
bbb.bbb.bbb.bbb = broadcast address
	Use this form of the command on older RH9 systems.
	
	Example:
	ifconfig eth0 172.30.4.149 netmask 255.255.255.0 broadcast 172.30.4.255
	(all on one line)
	Would configure eth0 with that IP address, mask and broadcast address.

ip address flush dev eth_n_ _n_ = interface number	Removes all settings from the selected interface. Example: **ip address flush dev eth0** will remove all interface settings, including the IP address, from eth0.

top

Interfaces - permanent configuration (Red Hat family)	
Edit **/etc/sysconfig/network-scripts/ifcfg-eth*n*** and add or modify these lines: **NM_CONTROLLED="***xx***"** **ONBOOT="***xx***"** **BOOTPROTO="***xx***"** **IPADDR=** *xxx.xxx.xxx.xxx* **NETMASK=** *xxx.xxx.xxx.xxx* For the new interface settings to take effect without restarting the system, use: **service network restart** or **/etc/init.d/network restart**	Each interface has an associated ifcfg-**eth***n* file in the **/etc/sysconfig/network-scripts** directory. These files are used at system startup to configure the interfaces. Set NM_CONTROLLED to “yes” or “no” to use or not use Red Hat NetworkManager utility. Set ONBOOT to “yes” to bring up the interface or “no” to disable the interface at system startup. Set BOOTPROTO to “static” to configure a static IP address or “dhcp” to configure a dynamic IP address. For static IP addresses, set IPADDR to the static IP address. Be sure this is a unique IP address for your system to avoid duplicate IPs on the network! Set NETMASK to the subnet mask. Example 1 - eth0 is not configured: **/etc/sysconfig/network-scripts/ifcfg-eth0** **DEVICE="eth0"** **NM_CONTROLLED="yes"** **ONBOOT="no"** Example2 - eth0 has a static IP configured: **/etc/sysconfig/network-scripts/ifcfg-eth0** **DEVICE="eth0"** **NM_CONTROLLED="no"** **ONBOOT="yes"** **BOOTPROTO=”static”** **IPADDR=172.30.4.149** **NETMASK=255.255.255.0** Example 3 - eth0 is configured for DHCP: **/etc/sysconfig/network-scripts/ifcfg-**

	eth0 **DEVICE="eth1"** **NM_CONTROLLED="no"** **ONBOOT="yes"** **BOOTPROTO=”dhcp”** Example 4 - eth0:1 is configured for an alias: **/etc/sysconfig/network-scripts/ifcfg-eth0:1** **DEVICE="eth0:1"** **NM_CONTROLLED="no"** **ONBOOT="yes"** **BOOTPROTO=”static”** **IPADDR=172.30.4.224** **NETMASK=255.255.255.0**

top

Routing table configuration (temporary)	
route -n or **ip route show**	Show the current routing table. The -n (numerical) option makes it faster. This option disables DNS lookups to replace IP addresses with hostnames in the output.
route -C	Show the routing table cache
ip route flush cache	Flush the routing table cache
route add default gw *xxx.xxx.xxx.xxx*	Adds the default gateway to the routing table. Unless there is another more specific route in the routing table this is the route will be used to send outbound packets. Example: **route add default gw 172.30.4.1** adds the lab router as the default gateway.
route del default gw *xxx.xxx.xxx.xxx*	Deletes the default gateway in the routing table. Example: **route del default gw 172.30.4.1** deletes the lab router as the default gateway.
route add -net *xxx.xxx.xxx.xxx/pp* **gw** *xxx.xxx.xxx.xxx*	Add static route Example: **route add -net 192.168.20.0/22 gw 172.30.4.250** *(all on one line)*
route del -net *xxx.xxx.xxx.xxx/pp* **gw** *xxx.xxx.xxx.xxx*	Delete static route

top

Routing table permanent configuration (Red Hat family)	
Edit **/etc/sysconfig/network** with: **GATEWAY=** *xxx.xxx.xxx.xxx*	Edit this file to add a permanent default gateway to the routing table. The new settings do not take effect until the system or network service is restarted. Example: **/etc/sysconfig/network** **NETWORKING=yes** **HOSTNAME=elrond.localdomain** **GATEWAY=172.30.4.1** The default gateway on Elrond has been set to the CIS Lab router (172.30.4.1). For the new interface settings to take effect without restarting the system, use: **service network restart** or **/etc/init.d/network restart**
Edit **/etc/sysconfig/network-scripts/route-eth***n* with: *xxx.xxx.xxx.xxx/pp* **via** *xxx.xxx.xxx.xxx*	Add static route permanently Example: **/etc/sysconfig/network-scripts/route-eth0** **192.168.20.0/22 via 172.30.4.250** to route traffic to the 192.168.20/22 network out the eth0 interface to the 172.30.4.250 “next hop” gateway router.

top

Network configuration - Debian family (permanent)	
Edit **/etc/network/interfaces** Use this “deprecated” script to restart network services: **/etc/init.d/networking restart** It seems this script in now deprecated and each interface must be manually shut down then brought back up! See: http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=565187	Edit this file to permanently configure networking on Debian and Ubuntu systems. **EXAMPLE - DHCP:** **/etc/network/interfaces** auto lo iface lo inet loopback auto eth0 iface eth0 inet dhcp **EXAMPLE - static IP:** **/etc/network/interfaces** auto lo iface lo inet loopback auto eth0 iface eth0 inet static address 172.30.4.222 netmask 255.255.255.0 gateway 172.30.4.1 **EXAMPLE - IP alias:** **/etc/network/interfaces** auto lo iface lo inet loopback auto eth0 iface eth0 inet static address 172.30.4.222 netmask 255.255.255.0 auto eth0:1 iface eth0:1 inet static address 172.30.4.223 netmask 255.255.255.0 gateway 172.30.4.1

	EXAMPLE - static IP and routes: **/etc/network/interfaces** auto lo iface lo inet loopback auto eth0 iface eth0 inet static address 172.30.4.222 netmask 255.255.255.0 gateway 172.30.4.1 up route add -net 192.168.2.0/24 gw 172.30.4.107 *(all on one line)* up route add -net 192.168.3.0/24 gw 172.30.4.107 *(all on one line)*

Name resolution	
cat /etc/resolv.conf	Show the DNS servers to use for resolving hostnames to IP addresses.
The **/etc/resolv.conf** file **nameserver** *xxx.xxx.xxx.xxx*	Edit this file to specify one or more DNS server. The first server listed will be the primary name server. The second will be the secondary name server and so forth. Example: **/etc/resolv.conf** **nameserver 192.168.0.8** **nameserver 10.240.1.2** configures the CIS VLab DNS server (192.168.0.8) as the primary and the campus DNS server (10.240.1.2) as the secondary.
> /etc/resolv.conf	Clears all DNS name servers
/etc/hosts file	You can locally add name resolution with the /etc/hosts file. Each line is this file starts with an IP address and is followed by one or more hostnames. Example: **echo " 192.168.23.200 sauron " >> /etc/hosts** *(all on one line)* allows you to ping sauron no by name in addition to by IP address.

top

Packet forwarding	
echo 1 > /proc/sys/net/ipv4/ip_forward	Enable packet forwarding
echo 0 > /proc/sys/net/ipv4/ip_forward	Disable packet forwarding
cat /proc/sys/net/ipv4/ip_forward	Show packet forwarding
The **/etc/sysctl.conf** file **net.ipv4.ip_forward =** *n* use *n*=0 to disable, use *n*=1 to enable	To permanently enable or disable packet forwarding. Example: **/etc/sysctl.conf** <snipped> **net.ipv4.ip_forward = 1** <snipped> will enable packet forwarding during system start or when the network service is restarted.

top

Firewalls	
iptables -L	Show the current firewall
iptables -F	Disables the firewall by flushing (deleting) all rules on all chains in memory.
iptables -D *chain rulenum*	Delete a rule on a chain in memory. Example: **iptables -D FORWARD 1** Delete the first rule on the FORWARD chain. This will modify the default CentOS firewall to allow packet forwarding.
iptables -P *chain target*	Set the policy on a chain to a target (e.g. ACCEPT, REJECT, DROP, etc) for the packet, if no rules apply. Example: **iptables -P FORWARD ACCEPT** sets the policy on the FORWARD chain to accept the packet, if no rules have applied.
service iptables restart	Loads the firewall rules form /etc/sysconfig/iptables
service iptables save	Copy the current firewall rules in memory and saves them to /etc/sysconfig/iptables.
iptables-save > iptables.bak	Copy the current firewall rules in memory to a file. Note: This can fail now due to default SELinux settings now (see /var/log/messages to verify). A partial workaround is to use: **service iptables save** but as this clobbers /etc/sysconfig/iptables be sure to back it up first.
iptables-restore < iptables.bak	Restore the current firewall in memory

	from a file.
iptables -A FORWARD -j REJECT --reject-with icmp-host-prohibited	Adds default CentOS rule for FORWARD chain. This will block packet forwarding.

top

Firewalls (Red Hat Family)	
Firewall configuration file: **/etc/sysconfig/iptables**	This file is not intended to be directly edited. You can copy this file to back it up. The contents are useful as they show how to form the actual iptables commands that could be entered from the command line Example: **cd /etc/sysconfig** **cp iptables iptables.bak** will backup the current firewall configuration file. Example: **cd /etc/sysconfig** **cp iptables.bak iptables** will restore the current firewall configuration file from the backup file. Example: **service iptables save** will replace /etc/sysconfig/iptables file with the current rules in memory. Example: **service iptables restart** loads the firewall rules into memory from /etc/sysconfig/iptables.

top

Network Testing	
ping *hostname* **ping** *xxx.xxx.xxx.xxx*	Test connectivity with another computer on the network. Use **Ctrl-C** to stop pinging. Options: **-c** *num* (limit the number of pings) -R (shows route travelled) -b (broadcast ping) Example: **ping -c3 google.com** will ping Google three times then stop. Example: **ping -Rc3 172.30.4.150** will show the route and do three pings. Example: **ping -b 172.30.4.255** will do a broadcast ping on the 172.30.4.0/24 network.
echo 0 > **/proc/sys/net/ipv4/icmp_echo_ignore_broadcasts** *(all on one line)*	Enables Linux system to respond to broadcast pings.
ping6 -I eth*n IPv6-address*	Works like the IPv4 ping except the outgoing interface must be specified. Example: **ping6 -I eth0 fe80::20c:29ff:fe2a:5717**
mtr *hostname* **or mtr** *xxx.xxx.xxx.xxx* Use q to quit	Displays the full route to the host and will refresh travels times.
traceroute *hostname* **or traceroute** *xxx.xxx.xxx.xxx* Use q to quit	Displays the full route to the host and will refresh travels times. Options: -I (use ICMP to get past some firewalls) Example: **traceroute google.com** Example:

	traceroute -I opus.cabrillo.edu
tcpdump	Will start sniffing packets. http://www.alexonlinux.com/tcpdump-for-dummies
tcpdump -n arp or icmp Use **-n** to prevent DNS lookups Use **Ctrl-s** or **Ctrl-q** to pause and continue	Packet sniffing command to capture only arp and icmp packets
tcpdump -n host *xxx.xxx.xxx.xxx* **and host** *xxx.xxx.xxx.xxx* *(all on one line)* Use **-n** to prevent DNS lookups Use **Ctrl-s** or **Ctrl-q** to pause and continue	Packet sniffing command to capture only traffic between two hosts. Example: tcpdump -n host 172.30.4.25 and host 172.30.4.1 *(all on one line)*
tcpdump -ne -i eth*n* **port** *nn* **or port** *nn*	Example: **tcpdump -ne -i eth1 port 80 or port 22** • no DNS lookups (-n) • shows mac addresses (-e) • will listen on eth1 interface (-i eth1) • only captures ssh and http traffic (port 80 or 22)

top

ARP commands	
arp -n	Display arp cache
ip neigh flush all	Flush arp cache
arpwatch (Red Hat family) Install arpwatch if necessary: • rpm –qa | grep arpwatch • yum install arpwatch Install /bin/mail if necessary: • rpm –qa | grep mailx • yum install mailx **service arpwatch start** *<Collection runs in the background>* **service arpwatch restart** **cat /var/lib/arpwatch/arp.dat**	arwatch (Debian family) Install arpwatch if necessary: • dpkg –l | grep arpwatch • apt-get install arpwatch Install /bin/mail if necessary: • dpkg –l | grep sendmail • apt-get install sendmail • dpkg –l | grep heirloom-mail • apt-get install heirloom-mail **/etc/init.d/arpwatch start** *<Collection runs in the background>* **/etc/init.d/arpwatch restart** **cat /var/lib/arpwatch/arp.dat**

top

Linux hardware and driver commands	
lspci or **/sbin/lspci**	Shows PCI devices including what NIC or NICs (Network Interface Controllers) are being used to physically connect the system to the network. The full absolute pathname may be required if user is not logged in as root and /sbin is not in the user's path. Example: **lspci | grep -i ether** will show all the ethernet NICs on the system.
lspci -k	Show the drivers kernel modules used by the PCI devices including any NICs. Example: **lspci -k | grep -iA4 ether** will show the drivers used by the NICs on your system.
lsmod or **/sbin/lsmod**	Shows the kernel modules that are currently loaded. Example NIC drivers (implemented as kernel modules) are e100 (Intel), e1000 (Intel), pcnet32 (AMD) and vmxnet (VMware). The full absolute pathname may be required if user is not logged in as root and /sbin is not in the user's path.
rmmod *module*	Use to unload (remove) a running kernel module (e.g. a NIC driver). Example: **rmmod e1000** would unload the Intel gigabit NIC driver if it was loaded.
modprobe *module*	Use to load a kernel module (e.g. NIC driver). Example: **modprobe e1000** would load the Intel gigabit NIC driver if not loaded already.
ls /lib/modules/$(uname -r)/kernel/drivers/net/	List all NIC drivers. These drivers are implemented as kernel modules and have a .ko suffix Information on older NIC drivers can be

	found here: http://www.tldp.org/HOWTO/text/Ethernet-HOWTO Example: **ls /lib/modules/2.6.32-71.el6.i686/kernel/drivers/net/** *(all on one line)* will list all the network drivers on the CentOS VMs used in the Fall 2011 term.

top

General Linux commands	
su -	To become root (superuser). The - is very important as it provides root’s shell environment.
sudo su - or **sudo -i**	To become root on the Ubuntu VMs.
exit	End a terminal login session
init 0 or **shutdown** *options time warning*	init 0 is a fast way to gracefully shutdown a VM. Note: no warning is given to users that the system will be shut down. The shutdown command is much more friendly in that it warns users before shutting down in the specified time interval. Example: **shutdown -h +5 ‘Save your work!’** Tells all users the system will shut down in 5 minutes and warns then to save their work. The h option performs a halt after the shutdown.
General Linux commands - basic inventory	
hostname	Shows the hostname of the system being used.
tty	Shows the current terminal being used.
uname -r	Print the version of the kernel being used.
who	Show logged in users and the IP address or hostnames they logged in from.
echo $PATH	Shows your path. The shell uses the path to locate any commands entered. Entering a command that is not located on the path will result in a “command not found” error.
cat /etc/*-release	Shows the name of the Linux distribution being run.

top

General Linux commands - files	
ls *[pathname]*	Short listing of files in current directory or pathname if specified.
ls -l *[pathname]*	Short listing of files in current directory or pathname if specified.
cat *pathname* **head** *pathname* **tail** *pathname* **more** *pathname* **less** *pathname*	Commands to display text files.
tail -f /var/log/messages	Useful for monitoring log files in real time.
vi *pathname*	Run the vi text editor on the specified file. Example: **vi lab01**
General Linux commands - redirection	
> *filename*	*filename* is created if it does not exist and emptied. Example: **> output** would empty the file named output or create it if it did not exist already.
command **>** *filename*	*filename* is emptied, then the output of the command is redirected into *filename.* Example: **ifconfig > output** would save the output of the ifconfig command in a file named output.
command **>>** *filename*	Output of the command is appended to the end of *filename.* Example: **route -n >> output** would append the routing table to the end of the file named output.

top

General Linux commands - logging in to a remote system	
ssh *account***@***hostname* **ssh** *account***@***xxx.xxx.xxx.xxx*	Login to a remote Linux computer on the network. Example: **ssh cis192@172.30.4.153**
ssh *account***@***hostname* **'***command***'**	Run a command on a remote system. Example: **ssh root@172.30.4.164 'ifconfig'** would run the ifconfig command on the remote system and show the output of the command on the local system.
ssh *account***@***IPv6address***%eth***n*	**ssh** works with IPv6 addresses too but the outgoing interface being specified. **ssh cis192@fe80::20c:29ff:fe2a:5717&eth0** *(all on one line)*
General Linux commands - copying files	
cp *source destination*	Linux command to copy file(s) from the source pathname to the destination pathname. Example: **cp /home/cis192/depot/lab01 .** will copy the file named lab01 in the /home/cis192/depot directory to your current directory.
scp *pathname account***@***host***:***pathname* **scp** *account***@***host***:***pathname pathname*	Copy files from one system to another. Example: **scp output simben192@opus.cabrillo.edu:** (above all on one line) would copy the local file named output to the user simben192's home directory on Opus.

top

General Linux commands - installing more commands or other software	
yum provides *command*	Find the package containing the command or program to install. Example: **yum provides mail**
yum install *package*	Download and install the software package on Red Hat family distributions. Just specify the name of the package to get the correct version for your distribution. Examples: **yum install traceroute** **yum install mtr tcpdump** **yum install mailx**
apt-get install *package*	Download and install the software package on Debian family distributions. Just specify the name of the package to get the correct version for your distribution. Examples: **apt-get install traceroute** **apt-get install mtr tcpdump** **apt-get install wireshark ipcalc**
General Linux commands - useful scripts	
while true; do *command*; **sleep** *seconds*; **done**	Repeatedly issue the same command over and over. Example: **while true; do ping sauron -c1; sleep 30; done** will ping sauron once every 30 seeonds

top

VMware commands and operations	
Change virtual terminals On PC Keyboard: • Method 1: While holding down the **Ctrl-Alt** keys, tap **spacebar** then tap f1, f2, ... or f7. • Method 2: While holding down **Alt** key, tap f1, f2, ... or f7. Does not always work but simpler than method 1. On Mac keyboard: • Hold down **Control** and **Option** keys, tap the **spacebar**, hold down **fn** key (in addition to **Control** and **Option** keys) and tap f1, f2, ... or f7.	Change to a different virtual terminal on the VM. F7 is graphics mode for the Ubuntu VMs. The Centos VMs do not have graphics mode (init level 3 only) Note: the spacebar does not need to be tapped on a physical (non-VM) system. This is just required for changing virtual terminals on VMware VMs.
Copy/Paste (vSphere Client 4.1) To enable this option for a specific virtual machine: 1. Log into a vCenter Server system using the vSphere Client and power off the virtual machine. 2. Select the virtual machine and click the **Summary** tab. 3. Click **Edit Settings**. 4. Navigate to **Options** > **Advanced** > **General** and click **Configuration Parameters**. 5. Click **Add Row**. 6. Type these values in the Name and Value columns: ○ isolation.tools.copy.disable – false ○ isolation.tools.paste.disable – false **Note**: These options override any settings made in the VMware Tools control panel of the guest operating system. 7. Click **OK** to close the Configuration Parameters dialog, and click **OK** again to close the Virtual Machine Properties dialog. 8. Power on the virtual machine.	**Copy/Paste (**ESXi server) To enable this option for all the virtual machines in the ESX/ESXi host: 1. Log in to the ESX/ESXi host as a root user and open the /etc/vmware/config file using a text editor. 2. Add these entries to the file: isolation.tools.copy.disable="FALSE" isolation.tools.paste.disable="FALSE" Save and close the file. The Copy and Paste options are only enabled when the virtual machines restart or resume the next time.

top