Lesson Module Status

- Slides
- Whiteboard with 1st minute quiz

- Flashcards
- Web Calendar summary
- Web book pages
- Commands
- Howtos

- Lab tested
- Lab template in depot
- Extra credit lab tested
- Lab template in depot

- Backup slides, Confer links, handouts on flash drive
- 9V backup battery for microphone

Course history and credits

Jim Griffin

- Jim created the original version of this course
- Jim's site: http://cabrillo.edu/~jgriffin/

Rick Graziani

- Thanks to Rick Graziani for the use of some of his great network slides
- Rick's site: http://cabrillo.edu/~rgraziani/

CIS 192 – Lesson 8

Instructor: **Rich Simms**
Dial-in: **888-450-4821**
Passcode: **761867**


Solomon


Sean C.


Chris


Corey


Bryan


Sean F.


Tony


David


Donna


Dave


Evan


Gabriel


Elia


Tajvia


Carlos


Adam


Ben


Laura

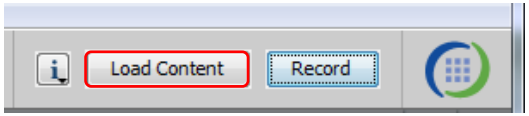
VMs for tonight

Frodo, Elrond

Email me (risimms@cabrillo.edu) a relatively current photo of your face for 3 points extra credit

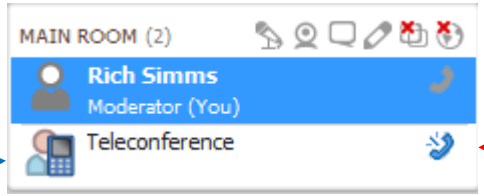
CCC Confer

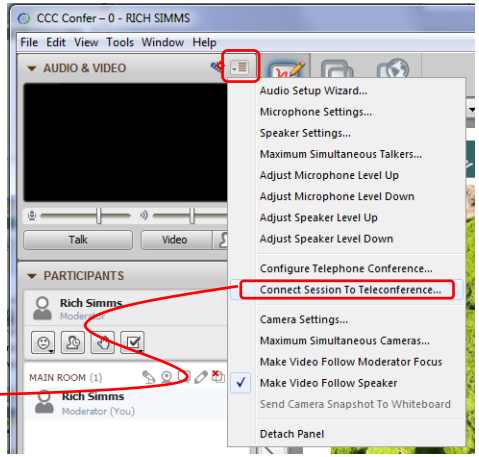
[ ] Preload White Board with *cis*lesson??*-WB*



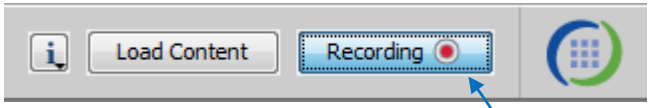
[ ] Connect session to Teleconference

Session now connected to teleconference





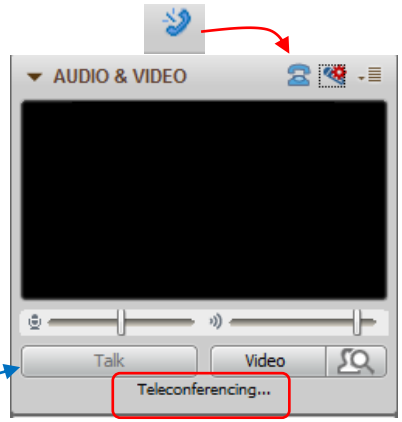
[ ] Is recording on?



Red dot means recording

[ ] Use teleconferencing, not mic

Should be greyed out

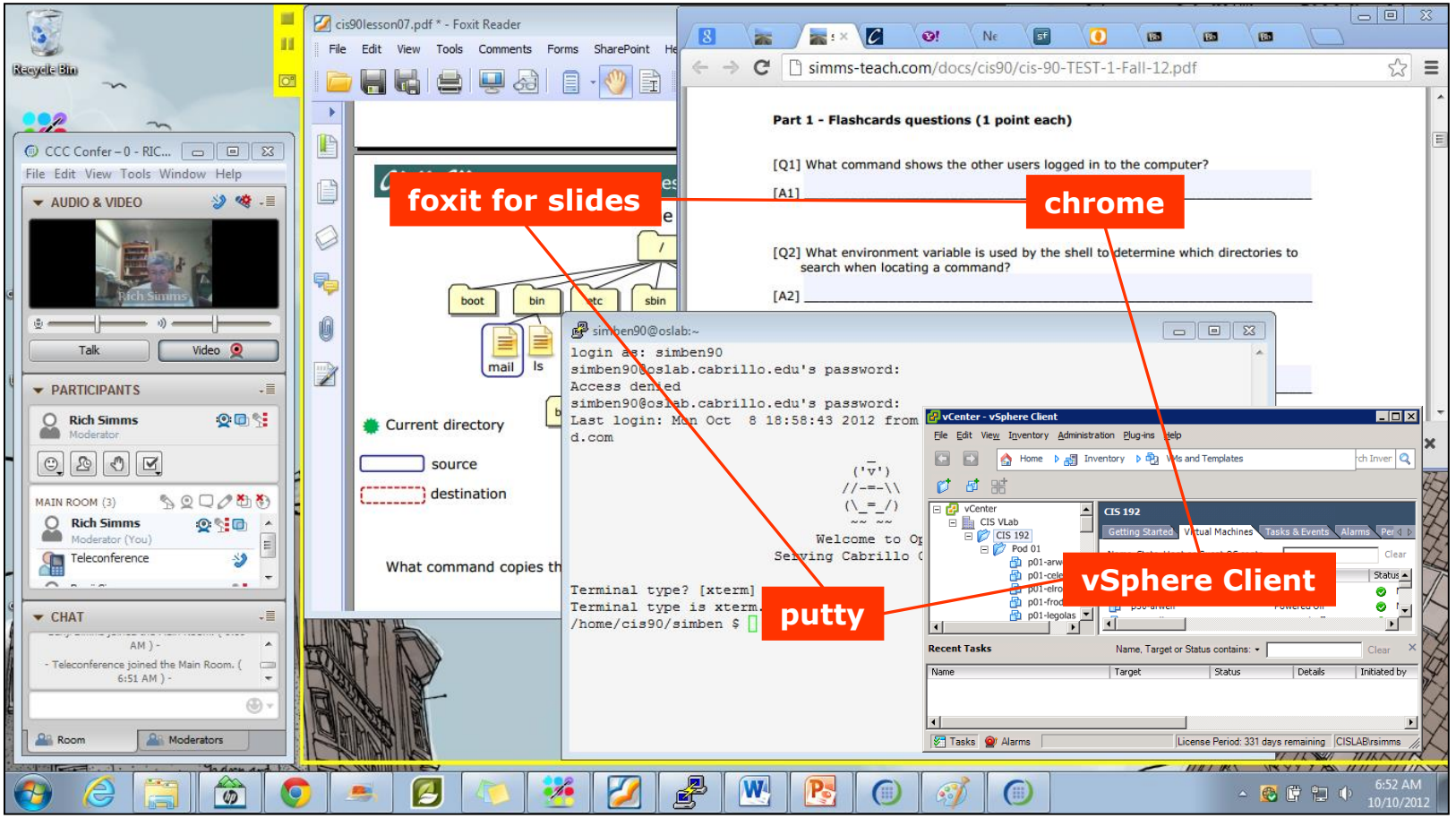


CCC Confer

Don't Forget

[ ] **Video (webcam) optional**

[ ] **layout and share apps**

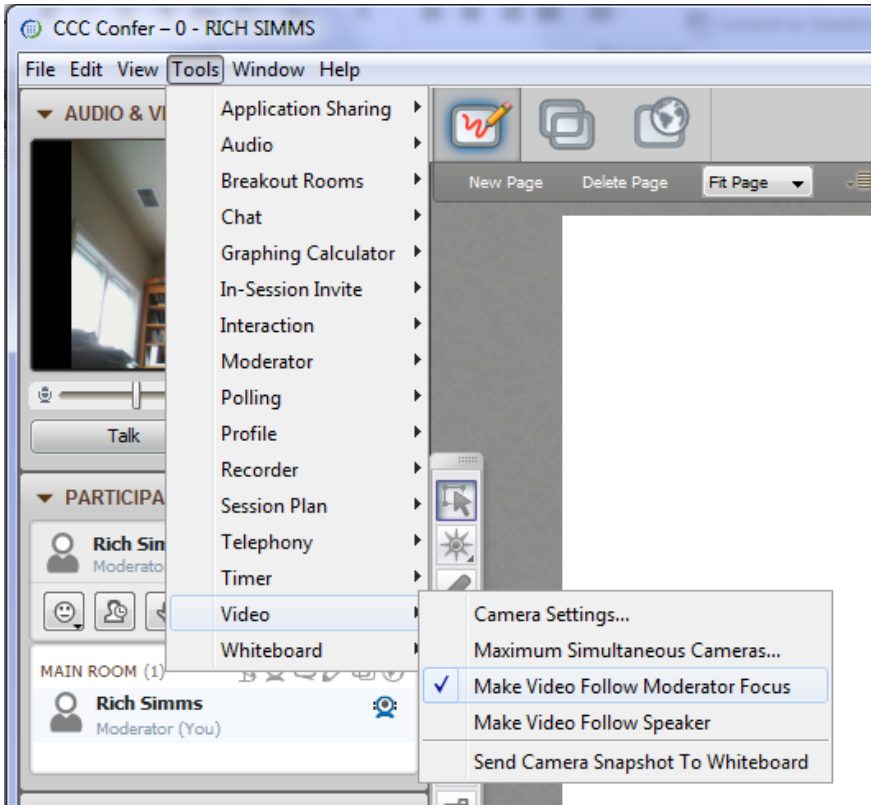


CCC Confer

[ ] Video (webcam) optional

[ ] Follow moderator

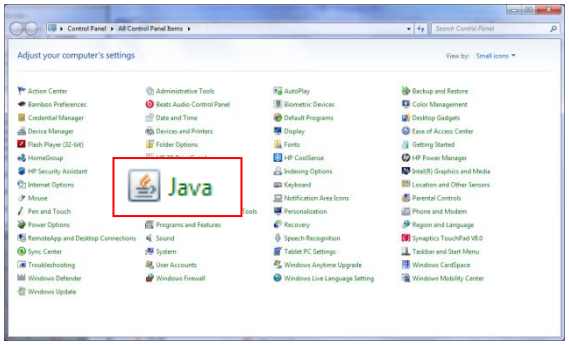
[ ] Double-click on postages stamps



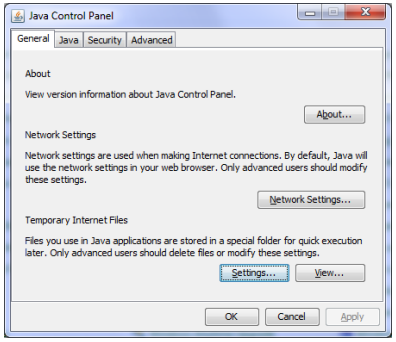
Universal Fix for CCC Confer:

1) Shrink (500 MB) and delete Java cache
2) Uninstall and reinstall latest Java runtime

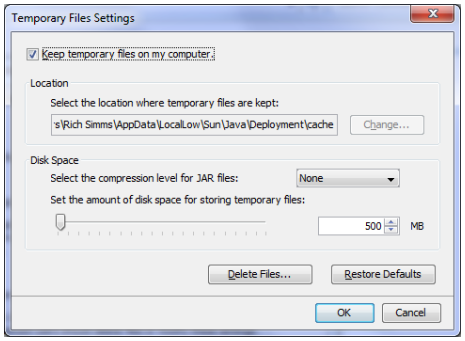
Control Panel (small icons)



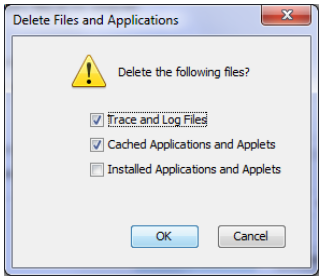
General Tab > Settings…



500MB cache size



Delete these



Google Java download

First Minute Quiz

Please answer these questions **in the order** shown:

Use CCC Confer White Board

For credit email answers to:
risimms@cabrillo.edu
within the first few minutes of class

PPP and WAN protocols

Objectives	Agenda
• Review lessons 5 - 8 • Implement serial connection using PPP	• Quiz • Questions on previous material • Housekeeping • Review for next test on Lessons 5-8 • PPP • Wrap

Questions on previous material

Questions

Lesson material?

Labs?

How this course works?

Chinese Proverb	他問一個問題，五分鐘是個傻子，他不問一個問題仍然是一個傻瓜永遠。 *He who asks a question is a fool for five minutes; he who does not ask a question remains a fool forever.*

Housekeeping

- DHCP Lab 6 due by 11:59pm tonight!
- Five posts due 11:59pm tonight!
- Test (no quiz) next week

Grades Web Page

http://simms-teach.com/cis192grades.php

Code Name	Grading Choice	Quizzes & Tests													Forum				Labs										Final	Extra Credit	Total	Grade
		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	T1	T2	T3	F1	F2	F3	F4	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10				
Max Points		3	3	3	3	3	3	3	3	3	3	30	30	30	20	20	20	20	30	30	30	30	30	30	30	30	30	30	60	90	560	
Aragorn	Grade	2		3	3							25			20				30	30	23	30	30							11		
Bilbo	Grade	3	3	3	3	1						29			20				29	28	29	30	24							19		
Denethor	P/NP	3	3	3		3						14			16				8	13	26	26	18							8		
Dwalin																																
Elrohir																																
Elrond																																
Faramir																																
Frodo																																
Gwaihir																																
Ioreth																																
Legolas																																
Nazgul																																
Pippin																																
Samwise																																
Saruman	Grade	3	3		3	3						28			20				30	30	30	30	30							8		
Strider	Grade	3	3	2		3						19			20				29	30		21	30							7		
Theoden	Grade	3	3	3	3	3						26			20				30	29	27	30	28							9		
Treebeard	P/NP																															

Please check your:

- Grading Choice
- Quiz points
- Forum points
- Test points
- Lab points
- Extra Credit points

Don't know you secret LOR code name?

… then email me your student survey to get it!

Help with labs

Like some help with labs?

I'm in the CIS Lab Monday afternoons
- See schedule at http://webhawks.org/~cislab/

or see me during office hours

or contact me to arrange another time online

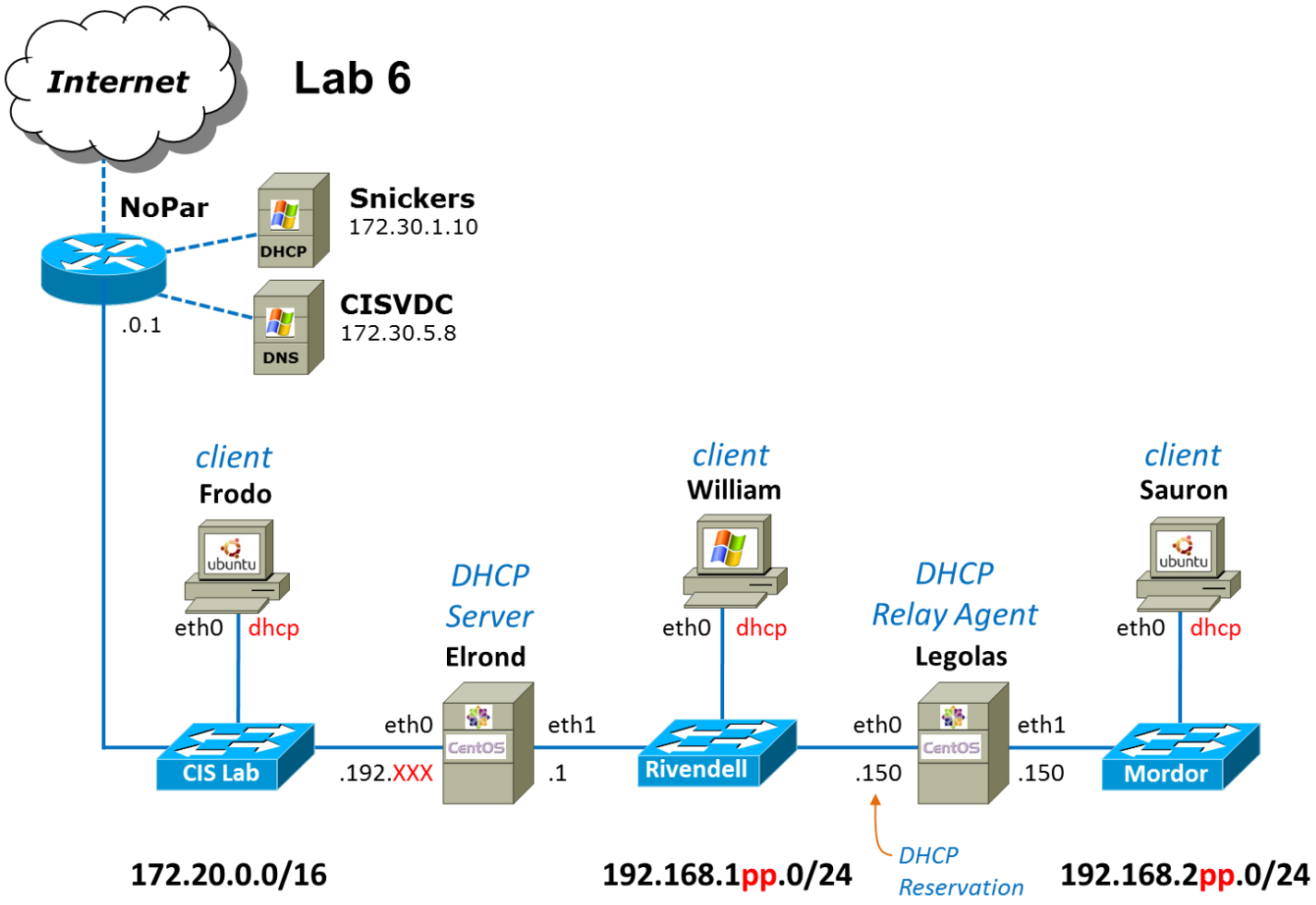
Selected Review

(based on Lab 6 Reference Implementation in Pod 28)

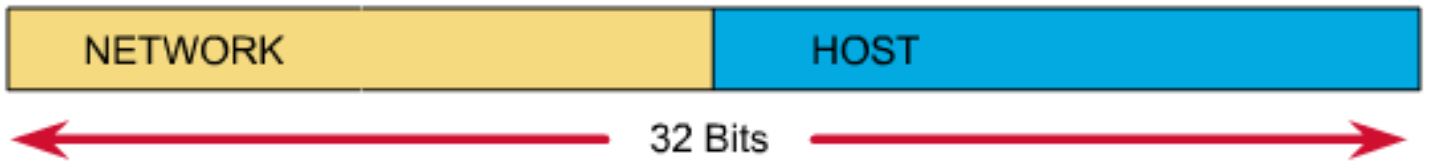
Elrond Setup

Challenge: See if you can do each of the following configuration steps using your crib sheets from previous labs and lesson slides. If you get stuck you can peek at the reference implementation in the Appendix.

- ☐ Permanently configure eth0 with your first static IP address for your pod. Save the remainder for a pool of CIS Lab network addresses for your DHCP server to use.
- ☐ Permanently configure eth1 as shown on the diagram.
- ☐ Permanently configure Nopar as the default gateway.
- ☐ Permanently configure the DNS nameserver and search domain(s).


Internet
Lab 6
NoPar
.0.1
Snickers
172.30.1.10
DHCP
CISVDC
172.30.5.8
DNS
client
Frodo
ubuntu
eth0
dhcp
CIS Lab
DHCP
Server
Elrond
eth0
.192.XXX
CentOS
eth1
.1
client
William
eth0
dhcp
Rivendell
DHCP
Relay Agent
Legolas
eth0
.150
CentOS
eth1
.150
client
Sauron
ubuntu
eth0
dhcp
Mordor
DHCP
Reservation
172.20.0.0/16
192.168.1pp.0/24
192.168.2pp.0/24

IPv4 Addresses



An IP address has two parts:

- **network number**
- **host number**

The **netmask** specifies the number of bits used to designate the network portion of the IP address

We will need the **netmask** when configuring permanent IP address settings.

Activity

172.20.0.0/16

#Bits in address for the network number =

Netmask =

192.168.128.0/24

#Bits in address for the network number =

Netmask =

192.168.228.0/24

#bits in address for the network number =

Netmask =

172.20.0.0/16

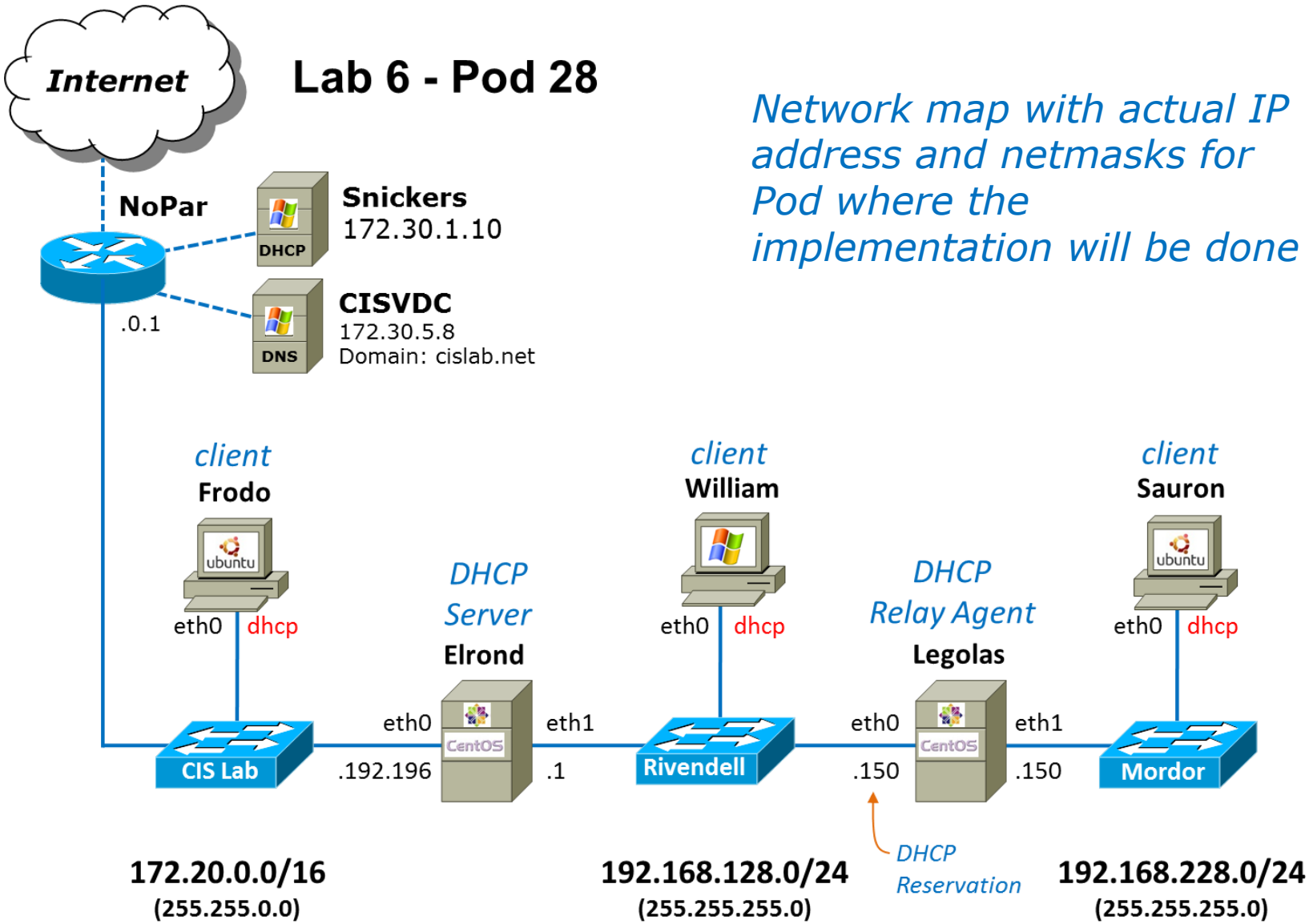
```
root@p28-frodo:~# ipcalc 172.20.0.0/16
Address:   172.20.0.0           10101100.00010100. 00000000.00000000
Netmask:   255.255.0.0 = 16     11111111.11111111. 00000000.00000000
Wildcard:  0.0.255.255          00000000.00000000. 11111111.11111111
=>                                          #bits
Network:   172.20.0.0/16        10101100.00010100. 00000000.00000000
HostMin:   172.20.0.1           10101100.00010100. 00000000.00000001
HostMax:   172.20.255.254       10101100.00010100. 11111111.11111110
Broadcast: 172.20.255.255       10101100.00010100. 11111111.11111111
Hosts/Net: 65534                 Class B, Private Internet
```

192.168.128.0/24

```
root@p28-frodo:~# ipcalc 192.168.128.0/24
Address:   192.168.128.0        11000000.10101000.10000000. 00000000
Netmask:   255.255.255.0 = 24   11111111.11111111.11111111. 00000000
Wildcard:  0.0.0.255            00000000.00000000.00000000. 11111111
=>                                          #bits
Network:   192.168.128.0/24     11000000.10101000.10000000. 00000000
HostMin:   192.168.128.1        11000000.10101000.10000000. 00000001
HostMax:   192.168.128.254      11000000.10101000.10000000. 11111110
Broadcast: 192.168.128.255      11000000.10101000.10000000. 11111111
Hosts/Net: 254                   Class C, Private Internet
```

192.168.228.0/24

```
root@p28-frodo:~# ipcalc 192.168.228.0/24
Address:   192.168.228.0        11000000.10101000.11100100. 00000000
Netmask:   255.255.255.0 = 24   11111111.11111111.11111111. 00000000
Wildcard:  0.0.0.255            00000000.00000000.00000000. 11111111
=>                                          #bits
Network:   192.168.228.0/24     11000000.10101000.11100100. 00000000
HostMin:   192.168.228.1        11000000.10101000.11100100. 00000001
HostMax:   192.168.228.254      11000000.10101000.11100100. 11111110
Broadcast: 192.168.228.255      11000000.10101000.11100100. 11111111
Hosts/Net: 254                   Class C, Private Internet
```



Network map with actual IP address and netmasks for Pod where the implementation will be done

Activity

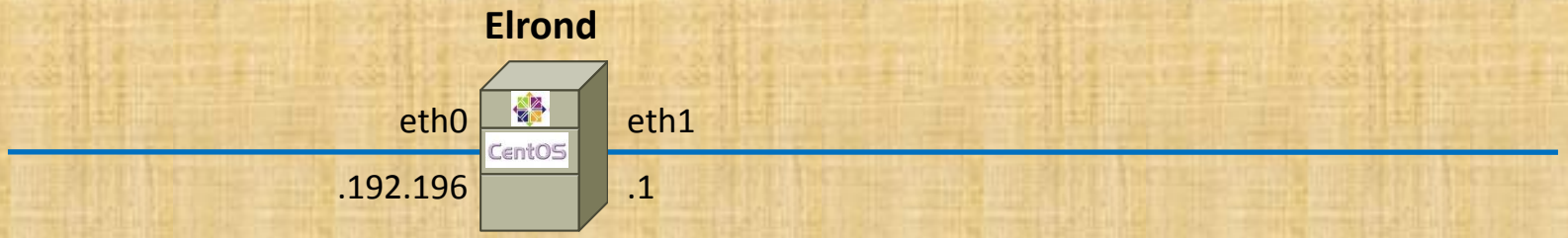
What are the files we need to configure on Elrond to configure the interfaces, default route and DNS nameservers?

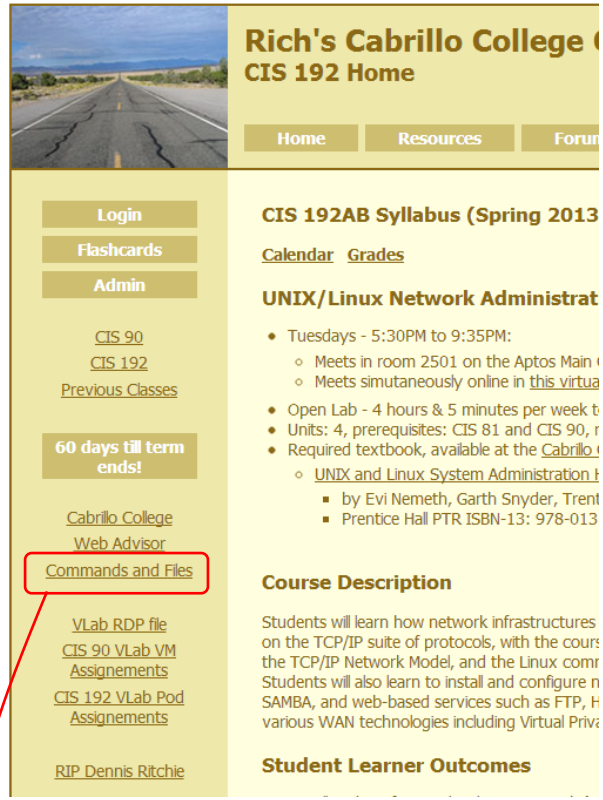
eth0:

eth1:

Default gateway:

DNS nameservers:





redhat.

Permanent Interface Configuration
Permanent Routing Table Configuration
Permanent Hostname Configuration

Remembering how to configure interfaces using crib sheet

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*

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. Since we don't use this in CIS192 set to "no".

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.

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.

Example: eth0 not configured
/etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE="eth0"
NM_CONTROLLED="yes"
ONBOOT="no"

Example: eth0 has static IP
/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: eth0 is DHCP
/etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE="eth0"
NM_CONTROLLED="no"
ONBOOT="yes"
BOOTPROTO="dhcp"

Example: IP alias on eth0
/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

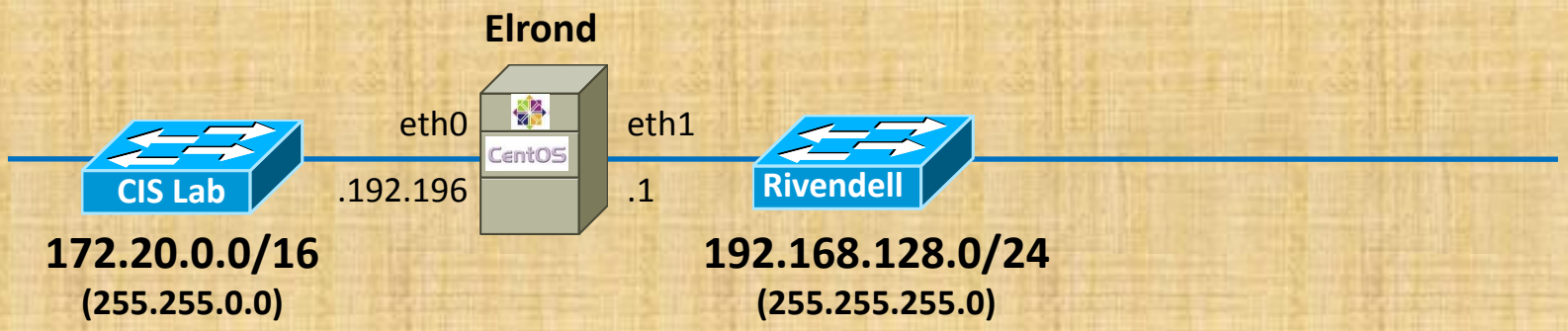
top

http://simms-teach.com/docs/cis192/cis192QuickRef.pdf

Activity - configuring eth0 and eth1

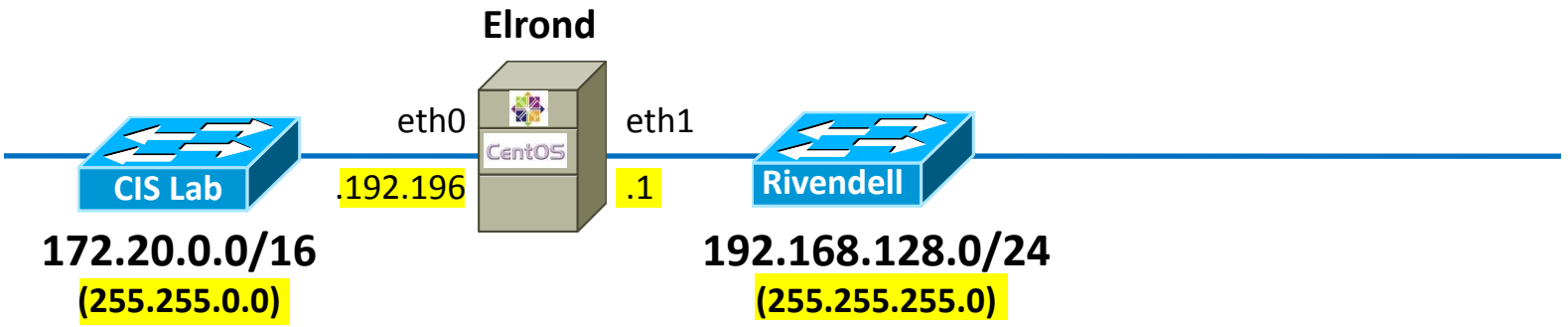
```
[root@p28-elrond ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE="eth0"
BOOTPROTO="
"
NM_CONTROLLED="no"
ONBOOT="
"
TYPE="Ethernet"
IPADDR=
NETMASK=
```

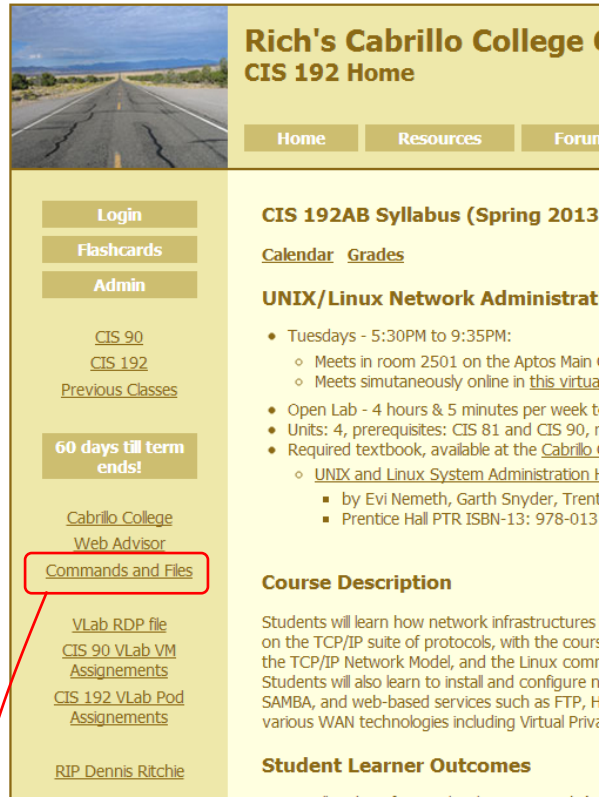
```
[root@p28-elrond ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE="eth1"
BOOTPROTO="
"
NM_CONTROLLED="no"
ONBOOT="
"
TYPE="Ethernet"
IPADDR=
NETMASK=
```



```
[root@p28-elrond ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE="eth0"
BOOTPROTO="static"
NM_CONTROLLED="no"
ONBOOT="yes"
TYPE="Ethernet"
IPADDR=172.20.192.196
NETMASK=255.255.0.0
```

```
[root@p28-elrond ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE="eth1"
BOOTPROTO="static"
NM_CONTROLLED="no"
ONBOOT="yes"
TYPE="Ethernet"
IPADDR=192.168.128.1
NETMASK=255.255.255.0
```





redhat

Permanent Interface Configuration
Permanent Routing Table Configuration
Permanent Hostname Configuration

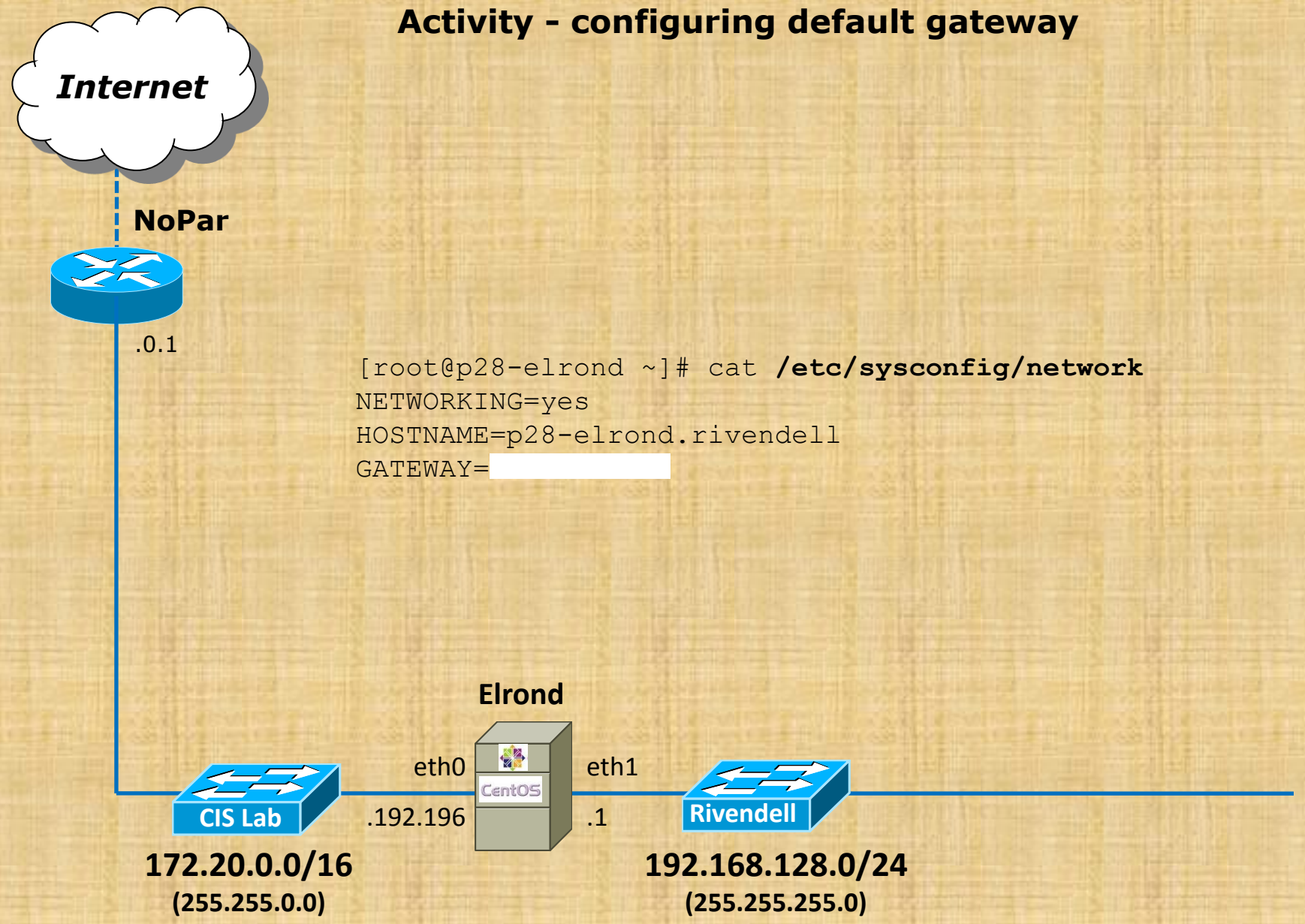
Remembering how to configure the default gateway using crib sheet

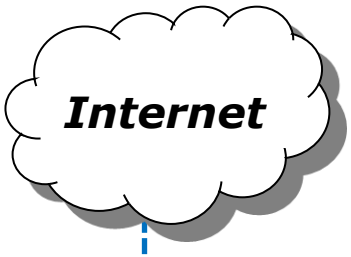
Routing table permanent configuration (Red Hat family)	
Edit **/etc/sysconfig/network** with: **GATEWAY=** *xxx.xxx.xxx.xxx*	Edit this file to add a permanent default gateway to the routing table. The new settings do not take effect until the system or network service is restarted. **Example:** **/etc/sysconfig/network** **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 192.168.20.0/22 network out the eth0 interface to the 172.30.4.250 "next hop" gateway router.

top

http://simms-teach.com/docs/cis192/cis192QuickRef.pdf

Activity - configuring default gateway





NoPar

.0.1

```
[root@p28-elrond ~]# cat /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=p28-elrond.rivendell
GATEWAY=172.20.0.1
```

Elrond

eth0

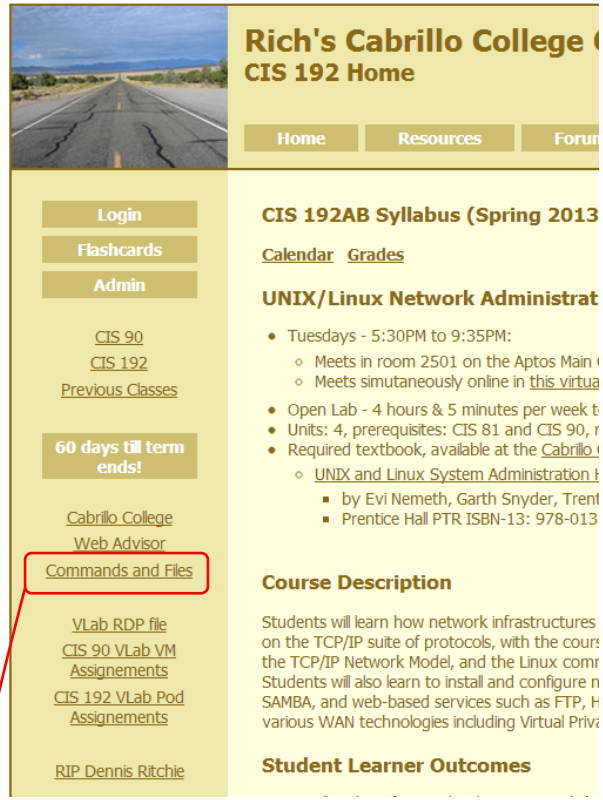
.192.196

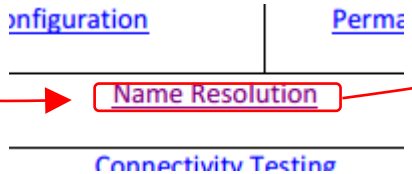
eth1

.1

172.20.0.0/16
(255.255.0.0)

192.168.128.0/24
(255.255.255.0)





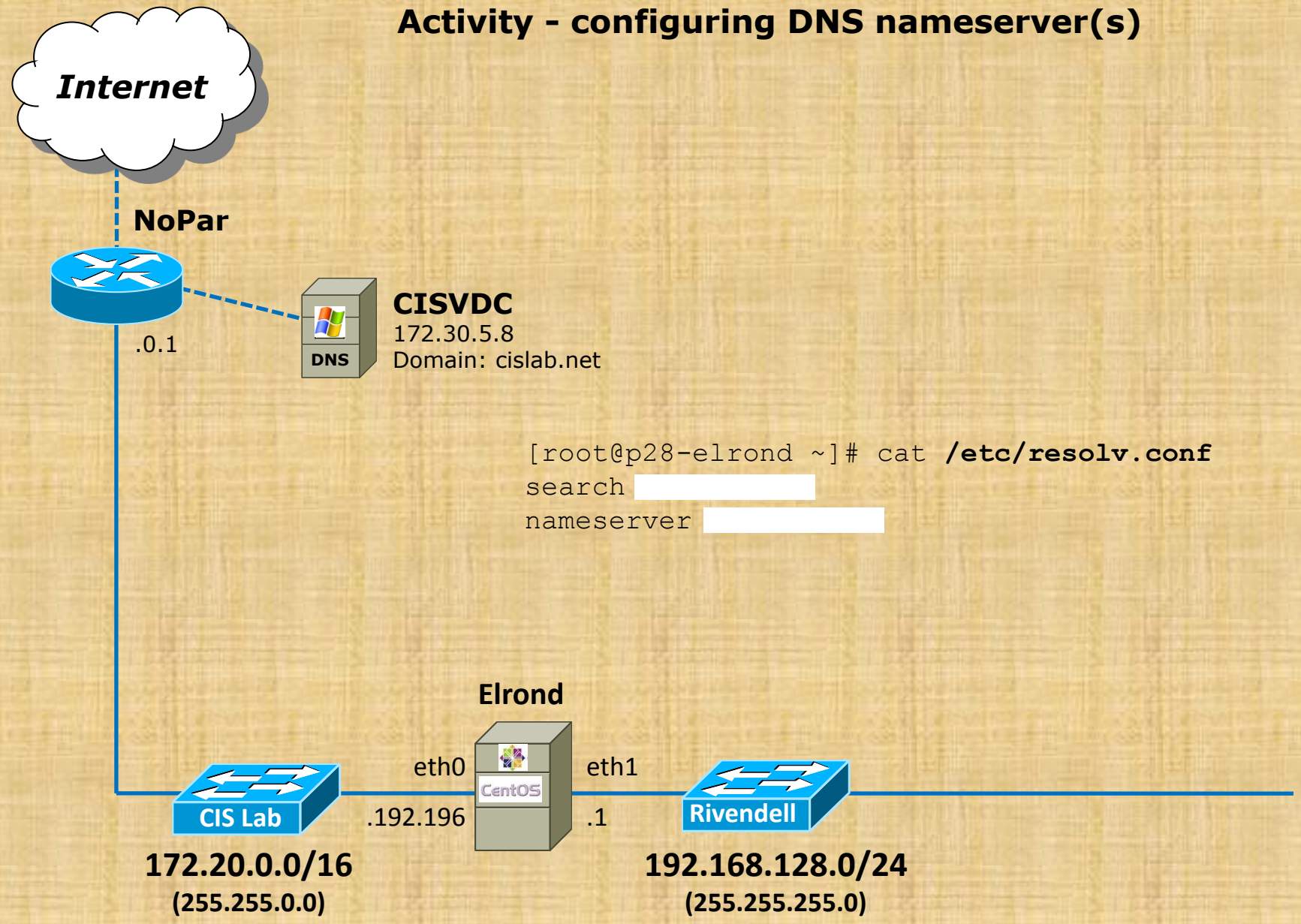
Remembering how to configure the DNS nameserver(s) using crib sheet

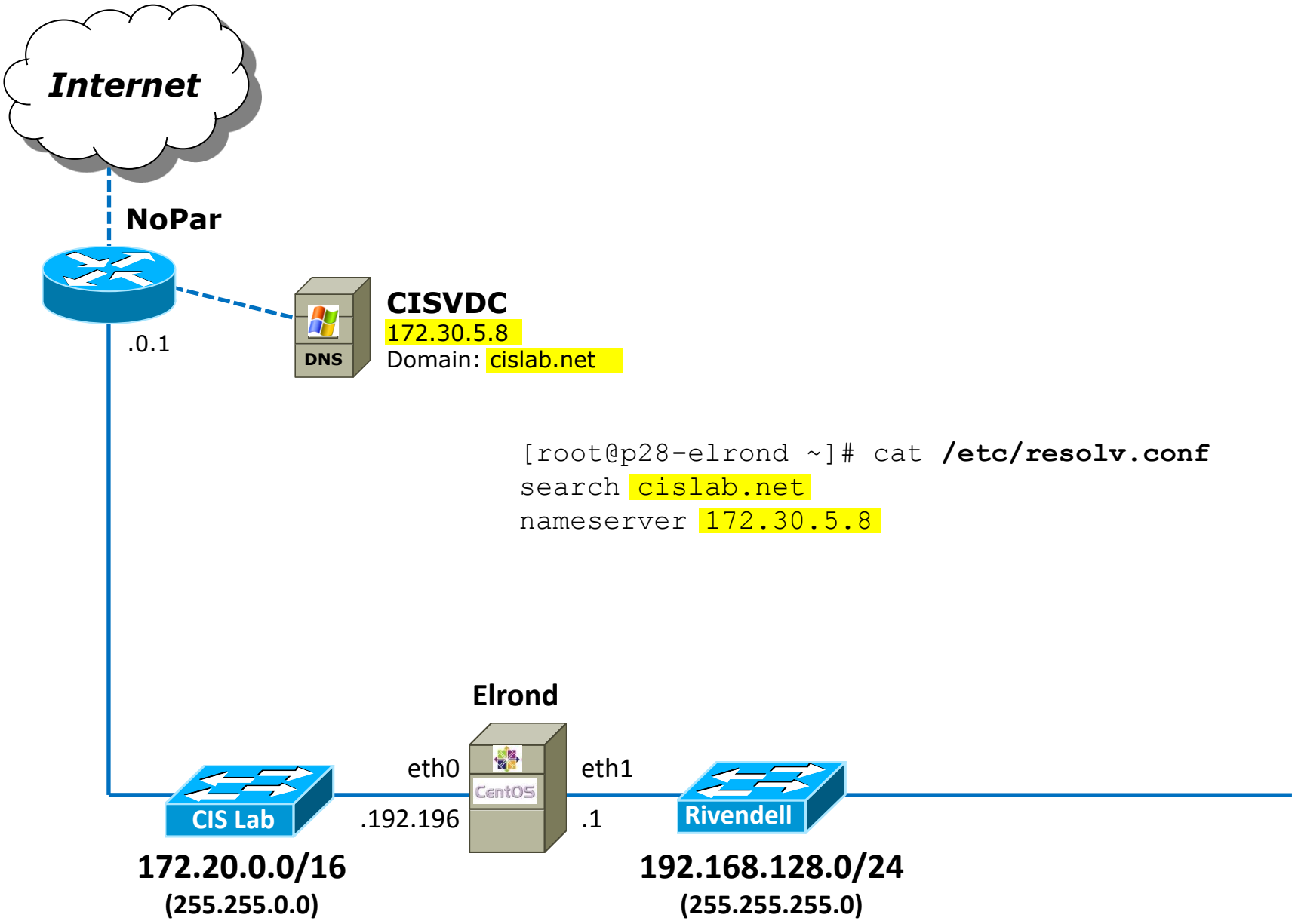
Name resolution	
On Red Hat family and some Debian family: The **/etc/resolv.conf** file: **search** *domain* **nameserver** *<ip address>*	Edit this file to specify one or more DNS server. The first server listed will be the primary name server. The second will be the secondary name server and so forth. Example: **/etc/resolv.conf** **search cislab.net** **nameserver 172.30.5.8** **nameserver 10.240.1.2** configures the CIS VLab DNS server (172.30.5.8) as the primary and the campus DNS server (10.240.1.2) as the secondary. Allows users to use shortnames for the cislab.net domain. For example **ping opus** will be treated as if the user typed **ping opus.cislab.net.**
On Debian family: Check to see if **/etc/resolv.conf** is symbolically linked to **../run/resolvconf/resolv.conf** and if it is DO NOT MODIFY **/etc/resolv.conf**. Instead add the equivalent lines to the **/etc/network/interfaces** file: **dns-search** *domain* **dns-nameservers** *<ip address> <ip address>* then restart networking service.	
> /etc/resolv.conf	Clears all DNS name servers
The **/etc/hosts** file: xxx.xxx.xxx.xxx name1 name2 ...	Edit this file to locally add name resolution for commonly used hosts. 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 by name in addition to by IP address.

top

http://simms-teach.com/docs/cis192/cis192QuickRef.pdf

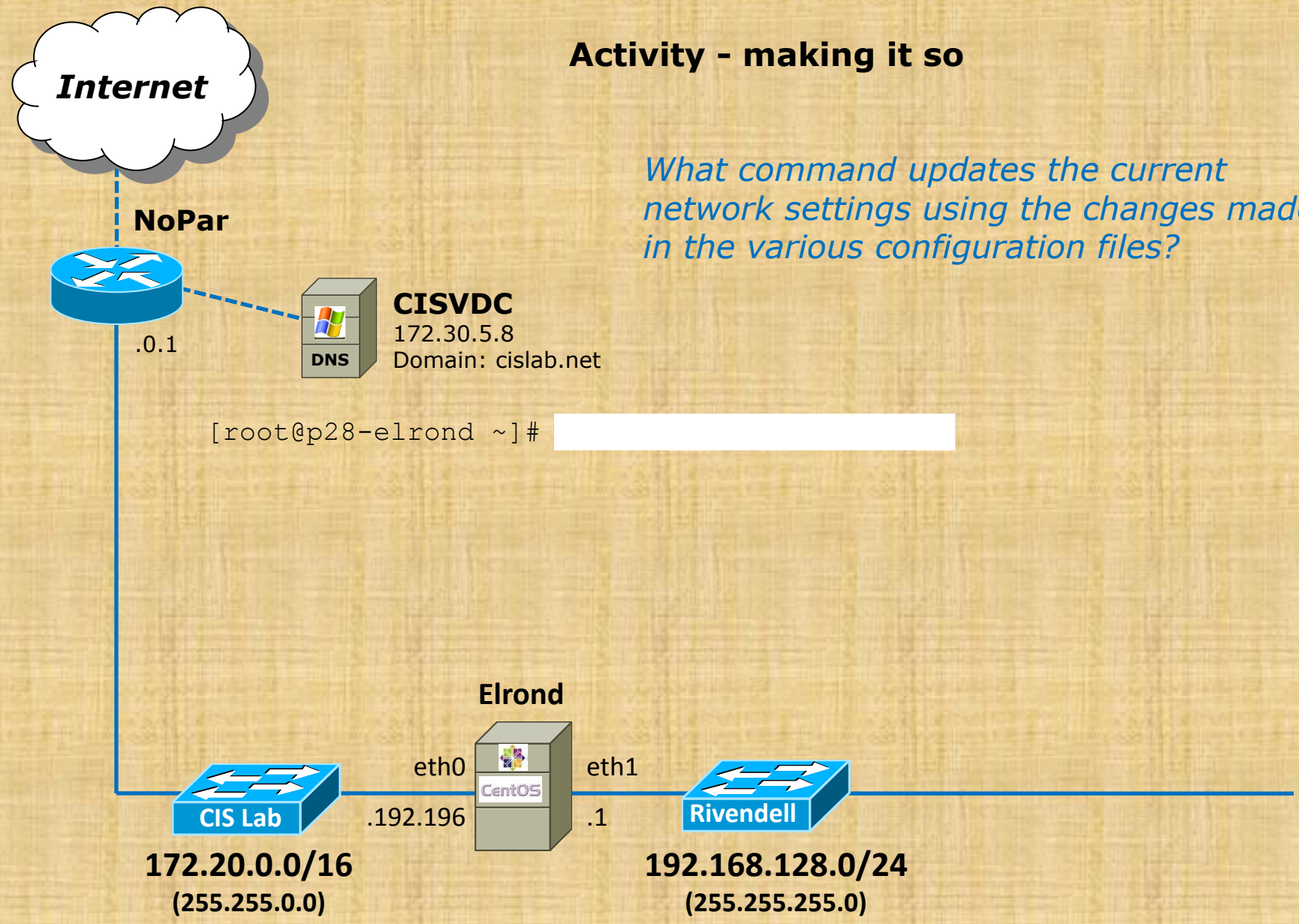
Activity - configuring DNS nameserver(s)



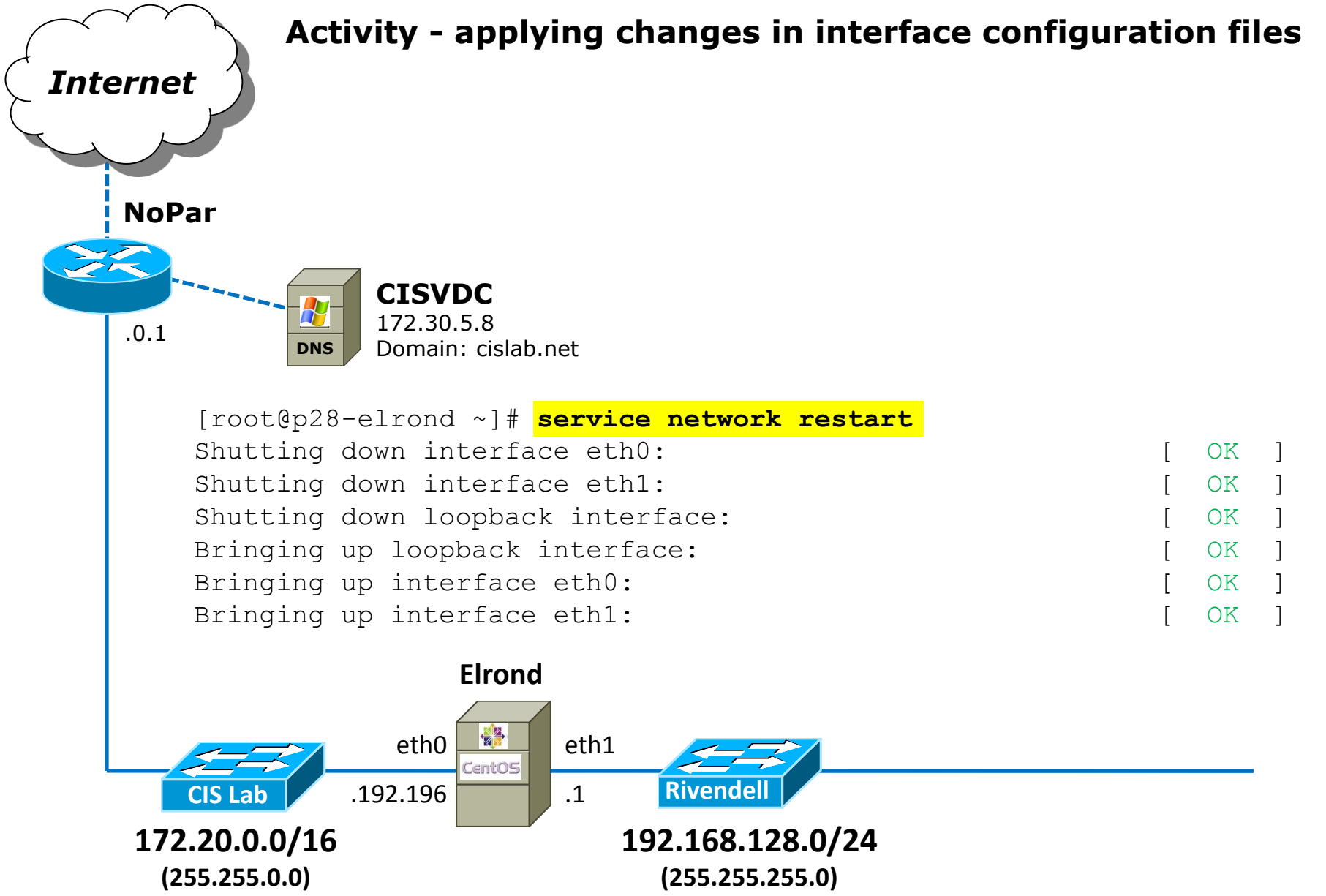

Internet
NoPar
.0.1
DNS
CISVDC
172.30.5.8
Domain: cislab.net
[root@p28-elrond ~]# cat /etc/resolv.conf
search cislab.net
nameserver 172.30.5.8
Elrond
eth0
eth1
CentOS
CIS Lab
Rivendell
.192.196
.1
172.20.0.0/16
(255.255.0.0)
192.168.128.0/24
(255.255.255.0)

Activity - making it so

What command updates the current network settings using the changes made in the various configuration files?

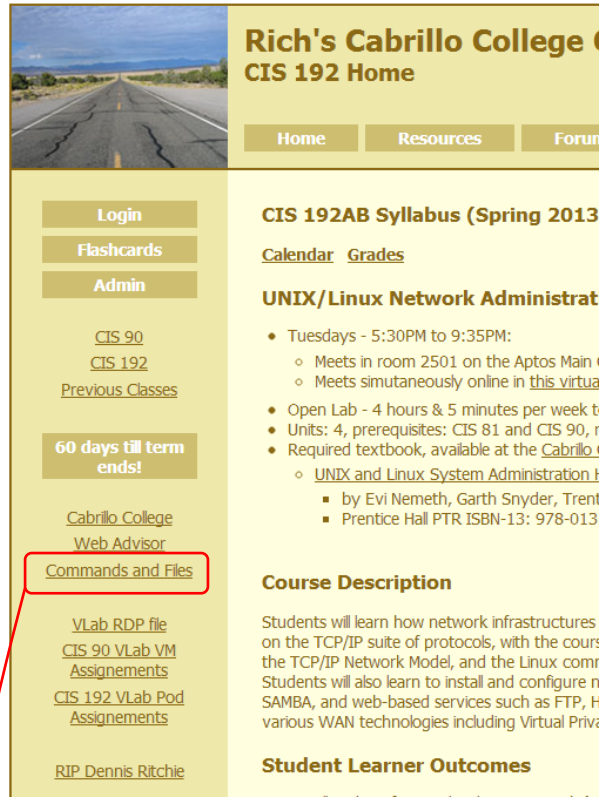


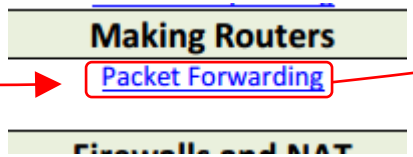
Activity - applying changes in interface configuration files



Elrond Setup

- Permanently configure ip_forward to enable packet forwarding.





Remembering how to configure packet forwarding using crib sheet

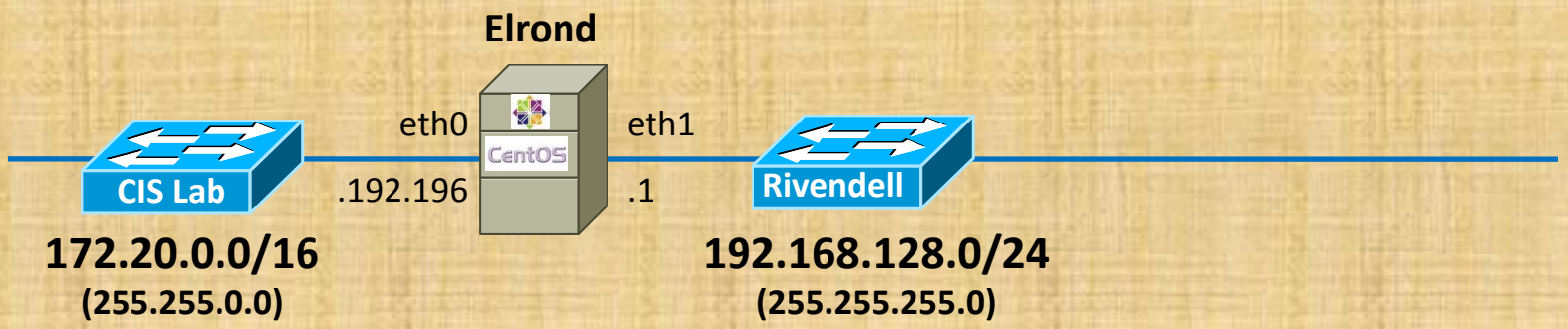
Packet forwarding	
echo 1 > /proc/sys/net/ipv4/ip_forward	Temporarily enable packet forwarding
echo 0 > /proc/sys/net/ipv4/ip_forward	Temporarily disable packet forwarding
cat /proc/sys/net/ipv4/ip_forward	Show packet forwarding status 0 = off (disabled) 1 = on (enabled)
The **/etc/sysctl.conf** file **net.ipv4.ip_forward** = *n* use *n*=0 to disable, use *n*=1 to enable For the new settings to take effect without restarting the system, use: **sysctl -p**	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

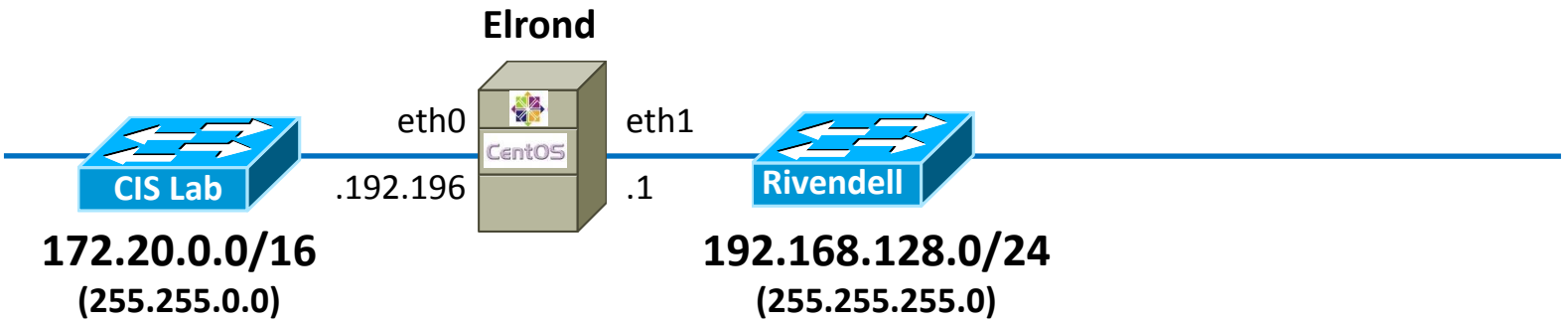
http://simms-teach.com/docs/cis192/cis192QuickRef.pdf

Activity - configuring packet forwarding

```
[root@p28-elrond ~]# cat /etc/sysctl.conf
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward =

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1
< SNIPPED >
```



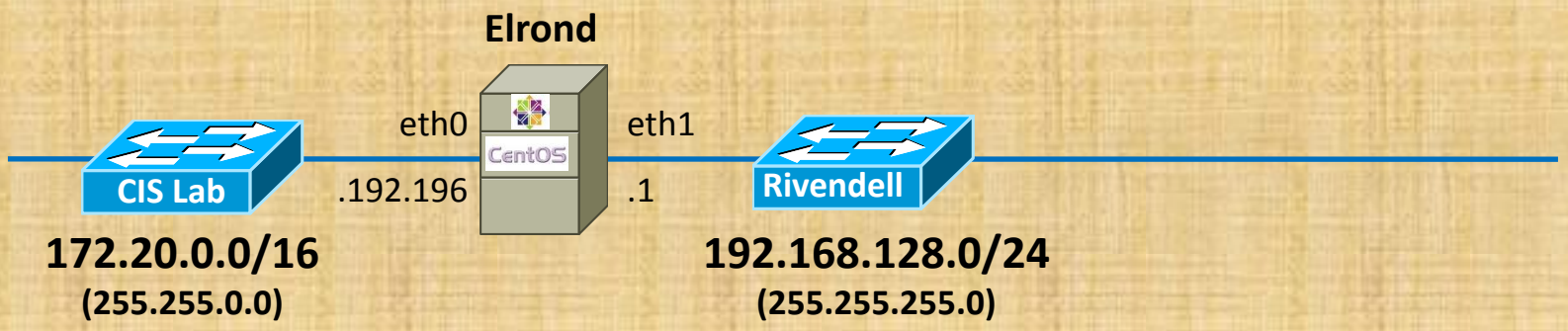
```
[root@p28-elrond ~]# cat /etc/sysctl.conf
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 1

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1
< SNIPPED >
```



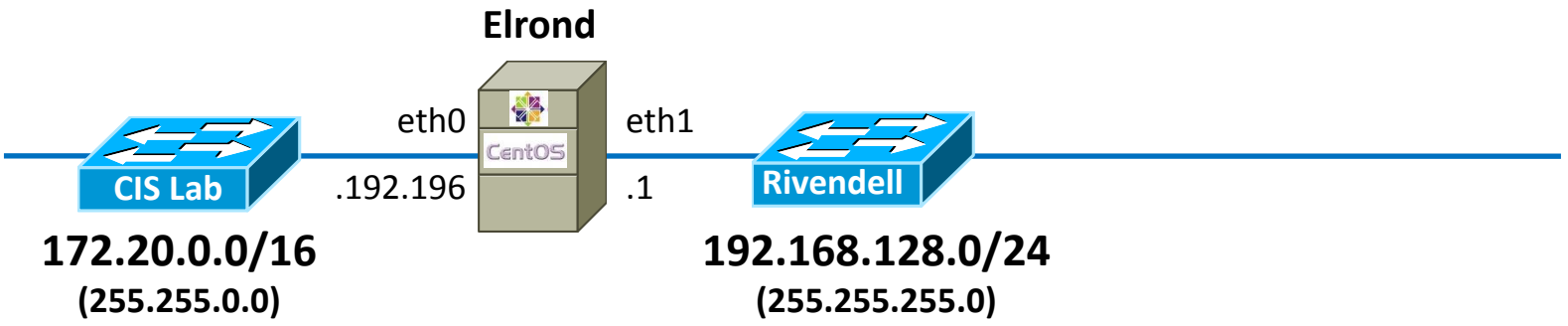
Activity - making it so

```
[root@p28-elrond ~]#
```

What command starts packet forwarding (by using the changes made in the configuration file)?

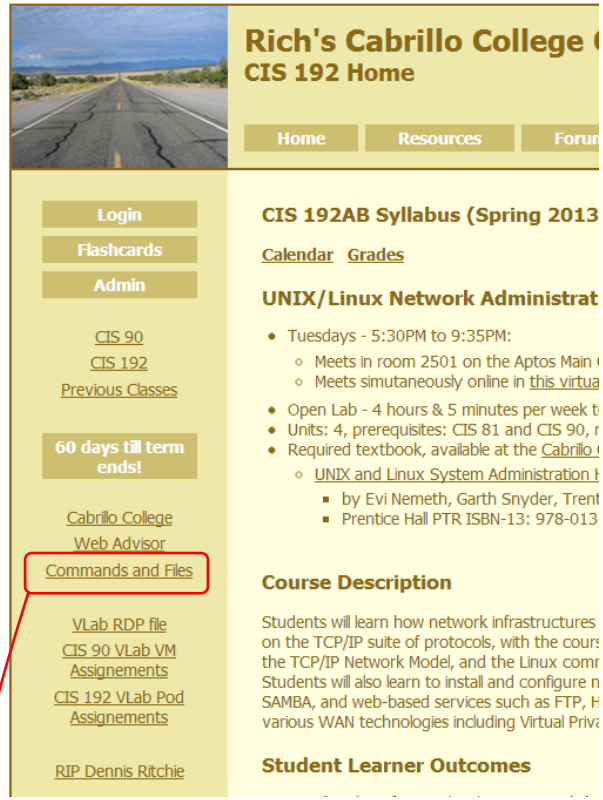


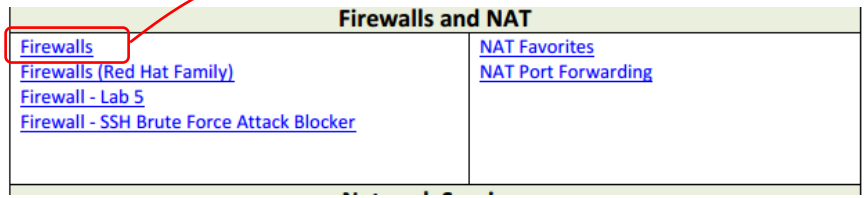
```
[root@p28-elrond ~]# sysctl -p
net.ipv4.ip_forward = 1
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.default.accept_source_route = 0
kernel.sysrq = 0
kernel.core_uses_pid = 1
net.ipv4.tcp_syncookies = 1
error: "net.bridge.bridge-nf-call-ip6tables" is an unknown key
error: "net.bridge.bridge-nf-call-iptables" is an unknown key
error: "net.bridge.bridge-nf-call-arptables" is an unknown key
kernel.msgmnb = 65536
kernel.msgmax = 65536
kernel.shmmax = 68719476736
kernel.shmall = 4294967296
```



Elrond Setup

- Permanently modify the firewall to:
  - Allow packet forwarding.
  - Allow DHCP requests with:
    iptables -I INPUT n -p udp -m udp --dport 67 -j ACCEPT
- Permanently provide NAT services for Rivendell and Mordor hosts.





Remembering how to configure the firewall to allow packet forwarding

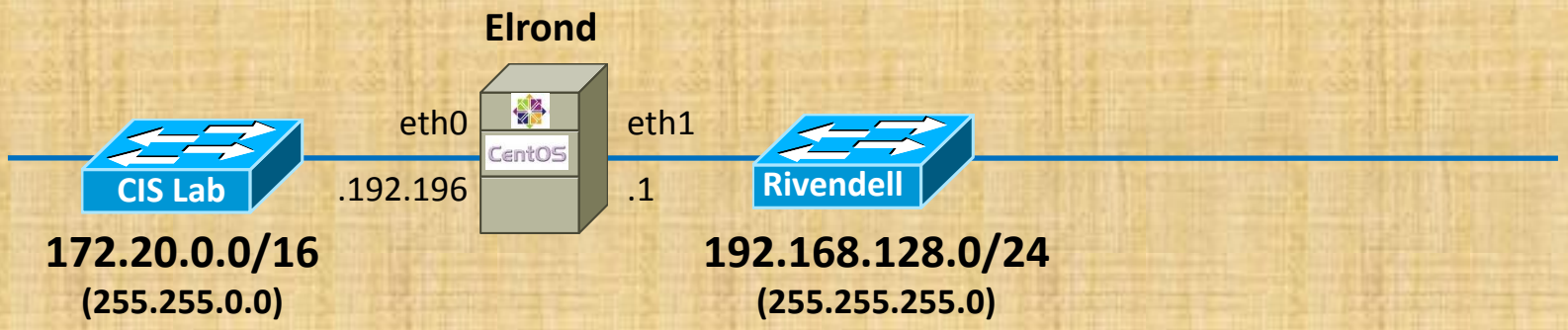
Firewalls	
iptables -L	Show the current firewall rules.
iptables -nL	Show the current firewall in numerical form, e.g. the ssh port shows as 22 instead of ssh.
iptables -nL --line-numbers	Same as above but shows line numbers.
iptables -F	Disables the firewall by flushing (deleting) all rules on all chains 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 from the /etc/sysconfig/iptables
service iptables save	Make the current firewall rules in memory permanent. The rules are saved in the /etc/sysconfig/iptables file.
iptables-save > iptables.bak	Copy the current firewall rules in memory to a file. Note: This may fail now due to SELinux (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.

http://simms-teach.com/docs/cis192/cis192QuickRef.pdf

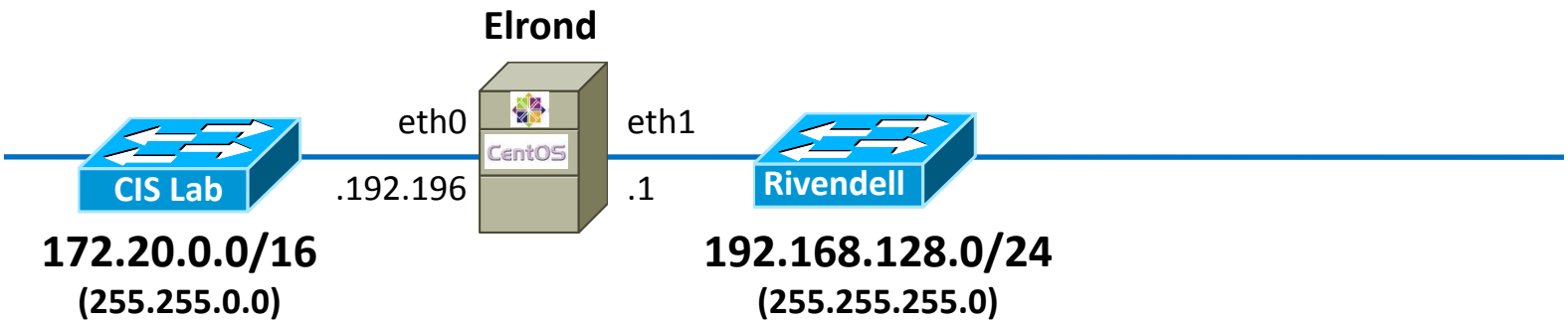
Activity - clearing the FORWARD firewall chain

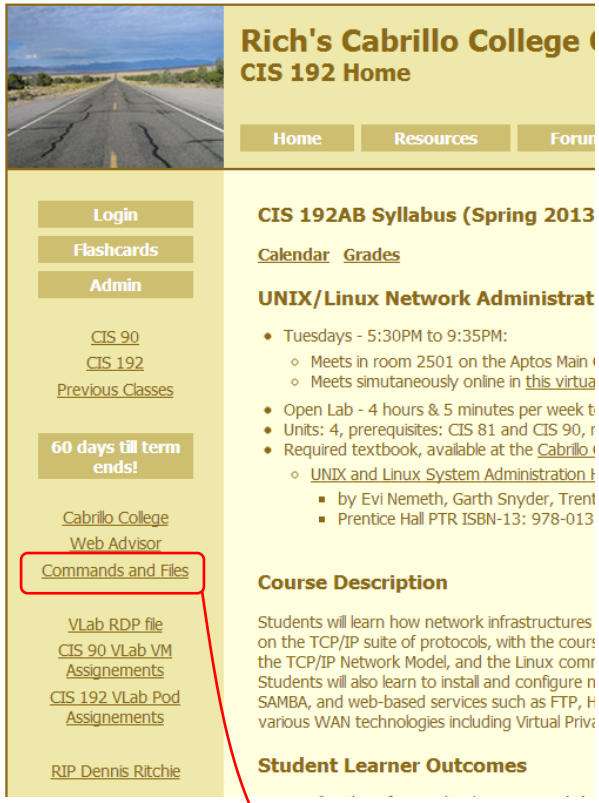
```
[root@p28-elrond ~]#
[root@p28-elrond ~]#
```

What commands delete the first rule on the iptables FORWARD chain and make the change permanent?



```
[root@p28-elrond ~]# iptables -D FORWARD 1
[root@p28-elrond ~]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[  OK  ]
[root@p28-elrond ~]#
```





Remembering how to configure the firewall provide NAT service to inside networks using the crib sheet

NAT Favorites
Example: **iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE** Adds NAT to a router whose eth0 interface is on the public side
Example: **iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to-source** *< ip address on eth0 >* Adds NAT to a router whose eth0 interface is on the public side

top

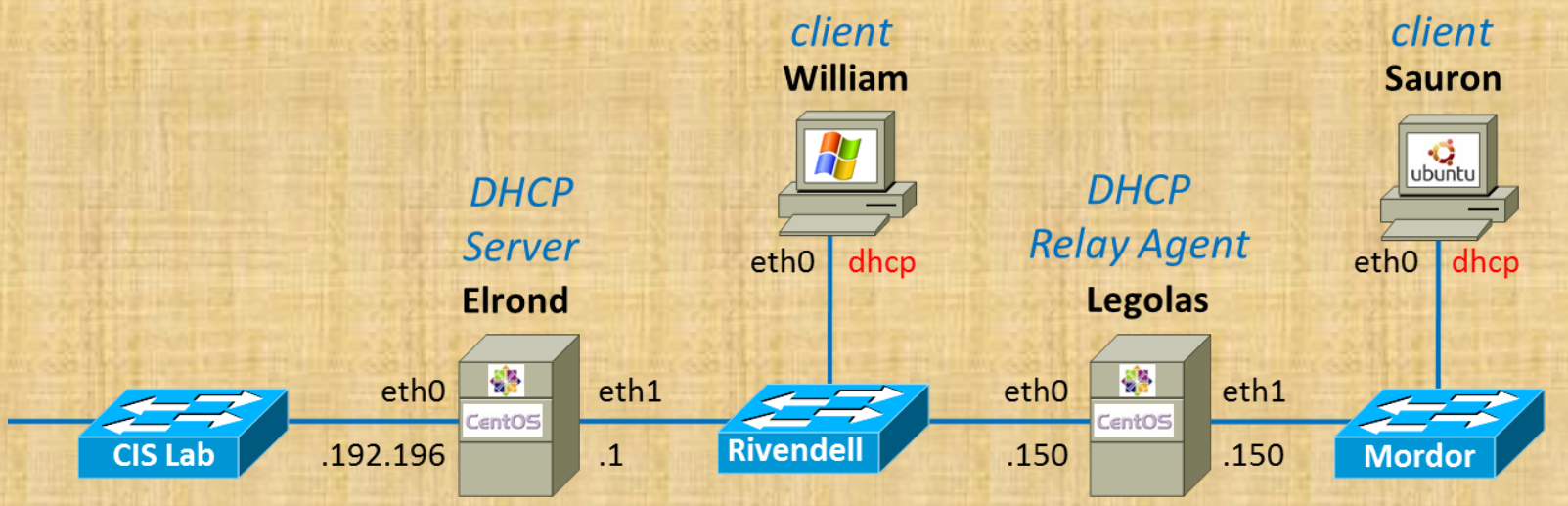
http://simms-teach.com/docs/cis192/cis192QuickRef.pdf

Firewalls and NAT	
Firewalls Firewalls (Red Hat Family) Firewall - Lab 5 Firewall - SSH Brute Force Attack Blocker	NAT Favorites NAT Port Forwarding

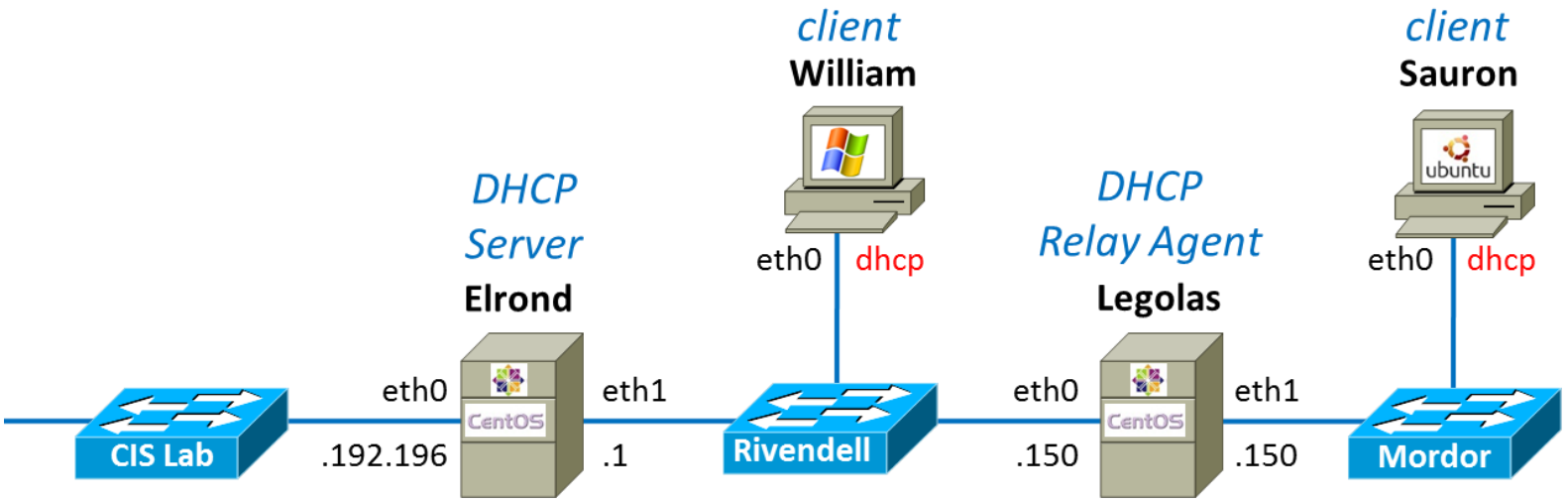
Activity - Providing NAT services with MASQUERADE

```
[root@p28-elrond ~]#
[root@p28-elrond ~]#
```

What commands on Elrond will configure permanent NAT service to enable all Rivendell and Mordor hosts to have Internet access?

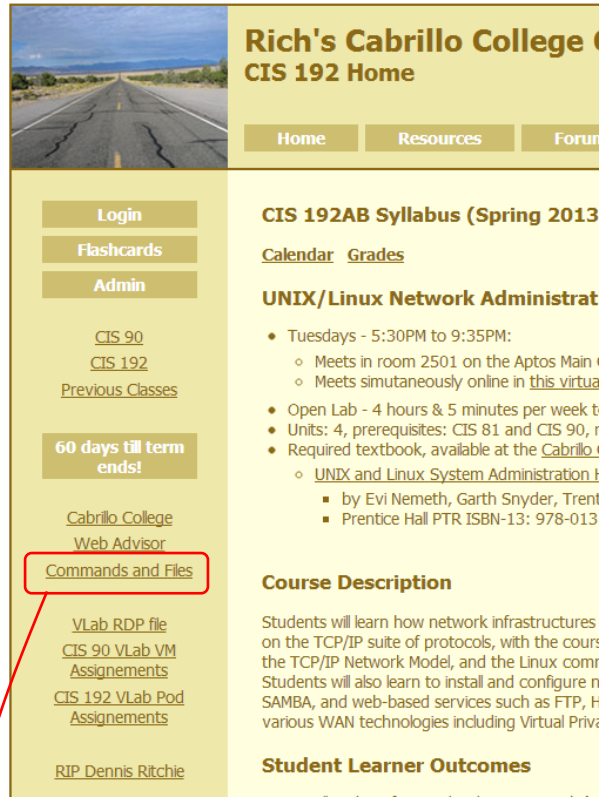


```
[root@p28-elrond ~]# iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
[root@p28-elrond ~]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[  OK  ]
[root@p28-elrond ~]#
```



Elrond Setup

- Permanently add a static route to Mordor.



redhat

Permanent Interface Configuration

Permanent Routing Table Configuration

Permanent Hostname Configuration

Remembering how to add a static route using the crib sheet

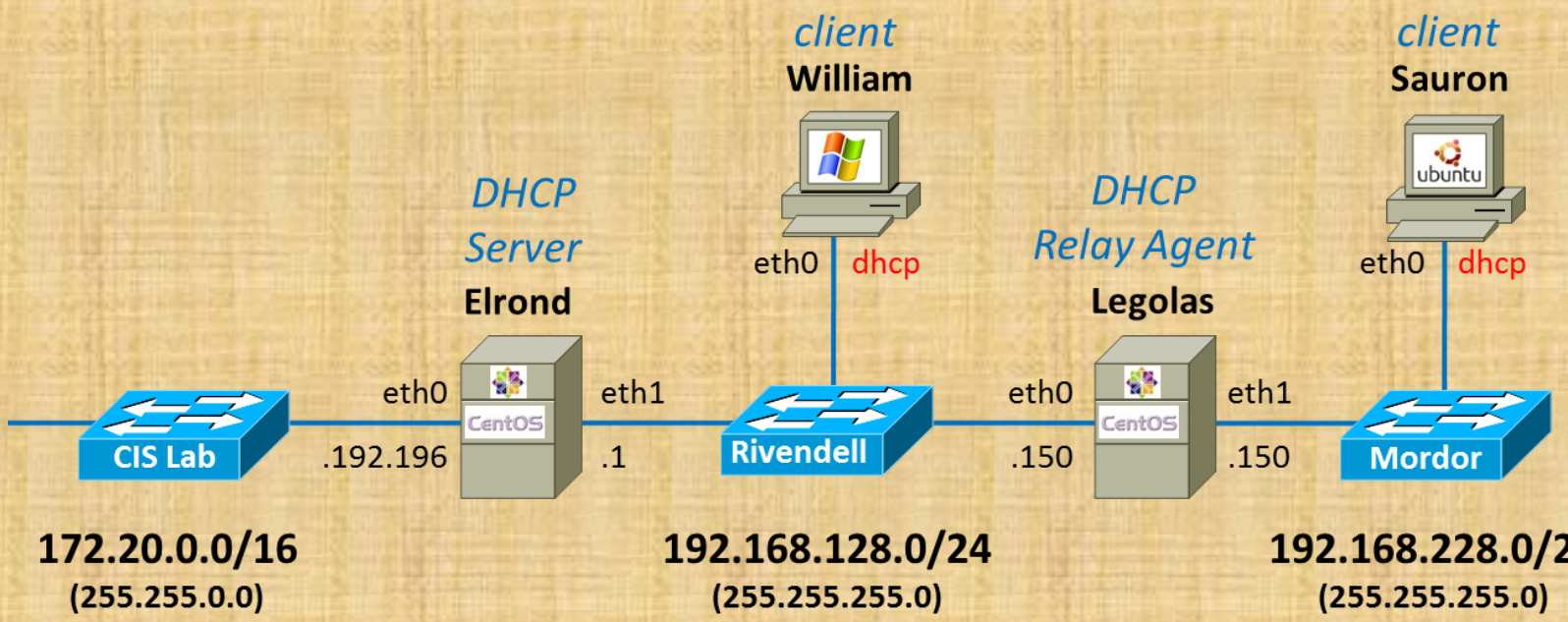
Routing table permanent configuration (Red Hat family)	
Edit **/etc/sysconfig/network** with: **GATEWAY=** *xxx.xxx.xxx.xxx*	Edit this file to add a permanent default gateway to the routing table. The new settings do not take effect until the system or network service is restarted. **Example:** **/etc/sysconfig/network** **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 192.168.20.0/22 network out the eth0 interface to the 172.30.4.250 "next hop" gateway router.

top

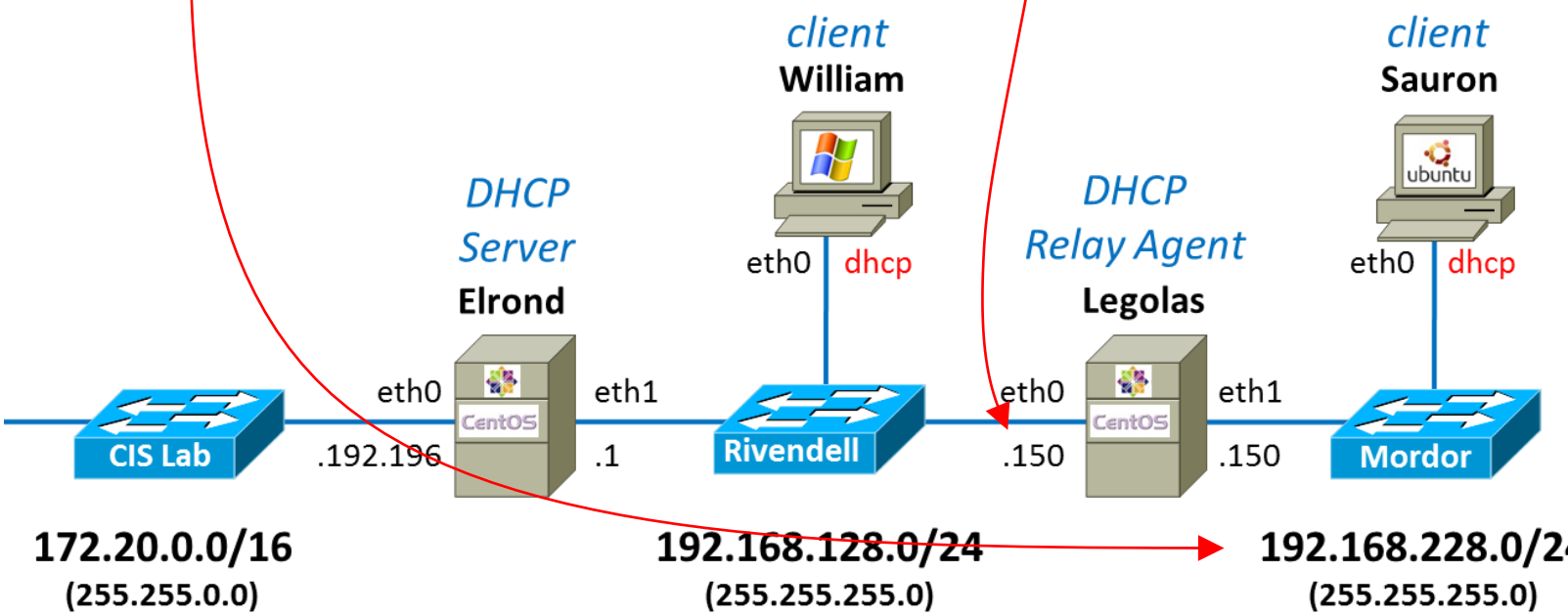
http://simms-teach.com/docs/cis192/cis192QuickRef.pdf

Activity - Add static route to Mordor

```
[root@p28-elrond ~]# cat /etc/sysconfig/network-scripts/route-eth1
                    via
```



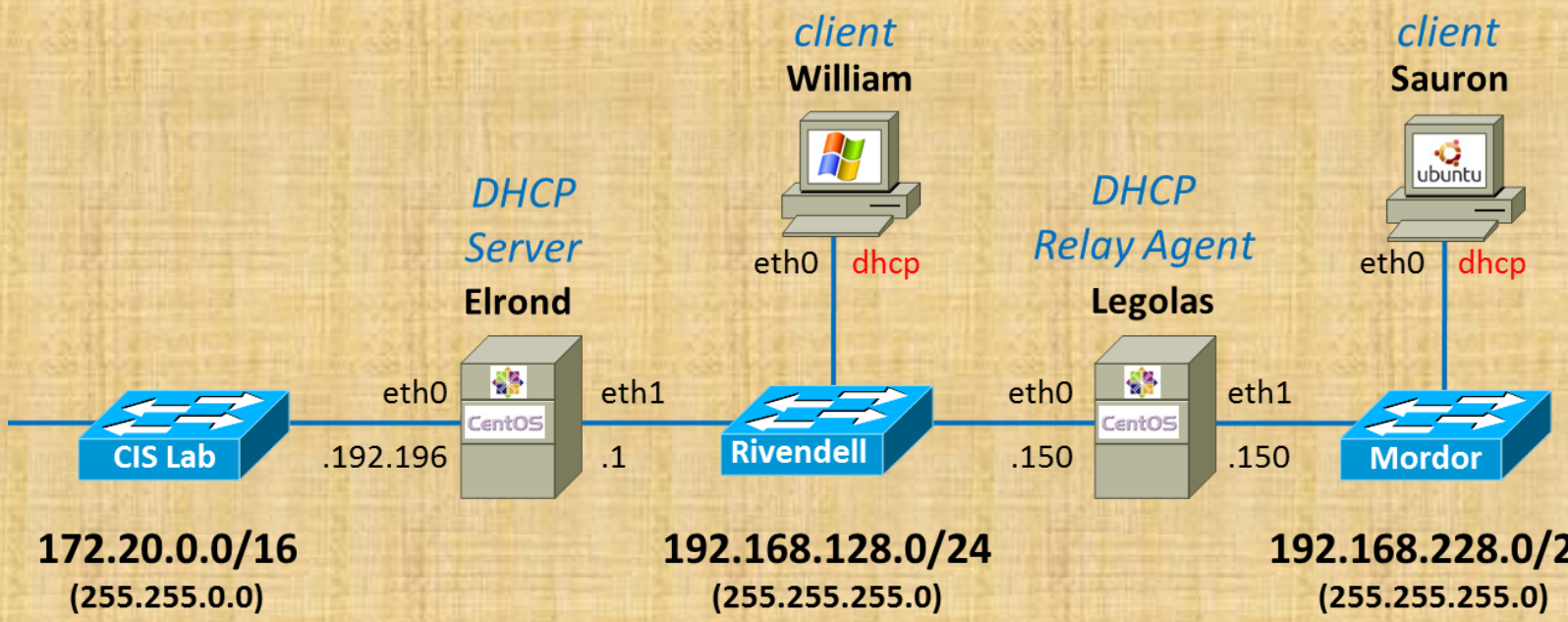
```
[root@p28-elrond ~]# cat /etc/sysconfig/network-scripts/route-eth1
192.168.228.0/24 via 192.168.128.150
```



Activity - making it so

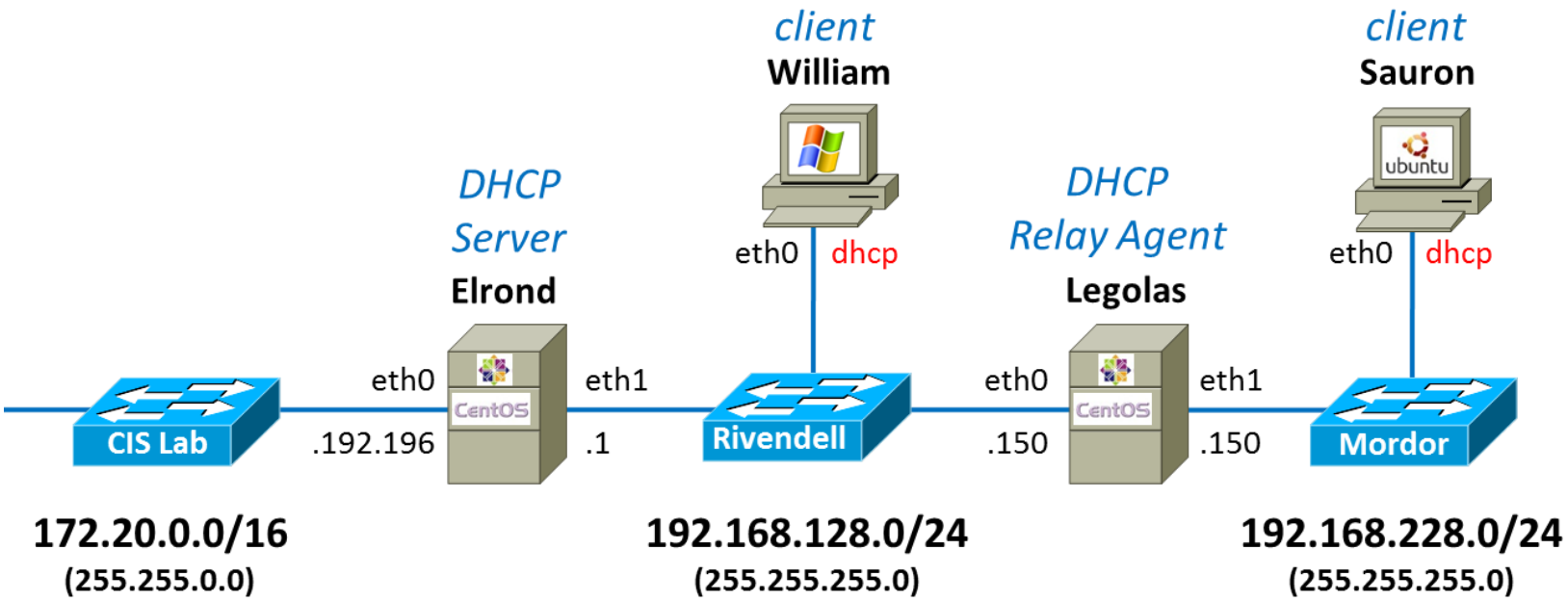
What command updates the current network settings using the changes made in the various configuration files?

```
[root@p28-elrond ~]#
```



Activity - applying changes in interface configuration files

```
[root@p28-elrond ~]# service network restart
Shutting down interface eth0:                              [  OK  ]
Shutting down interface eth1:                              [  OK  ]
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:                                [  OK  ]
Bringing up interface eth1:                                [  OK  ]
```



Understanding DHCP configuration files

elrond

DHCP

Global configuration

```
[root@p28-elrond ~]# cat /etc/dhcp/dhcpd.conf
#
# DHCP Server Configuration file.
#   see /usr/share/doc/dhcp*/dhcpd.conf.sample
#   see 'man 5 dhcpd.conf'
#

option domain-name-servers 172.30.5.8, 10.240.1.2;
default-lease-time 3600; # 60 minutes
max-lease-time 7200; # 2 hours
ddns-update-style none;
```

elrond

DHCP

Rivendell configuration

```
[root@p28-elrond ~]# cat /etc/dhcp/dhcpd.conf
< snipped >
#
#   R I V E N D E L L
#
subnet 192.168.128.0 netmask 255.255.255.0 {
  authoritative;
  option routers 192.168.128.1; # Default GW
  option subnet-mask 255.255.255.0;
  option domain-name "rivendell";
  option domain-search "cislab.net";
  range 192.168.128.50 192.168.128.99;

  # reservations
  host p28-legolas {
    hardware ethernet       00:50:56:B7:CF:0B;
    fixed-address           192.168.128.150;
  }
}
< snipped >
```

elrond

DHCP

CIS Lab configuration

```
[root@p28-elrond ~]# cat /etc/dhcp/dhcpd.conf
< snipped >
#
#   C I S   L A B
#
subnet 172.20.0.0 netmask 255.255.0.0 {
  option routers 172.20.0.1;
  option subnet-mask 255.255.0.0;
  option domain-name "cislab.net";

  range 172.20.192.198 172.20.192.202;
}
< snipped >
```

elrond

DHCP

Mordor configuration

```
[root@p28-elrond ~]# cat /etc/dhcp/dhcpd.conf
< snipped >
#
#   M O R D O R
#
subnet 192.168.228.0 netmask 255.255.255.0 {
  option routers 192.168.228.150; # Default GW
  option subnet-mask 255.255.255.0;
  option domain-name "mordor";
  option domain-search "cislab.net";

  range 192.168.228.50 192.168.228.99;
}
< snipped >
```

Selected Review

Debian/Ubuntu Network Settings

Debian/Ubuntu NIC Config (permanent)

hostname

```
root@p02-sawyer:~# cat /etc/hostname
p02-sawyer
```

*Be sure and update **/etc/hosts** after changing hostname*

Network Manager

To temporarily disable NetworkManager use:
service network-manager stop

To stop it from ever running again, edit:
/etc/init/network-manager.conf
and comment out the "start on ..." line

static

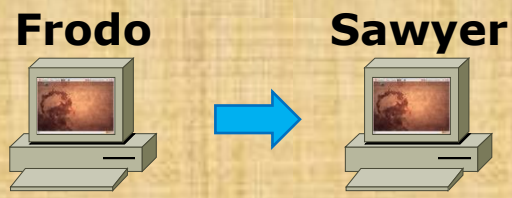
```
root@p02-sawyer:~# cat /etc/network/interfaces
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 172.20.192.20
netmask 255.255.0.0

gateway 172.20.0.1

up route add -net 192.168.128.0/24 gw
172.20.192.196

dns-search cislab.net
dns-nameservers 172.30.5.8 10.240.1.2
```

dhcp

```
root@p02-sawyer:~# cat /etc/network/interfaces
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp
```

Be sure to apply configuration file changes with:
/etc/init.d/networking restart

Exercise - Debian/Ubuntu NIC Config (permanent)



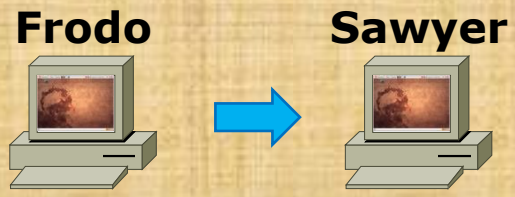
1. Backup your Lab 6 network settings:
   - cp /etc/network/interfaces /etc/network/interfaces.lab06

2. Configure Frodo permanently:
   - Hostname = p2-sawyer
   - Static IP = 172.20.90.xxx/16
   - Default gateway = 172.20.0.1
   - Static route to 192.168.128.0/24 via 172.20.192.196
   - DNS servers: 172.30.5.8 10.240.1.2 (search cislab.net)

3. Reboot Frodo

```
cis192@p02-sawyer:~$ cat /etc/hostname
p02-sawyer



cis192@p02-sawyer:~$ cat /etc/network/interfaces
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 172.20.90.20
netmask 255.255.0.0

gateway 172.20.0.1

up route add -net 192.168.128.0/24 gw 172.20.192.196

dns-search cislab.net
dns-nameservers 172.30.5.8 10.240.1.2
```

Exercise - Debian/Ubuntu NIC Config (permanent)



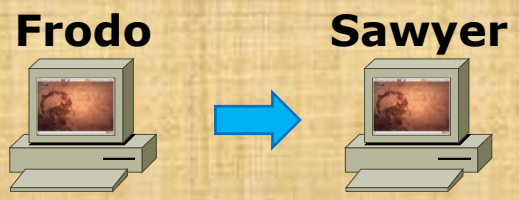
1. Add to /etc/hosts:
   - Update 127.0.1.1 entry
   - Add: 192.168.128.150 p28-legolas

2. Test your permanent network settings:
   - ping sawyer
   - ping p28-legolas
   - ping google.com

```
cis192@p02-sawyer:~$ cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       p02-frodo p02-sawyer

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

192.168.128.1 p28-elrond
192.168.128.150 p28-legolas
```

```
cis192@p02-sawyer:~$ ping p02-sawyer -c1
PING p02-frodo (127.0.1.1) 56(84) bytes of data.
64 bytes from p02-frodo (127.0.1.1): icmp_req=1 ttl=64 time=0.037 ms

cis192@p02-sawyer:~$ ping p28-legolas -c1
PING p28-legolas (192.168.128.150) 56(84) bytes of data.
64 bytes from p28-legolas (192.168.128.150): icmp_req=1 ttl=63 time=0.606 ms

cis192@p02-sawyer:~$ ping google.com -c1
PING google.com (74.125.224.135) 56(84) bytes of data.
64 bytes from nuq04s09-in-f7.1e100.net (74.125.224.135): icmp_req=1 ttl=55 time=6.31 ms
```

Exercise - Debian/Ubuntu NIC Config (permanent)



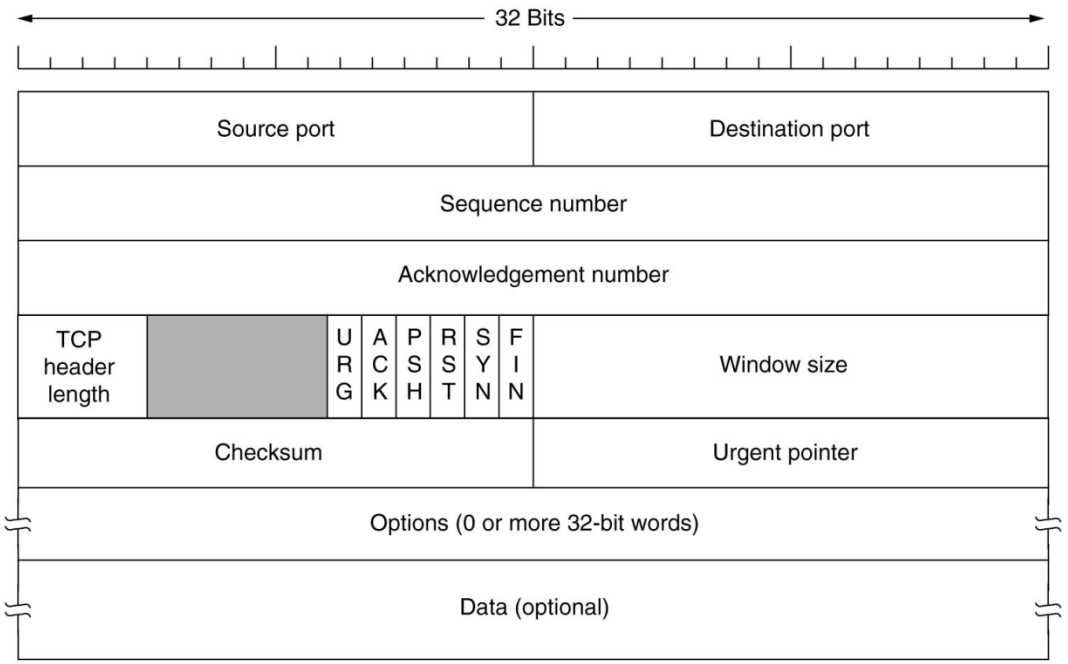
1. Restore your Lab 6 network settings:
   - **cp /etc/network/interfaces.lab06 /etc/network/interfaces**
   - Edit /etc/hostname and change back to pxx-frodo
   - Edit /etc/hosts and remove additions

TCP Connections

Transport Layer

The Transmission Control Protocol

TCP Header



Sequence and acknowledgement numbers are used for flow control.

ACK, SYN and FIN flags are used for initiating connections, acknowledging data received and terminating connections

Window size is use to communicate buffer size of recipient.

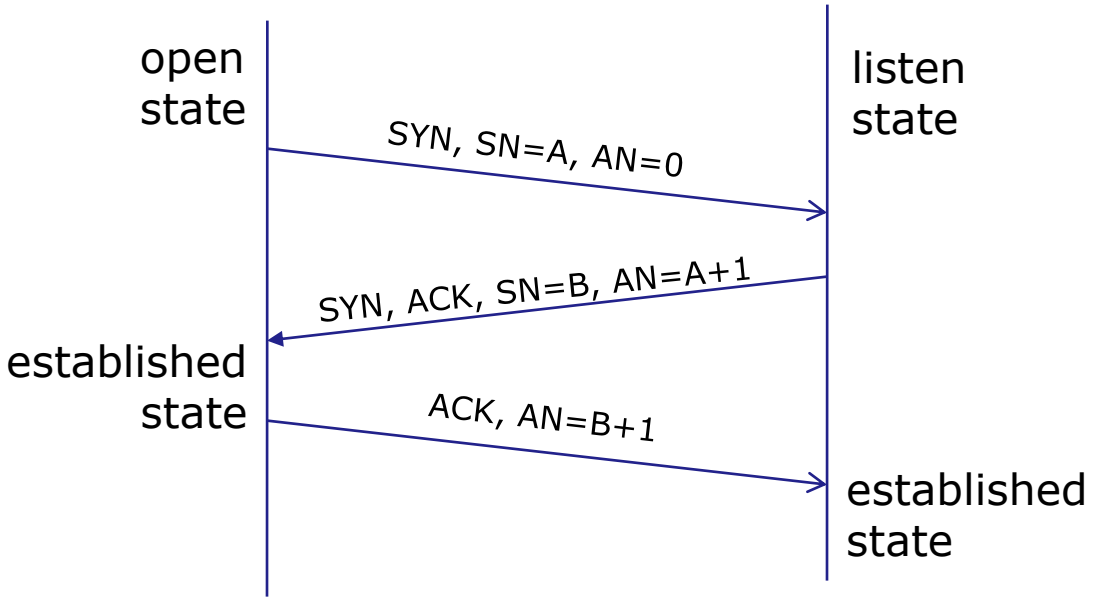
Options like SACK permit selective acknowledgement

Transport Layer

3-Way Handshake

Initiating a new TCP Connection

1. SYN
2. SYN-ACK
3. ACK



AN=Acknowledgment Number
SN=Sequence Number
ACK=ACK flag set
SYN=SYN flag set

Transport Layer

Sockets

Sockets are communication endpoints which define a network connection between two computers (RFC 793).

Source	Destination
IP Address	IP Address
Port	Port

The socket is associated to a port number so that the TCP layer can identify the application to send data to.

Application programs can read and write to a socket just like they do with files.

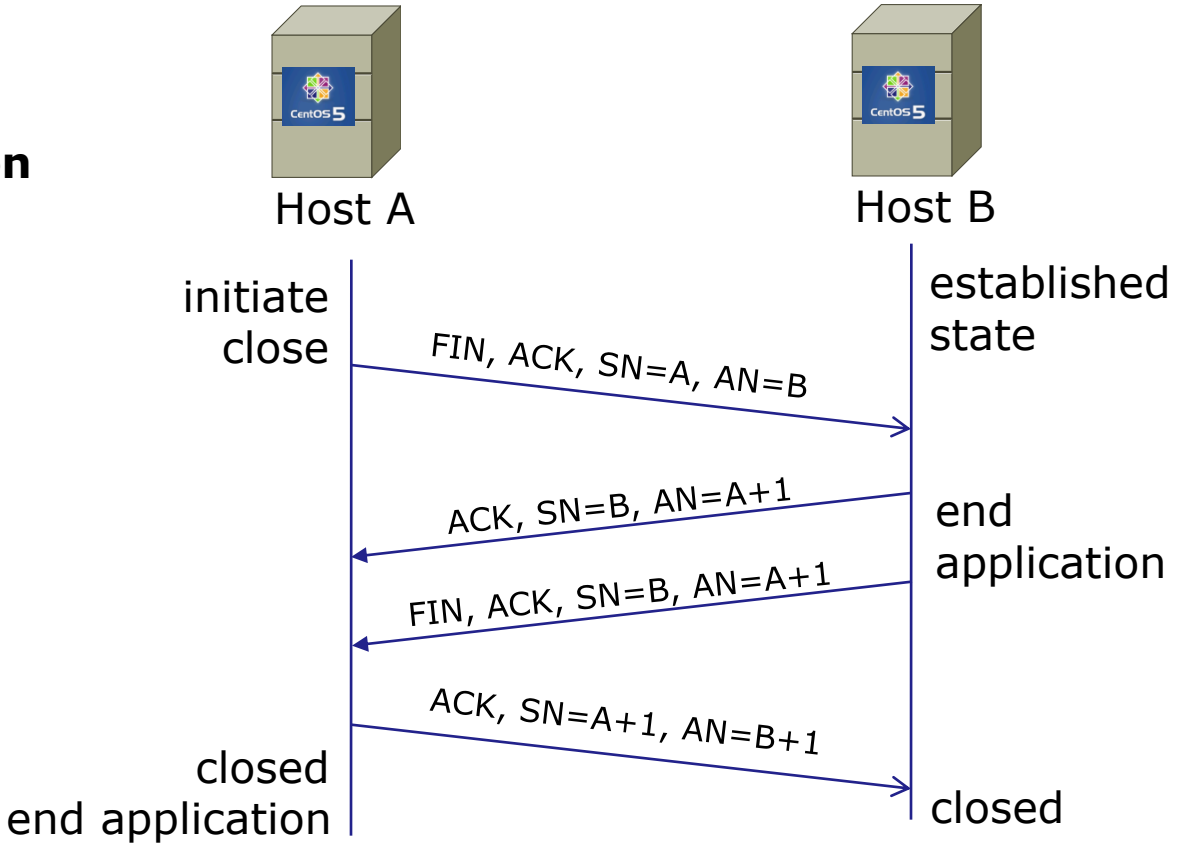
Transport Layer

Closing a TCP Connection

Four-Way Handshake

1. FIN, ACK
2. ACK
3. FIN, ACK
4. ACK

Closing with a shorter three-way handshake is also possible, where the Host A sends a FIN and Host B replies with a FIN & ACK (combining two steps into one) and Host A replies with an ACK.



AN=Acknowledgment Number
SN=Sequence Number
ACK=ACK flag set
FIN=FIN flag set

TCP connection exercise

SIP	SP	DIP	DP	Protocol	Info	*Packet Numbers*
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PASV	*1*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 227 Entering Passive Mode (192,168,2,150,200,83	*2*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=88 Ack=313 Win=5856 Len=0	*3*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 WS=5	*4*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1	*5*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=1 Win=5856 Len=0	*6*
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas	*7*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg	*8*
192.168.2.150	51283	172.30.4.83	41025	FTP-DATA	FTP Data: 18 bytes	*9*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0	*10*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=19 Win=5856 Len=0	*11*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=378 Win=5856 Len=0	*12*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0	*13*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [ACK] Seq=20 Ack=2 Win=5888 Len=0	*14*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.	*15*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=397 Win=5856 Len=0	*16*

What is the socket being used for the FTP data transfer?

TCP connection exercise

SIP	SP	DIP	DP	Protocol	Info	*Packet Numbers*
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PASV	*1*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 227 Entering Passive Mode (192,168,2,150,200,83	*2*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=88 Ack=313 Win=5856 Len=0	*3*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 WS=5	*4*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1	*5*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=1 Win=5856 Len=0	*6*
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas	*7*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg	*8*
192.168.2.150	51283	172.30.4.83	41025	FTP-DATA	FTP Data: 18 bytes	*9*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0	*10*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=19 Win=5856 Len=0	*11*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=378 Win=5856 Len=0	*12*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0	*13*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [ACK] Seq=20 Ack=2 Win=5888 Len=0	*14*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.	*15*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=397 Win=5856 Len=0	*16*

After which packet number is the FTP data transfer connection considered *Established*?

TCP connection exercise

SIP	SP	DIP	DP	Protocol	Info	*Packet Numbers*
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PASV	*1*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 227 Entering Passive Mode (192,168,2,150,200,83	*2*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=88 Ack=313 Win=5856 Len=0	*3*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 WS=5	*4*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1	*5*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=1 Win=5856 Len=0	*6*
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas	*7*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg	*8*
192.168.2.150	51283	172.30.4.83	41025	FTP-DATA	FTP Data: 18 bytes	*9*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0	*10*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=19 Win=5856 Len=0	*11*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=378 Win=5856 Len=0	*12*
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0	*13*
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [ACK] Seq=20 Ack=2 Win=5888 Len=0	*14*
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.	*15*
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=397 Win=5856 Len=0	*16*

What service makes use of the state of a connection?

Tunable kernel parameters

Tunable Kernel Parameters

tcp_fin_timeout	*how long to keep in FIN-WAIT-2 state*
tcp_keepalive_time	*how long to keep an unused connection alive*
tcp_sack	*enable/disable selective acknowledgments*
tcp_timestamps	*enable RFC 1323 definition for round-trip measurement*
tcp_window_scaling	*enable RFC 1323 window scaling*
tcp_retries1	*how many times to retry before reporting an error*
tcp_retries2	*how many times to retry before killing connection*
tcp_syn_retries	*how many times to retransmit the SYN, ACK reply*

In the same directory:

ip_forward	*enable/disable selective acknowledgments*

Found in the **/proc/sys/net/ipv4** directory

Tunable Kernel Parameters

```
[cis192@arwen ~]$ cat /etc/sysctl.conf
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0

< snipped >

[cis192@arwen ~]$

[cis192@arwen ~]$ cat /proc/sys/net/ipv4/conf/default/accept_source_route
0
[cis192@arwen ~]$ cat /proc/sys/net/ipv4/conf/default/rp_filter
1
[cis192@arwen ~]$ cat /proc/sys/net/ipv4/ip_forward
0
```

*Note: Use **sysctl -p** to put in effect any changes made to /etc/sysctl.conf*

TCP Tunable Parameters Exercise

- Revert Arwen to snapshot

For Arwen:

How many retries (tcp_retries2) will Arwen do on a TCP connection before killing it?

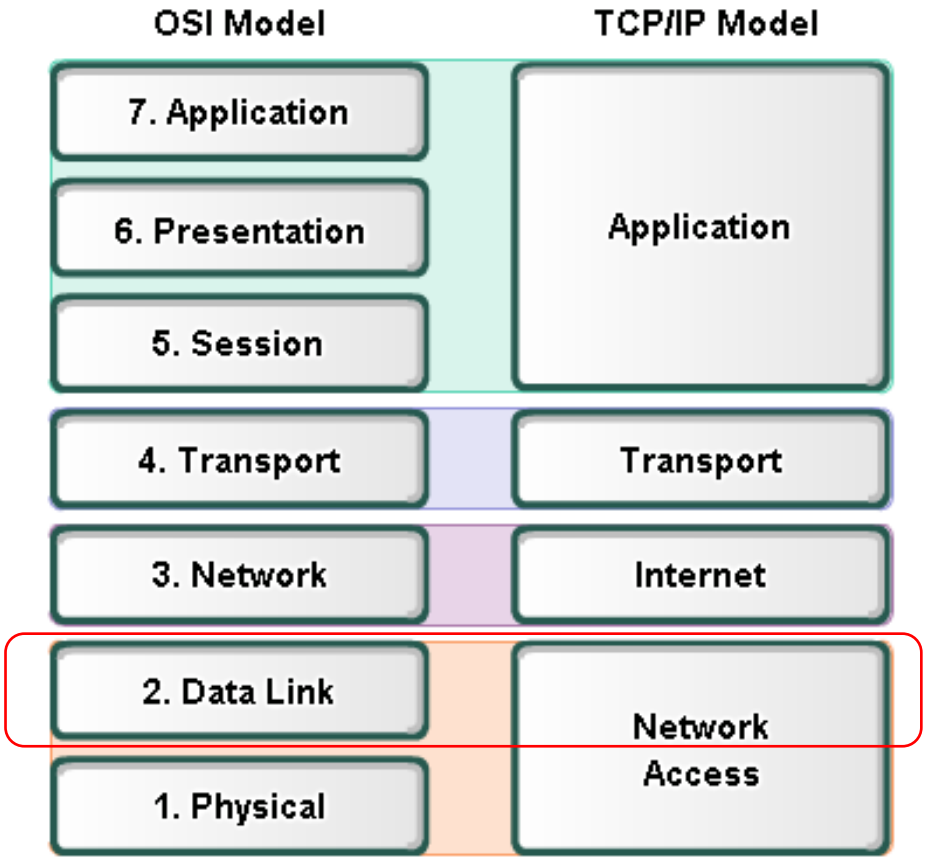
Is TCP Selective acknowledgment (tcp_sack) enabled or disabled?

How would you enable IP packet forwarding (ip_forward) temporarily?

How would you enable IP packet forwarding (ip_forward) permanently?

PPP

Layer 2 Technologies

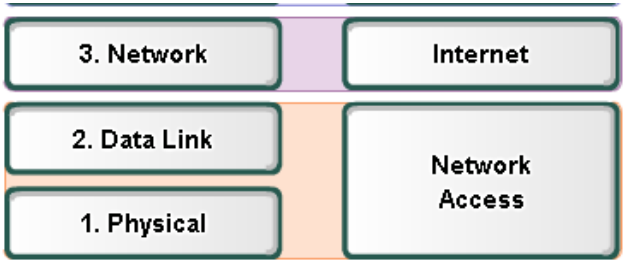


Layer 2 technologies

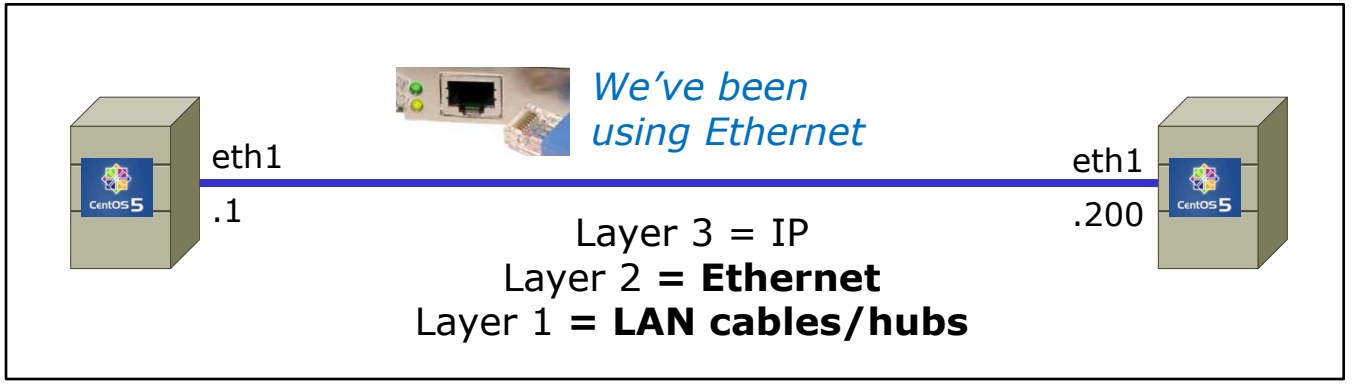
- X.25
- HIPPI
- Ethernet/IEEE 802.3
- Token Ring
- FDDI/CDDI
- Fibre Channel
- ATM
- PPP

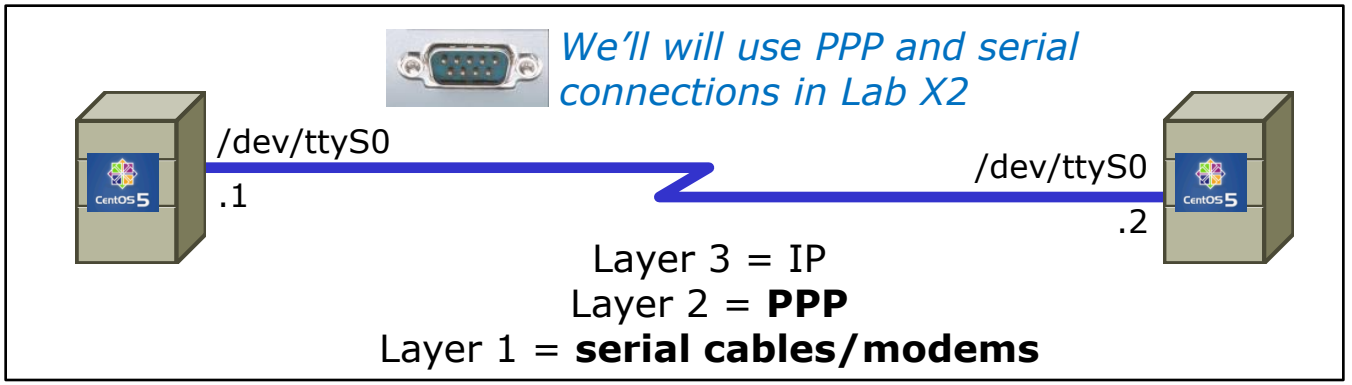
*Up to now we have been using **Ethernet** for Layer 2.*

*In LabX2 we will implement **PPP** over a serial connection.*



Layer 2 Technologies

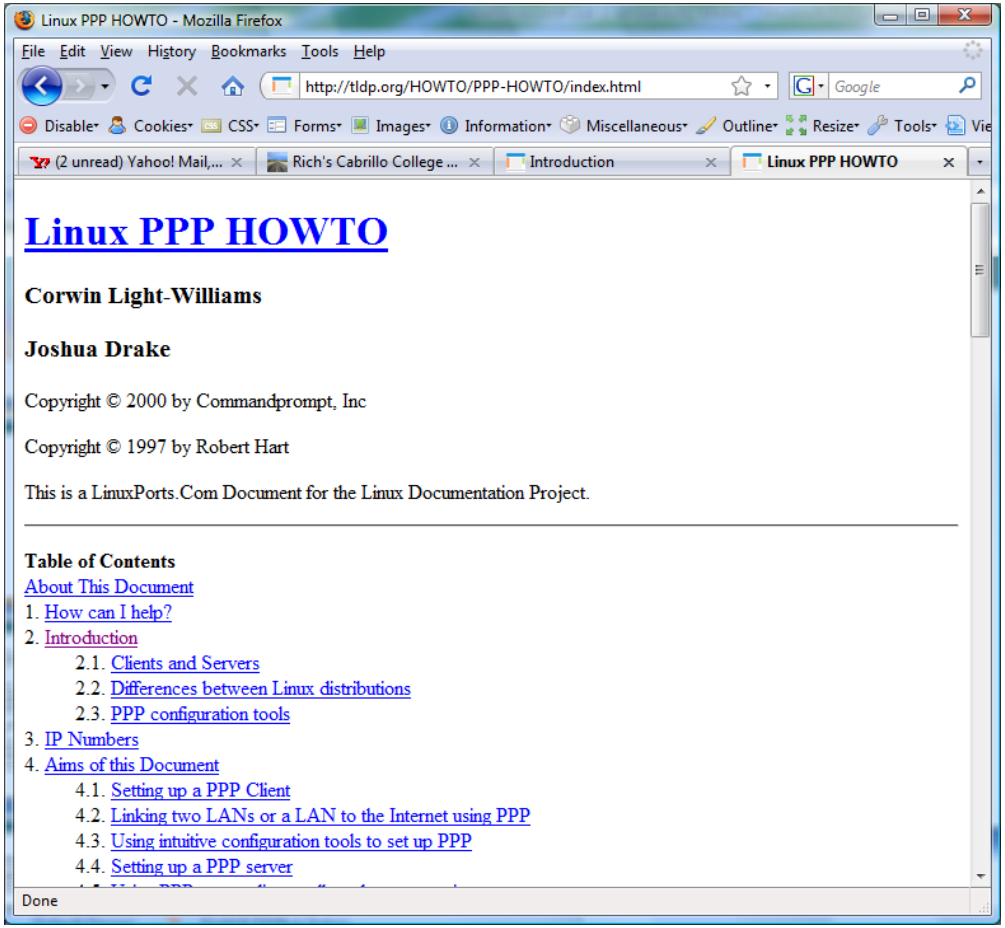




PPP is used rather than Ethernet for serial lines

PPP

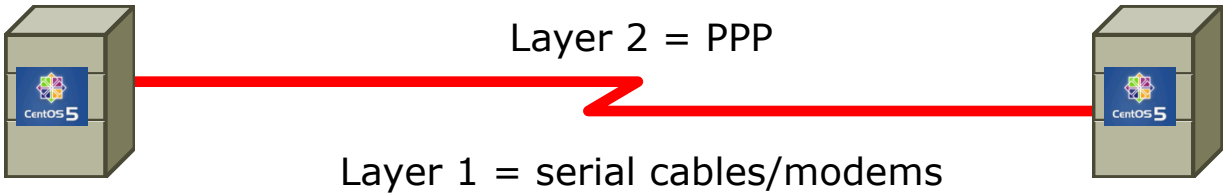
http://tldp.org/HOWTO/PPP-HOWTO/index.html



Old, buts lots of good information on PPP here!

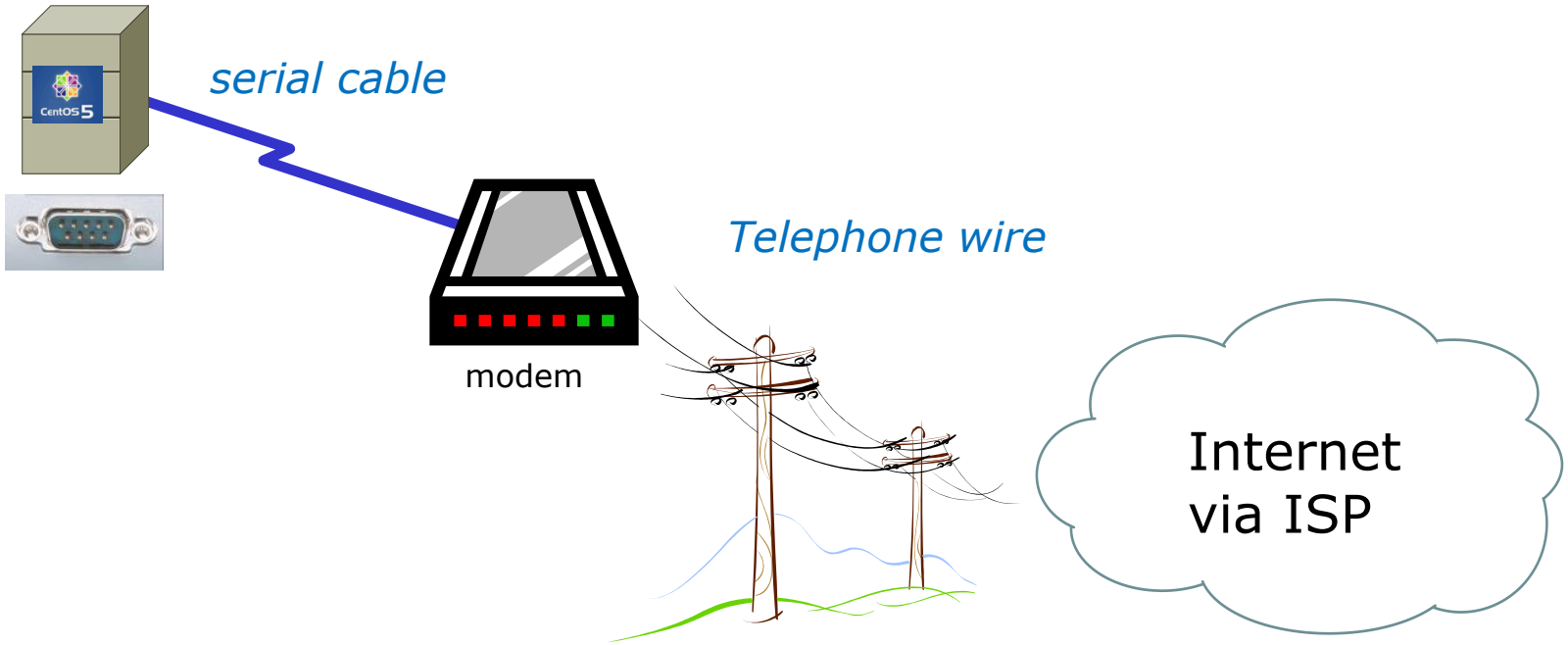
PPP

- PPP = Point to Point protocol (RFC 1331)
- A point to point network has only two hosts (at each end of the serial connection)
- PPP allows running IP and other network protocols over a serial link
- Serial links can be:
  - Direct connections using a null-modem cable
  - Using modems and telephones lines



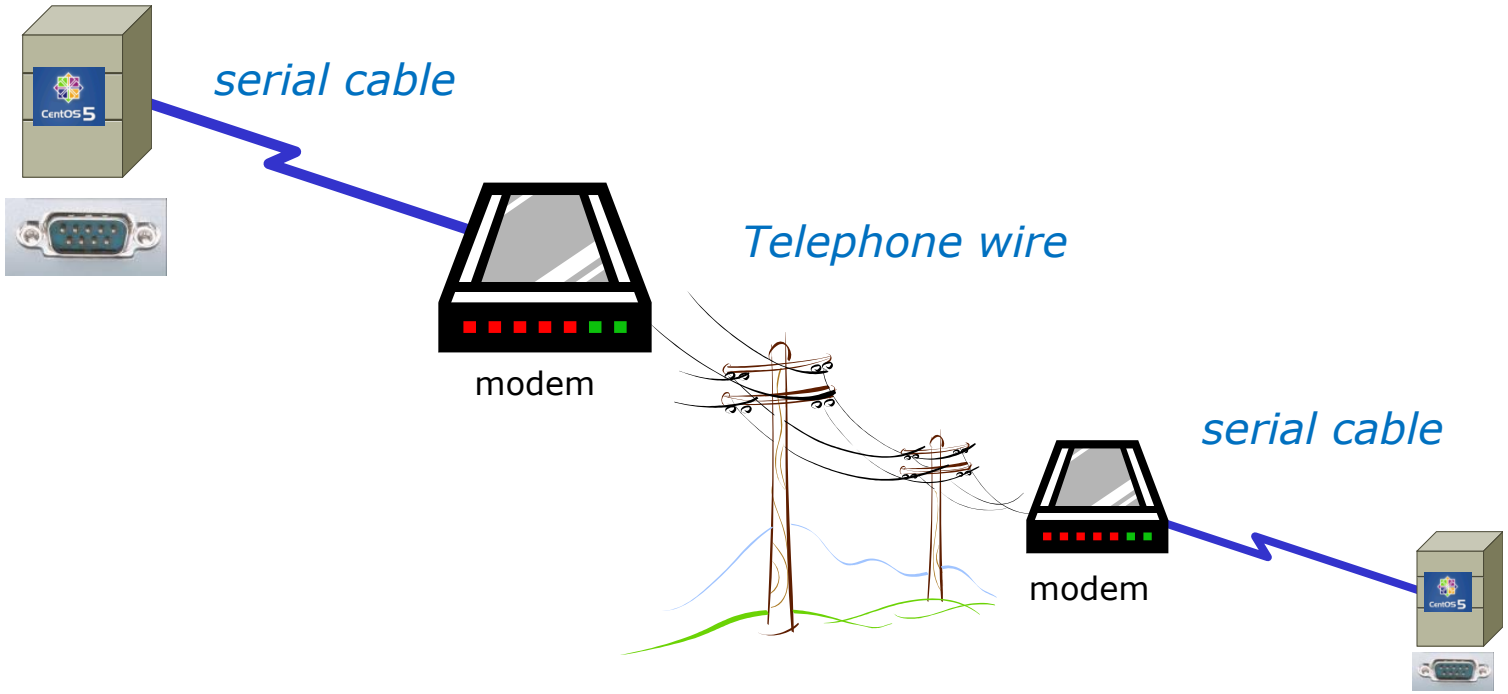
PPP

- PPP can be used as a dial-up connection to the Internet via your ISP



PPP

- PPP can be used as a WAN technology to connect LANs together



Features of PPP and SLIP

Both protocols offer the ability to send datagrams over a serial-line connection.

SLIP

- Works only with TCP/IP
- No error detection unless SLIP headers become corrupted
- Supports header compression only
- Supports only *clear-text* authentication

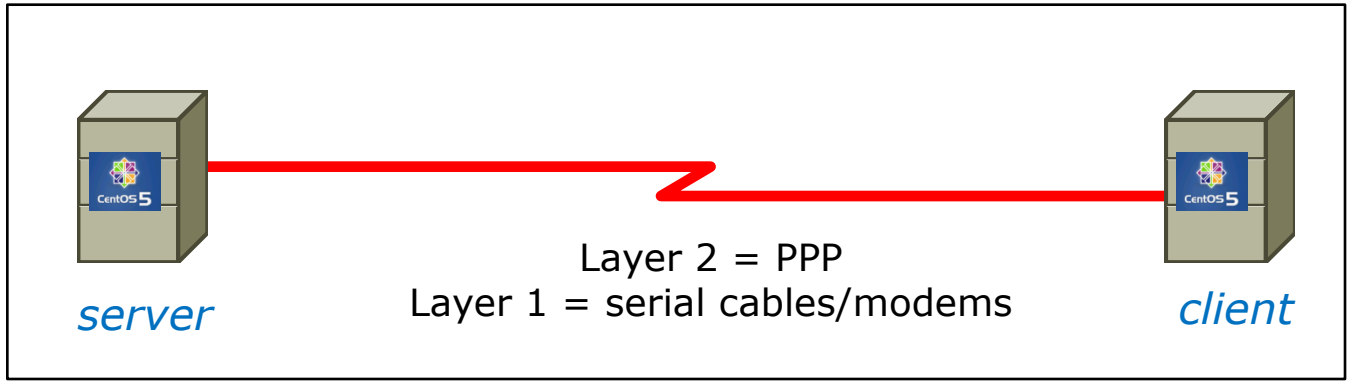
PPP

- Supports TCP/IP as well as UDP/IP, IPX/SPX, and Appletalk
- Built-in error detection
- Supports built-in data compression using the Van Jacobson compression algorithm
- Supports various authentication mechanisms e.g. PAP and CHAP

Password Authentication Protocol

Challenge Handshake Authentication Protocol

PPP Architecture



- PPP is also called a *Peer-to-Peer* protocol because there is fundamentally no difference between the server and the client.
- The ppp daemons (services) must be running on both sides of the connection.
- The computer that initiates the call is called the client, the one who answers the call is the server.

PPP Architecture

PPP runs as two major components:

1. Kernel portion - consists of and manages low-level protocols

```
[root@gothmog ~]# lsmod | grep  "^ppp"
ppp_deflate            9793  2
ppp_async             15169  1
ppp_generic           30037  6 ppp_deflate,ppp_async
```

2. User portion - consists of and manages the authentication protocols
   - **pppd** - runs the various protocols
   - **chat** - provides automated dialing management for modem connections

Both of these programs rely on command line options and/or shell scripts to configure how they operate

Setting Up PPP

- Install the software if necessary which may require building and adding kernel modules:
  - Red Hat, CentOS and Ubuntu already have PPP kernel support out of the box.
  - Make sure the pppd service has been installed:

```
[root@gothmog ~]# rpm -qa | grep ppp
ppp-2.4.4-2.el5
rp-pppoe-3.5-32.1
```

- Check your serial port
  - **setserial /dev/ttyS0** to look for modern, higher speed 16450A/16550A UART chip
  - **stty –a** to look for baud rate, parity and stop bits
- Configure your modem

setserial and stty commands

```
[root@gothmog ~]# setserial /dev/ttyS0
/dev/ttyS0, UART: 16450, Port: 0x03f8, IRQ: 4
[root@gothmog ~]#
```

Has modern UART chip

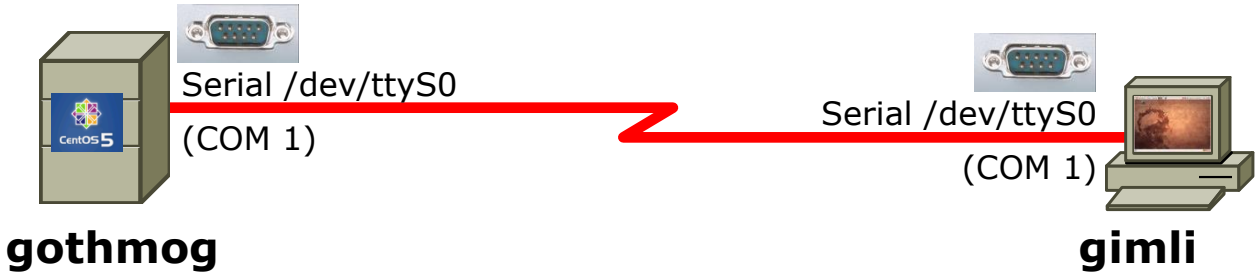
```
[root@gothmog ~]# stty -a
speed 38400 baud; rows 24; columns 80; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = M-^?; eol2 = M-^?;
swtch = M-^?; start = ^Q; stop = ^S; susp = ^Z; rprnt = ^R; werase = ^W;
lnext = ^V; flush = ^O; min = 1; time = 0;
-parenb -parodd cs8 hupcl -cstopb cread -clocal -crtscts -cdtrdsr
-ignbrk brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl ixon -ixoff
-iuclc ixany imaxbel iutf8
opost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0 bs0 vt0 ff0
isig icanon iexten echo echoe echok -echonl -noflsh -xcase -tostop -echoprt
echoctl echoke
[root@gothmog ~]#
```

38400 baud, no parity, data 8 bits, one stop bit, XON/XOFF flow control
*(use **man stty** for complete details)*

Lab X2

Exploring Serial Connections

Console port example with **minicom**



On gothmog, add this line to /etc/inittab:

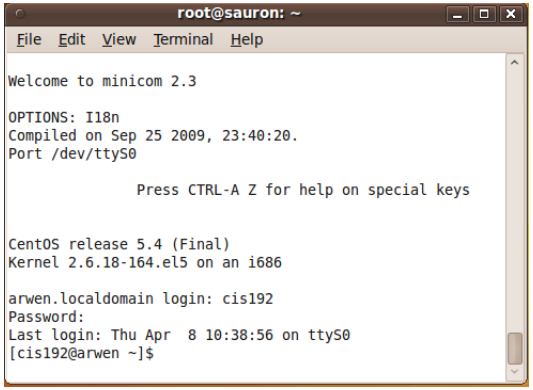
```
s1:35:respawn:/sbin/agetty 38400 ttyS0
```

This enables the login process for any connections to the serial port /dev/ttyS0

On gimli, configure minicom (a terminal emulator) to use:

- /dev/ttyS0
- 38400 baud
- 8 bits data
- no parity
- 1 stop bit
- hardware flow control

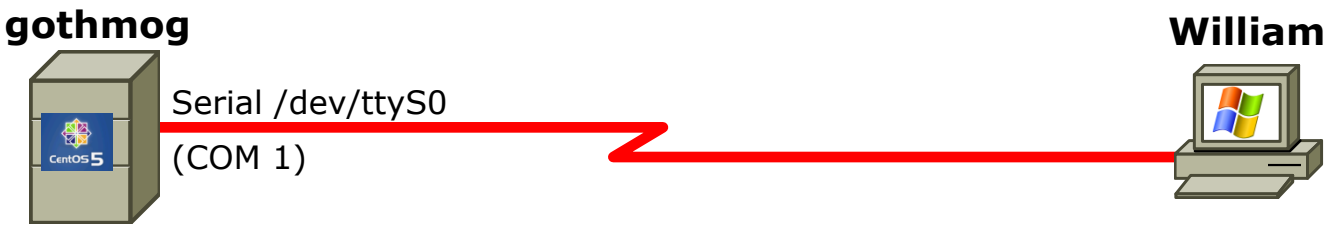
Note: PPP is not used yet in this example, just using the serial connection for console access



Login to gothmog using ***minicom -o***

Exploring Serial Connections

Console port example using **Putty**

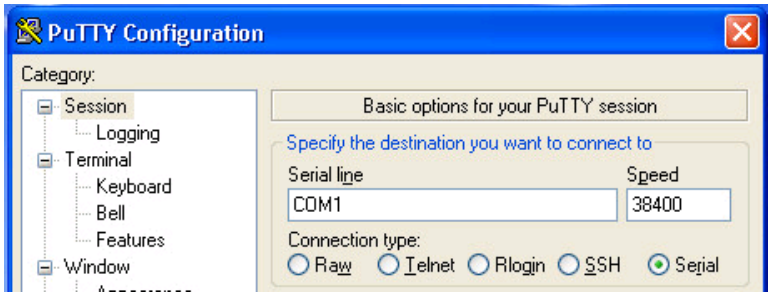


Server

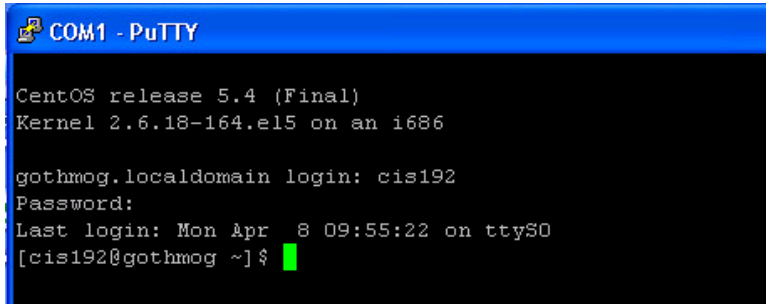
On windows station, configure Putty to use com port or pipe

On gothmog, add this line to /etc/inittab:

```
s1:35:respawn:/sbin/agetty 38400 ttyS0
```

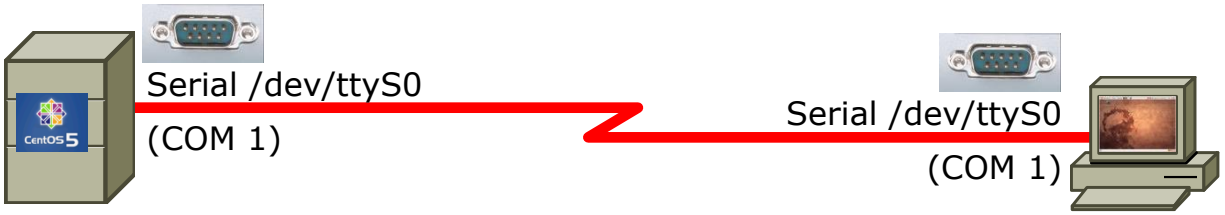


Note: PPP is not used for this, just using the serial connection for console access



Exploring Serial Connections

PPP example with bash_profile script on server, minicom on client (part 1)



gothmog **gimli**

On gothmog,
Add this line to /etc/inittab:

```
s1:35:respawn:/sbin/agetty 38400 ttyS0
```

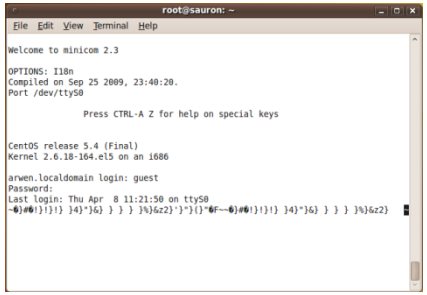
Add a user guest that runs this command at login (added to bash_profile):

```
/usr/sbin/pppd -detach crtscts
proxyarp 10.0.0.1:10.0.0.2 /dev/ttyS0
38400
```

(all on one line)

pppd must be run on both ends to establish the connection

On gimli,



Login as guest on gothmog using ***minicom -o***

Exit minicom and run this command quickly:

```
pppd -detach crtscts
/dev/ttyS0 38400 &
```

(all on one line)

Exploring Serial Connections

PPP example with bash_profile script on server, minicom on client (part 2)

On gimli,

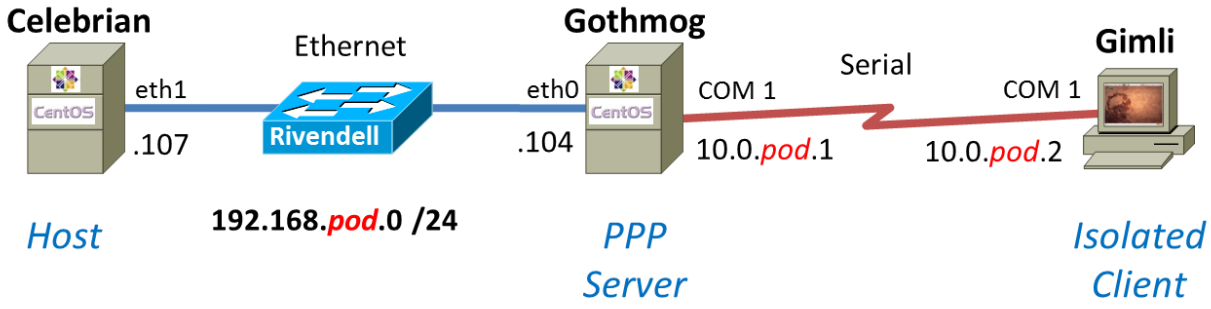
```
root@gimli:~# pppd -detach crtscts /dev/ttyS0 38400 &
[1] 1675
root@gimli:~# Using interface ppp0
Connect: ppp0 <--> /dev/ttyS0
Deflate (15) compression enabled
Cannot determine ethernet address for proxy ARP
local  IP address 10.0.0.2
remote IP address 10.0.0.1

root@gimli:~# ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:240 (240.0 B)  TX bytes:240 (240.0 B)

ppp0      Link encap:Point-to-Point Protocol
          inet addr:10.0.0.2  P-t-P:10.0.0.1  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:5 errors:0 dropped:0 overruns:0 frame:0
          TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:69 (69.0 B)  TX bytes:75 (75.0 B)
```

PPP connection established

Note both the local IP address and remote IP address are shown in ifconfig output

Lab X2

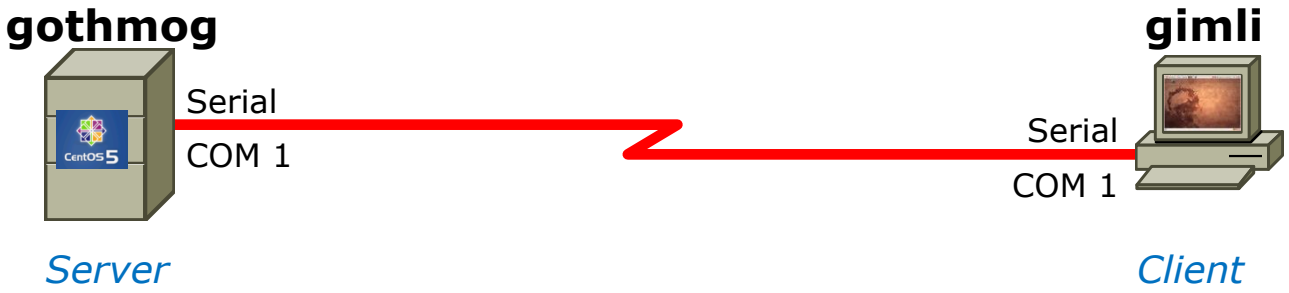
Using a named pipe for the virtual null modem cable between the two serial COM ports



Using Ethernet as the LAN layer 2 protocol over the hub and LAN cables

Using PPP as the WAN layer 2 protocol over the serial connection

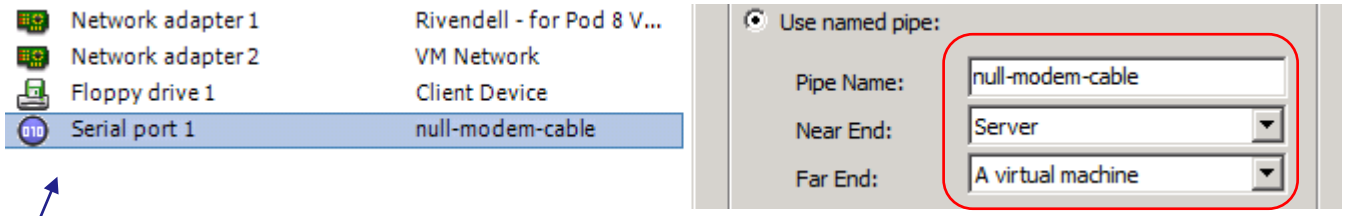
Lab X2 – Serial connections



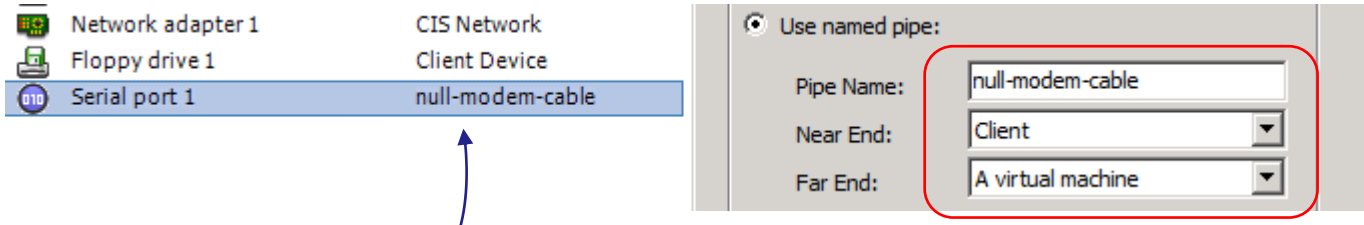
- *If you use real computers to do Lab X2, then you would connect the COM ports using a **null modem cable***
- *If you use VMware or VirtualBox VMs, then you would make a virtual serial connection using OS **pipes***

Lab X2 – Serial connections with VMware ESXi/vSphere

gothmog (the server end)



Gimli (the client end)



Use the Hardware Wizard to add serial ports

Lab X2

In the DOS/Windows world serial ports are called COM 1, COM 2, etc.

```
[root@gothmog ~]# ls -l /dev/ttyS?
crw--w---- 1 ppp  tty  4, 64 Mar 25 06:56 /dev/ttyS0
crw-rw---- 1 root uucp 4, 65 Mar 24 16:39 /dev/ttyS1
crw-rw---- 1 root uucp 4, 66 Mar 24 16:39 /dev/ttyS2
crw-rw---- 1 root uucp 4, 67 Mar 24 16:39 /dev/ttyS3
[root@gothmog ~]#
```

Each serial port is considered by UNIX to be a device. In the past these serial ports were used to connect terminals. Teletypes were terminals without a screen (had a keyboard and printer).

Note: DOS COM1 = Linux /dev/ttyS0

Lab X2 Commands

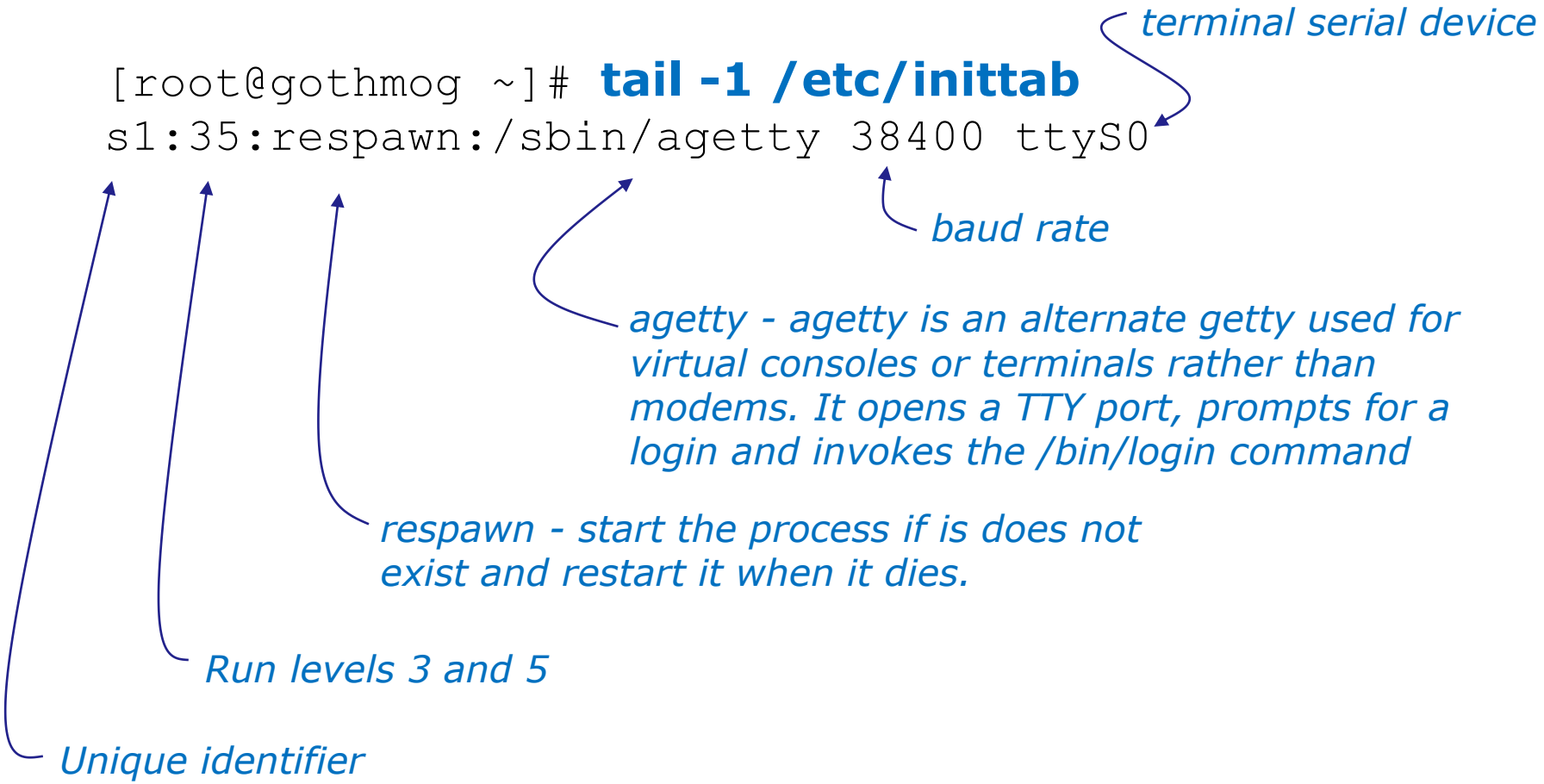
Lab X2


```
[root@gothmog ~]# setserial /dev/ttyS0
/dev/ttyS0, UART: 16550A, Port: 0x03f8, IRQ: 4
[root@gothmog ~]#
```

The setserial command sets or reports on serial port configuration.

Lab X2

Handling the login process on the pppd server



Lab X2

Handling the login process on the pppd server

```
[root@gothmog ~]# telinit q
```

*Tells init to reread the **/etc/inittab** file after making changes*

Lab X2

```
[root@gothmog ~]# chmod u+s /usr/sbin/pppd
[root@gothmog ~]# ls -l /usr/sbin/pppd
-r-sr-xr-x 1 root root 312236 Mar 14  2007 /usr/sbin/pppd
```

*This sets a special permission called the **setuid** bit. This allows users to run an executable with the permissions of the executable's owner.*

```
[root@gothmog ~]# stat /usr/sbin/pppd
  File: `/usr/sbin/pppd'
  Size: 312172          Blocks: 632        IO Block: 4096
regular file
Device: fd00h/64768d    Inode: 308263      Links: 1
Access: (4555/-r-sr-xr-x)  Uid: (    0/    root)   Gid: (
0/    root)
Access: 2010-04-04 03:20:12.000000000 -0700
Modify: 2009-01-20 20:27:13.000000000 -0800
Change: 2010-04-04 19:45:23.000000000 -0700
```

*FYI, the **stat** command provides additional inode information about a file than a long listing (ls –l) does.*

Lab X2

minicom
is a small terminal emulator with a dialing capability

```
[root@gothmog ~]# minicom -s
                  -o
```

-s option is used to setup defaults which are saved in ***/etc/minicom/minirc.dfl***

-o option prevents initialization. Useful for restarting a session

Use ***apt-get install minicom*** *to install on Ubuntu*

Lab X2

minicom
is a small terminal emulator with a dialing capability

```
root@gimli:~# minicom -s
```

Select choice and hit Enter

```
+-----[configuration]------+
| Filenames and paths      |
| File transfer protocols  |
| Serial port setup        |
| Modem and dialing        |
| Screen and keyboard      |
| Save setup as dfl        |
| Save setup as..          |
| Exit                     |
| Exit from Minicom        |
+--------------------------+
```

Use Escape to go back up one level
Use Enter to make sections
Use Letters to make choices

```
+-----------------------------------------------------------------------+
| A -    Serial Device      : /dev/tty8                                 |
| B - Lockfile Location     : /var/lock                                 |
| C -   Callin Program      :                                           |
| D -  Callout Program      :                                           |
| E -    Bps/Par/Bits       : 115200 8N1                                |
| F - Hardware Flow Control : Yes                                       |
| G - Software Flow Control : No                                        |
|                                                                       |
|    Change which setting?                                              |
+-----------------------------------------------------------------------+
        | Screen and keyboard      |
        | Save setup as dfl        |
        | Save setup as..          |
        | Exit                     |
        | Exit from Minicom        |
        +--------------------------+
```

Select option and type new configuration value

Lab X2

```
+-----------------------------------------------------------------------+
| A -    Serial Device      : /dev/ttyS0                                |
| B - Lockfile Location     : /var/lock                                 |
| C -   Callin Program      :                                           |
| D -  Callout Program      :                                           |
| E -    Bps/Par/Bits       : 38400 8N1                                 |
| F - Hardware Flow Control : Yes                                       |
| G - Software Flow Control : No                                        |
|                                                                       |
|    Change which setting?                                              |
+-----------------------------------------------------------------------+
        | Screen and keyboard      |
        | Save setup as dfl        |
        | Save setup as..          |
        | Exit                     |
        | Exit from Minicom        |
        +--------------------------+
```

When finished use Esc to exit menu

```
+-----[configuration]------+
| Filenames and paths      |
| File transfer protocols  |
| Serial port setup        |
| Modem and dialing        |
| Screen and keyboard      |
| Save setup as dfl        |
| Save setup as..          |
| Exit                     |
| Exit from Minicom        |
+--------------------------+
```

Use Save setup as dfl to save

```
+-----[configuration]------+
| Filenames and paths      |
| File transfer protocols  |
| Serial port setup        |
| Modem and dialing        |
| Screen and keyboard      |
| Save setup as dfl        |
| Save setup as..          |
| Exit                     |
| Exit from Minicom        |
+--------------------------+
```

Use Exit from Minicom to exit

Lab X2

```
root@gimli:~# minicom -o

Welcome to minicom 2.3

OPTIONS: I18n
Compiled on Oct 24 2008, 06:37:44.
Port /dev/ttyS0

               Press CTRL-A Z for help on special keys


CentOS release 5.2 (Final)
Kernel 2.6.18-92.1.22.el5 on an i686

gothmog.localdomain login: cis192
Password:
Last login: Tue Mar 24 17:27:32 on ttyS0
[cis192@gothmog ~]$ hostname
gothmog.localdomain
[cis192@gothmog ~]$



CentOS release 5.2 (Final)
Kernel 2.6.18-92.1.22.el5 on an i686

gothmog.localdomain login:

                          +----------------------+
                          | Leave without reset? |
                          |     Yes       No     |
                          +----------------------+


 CTRL-A Z for help |115200 8N1 | NOR | Minicom 2.3    | VT102 | Online 00:01
 root@gimli:~#
```

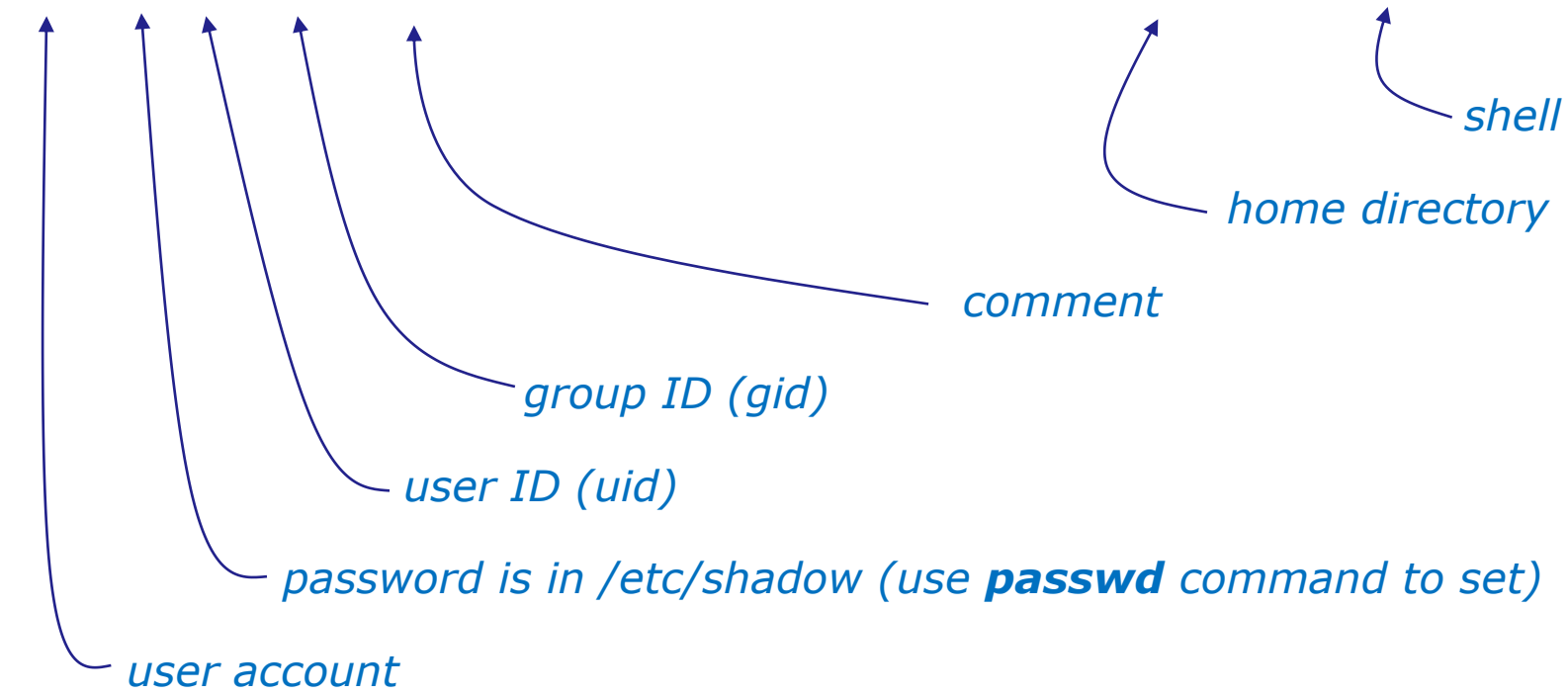
Example session using minicom –o to log into gothmog at other end of the serial connection

Ctrl-A z q
(press Ctrl and A keys together, then z then q)

Lab X2

Creating a new user account on the server side with **useradd**

```
[root@gothmog ~]# useradd -c "Guest account for serial access" guest
[root@gothmog ~]# cat /etc/passwd | grep guest
guest:x:501:501:Guest account for serial access:/home/guest:/bin/bash
```



Lab X2

The .bash_profile file for the guest user

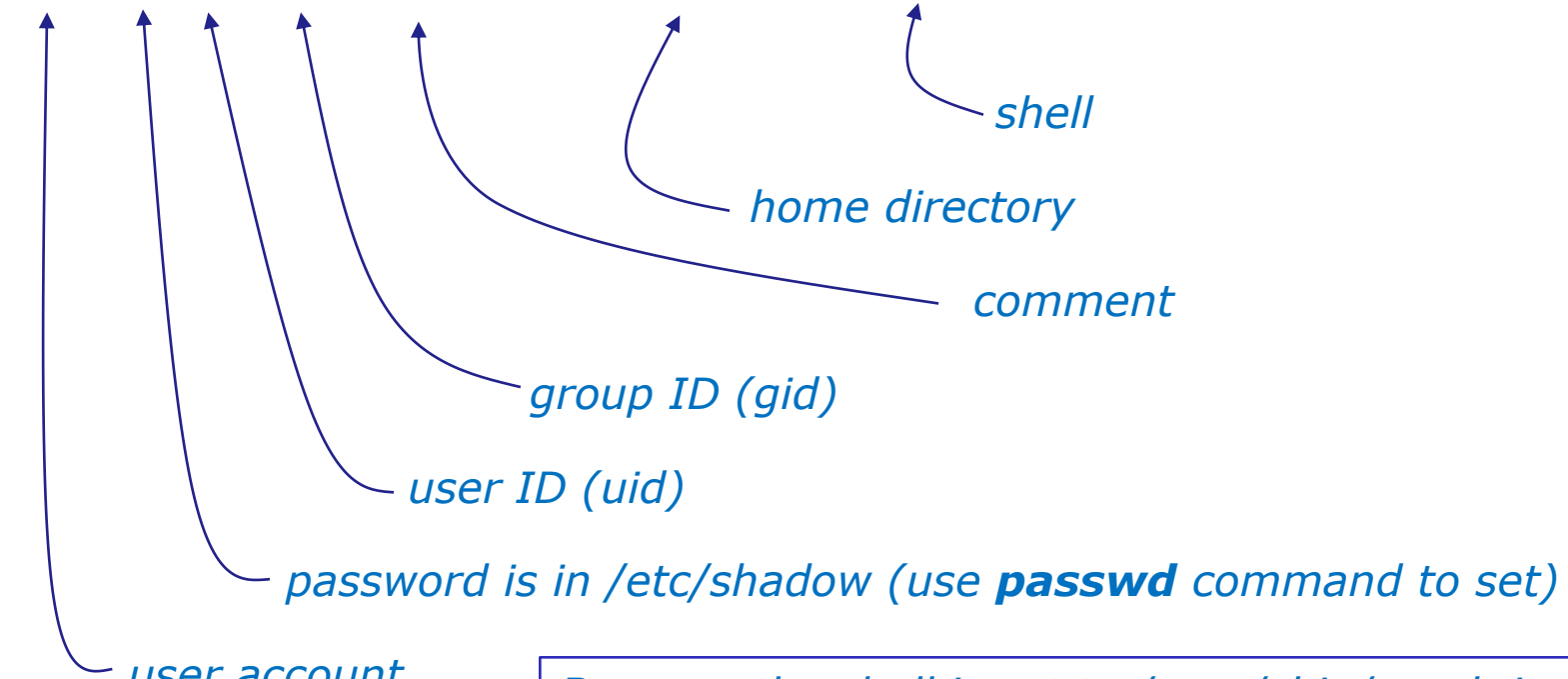
```
[root@gothmog ~]# cat /home/guest/.bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin

export PATH
/usr/sbin/pppd -detach crtscts proxyarp 10.0.0.1:10.0.0.2 /dev/ttyS0 38400
[root@gothmog ~]#
```

This is used in Part 3 of Lab X2. As soon as guest logs in, the pppd service is run automatically on the server.

Lab X2

Creating a new user account on the server side with **useradd**

```
[root@gothmog ~]# useradd  -c "PPP Account" -d /etc/ppp -s /usr/sbin/pppd ppp
[root@gothmog ~]# cat /etc/passwd | grep ppp
ppp:x:502:502:PPP Account:/etc/ppp:/usr/sbin/pppd
```



Because the shell is set to /user/sbin/pppd, it is run as soon as the ppp user logs in using the option in /etc/ppp/options

Lab X2

The server side options can be put on the command line

/usr/sbin/pppd -detach crtscts proxyarp 10.0.0.1:10.0.0.2 /dev/ttyS0 38400

or in the configuration file

```
[root@gothmog ~]# cat /etc/ppp/options
-detach
crtscts
lock
proxyarp
10.0.0.1:10.0.0.2
/dev/ttyS0
38400
```

Don't fork to become a background process (otherwise pppd will do so if a serial device is specified).

Use hardware flow control using RTS and CTS signals to control the flow of data on the serial port.

Specifies that pppd should use a UUCP-style lock on the serial device to ensure exclusive access to the device.

Add an entry to this system's ARP [Address Resolution Protocol] table with the IP address of the peer and the Ethernet address of this system.

IP address for server-end: client-end

Serial device

Desired baud rate

Refer to **pppd** man page for full details

Lab X2

Command line (client side) to make a connection

With this option, pppd will detach (run in the background) once it has successfully established the ppp connection (to the point where the first network control protocol, usually the IP control protocol, has come up).

Add a default route to the system routing tables, using the peer as the gateway, when IPCP negotiation is successfully completed. This entry is removed when the PPP connection is broken.

```
pppd updetach crtscts defaultroute /dev/ttyS0 38400 connect \
"exec chat -v TIMEOUT 3 ogin:--ogin: ppp assword: secret"
```

command line (client side)

Lab X2

Command line (client side) to make a connection

```
root@gimli:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
root@gimli:~#
root@gimli:~# pppd updetach crtscts defaultroute /dev/ttyS0 38400 connect
"exec chat -v TIMEOUT 3 ogin:--ogin: ppp assword: secret"
Serial connection established.
Using interface ppp0
Connect: ppp0 <--> /dev/ttyS0
Deflate (15) compression enabled
Cannot determine ethernet address for proxy ARP
local  IP address 10.0.0.2
remote IP address 10.0.0.1
root@gimli:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.1        0.0.0.0         255.255.255.255 UH    0      0        0 ppp0
0.0.0.0         0.0.0.0         0.0.0.0         U     0      0        0 ppp0
root@gimli:~#
```

***updetach** option:*
Makes pppd run in the background when link comes up

***defaultroute** option:*
Adds a route to the peer for all traffic

Lab X2

Command line (client side) to make a connection

pppd updetach crtscts defaultroute /dev/ttyS0 38400 connect \ "exec chat -v TIMEOUT 3 ogin:--ogin: ppp assword: secret"

*The **connect option** can be used to run a script which in this case runs the chat command.*

The chat command is used to handle the login automatically.

Lab X2

Command line (client side) to make a connection

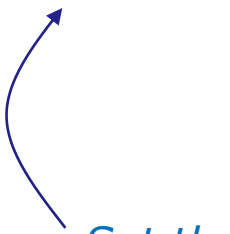
**pppd updetach crtscts defaultroute /dev/ttyS0 38400 connect **
"exec chat -v TIMEOUT 3 ogin:--ogin: ppp assword: secret"

Requests verbose mode for logging purposes.

Lab X2

Command line (client side) to make a connection

```
pppd updetach crtscts defaultroute /dev/ttyS0 38400 connect \
"exec chat -v TIMEOUT 3 ogin:--ogin: ppp assword: secret"
```



Set the timeout to 3 seconds

Lab X2

Command line (client side) to make a connection

**pppd updetach crtscts defaultroute /dev/ttyS0 38400 connect **
"exec chat -v TIMEOUT 3 ogin:--ogin: ppp assword: secret"

***expect:send** pairs:*
*expect …**ogin** then send **ppp**,*
*expect …**assword** then send **secret***

*Note: the **--ogin** is **sub-expect:sub-send** pair. If the first login is not received, send a single return (empty line) and look again for another login*

Note, because the beginning of the expected word may be garbled due to a flakey modem connection, just look for the end of the word (e.g. login to ogin, password to assword)

Lab X2

Troubleshooting

Tips

- Serial connection can only be used by one pair of computers at a time.
  E.g. Both minicom on gimli and Putty workstation cannot access serial COM 1 on gothmog at the same time.

- View log file:
  cat /var/log/messages | grep pppd

Lab X2

Troubleshooting

```
cis192@gimli:~$ su -
Password:
root@gimli:~# ./ppp-on
Serial connection established.
Using interface ppp0
Connect: ppp0 <--> /dev/ttyS0
LCP: timeout sending Config-Requests
Connection terminated.
Modem hangup
root@gimli:~#
```

Remove default gateway on gothmog

Lab X2

Troubleshooting

```
root@gimli:~# ./ppp-on
Connect script failed
root@gimli:~#
```

Make sure you have logged out from any previously made serial connections. You may need to run minicom –o again to see if you are still logged in as guest.

Wrap

New commands, daemons:

- pppd
- chat
- minicom

Configuration files

- /etc/ppp/options
- /etc/minicom/minirc.dfl

Next Class

Assignment: Check Calendar Page
http://simms-teach.com/cis192calendar.php

- No Quiz next week (test instead)

Lab 6 Practice Test Workshop

Backup