Lesson Module Status

- Wall updated and emailed

- Slides –
- Properties -
- Flashcards -
- 1st minute quiz –
- Web Calendar summary –
- Web book pages –
- Commands –
- Howtos –

- Lab tested –
- Lab template in depot -
- Youtube Videos uploaded –

- VM (Classroom PC) –
- VMs (VLab) -

- Headset charged –

- Special – practice test published

[ ] Has the phone bridge been added?

[ ] Is phone being used for voice input?

[ ] Is recording on?

[ ] Share slides, putty (rsimms, simben192), Chrome, vlab192.rdp, VMware Workstation, wireshark

[ ] Disable spelling on PowerPoint

[ ] Repeat all ?'s for remote students

[ ] Remote student proxy

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 192A Fall 2011 - Tuesdays, 1:00 to 5:10PM

Cabrillo College est. 1959

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

James Lars Daniel Elizabeth

Carlos V Branden Chad Donovan Leopoldo Jacob G Jeff Timothy Jacob S Laura

Gabriel V Jason Thomas Josh Carlos R Geoffrey Ellison Mark David Leandro

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

First Minute Quiz

Please answer these questions in the order shown:

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

Routing and Subnetting

Related Course Objectives	Agenda
• Configure appropriate IP addresses, network and subnet masks, and broadcast addresses based on the size and number of network segments required. • Connect multiple network segments together using Linux servers as routers and configuring the appropriate routing tables	• Quiz • Questions on previous material • Housekeeping • Permanent network configuration • Routing • IP forwarding • Static Routes • Routing table • Troubleshooting • Lab • Home network • Wrap

Questions

Questions?

- Previous lessons
- Labs
- How this class works

Snapshots

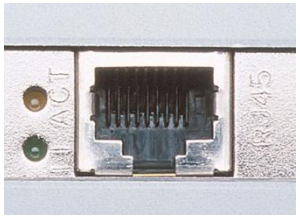
Revert to Snapshot

Live Demo

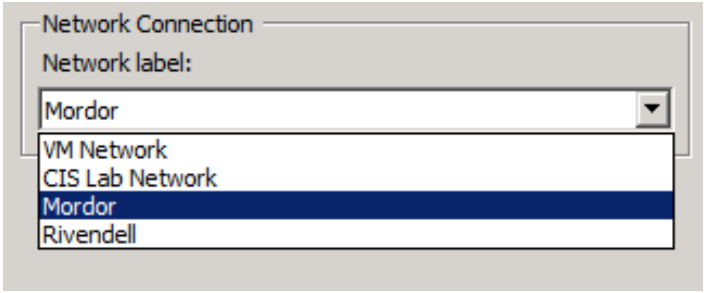
Cabling

Physical and virtual cabling

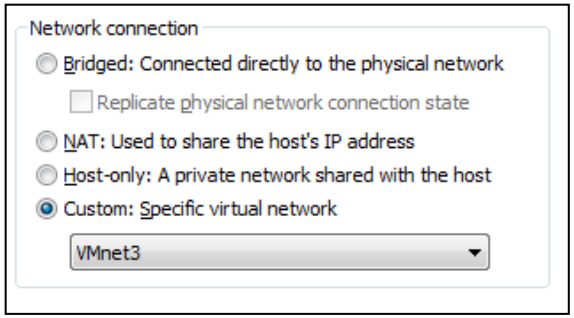
- In a physical environment we would connect Ethernet LAN cables between clients, servers, switches and routers.



- In a virtual environment cabling still must be done

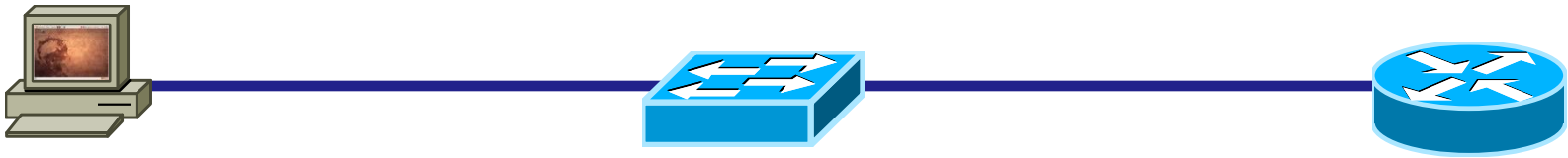


On VMware ESXi



On VMware Workstation

Cabling Devices on a Physical Network



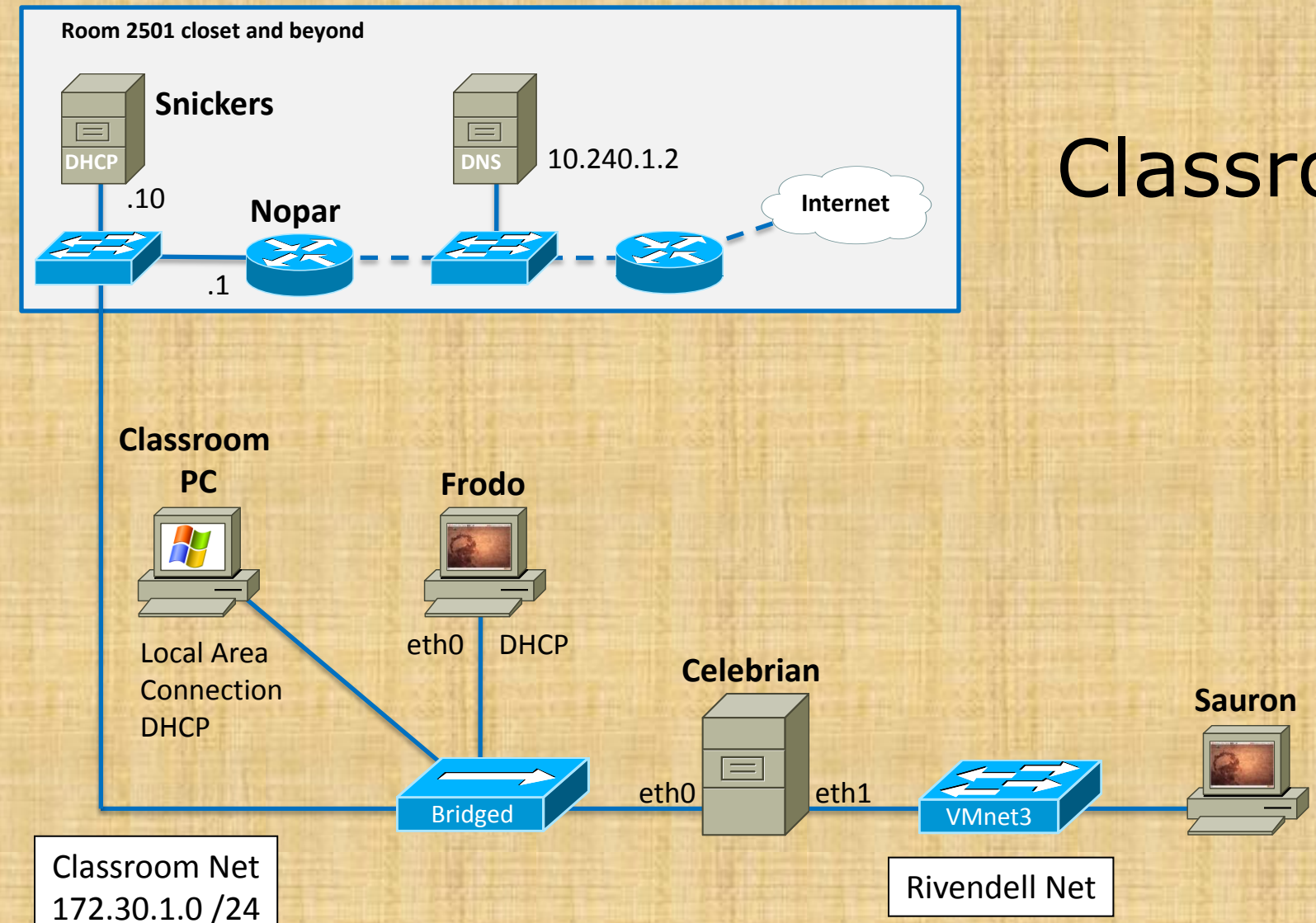
Desktop PC

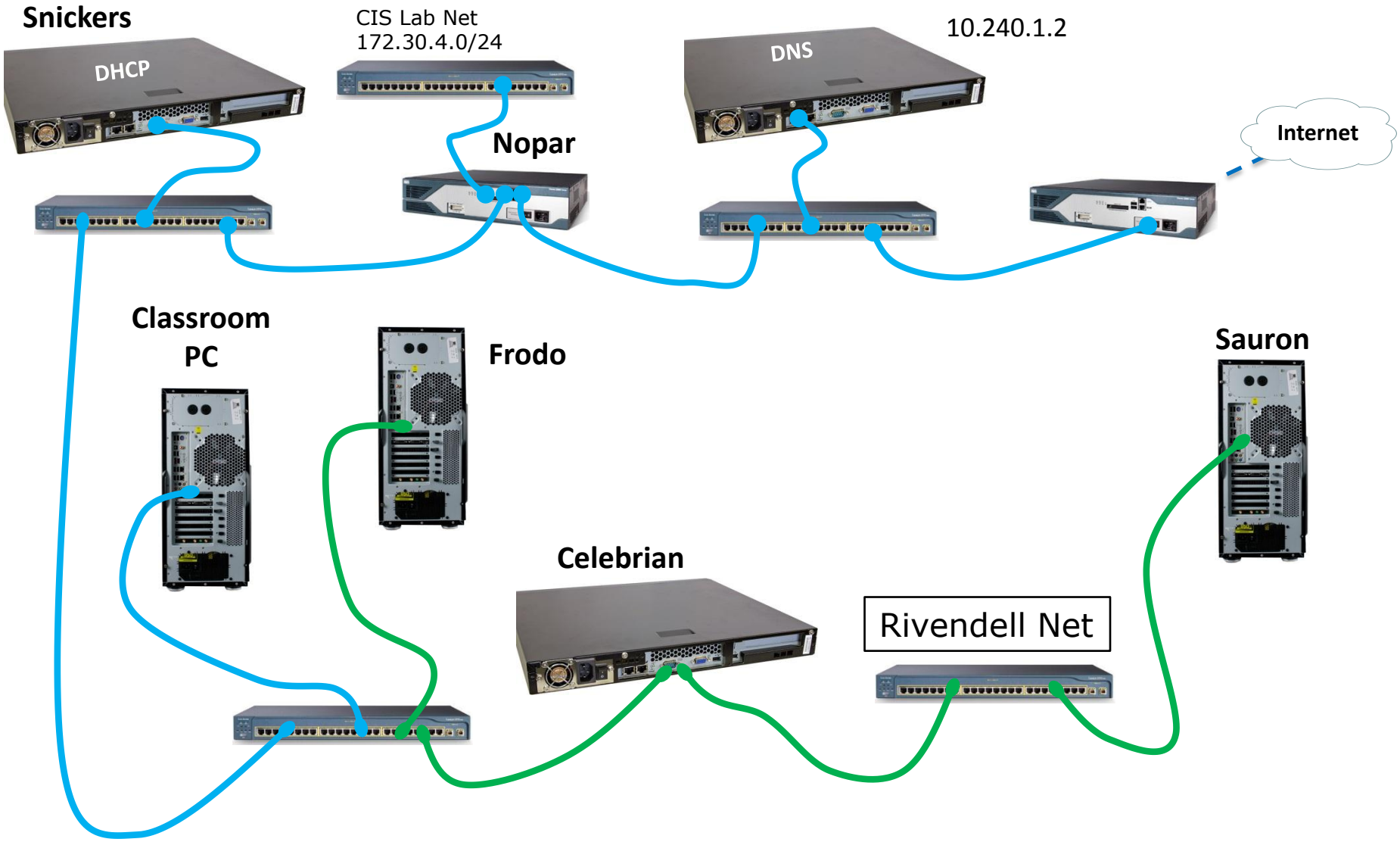
Switch

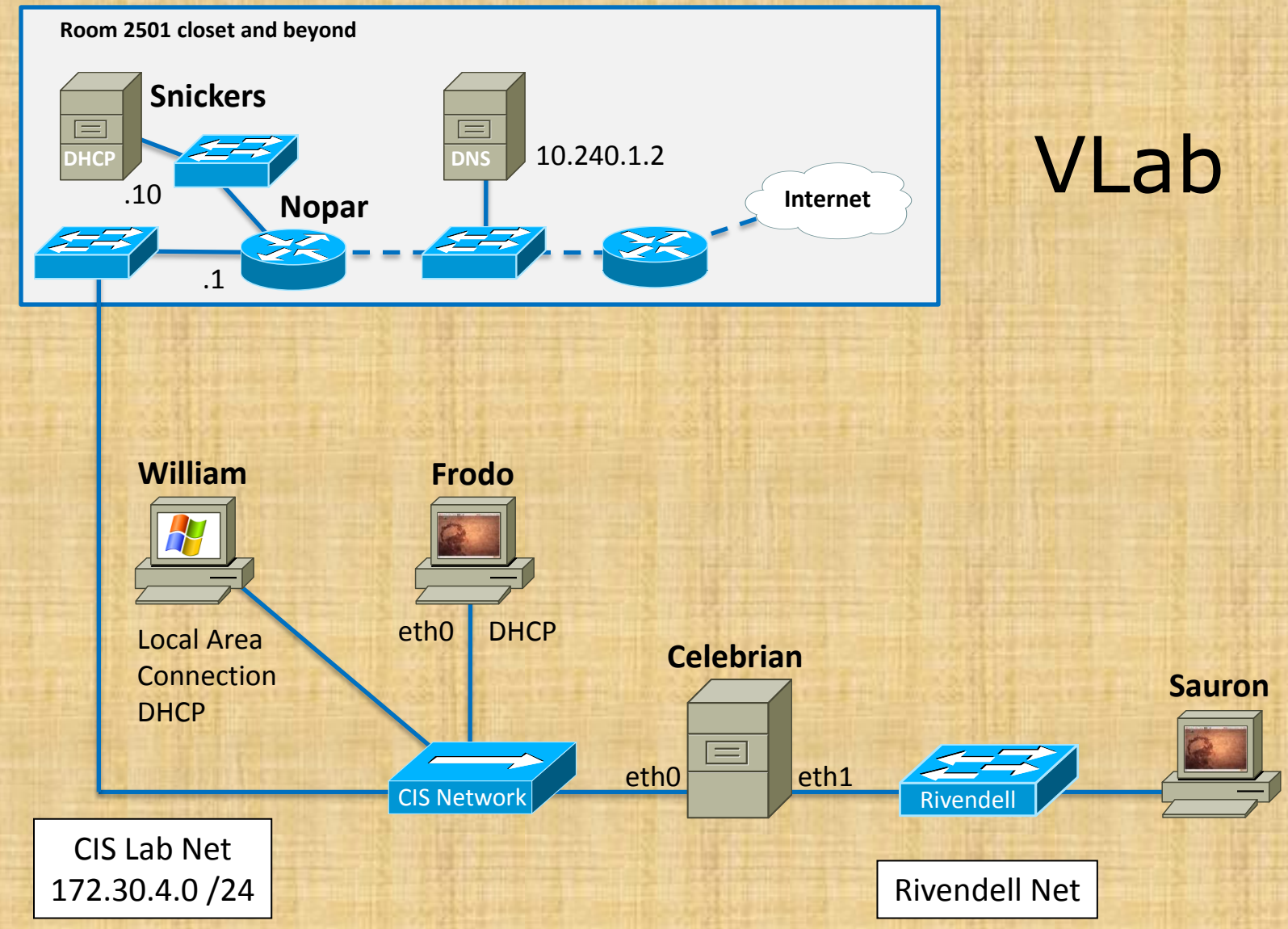
Router

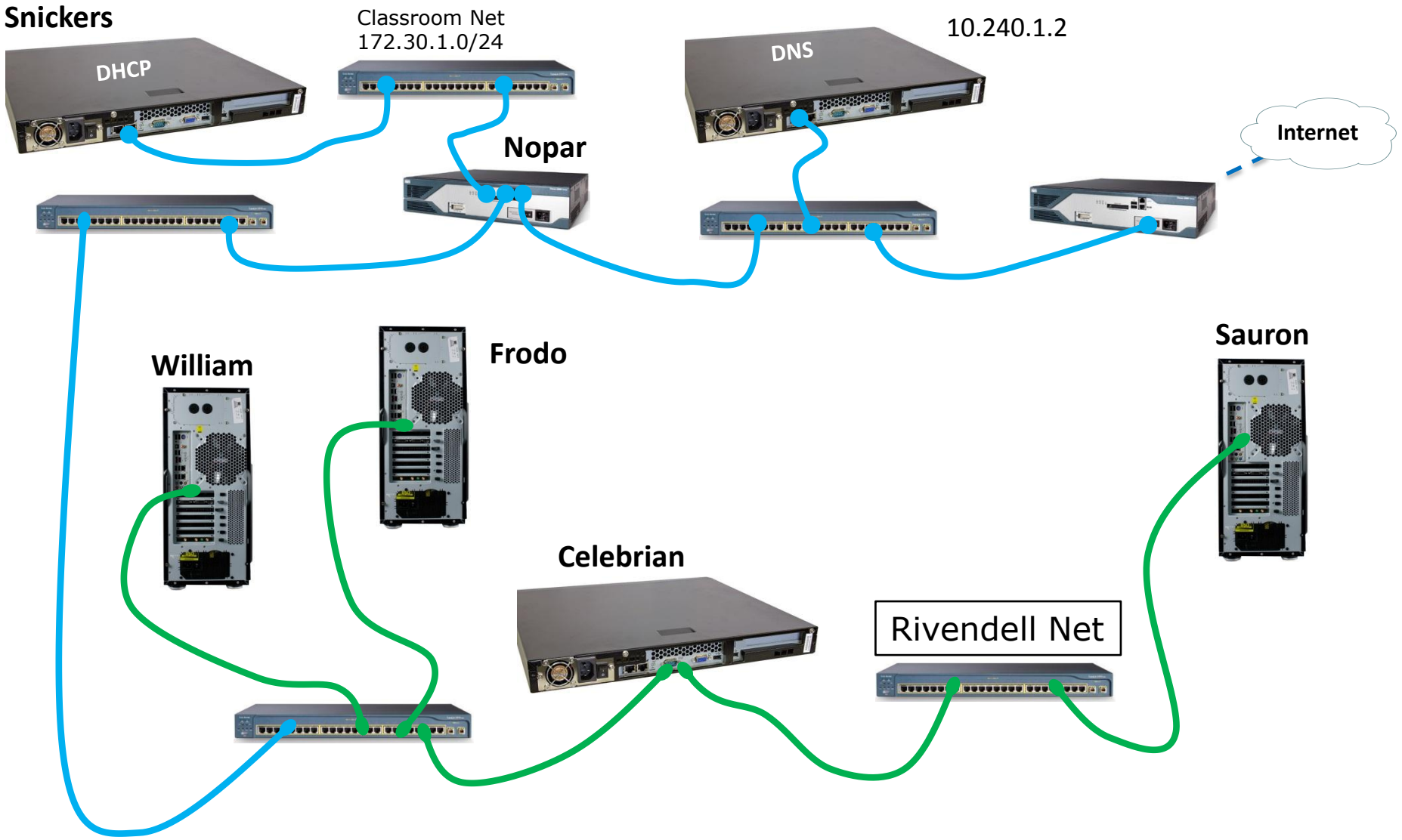
Cabling a PC to a router using a switch

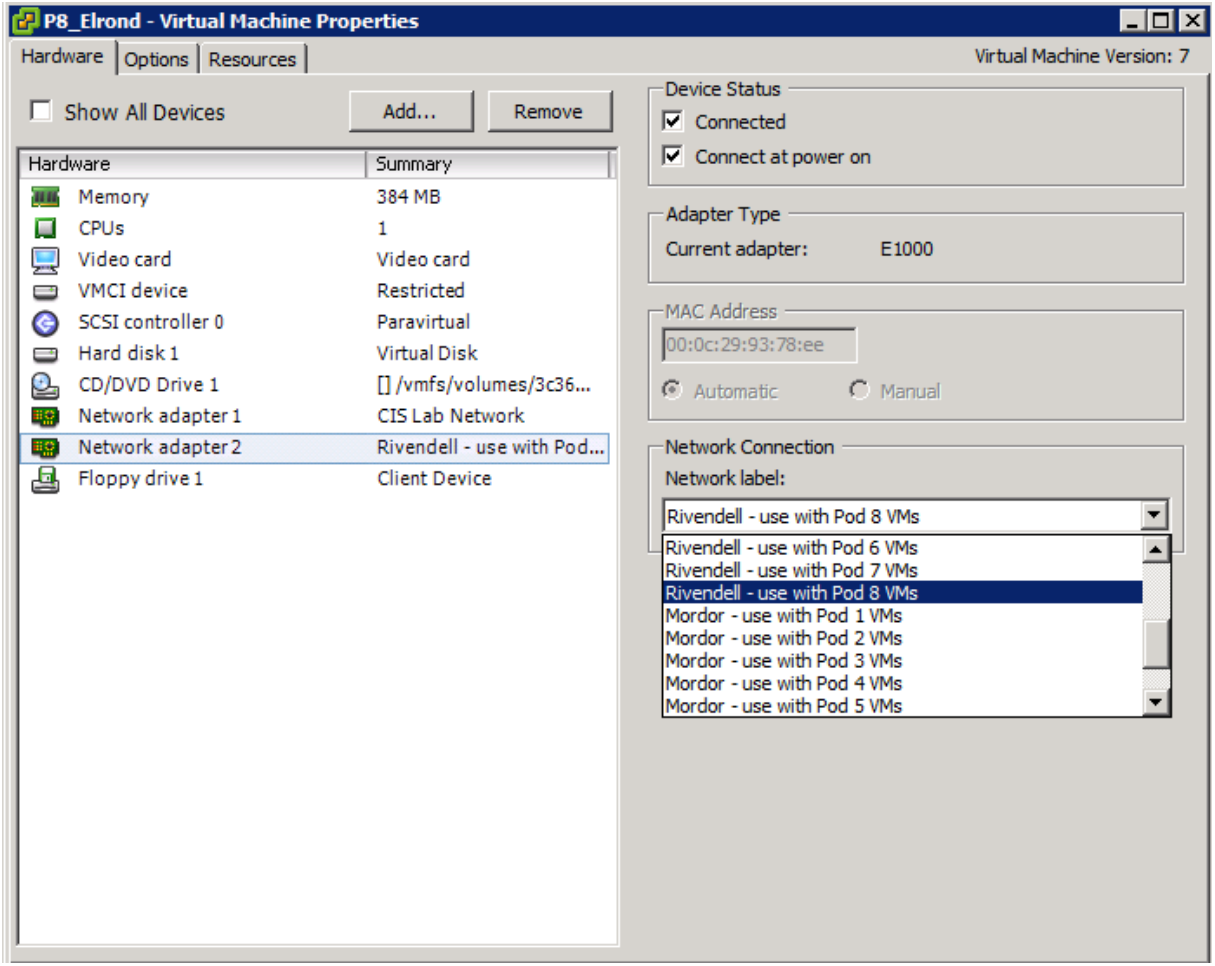
Classroom


Room 2501 closet and beyond
Snickers
DHCP
.10
Nopar
.1
DNS
10.240.1.2
Internet
Classroom PC
Local Area Connection DHCP
Frodo
eth0
DHCP
Celebrian
eth0
eth1
Sauron
Bridged
VMnet3
Classroom Net
172.30.1.0 /24
Rivendell Net


Snickers
DHCP
CIS Lab Net
172.30.4.0/24
Nopar
DNS
10.240.1.2
Internet
Classroom
PC
Frodo
Sauron
Celebrian
Rivendell Net
Classroom Net
172.30.1.0/24


Room 2501 closet and beyond
Snickers
DHCP
.10
Nopar
.1
DNS
10.240.1.2
Internet
VLab
William
Local Area Connection
DHCP
Frodo
eth0
DHCP
Celebrian
eth0
eth1
Sauron
CIS Network
Rivendell
CIS Lab Net
172.30.4.0 /24
Rivendell Net


Snickers
DHCP
Classroom Net
172.30.1.0/24
DNS
10.240.1.2
Internet
Nopar
William
Frodo
Sauron
Celebrian
Rivendell Net
CIS Lab Net
172.30.4.0/24

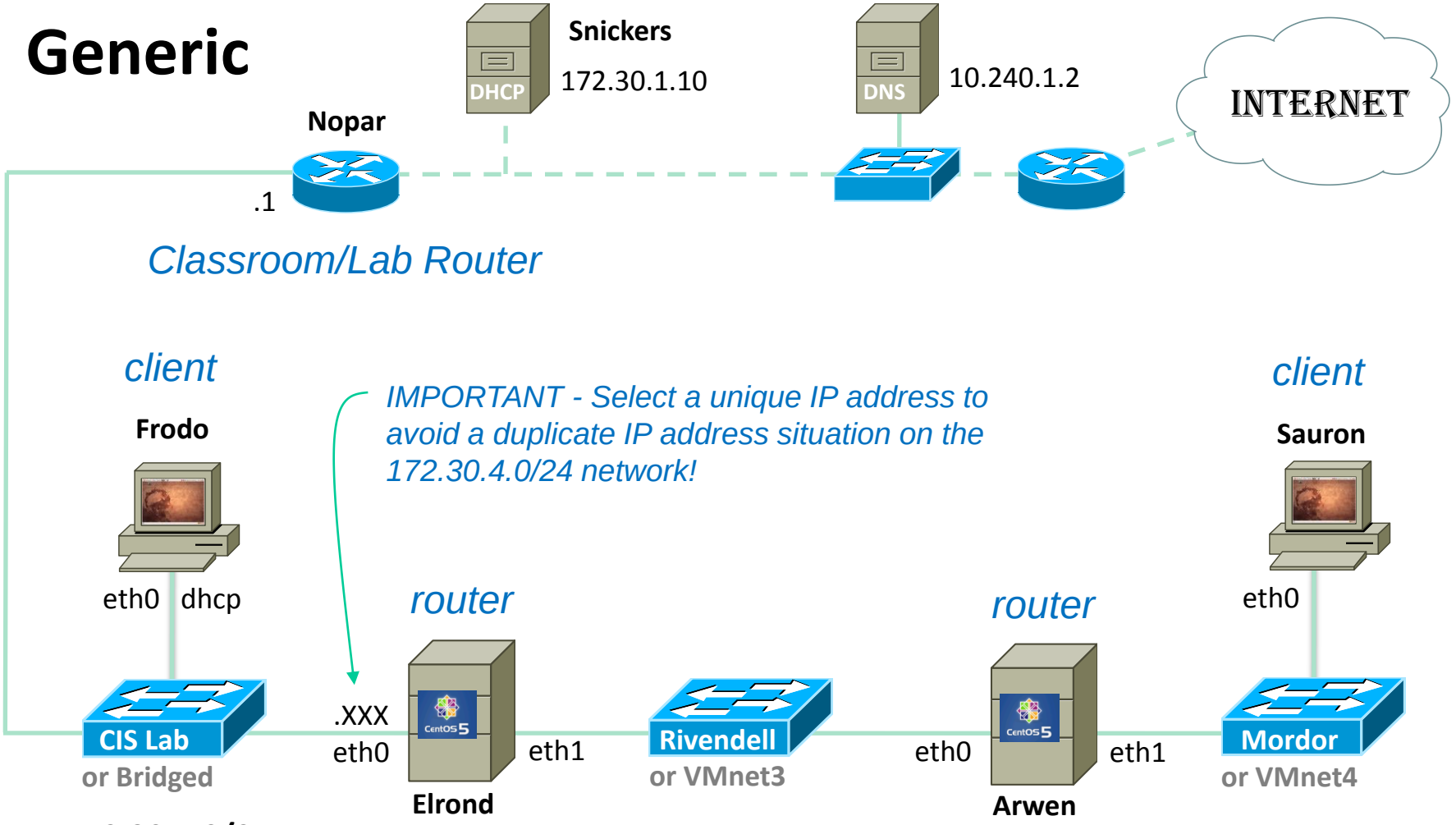


Notice: New virtual switches added for each pod to avoid duplicate IP addresses for future labs

Generic Diagram

Classroom
CIS Lab
CIS VLab

Generic

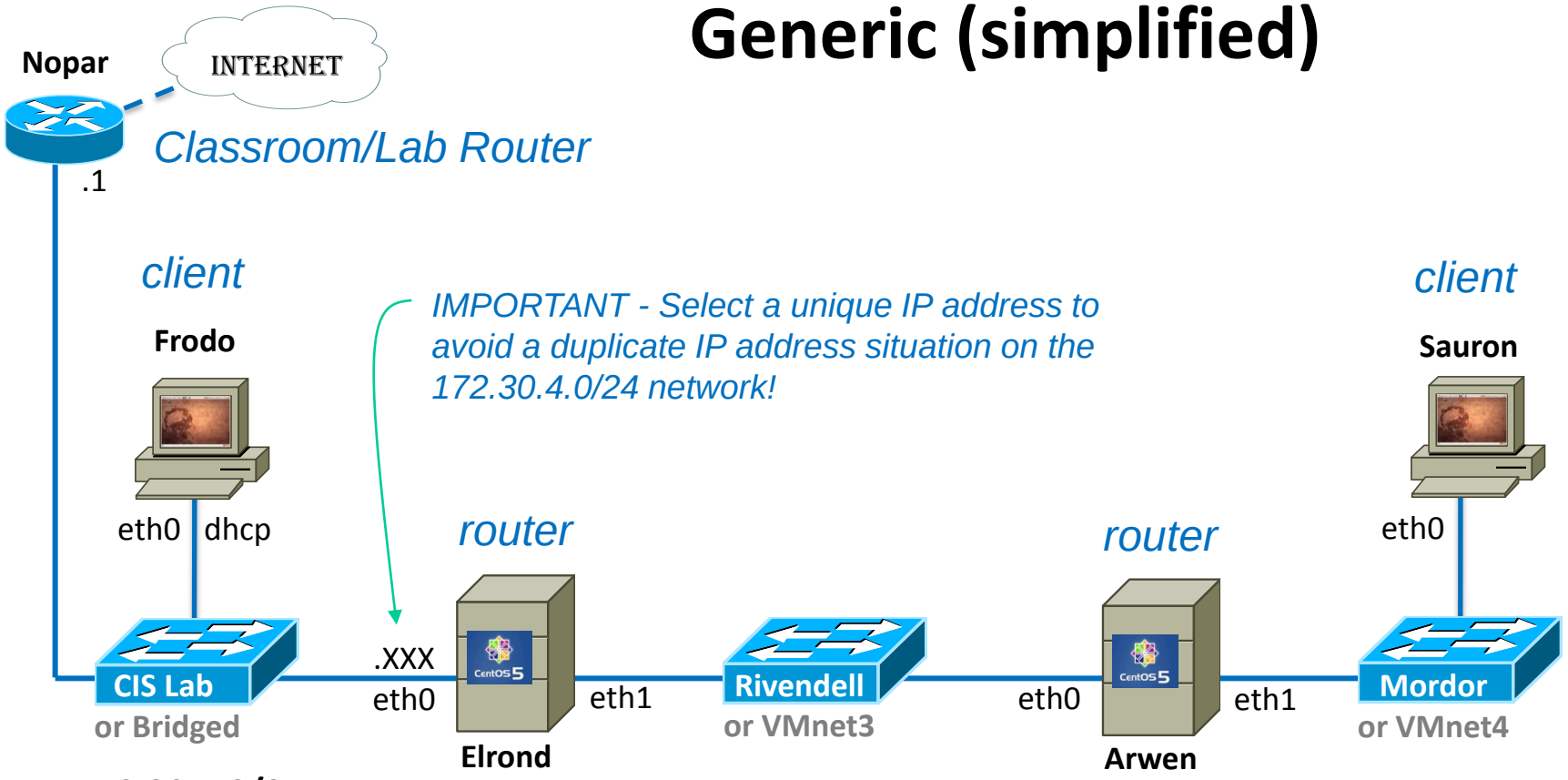


CIS LAB

RIVENDELL

MORDOR

Generic (simplified)

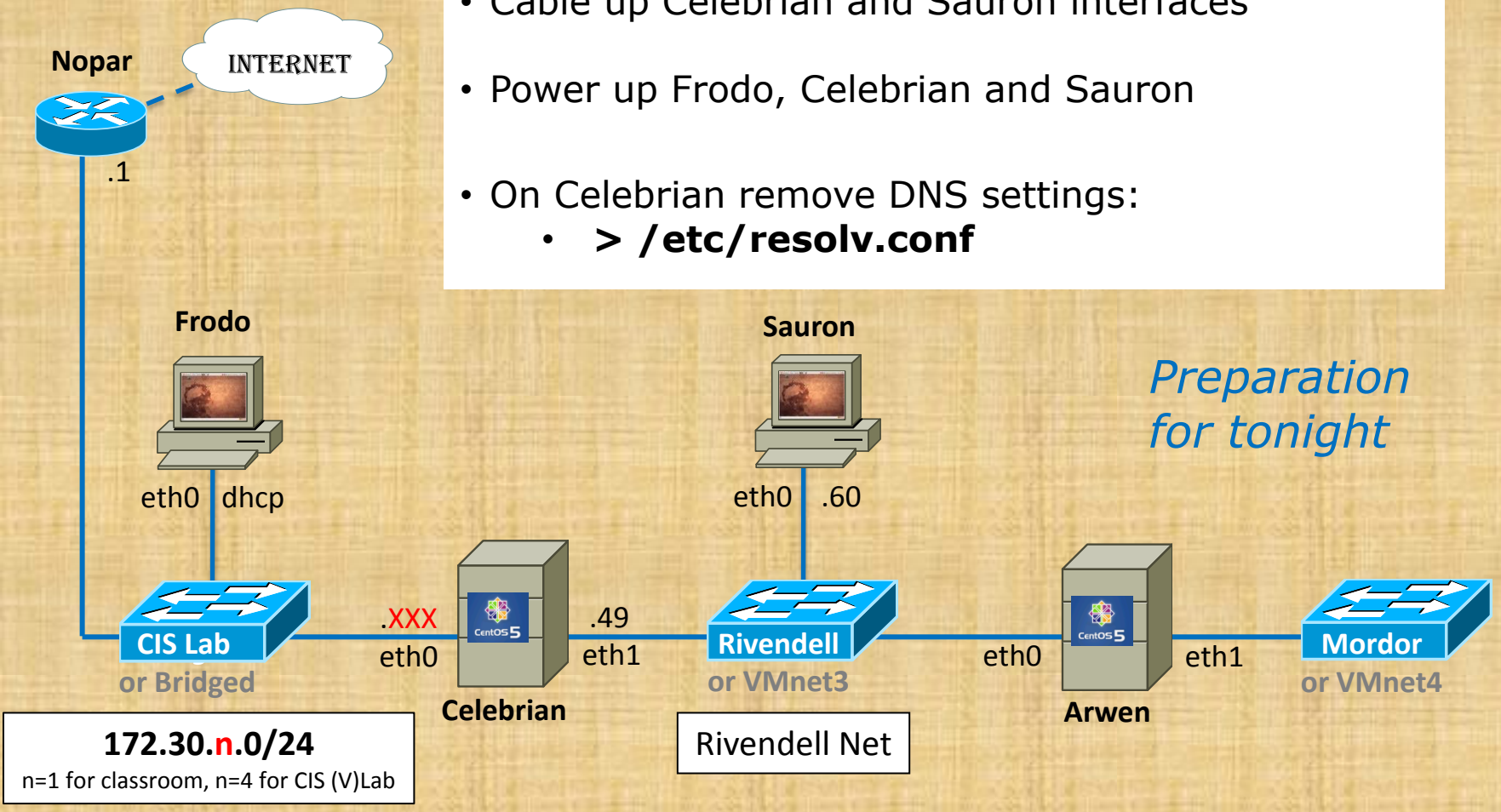


CIS CLASSROOM OR LAB

RIVENDELL

MORDOR

- Cable up Celebrian and Sauron interfaces
- Power up Frodo, Celebrian and Sauron
- On Celebrian remove DNS settings:
  - **> /etc/resolv.conf**



Housekeeping

Roll Call
and
Student Survey Check

VLab Changes

- Remade all pods (mix of clone and non-clone VMs)
- MAC/CoRD users must cancel to stop login to vcenter, then login to vmserver4 as usual.
- Pod 2 is experimental (uses NFS server)
- Separate Rivendell and Mordor switches now
- Don’t forget to use Fang to reserve VMs

Help with labs

Like some help with labs?

I'm in the CIS Lab Monday afternoons

- See schedule at http://webhawks.org/~cislab/

or contact me to arrange another time online

Turning in work and getting it graded work back on Opus

Benji (simben192) submits his lab2 for grading

```
[simben192@opus ~]$ cp lab02 /home/rsimms/turnin/lab02.$LOGNAME
```

Benji verifies his lab was submitted

```
[simben192@opus ~]$ ls /home/rsimms/turnin/
lab01  lab02.simben192  lab02.visgab192  lab02.wildan192  lab02.winjas192
```

Benji's lab1 has been graded and placed in his directory

```
[simben192@opus ~]$ ls -l
total 52
-rw-r--r-- 1 simben192 cis192  610 Nov  7 08:51 capture
-rw-r--r-- 1 simben192 cis192  360 Nov  1 09:26 lab01
-r-------- 1 simben192 staff  4170 Nov  2 16:05 lab01.graded
-rw------- 1 simben192 cis192 3702 Nov  7 08:49 lab02
-rw-r--r-- 1 simben192 cis192 1350 Oct 31 19:05 labnotes
-rw-r--r-- 1 simben192 cis192 1400 Nov  1 13:15 notes
[simben192@opus ~]$
```

Grades Web Page

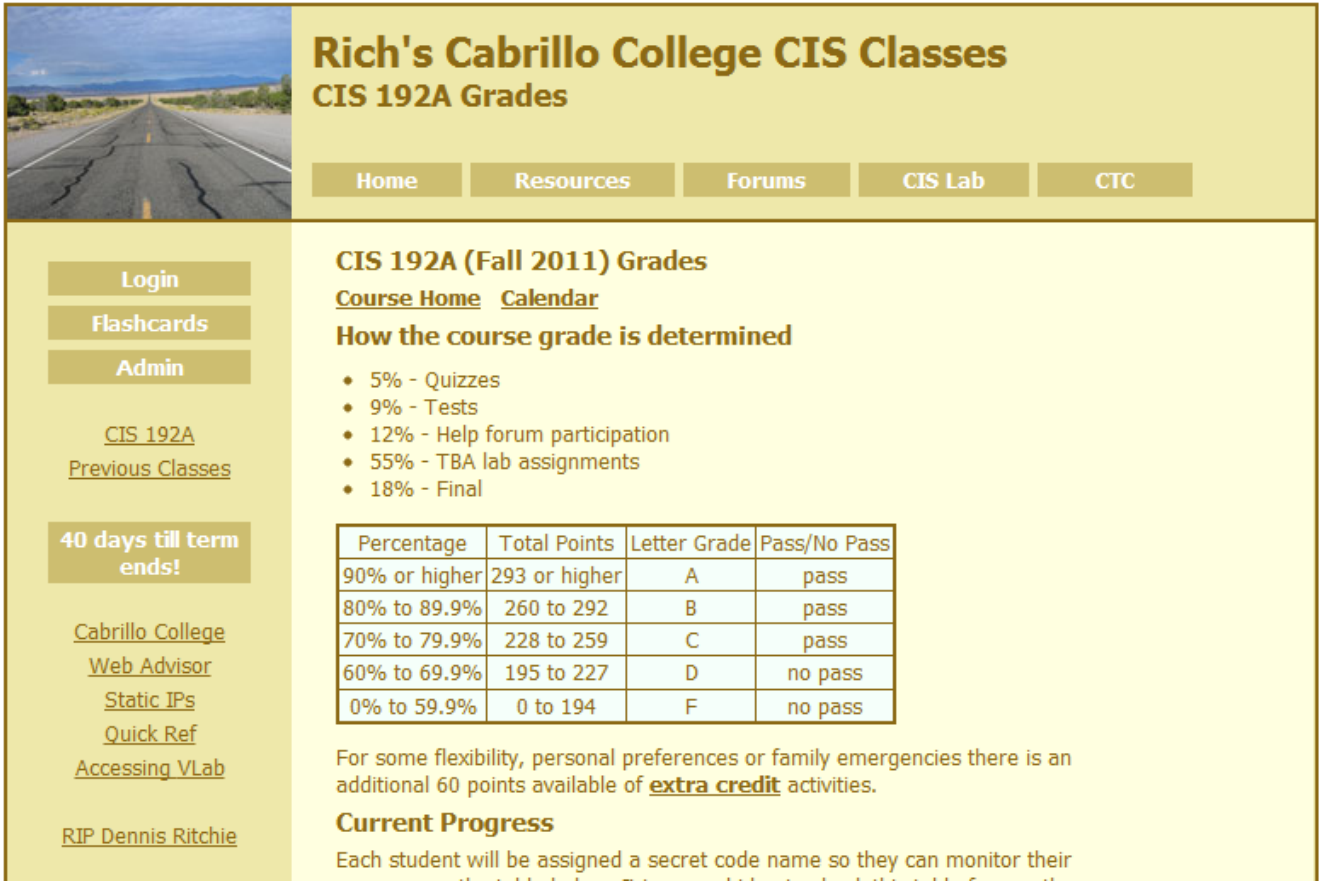
http://simms-teach.com/cis192Agrades.php

Code Name	Grading Choice	Quizzes & Tests						Forum		TBA Labs						Final	Extra Credit	Total	Grade
		Q1	Q2	Q3	Q4	Q5	T1	F1	F2	L1	L2	L3	L4	L5	L6				
Max Points		3	3	3	3	3	30	20	20	30	30	30	30	30	30	60	60	325	
Arwen	Grade	3								30									
Aragorn	Grade	3								30							9		
Bombadil	Grade	3								30							5		
Denethor	Grade	3								30									
Dwalin	Grade									27									
Elrohir	Grade	3								30									
Elrond	Grade	3								30							3		
Eomer	Grade	3								30									
Faramir	Grade																		
Frodo	Grade	3								30							4		
Gimli	Grade									25									
Glorfindel	Grade									30									
Goldberry	Grade																		
Gwaihir	Grade	3								25									
Ioreth	Grade	3								30									
Legolas	Grade	3								30							5		
Nazgul	Grade									24							1		
Pippen	Grade	3								30							3		
Samwise	Grade									27									
Saruman	Grade	3								27							5		
Sauron	Grade																		
Strider	Grade									30									
Theoden	Grade	3								30							7		
Treebeard	Grade	3								27							4		

Email **risimms@cabrillo.edu** for your code name

Check your Grading Choice, Quiz #1, Lab #1 and Extra Credit

Grades Web Page



Rich's Cabrillo College CIS Classes

CIS 192A Grades

Home | Resources | Forums | CIS Lab | CTC

Login | Flashcards | Admin

CIS 192A
Previous Classes

40 days till term ends!

Cabrillo College
Web Advisor
Static IPs
Quick Ref
Accessing VLab

RIP Dennis Ritchie

CIS 192A (Fall 2011) Grades

Course Home Calendar

How the course grade is determined

- 5% - Quizzes
- 9% - Tests
- 12% - Help forum participation
- 55% - TBA lab assignments
- 18% - Final

Percentage	Total Points	Letter Grade	Pass/No Pass
90% or higher	293 or higher	A	pass
80% to 89.9%	260 to 292	B	pass
70% to 79.9%	228 to 259	C	pass
60% to 69.9%	195 to 227	D	no pass
0% to 59.9%	0 to 194	F	no pass

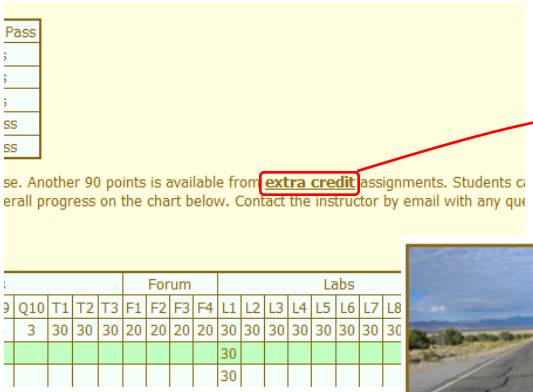
For some flexibility, personal preferences or family emergencies there is an additional 60 points available of **extra credit** activities.

Current Progress

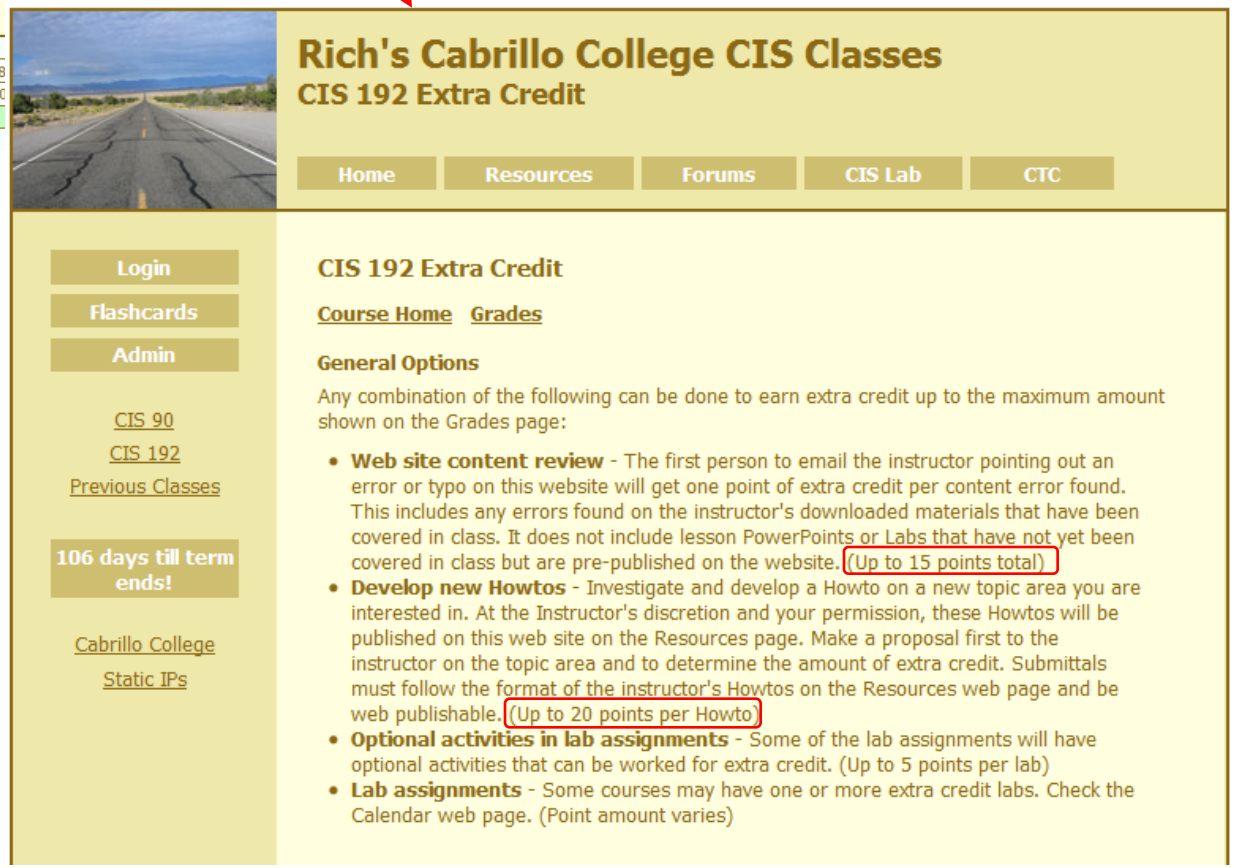
Each student will be assigned a secret code name so they can monitor their

Select the grade you want and earn that many points!

Extra Credit



Link to Extra Credit page is on the Grades page



Note the total extra credit cap is 90 points

3	11/8	**Quiz 2** **IP Routing and Subnetting** • Describe the fields of a datagram header • Describe the role of routing • Describe how packet forwarding is accomplished • Use the Simple Routing Algorithm to interpret a routing table • View, add, and delete entries in a routing table • By properly configuring routing tables on hosts and routers, configure a LAN of multiple segments which allows all hosts to communicate with each other. **Materials** • Presentation slides (download) **TBA Assignment** • Practice Test 1 • Lab 3 (Static Routing) **CCC Confer** • Enter virtual classroom • Class archives	13	Lab 2
4	11/15	**Test 1** **TCP, Transport and Application Layers** • Dynamic routing • Quagga routing suite • RIPv2 implementation skills • Transport layer • TCP and UDP protocols • Service ports and sockets • Super Daemons • Telnet, FTP and SSH services • SSH port forwarding • TCP Wrappers • Firewalls (first taste) **Materials** • Presentation slides (download) **TBA Assignment** • Lab 4 (Dynamic Routing) • Extra Credit Lab X1 (SSH Tunneling) **CCC Confer** • Enter virtual classroom • Class archives	12	5 posts Lab 3
		Quiz 3		

Plan ahead with the Calendar web page

Lab 2 is due midnight tonight

First test is next week!

Lab 3 and five forum posts are due midnight March 4th

Review

Commands

Treat VMs as real computers

Always shutdown any running VMs when you are finished

- Powering off a VM is the same as holding down the power button on a real computer. Any pending drive writes will be lost and open files may become corrupted.

- Shutting down your host VMware station before shutting down running VMs can also result in corrupted files for the same reason as above.

- The fastest way to shutdown Linux is to use: **init 0**

Linux Networking Review Examples

startx (start up X windows and desktop)
Alternate command Terminals
- Ctrl-alt-f1
- Ctrl-alt-f2
- Ctrl-alt-f3
- Ctrl-alt-f4
- Ctrl-alt-f5
- Ctrl-alt-f6

Graphical desktop
- Ctrl-alt-f7

shutdown now
init 0 (fastest way to shut down)
init 1 (minimal system)
init 3 (normal system)
init 5 (X windows graphics mode)

su - (gets you to root with root's path)
sudo –i (gets you to root on Ubuntu)

Linux Networking Review Examples

lspci	(list PCI hardware including NIC)
lspci –k	(shows drivers)
dmesg	(shows ton of HW info)
more /var/log/dmesg	(same as dmesg, but not the latest entries)
lsmod	(view already installed drivers)

To research Linux network driver info:

http://www.tldp.org/HOWTO/text/Ethernet-HOWTO
http://www.tldp.org/HOWTO/Hardware-HOWTO/nic.html

Network drivers (in /lib/modules/$(uname -r)/kernel/drivers/net)

Leave off the .o or .ko suffix when specifying drivers:

insmod 3c59x	(installs 3c59x driver - no dependencies)
modprobe pcnet32	(installs driver with/without dependencies)
rmmod pcnet32	(removes pcnet32 driver)

Linux Networking Review Examples

Controlling interfaces

ifconfig	(shows active interfaces)
ifconfig eth0 up	(brings up eth0)
ifconfig eth0 down	(brings up eth0)

Configuring interfaces

ifconfig eth1 172.30.4.106/24	(adds IP/mask)
ifconfig eth1:1 172.30.4.206/24	(adds alias IP/mask)

Configure default gateways with:

```
route add default gw 192.168.2.6
route del default gw 192.168.2.6
```

To display routing table:

```
route -n
```

TUI interface configuration tool:

netconfig	(RH9)
system-config-network	(newer versions of RH)

Linux Networking Review Examples

To show name servers:
cat /etc/resolv.conf

To configure nameservers:
echo "nameserver 10.240.1.2" > /etc/resolv.conf

or vi /etc/resolv.conf (and hand edit)

nameserver 10.240.1.2	(IP address of primary name server)
nameserver XXX.XXX.XXX.XXX	(IP address of secondary name server)
search cabrillo.edu	(domain suffix to add for short names)

Remote logins:

ssh 172.30.4.107	(uses current login name)
ssh root@172.30.4.107	(specifies a login name)

Linux Networking Review Examples

arp –n	(show arp cache)
service arpwatch start | stop	(Red Hat, monitor arp cache entries)
/etc/init.d/arpwatch start | stop	(Ubuntu, monitor arp cache entries)
/var/lib/arpwatch/arp.dat	(file of collected IP/MAC pairs)
yum install arpwatch	(Red Hat new package install, needs Internet)
apt-get install ipcalc	(Ubuntu new package install, needs Internet)

ping 172.30.4.1
ping –Rc3 opus.cabrillo.edu
ipcalc
traceroute <ip address>
mtr google.com
netstat –i (show interface stats)

tcpdump src 172.30.1.105 or dst 172.30.1.105
wireshark

Tangent

RH9

Tangent on the older Red Hat 9

(Red Hat Family)

What happens with RH9 if you don't specify the broadcast address for the 172 net in the classroom?

```
[root@rh9 root]# ifconfig eth0 172.30.1.201 netmask 255.255.255.0
[root@rh9 root]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:28:3A:0C
          inet addr:172.30.1.201  Bcast:172.30.255.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:926 (926.0 b)  TX bytes:830 (830.0 b)
          Interrupt:9 Base address:0x10a4
```

Answer: Trouble! We are using a /24 network in the classroom yet RH9 calculates a /16 based broadcast address!

RH9 calculates the broadcast address using the older classful method. The 172 network is a class B network where the first 16 bits form the network portion of the address. In CIDR this would be /16 network

Tangent on the older Red Hat 9

(Red Hat Family)

Specify both the netmask and broadcast address when using RH9

```
[root@rh9 root]# ifconfig eth0 172.30.1.201 netmask 255.255.255.0 broadcast 172.30.1.255
[root@rh9 root]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:28:3A:0C
          inet addr:172.30.1.201  Bcast:172.30.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:926 (926.0 b)  TX bytes:830 (830.0 b)
          Interrupt:9 Base address:0x10a4

[root@rh9 root]#
```

Note the broadcast address is calculated correctly with the newer VMs we are using in this course. The only VM that uses RH9 is Nosmo. For the rest of the class VMs you do not have to specify the broadcast address on the ifconfig command.

Review

Subnets

Subnetting – Lab 2 Rivendell network

```
cis192@frodo:~$ ipcalc 192.168.2.0/24
Address:   192.168.2.0          11000000.10101000.00000010. 00000000
Netmask:   255.255.255.0 = 24   11111111.11111111.11111111. 00000000
Wildcard:  0.0.0.255            00000000.00000000.00000000. 11111111
=>
Network:   192.168.2.0/24       11000000.10101000.00000010. 00000000
HostMin:   192.168.2.1          11000000.10101000.00000010. 00000001
HostMax:   192.168.2.254        11000000.10101000.00000010. 11111110
Broadcast: 192.168.2.255        11000000.10101000.00000010. 11111111
Hosts/Net: 254                   Class C, Private Internet
```

0000 0001 = 1 = 2^0	1000 0000 = 128
0000 0010 = 2 = 2^1	1100 0000 = 192
0000 0100 = 4 = 2^2	1110 0000 = 224
0000 1000 = 8 = 2^3	1111 0000 = 240
0001 0000 = 16 = 2^4	1111 1000 = 248
0010 0000 = 32 = 2^5	1111 1100 = 252
0100 0000 = 64 = 2^6	1111 1110 = 254
1000 0000 = 128 = 2^7	1111 1111 = 255

Subnetting – Classroom and CIS Lab Networks

```
cis192@frodo:~$ ipcalc 172.30.1.0/24
Address:   172.30.1.0           10101100.00011110.00000001. 00000000
Netmask:   255.255.255.0 = 24   11111111.11111111.11111111. 00000000
Wildcard:  0.0.0.255            00000000.00000000.00000000. 11111111
=>
Network:   172.30.1.0/24        10101100.00011110.00000001. 00000000
HostMin:   172.30.1.1           10101100.00011110.00000001. 00000001
HostMax:   172.30.1.254         10101100.00011110.00000001. 11111110
Broadcast: 172.30.1.255         10101100.00011110.00000001. 11111111
Hosts/Net: 254                   Class B, Private Internet

cis192@frodo:~$ ipcalc 172.30.4.0/24
Address:   172.30.4.0           10101100.00011110.00000100. 00000000
Netmask:   255.255.255.0 = 24   11111111.11111111.11111111. 00000000
Wildcard:  0.0.0.255            00000000.00000000.00000000. 11111111
=>
Network:   172.30.4.0/24        10101100.00011110.00000100. 00000000
HostMin:   172.30.4.1           10101100.00011110.00000100. 00000001
HostMax:   172.30.4.254         10101100.00011110.00000100. 11111110
Broadcast: 172.30.4.255         10101100.00011110.00000100. 11111111
Hosts/Net: 254                   Class B, Private Internet
```

Subnetting – Lab 3 Rivendell and Mordor networks

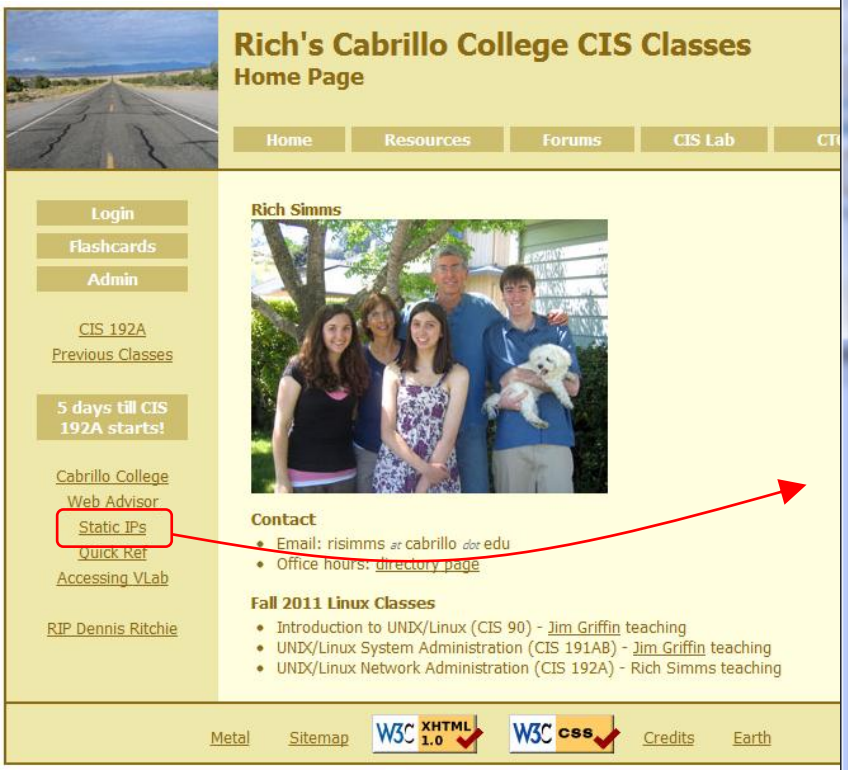
```
cis192@frodo:~$ ipcalc 192.168.16.0/22
Address:   192.168.16.0          11000000.10101000.000100 00.00000000
Netmask:   255.255.252.0 = 22    11111111.11111111.111111 00.00000000
Wildcard:  0.0.3.255             00000000.00000000.000000 11.11111111
=>
Network:   192.168.16.0/22       11000000.10101000.000100 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

cis192@frodo:~$ ipcalc 192.168.20.0/22
Address:   192.168.20.0          11000000.10101000.000101 00.00000000
Netmask:   255.255.252.0 = 22    11111111.11111111.111111 00.00000000
Wildcard:  0.0.3.255             00000000.00000000.000000 11.11111111
=>
Network:   192.168.20.0/22       11000000.10101000.000101 00.00000000
HostMin:   192.168.20.1          11000000.10101000.000101 00.00000001
HostMax:   192.168.23.254        11000000.10101000.000101 11.11111110
Broadcast: 192.168.23.255        11000000.10101000.000101 11.11111111
Hosts/Net: 1022                   Class C, Private Internet
```

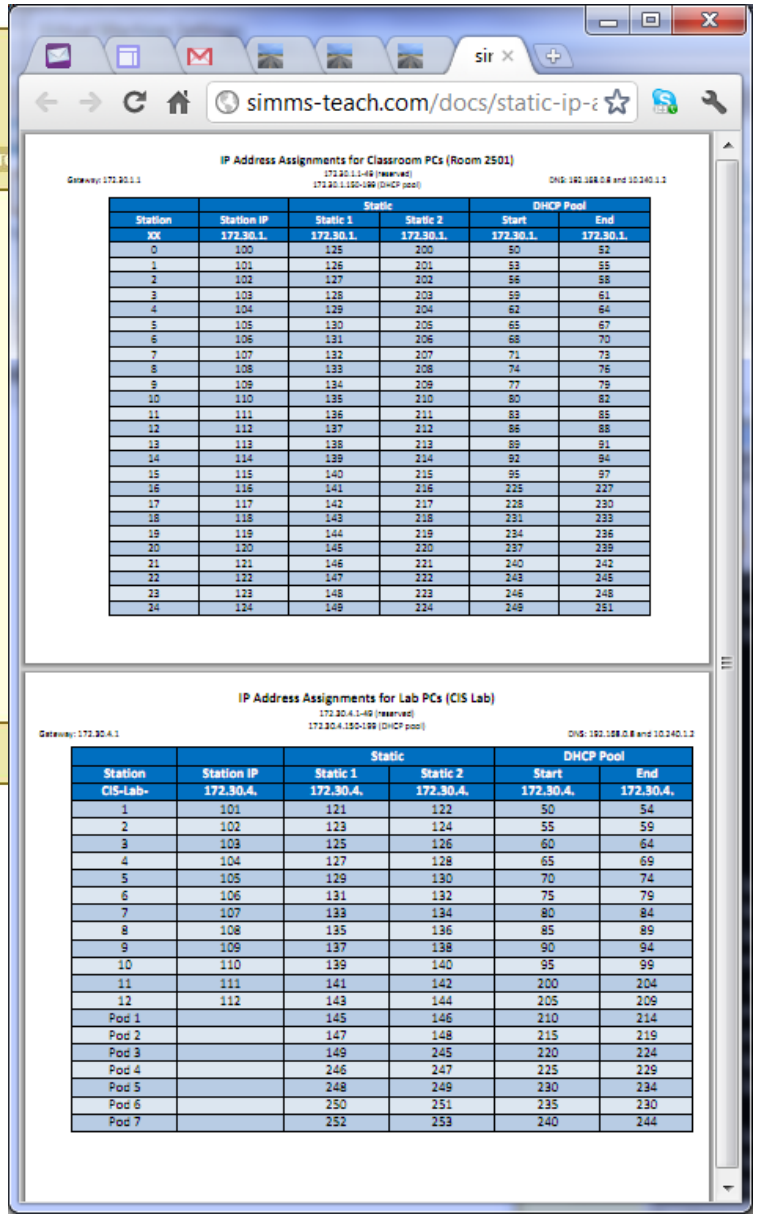
Supernetting – Lab 3 Rivendell and Mordor networks

```
cis192@frodo:~$ ipcalc 192.168.16.0/21
Address:   192.168.16.0          11000000.10101000.00010 000.00000000
Netmask:   255.255.248.0 = 21    11111111.11111111.11111 000.00000000
Wildcard:  0.0.7.255             00000000.00000000.00000 111.11111111
=>
Network:   192.168.16.0/21       11000000.10101000.00010 000.00000000
HostMin:   192.168.16.1          11000000.10101000.00010 000.00000001
HostMax:   192.168.23.254        11000000.10101000.00010 111.11111110
Broadcast: 192.168.23.255        11000000.10101000.00010 111.11111111
Hosts/Net: 2046                   Class C, Private Internet
```

Subnetting – For next activity

```
cis192@frodo:~$ ipcalc 10.10.15.48/28
Address:   10.10.15.48           00001010.00001010.00001111.0011 0000
Netmask:   255.255.255.240 = 28  11111111.11111111.11111111.1111 0000
Wildcard:  0.0.0.15              00000000.00000000.00000000.0000 1111
=>
Network:   10.10.15.48/28        00001010.00001010.00001111.0011 0000
HostMin:   10.10.15.49           00001010.00001010.00001111.0011 0001
HostMax:   10.10.15.62           00001010.00001010.00001111.0011 1110
Broadcast: 10.10.15.63           00001010.00001010.00001111.0011 1111
Hosts/Net: 14                     Class A, Private Internet
```





Station	Station IP	Static		DHCP Pool	
		Static 1	Static 2	Start	End
XX	172.30.1.	172.30.1.	172.30.1.	172.30.1.	172.30.1.
0	100	125	200	50	52
1	101	126	201	53	55
2	102	127	202	56	58
3	103	128	203	59	61
4	104	129	204	62	64
5	105	130	205	65	67
6	106	131	206	68	70
7	107	132	207	71	73
8	108	133	208	74	76
9	109	134	209	77	79
10	110	135	210	80	82
11	111	136	211	83	85
12	112	137	212	86	88
13	113	138	213	89	91
14	114	139	214	92	94
15	115	140	215	95	97
16	116	141	216	225	227
17	117	142	217	228	230
18	118	143	218	231	233
19	119	144	219	234	236
20	120	145	220	237	239
21	121	146	221	240	242
22	122	147	222	243	245
23	123	148	223	246	248
24	124	149	224	248	251

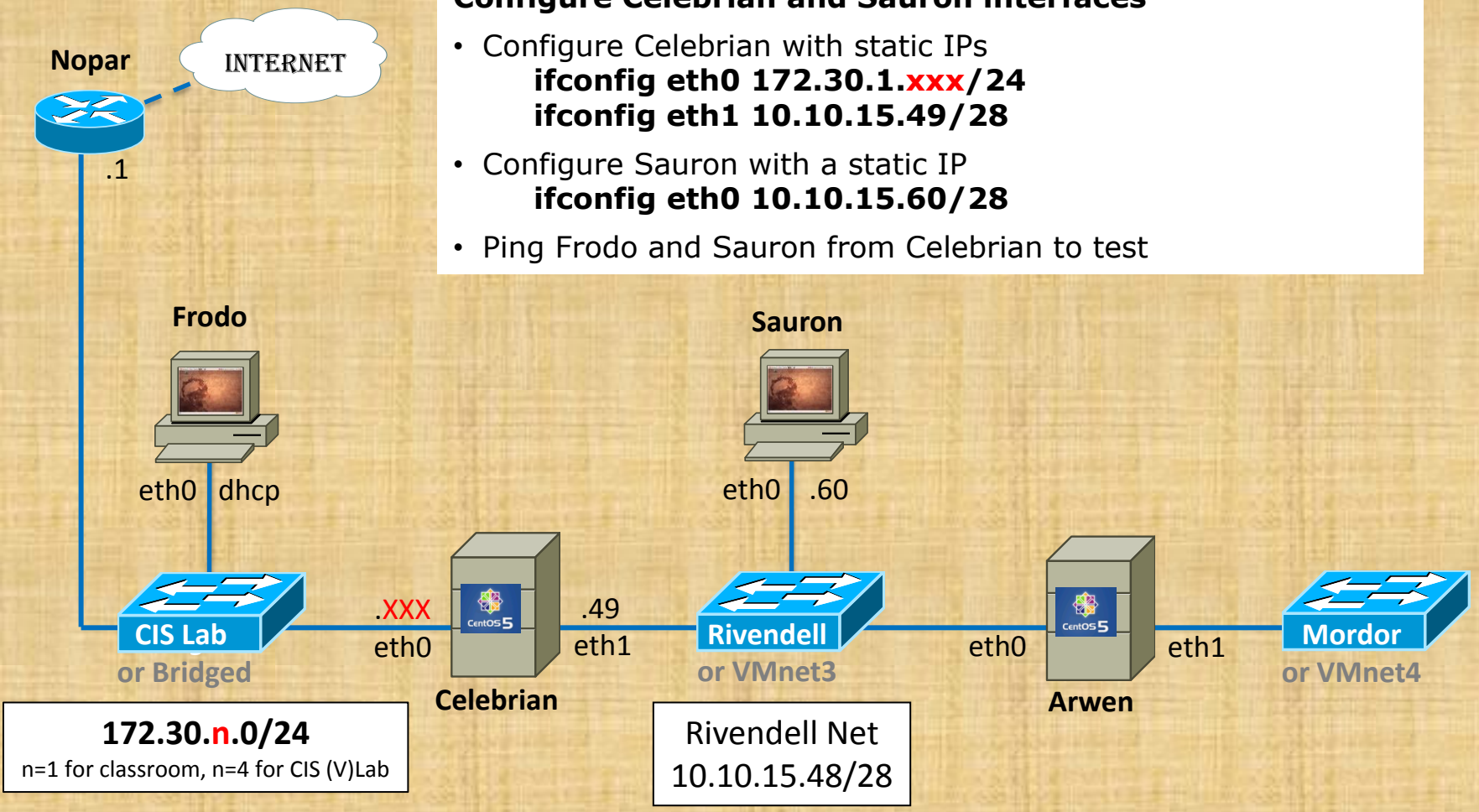
Station	Station IP	Static		DHCP Pool	
		Static 1	Static 2	Start	End
CIS-Lab-	172.30.4.	172.30.4.	172.30.4.	172.30.4.	172.30.4.
1	101	121	122	50	54
2	102	123	124	55	59
3	103	125	126	60	64
4	104	127	128	65	69
5	105	129	130	70	74
6	106	131	132	75	79
7	107	133	134	80	84
8	108	135	136	85	89
9	109	137	138	90	94
10	110	139	140	95	99
11	111	141	142	200	204
12	112	143	144	205	209
Pod 1		145	146	210	214
Pod 2		147	148	215	219
Pod 3		149	245	220	224
Pod 4		246	247	225	229
Pod 5		248	249	230	234
Pod 6		250	251	235	230
Pod 7		252	253	240	244

To avoid **TROUBLE**, use the Static IPs link on the web site to select IP addresses.

Only use static IPs assigned to the station you are using in the classroom or the lab!

Configure Celebrian and Sauron interfaces

- Configure Celebrian with static IPs
  ifconfig eth0 172.30.1.xxx/24
  ifconfig eth1 10.10.15.49/28
- Configure Sauron with a static IP
  ifconfig eth0 10.10.15.60/28
- Ping Frodo and Sauron from Celebrian to test




```
Network:   10.10.15.48/28
HostMin:   10.10.15.49
HostMax:   10.10.15.62
Broadcast: 10.10.15.63
```

Permanent Network Configuration

Configuring Network Settings

Different ways to configure network settings

1. **GUI tools**
   - Permanent,
   - Different for each distribution

2. The **ifconfig** and **route** commands
   - Temporary (till next restart)

3. Editing **configuration files** and restarting the network service
   - Permanent
   - Some variations between distributions
   - Requires network service to be restarted

Configuring Permanent Network Settings
(Red Hat Family)

Setting	File
IP address and subnet mask	/etc/sysconfig/network-scripts/ifcfg-eth*
Default gateway	/etc/sysconfig/network
DNS server(s)	/etc/resolv.conf
Hostname	/etc/sysconfig/network
Name/IP pairing	/etc/hosts

/etc/sysconfig/network-scripts/ifcfg-eth0

By the way - tab completes are wonderful

Managing System Services (daemons)
(Red Hat Family)

Manually starting and stopping

- service --status-all
- service *xxxxxx* <stop|start|restart|status>

Show all services

Control a specific service

System startup configuration

- chkconfig --list
- chkconfig [--level levels] *xxxxxx* <on|off>

Configuring Permanent Network Settings
(Red Hat Family)

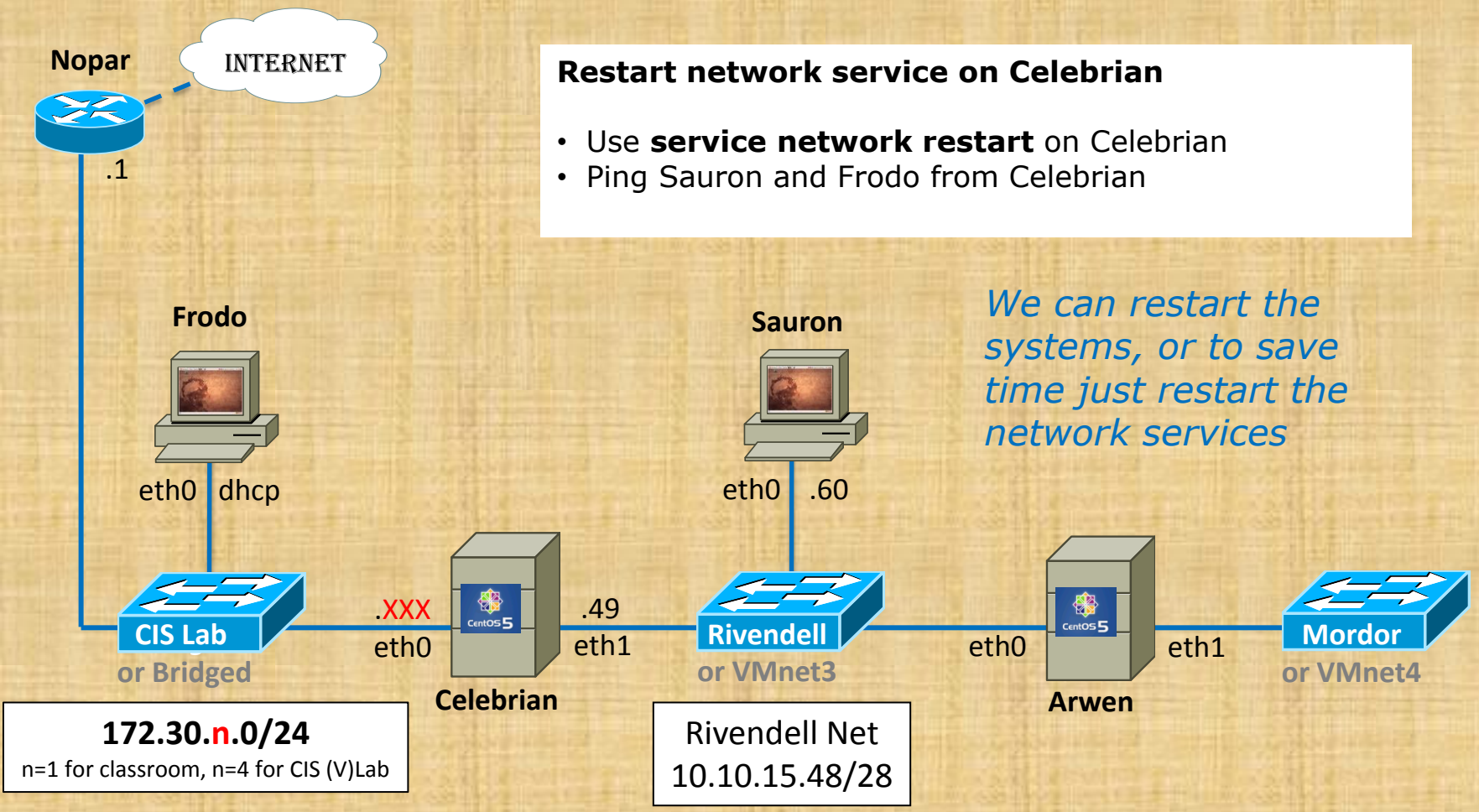
Restarting network services

```
[root@elrond ~]# service network restart
Shutting down interface eth0:                              [  OK  ]
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:
Determining IP information for eth0... done.
                                                           [  OK  ]
```

or

```
[root@elrond ~]# /etc/init.d/network restart
Shutting down interface eth0:                              [  OK  ]
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:
Determining IP information for eth0... done.
                                                           [  OK  ]
```

For Ubuntu 8.10: **/etc/init.d/networking restart**
For OpenSUSE 11.1: **rcnetwork restart**



Restart network service on Celebrian

- Use **service network restart** on Celebrian
- Ping Sauron and Frodo from Celebrian

We can restart the systems, or to save time just restart the network services

Why can't Celebrian ping Sauron or Frodo now?

Set Static IP Address and Subnet Mask

(Red Hat Family)

Temporary

- **ifconfig eth0 172.30.4.125/24**

Permanent

Use tab completion when typing!

- Edit /etc/sysconfig/network-scripts/ifcfg-eth0

There is a different file for each interface

```
[root@elrond ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth0
ONBOOT=yes
HWADDR=00:0c:29:ba:63:76
BOOTPROTO=static
IPADDR=172.30.4.125
NETMASK=255.255.255.0
BROADCAST=172.30.4.255
[root@elrond ~]#
```

Add these static IP settings

Specifying the broadcast is optional. It was required on older versions of Linux where the netmask differed from the classful mask.

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

For new settings to take effect

Set Static IP Address and Subnet Mask

(Red Hat Family)

Verify

- Use **ifconfig** and **ping** commands

```
[root@elrond ~]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:BA:63:76
          inet addr:172.30.4.125  Bcast:172.30.4.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:feba:6376/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:556 errors:0 dropped:0 overruns:0 frame:0
          TX packets:495 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:61635 (60.1 KiB)  TX bytes:82641 (80.7 KiB)
          Interrupt:177 Base address:0x1400

[root@elrond ~]# ping -c 1 172.30.4.1
PING 172.30.4.1 (172.30.4.1) 56(84) bytes of data.
64 bytes from 172.30.4.1: icmp_seq=1 ttl=255 time=3.61 ms

--- 172.30.4.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 3.617/3.617/3.617/0.000 ms
[root@elrond ~]#
```

Set Dynamic IP Address and Subnet Mask

(Red Hat Family)

Temporary

- **dhclient** and **dhclient –r** to release

Permanent

Use tab completion when typing!

- Edit /etc/sysconfig/network-scripts/ifcfg-eth0

There is a different file for each interface

```
[root@elrond ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth0
ONBOOT=yes
HWADDR=00:0c:29:ba:63:76
BOOTPROTO=dhcp
[root@elrond ~]#
```

Add this for DHCP

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

For new settings to take effect

Set Dynamic IP Address and Subnet Mask

(Red Hat Family)

Verify

- Use **ifconfig** and **ping** commands

```
[root@elrond ~]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:BA:63:76
          inet addr:172.30.4.168  Bcast:172.30.4.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:feba:6376/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3548 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2135 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:824033 (804.7 KiB)  TX bytes:287392 (280.6 KiB)
          Interrupt:177 Base address:0x1400

[root@elrond ~]# ping -c 1  google.com
PING google.com (74.125.67.100) 56(84) bytes of data.
64 bytes from gw-in-f100.google.com (74.125.67.100): icmp_seq=1 ttl=240 time=44.0 ms

--- google.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 44.088/44.088/44.088/0.000 ms
[root@elrond ~]#
```

Snickers will set the DNS server as well

Getting a dynamic address using dhclient

verbose option to see DORA

dhclient -v eth0

```
[root@elrond ~]# dhclient -v eth0
Internet Systems Consortium DHCP Client 4.1.1-P1
Copyright 2004-2010 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/00:0c:29:93:78:e4
Sending on   LPF/eth0/00:0c:29:93:78:e4
Sending on   Socket/fallback
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 4
DHCPOFFER from 172.30.4.1
DHCPREQUEST on eth0 to 255.255.255.255 port 67
DHCPACK from 172.30.4.1
bound to 172.30.4.153 -- renewal in 3506 seconds.
[root@elrond ~]# _
```

*Using **dhclient** to get an IP address*

Releasing a dynamic address using dhclient

dhclient -v –r eth0

```
[root@elrond ~]# dhclient -v -r eth0
Internet Systems Consortium DHCP Client 4.1.1-P1
Copyright 2004-2010 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/00:0c:29:93:78:e4
Sending on   LPF/eth0/00:0c:29:93:78:e4
Sending on   Socket/fallback
DHCPRELEASE on eth0 to 172.30.1.10 port 67
[root@elrond ~]# _
```

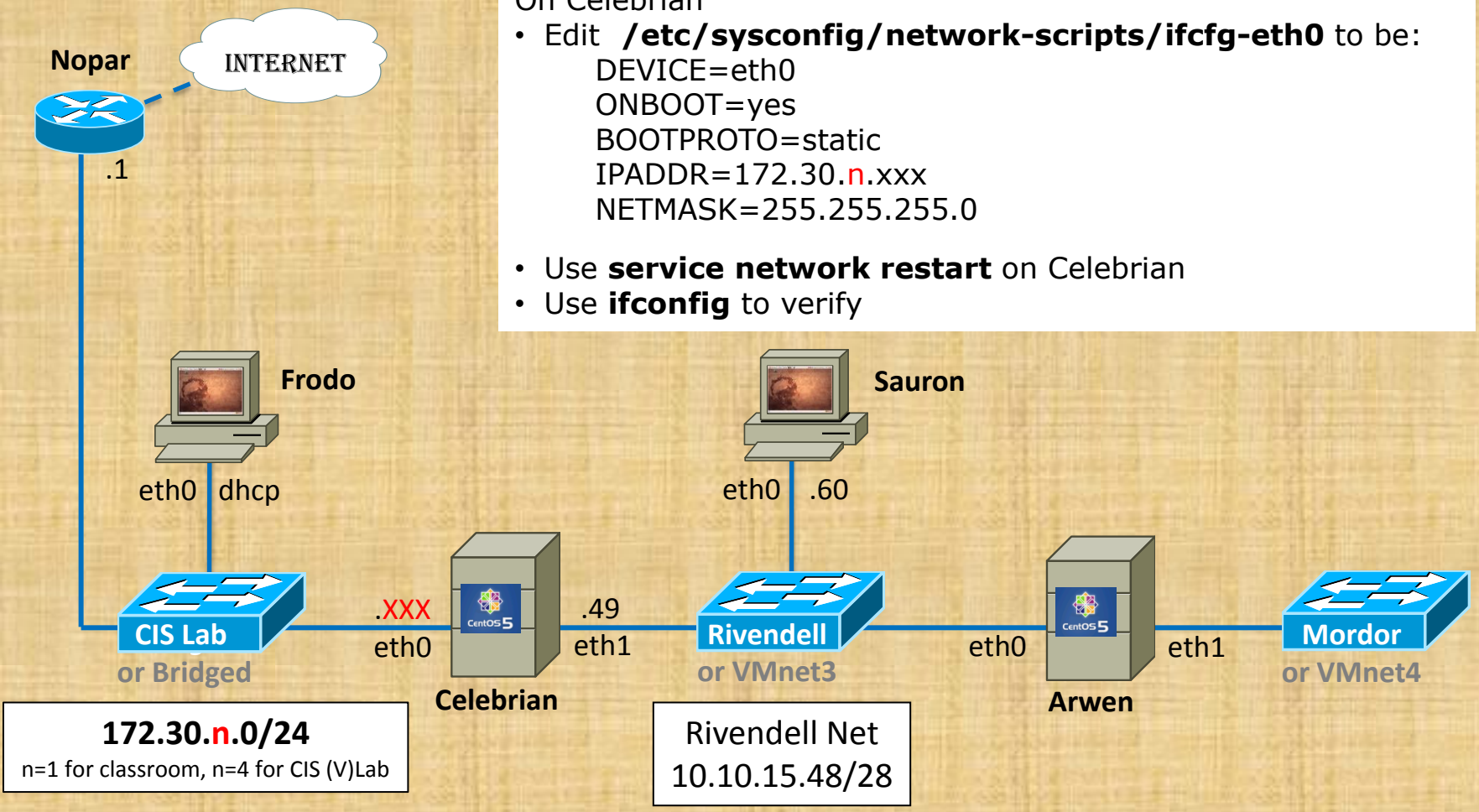
*Using **dhclient –r** to release an IP address*

On Celebrian

- Edit **/etc/sysconfig/network-scripts/ifcfg-eth0** to be:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=static
IPADDR=172.30.n.xxx
NETMASK=255.255.255.0
```

- Use **service network restart** on Celebrian
- Use **ifconfig** to verify



Configure eth0 interface

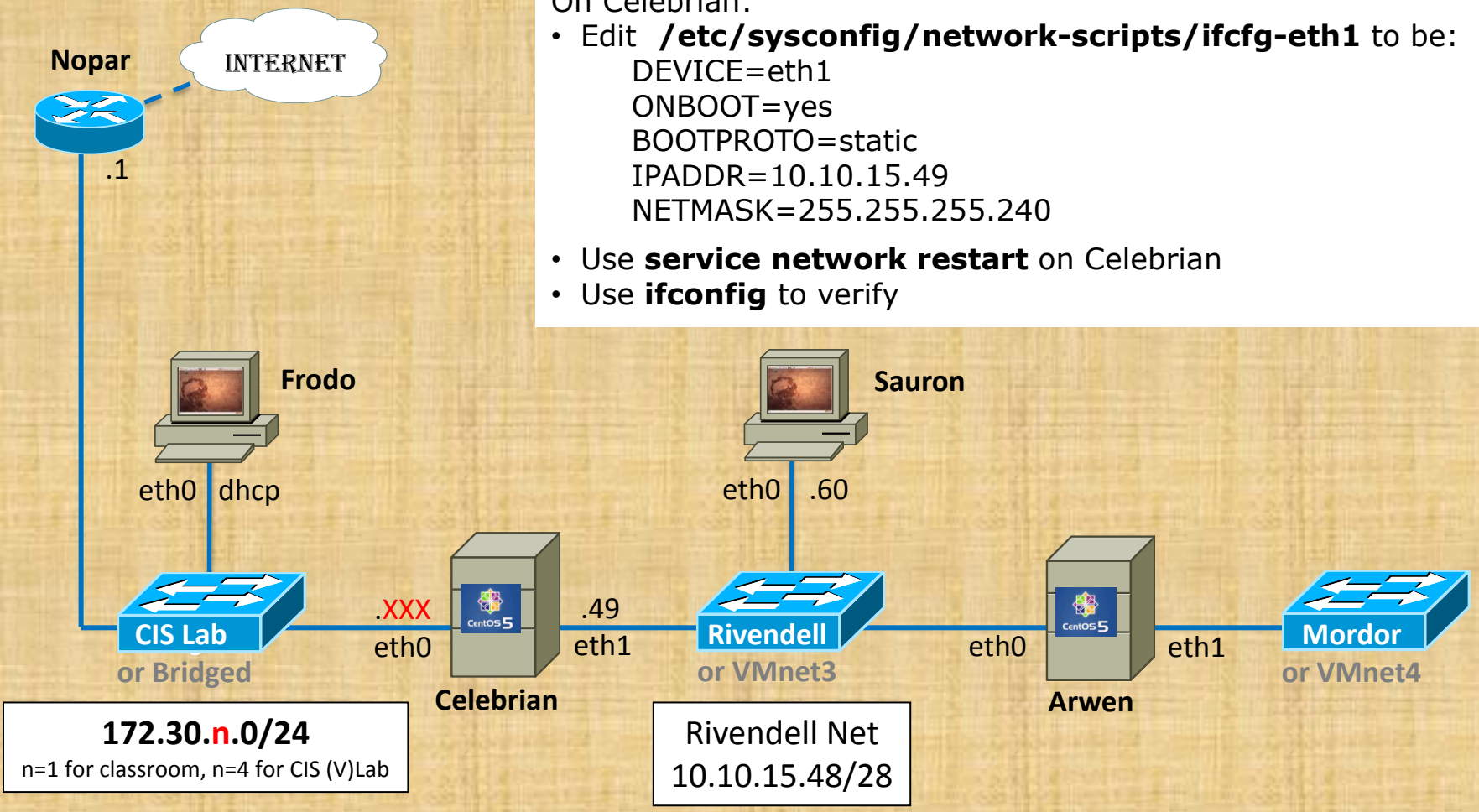
Can Celebrian ping Frodo's IP address?
Can Celebrian ping Sauron's IP address?

On Celebrian:

- Edit **/etc/sysconfig/network-scripts/ifcfg-eth1** to be:

```
DEVICE=eth1
ONBOOT=yes
BOOTPROTO=static
IPADDR=10.10.15.49
NETMASK=255.255.255.240
```

- Use **service network restart** on Celebrian
- Use **ifconfig** to verify



Configure eth1 interface

Can Celebrian ping Frodo and Sauron's IP addresses?
Can Celebrian ping 10.24.1.2 (Cabrillo DNS server)?

Configuring the default gateway
(Red Hat Family)

Temporary
- **route add default gw 172.30.4.1**

Permanent
- Edit **/etc/sysconfig/network**

```
[root@elrond ~]# cat /etc/sysconfig/network
NETWORKING=yes
NETWORKING_IPV6=no
HOSTNAME=elrond.localdomain
GATEWAY=172.30.n.1
[root@elrond ~]#
```

Add the gateway

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

For new settings to take effect

Configuring the default gateway
(Red Hat Family)

Verify

- Use **route** to verify

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth0
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

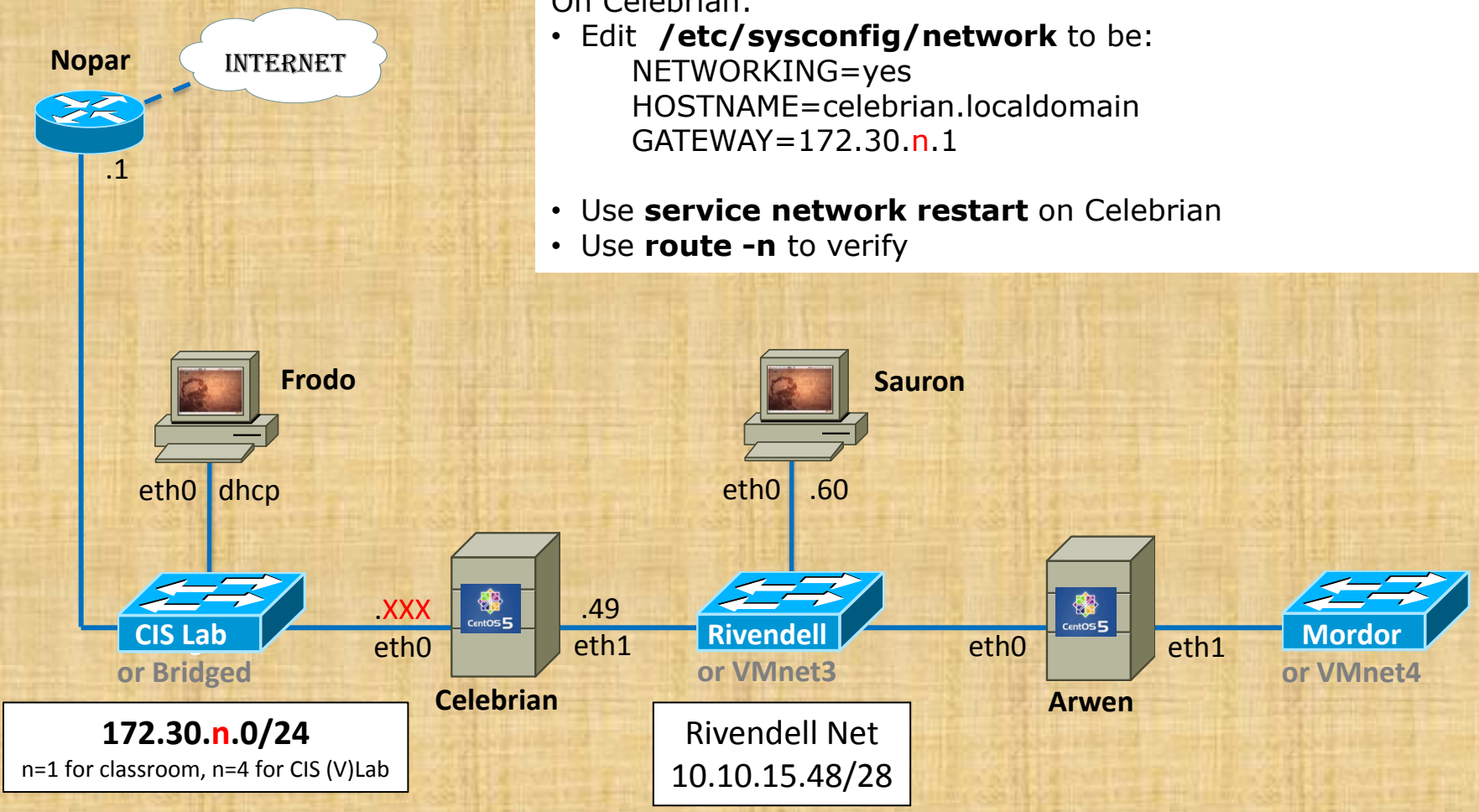
Default route to take

On Celebrian:

- Edit **/etc/sysconfig/network** to be:

```
NETWORKING=yes
HOSTNAME=celebrian.localdomain
GATEWAY=172.30.n.1
```

- Use **service network restart** on Celebrian
- Use **route -n** to verify



Let's configure the default gateway on Celebrian permanently -- can Celebrian ping 10.240.1.2 now?

Configuring the DNS

Permanent
• Edit **/etc/resolv.conf**

```
[root@elrond ~]# cat /etc/resolv.conf
search cabrillo.edu
nameserver 10.240.1.2
[root@elrond ~]#
```

This will be appended to host names when trying to resolve them

May add up to three of these for primary , secondary and tertiary DNS servers

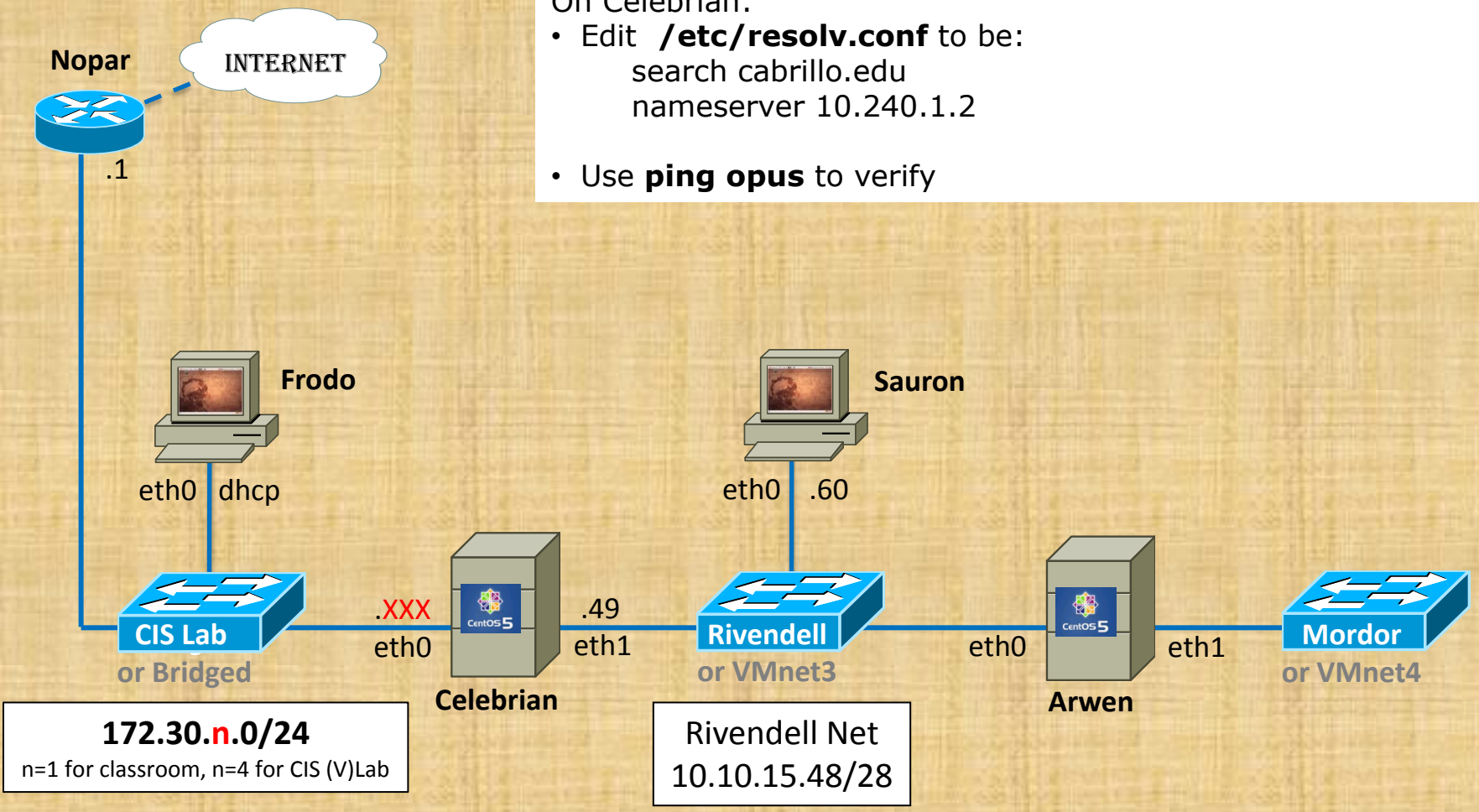
Verify
• Ping by hostname

```
[root@elrond ~]# ping -c 1 opus
PING opus.cabrillo.edu (207.62.186.9) 56(84) bytes of data.
64 bytes from opus.cabrillo.edu (207.62.186.9): icmp_seq=1 ttl=63 time=3.67 ms

--- opus.cabrillo.edu ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 3.671/3.671/3.671/0.000 ms
[root@elrond ~]#
```

Note: On the ping, we can leave the .cabrillo.edu off the hostname since we have it in the search string in /etc/resolv.conf

On Celebrian:

- Edit **/etc/resolv.conf** to be:
  search cabrillo.edu
  nameserver 10.240.1.2
- Use **ping opus** to verify



Let's configure the DNS settings on Celebrian to use the Cabrillo name server

Can Celebrian ping opus.cabrillo.edu?
Can Celebrian ping opus and lilly?

Configuring the hostname
(Red Hat Family)

Permanent: Step 1 - edit **/etc/sysconfig/network**

```
[root@elrond ~]# cat /etc/sysconfig/network
NETWORKING=yes
NETWORKING_IPV6=no
HOSTNAME=homer.localdomain
GATEWAY=172.30.4.1
[root@elrond ~]# init 6
```

change hostname

Restart

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

homer login: root
Password:
Last login: Fri Feb 20 01:23:44 from 172.30.4.103
[root@homer ~]# _
```

new hostname

Configuring the hostname
(Red Hat Family)

Permanent: Step 2 - edit **/etc/hosts**

```
[root@homer ~]# cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       elrond.localdomain elrond localhost.localdomain localhost
::1             localhost6.localdomain6 localhost6
[root@homer ~]#
```

Be sure and update /etc/hosts with new hostname

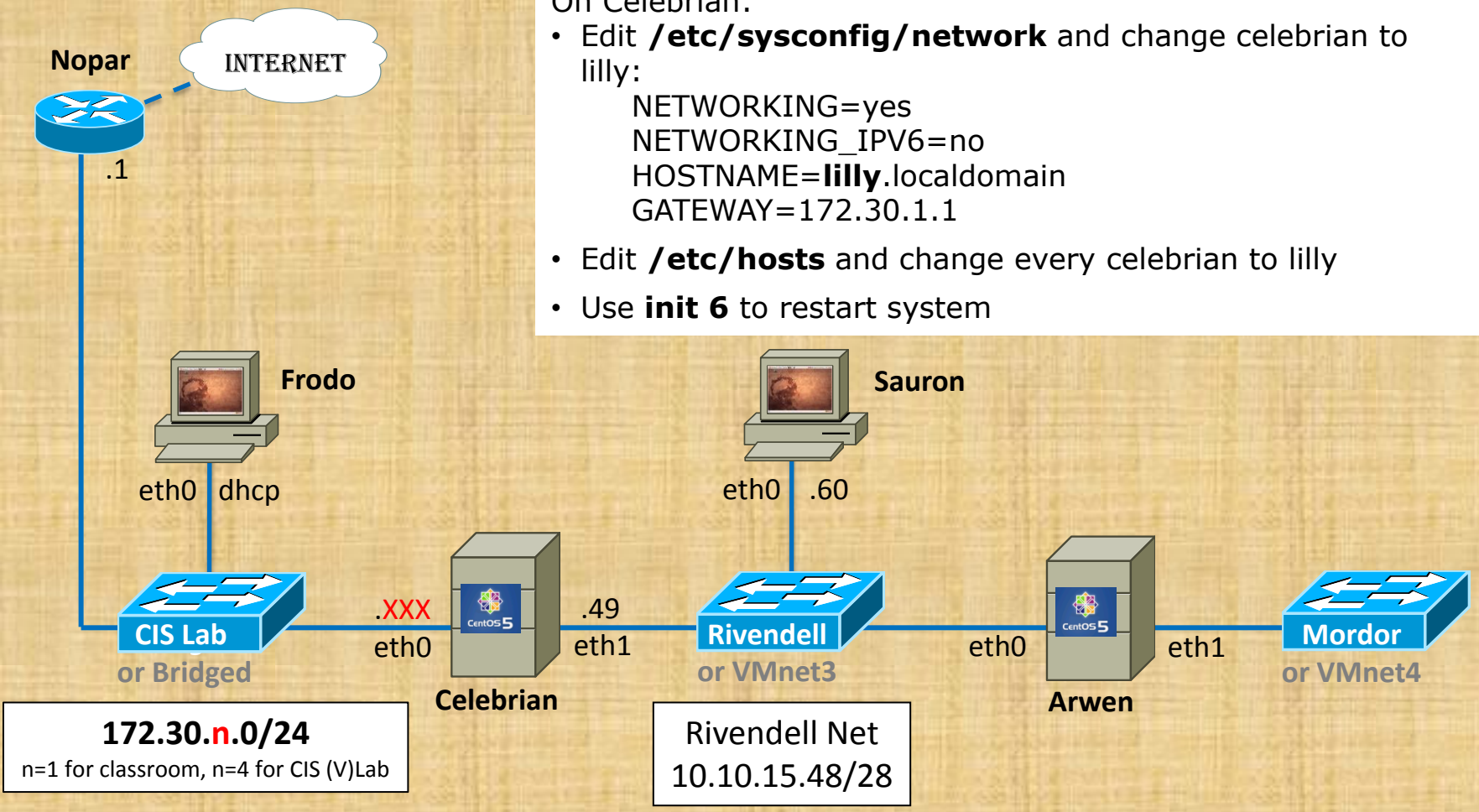
```
[root@homer ~]# cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       homer.localdomain homer localhost.localdomain localhost
::1             localhost6.localdomain6 localhost6
[root@homer ~]#
```

On Celebrian:

- Edit **/etc/sysconfig/network** and change celebrian to lilly:

  NETWORKING=yes
  NETWORKING_IPV6=no
  HOSTNAME=**lilly**.localdomain
  GATEWAY=172.30.1.1

- Edit **/etc/hosts** and change every celebrian to lilly
- Use **init 6** to restart system



Rename Celebrian to Lilly. Note we are changing the system hostname, not the VM name!

What happens to your prompt now after the Celebrian VM restarts?

/etc/hosts

ping frodo vs **ping 172.30.4.150**
ssh frodo vs **ssh 172.30.4.150**
scp myfile frodo: vs **ssh myfile 172.30.4.150:**

- Before the Domain Name System (DNS) arrived on the scene, text files with IP/hostname associations were maintained for name resolution.
- Hostnames are much easier to remember than IP addresses!
- The file **/etc/hosts** files is still available to quickly add local hostnames for systems not resolved by your DNS servers.

/etc/hosts

```
[root@elrond ~]# cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       elrond.localdomain elrond localhost.localdomain localhost
::1             localhost6.localdomain6 localhost6
192.168.2.123   legolas
172.30.4.150    frodo
172.30.4.1      nopar
192.168.3.200   sauron
[root@elrond ~]#
```

Use /etc/hosts to refer to hosts by name rather than IP address

/etc/hosts

using a name rather than an IP address

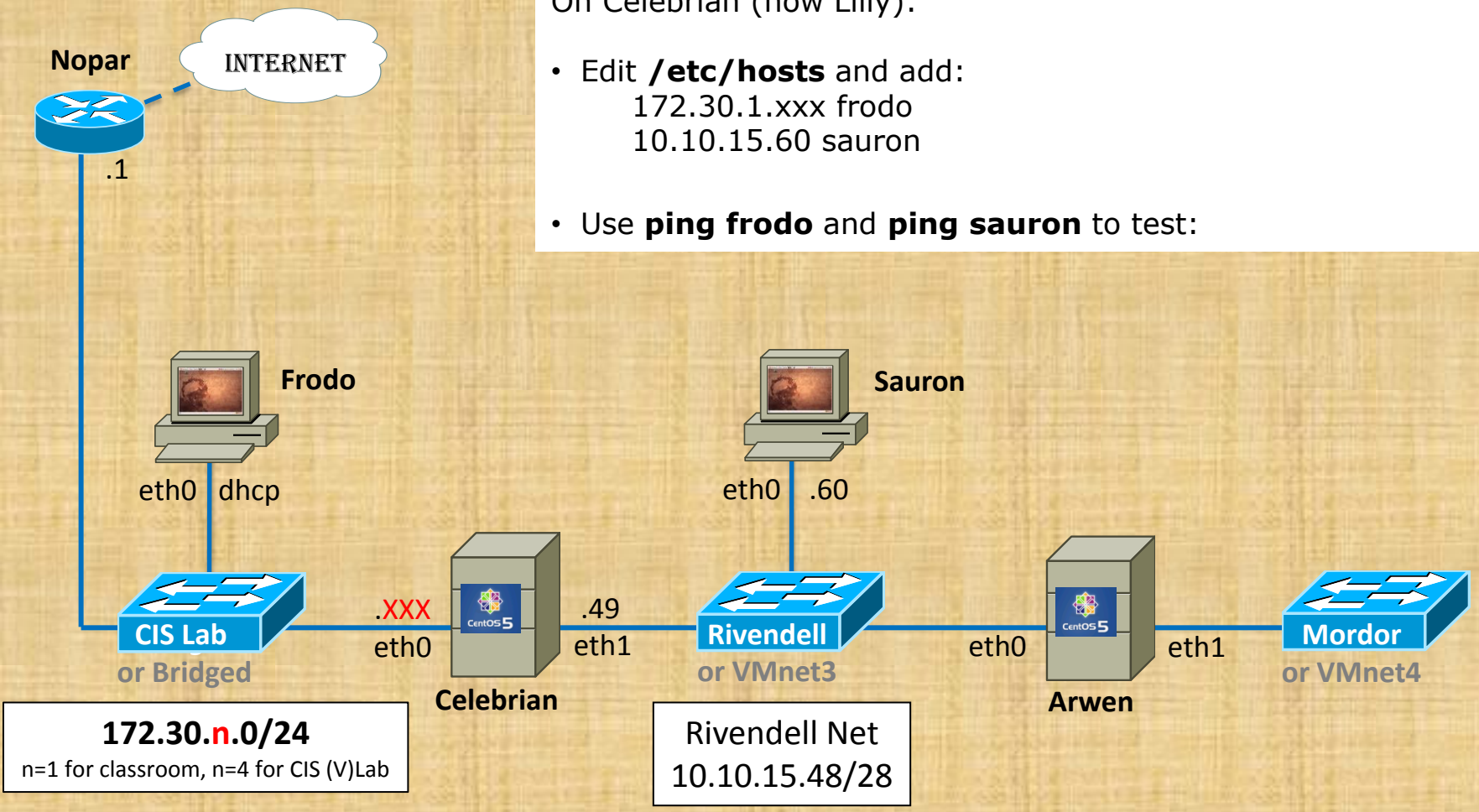
```
[root@elrond ~]# ping nopar
PING nopar (172.30.4.1) 56(84) bytes of data.
64 bytes from nosmo (172.30.4.1): icmp_seq=1 ttl=255 time=1.55 ms

--- nosmo ping statistics ---
2 packets transmitted, 1 received, 50% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.554/1.554/1.554/0.000 ms


[root@elrond ~]# ping sauron
PING sauron (192.168.3.200) 56(84) bytes of data.
64 bytes from sauron (192.168.3.200): icmp_seq=1 ttl=63 time=1.61 ms

--- sauron ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.616/1.616/1.616/0.000 ms
[root@elrond ~]#
```

/etc/hosts was updated with nosmo and sauron and their IP addresses

On Celebrian (now Lilly):

- Edit **/etc/hosts** and add:
  172.30.1.xxx frodo
  10.10.15.60 sauron

- Use **ping frodo** and **ping sauron** to test:



Let's add some remote hostnames so we don't have to type the IP address each time

What happens to your prompt now after the Celebrian VM restarts?

Debian/Ubuntu Network Config

Debian/Ubuntu
Permanent Network Configuration

Configuring DHCP on an interface

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

auto eth0
iface eth0 inet dhcp

root@sun:~#
```

Always need a loopback device on Linux systems

Setting a DHCP IP address

Debian/Ubuntu configuration differs from the Red Hat family. It is a little nicer in that there are fewer files to configure.

Debian/Ubuntu
Permanent Network Configuration

Configuring a static IP, mask and default gateway

```
root@sun:~# cat /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
broadcast 172.30.4.255
network 172.30.4.0

gateway 172.30.4.1

root@sun:~#
```

Always need a loopback device on Linux systems

Setting a static IP address

Setting the default gateway

Debian/Ubuntu
Permanent Network Configuration

Configuring DNS Settings

```
root@sun:~# cat /etc/resolv.conf
search cabrillo.edu
nameserver 10.240.1.2
root@sun:~#
```

Search string is optional. Will be appended to names being resolved if the name by itself does not resolve to an IP address

Same as the Red Hat family

Debian/Ubuntu
Permanent NIC Configuration

Restarting network service
(so new settings take effect)

/etc/init.d/networking restart

Debian/Ubuntu is a little different than Red Hat family

Debian/Ubuntu
System Hostname

Changing system hostname

```
root@jin:~# cat /etc/hostname
jin
root@jin:~#

root@jin:~# vi /etc/hostname
root@jin:~# cat /etc/hostname
sun
root@jin:~# init 6

< system restart snipped >

root@sun:~#
root@sun:~#
```

Change hostname to sun

Restart system

Prompt string changes after reboot

Important, be sure to update /etc/hosts with new hostname!

Debian/Ubuntu
Permanent NIC Configuration

System hostname

```
root@jin:~# cat /etc/hostname
jin
root@jin:~#
```

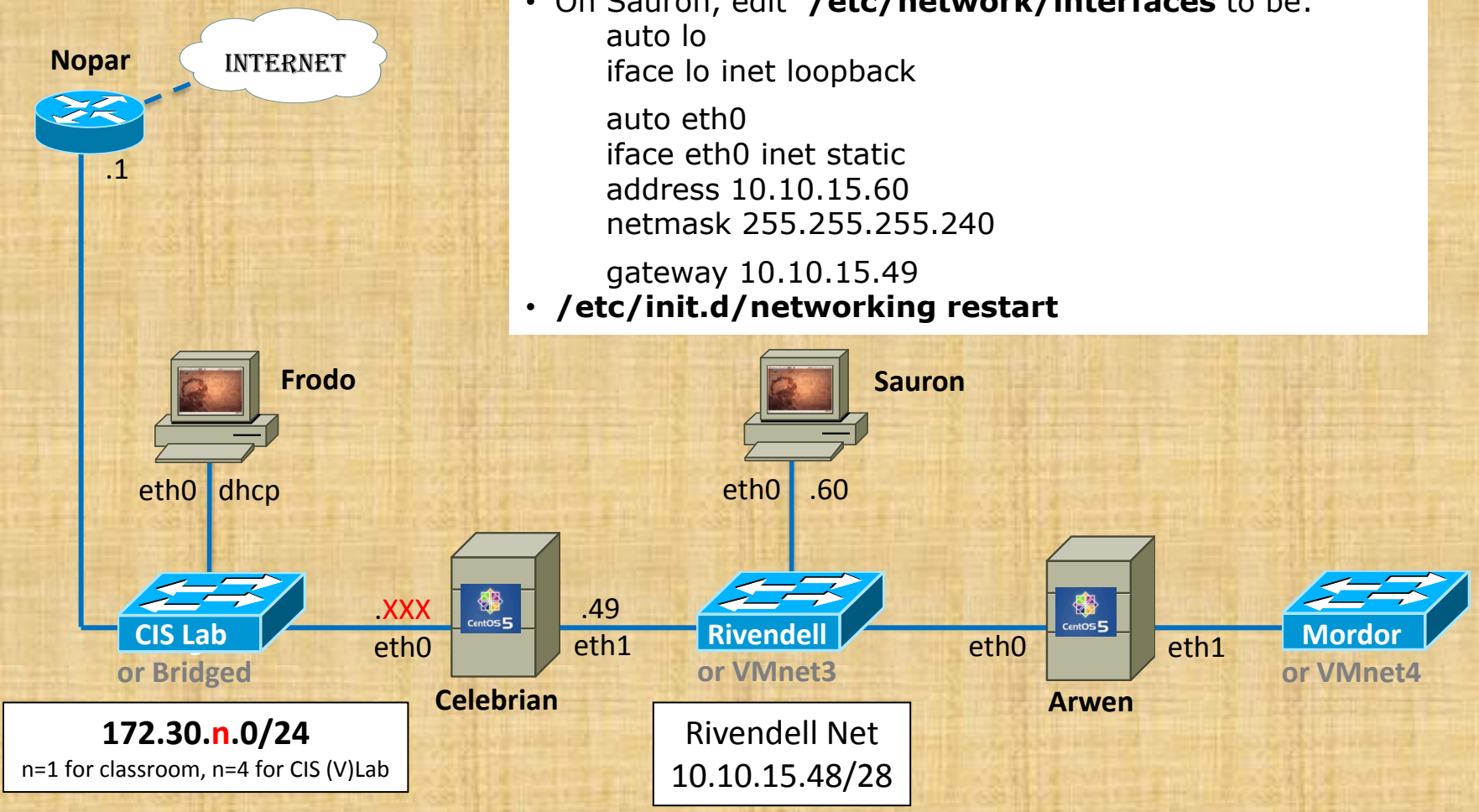
To change the hostname, edit this file.

- *Use just the domain name, not the fully qualified domain name.*
- *The new name will take effect after the next system restart.*

- On Sauron, edit **/etc/network/interfaces** to be:

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 10.10.15.60
netmask 255.255.255.240

gateway 10.10.15.49
```

- **/etc/init.d/networking restart**



Let's configure the eth0 interface on Sauron permanently

Can Celebrian ping Frodo now?

Routing

Routing Summary

sign post

```
[root@lilly ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.10.15.48     0.0.0.0         255.255.255.240 U     0      0        0 eth1
172.30.1.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.1.1      0.0.0.0         UG    0      0        0 eth0
[root@lilly ~]#
```

routing table

- **Routers** operate at **layer 3** and make decisions on where to send a packet.
- The routing decision is based on the routing table.
- If there is no route, the packet is dropped

Now lets looks at some of the details

Routers and the Network Layer (Layer 3)

Cisco commercial Router (IOS)

Linux Server with multiple NICs

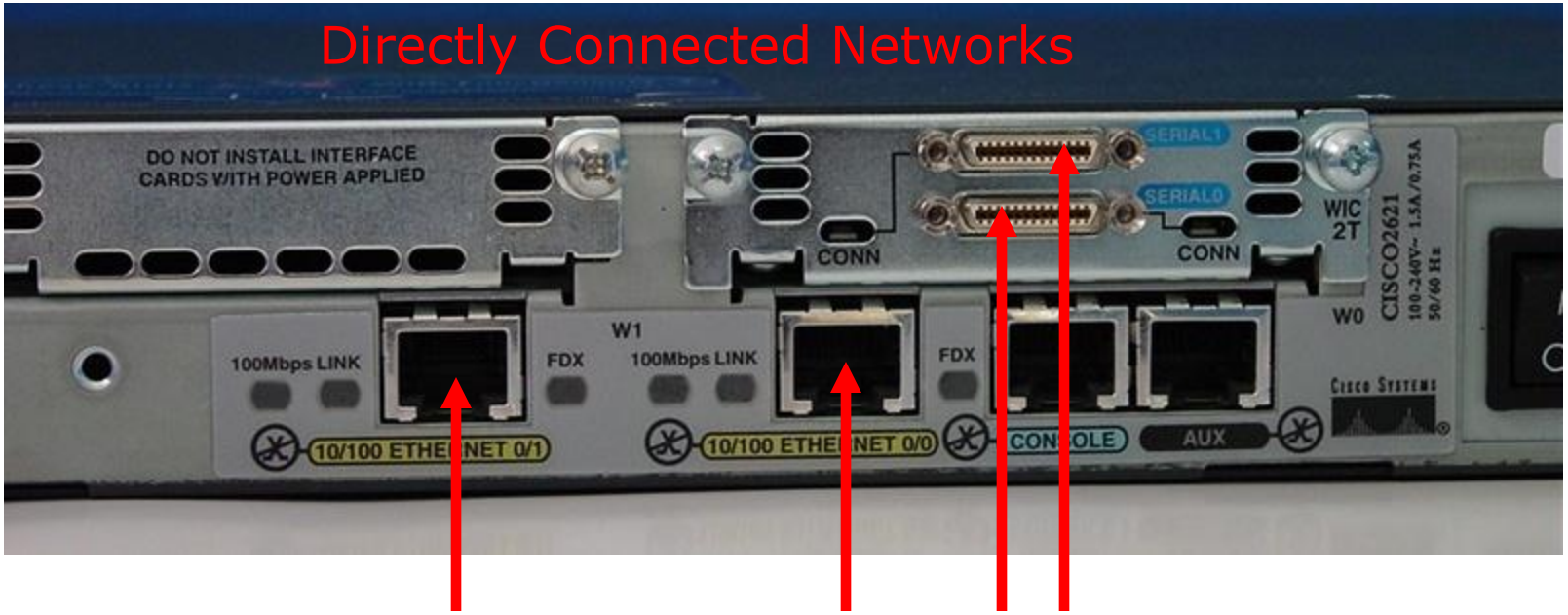
Linksys home Router

Routerboard “kit” with MikroTik RouterOS (based on Linux 2.6 kernel)

Routers and the Network Layer (Layer 3)

Routers

- Networking devices that make best path decisions (which interface to forward the IP packet) based in Layer 3 IP Destination Address.
- Routers connect multiple networks.

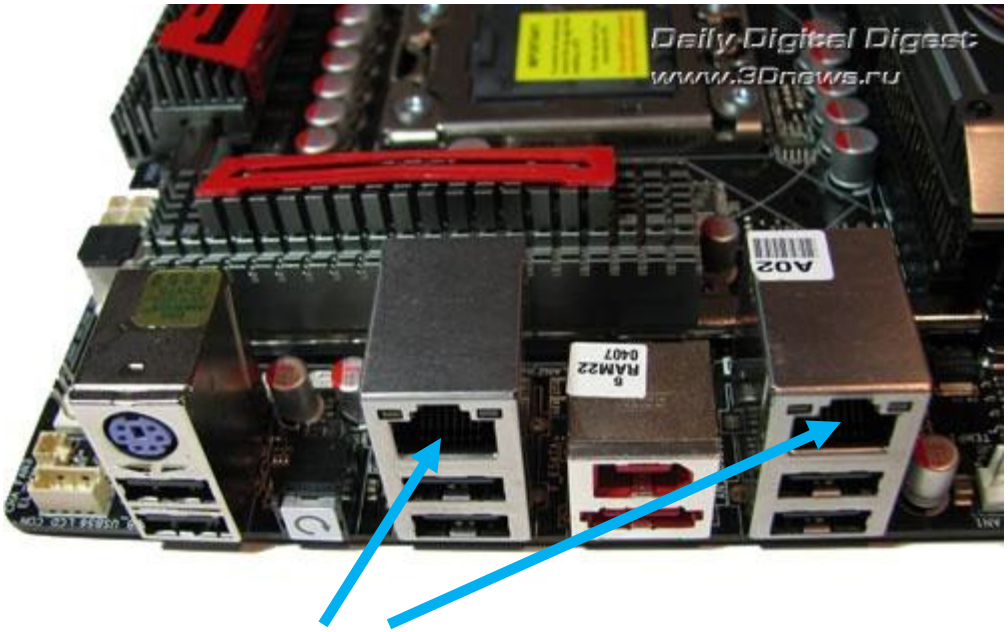


Each interface connects to a different network. Each interface has an IP address/mask for that network.

Routers and the Network Layer (Layer 3)

Linux routers (a computer with multiple NICs)

- Networking devices that make best path decisions (which interface to forward the IP packet) based in Layer 3 IP Destination Address.
- Linux routers connect multiple networks.



Directly Connected Networks

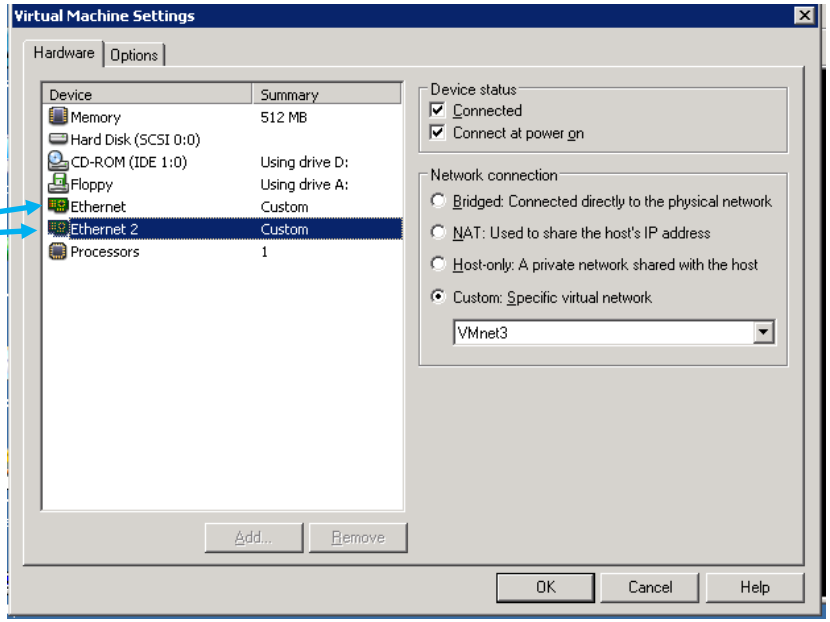
Each interface connects to a different network. Each interface has an IP address/mask for that network.

Routers and the Network Layer (Layer 3)

Virtual Linux routers

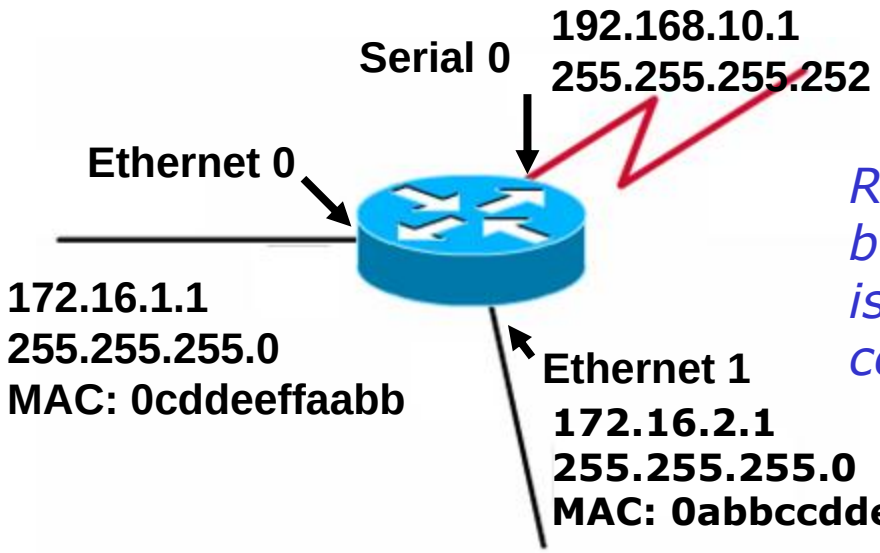
- Networking devices that make best path decisions (which interface to forward the IP packet) based in Layer 3 IP Destination Address.
- Virtual Linux routers connect multiple networks.

Each interface connects to a different network. Each interface has an IP address/mask for that network.



Directly Connected Networks

Routers belong to networks

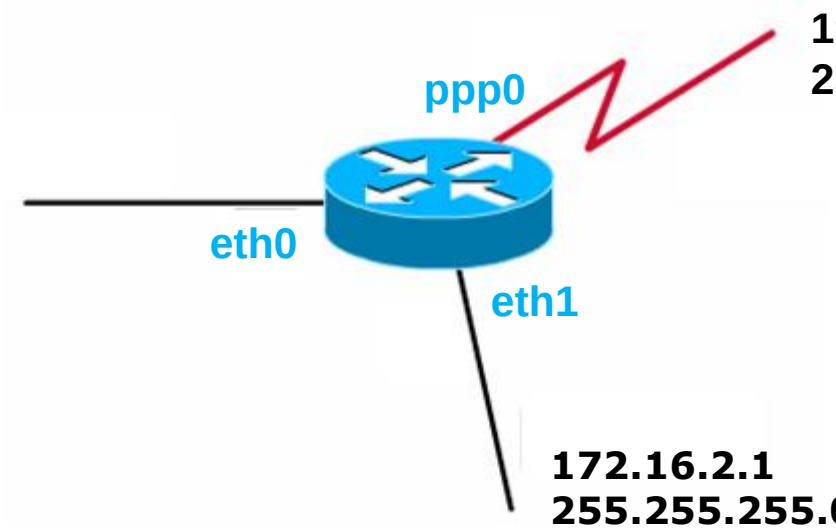


Router interfaces belonging to networks is a very important concept!

Directly Connected Networks

- Router interfaces must be members of different networks.
- Router interfaces participate in the network like other hosts on that network.
- Ethernet interfaces:
  - Have MAC Addresses
  - ARP Tables
  - Participate in the ARP Request and ARP Reply process like other hosts on that network.

Linux routers belong to networks

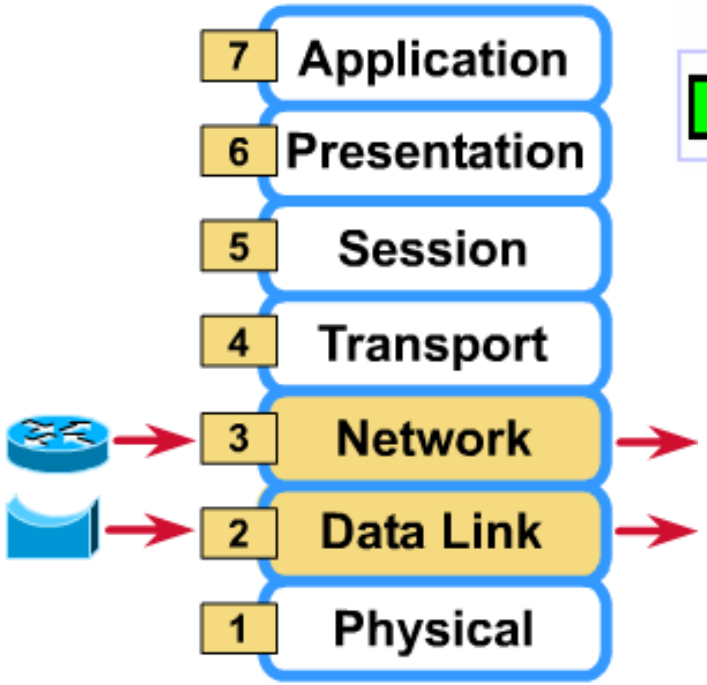


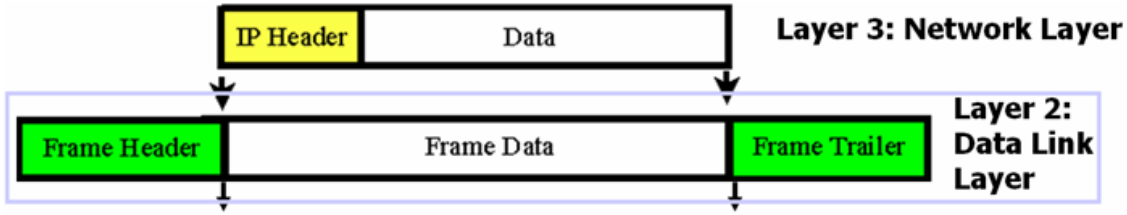
Directly Connected Networks

- Linux router interfaces must be members of different networks.
- Router interfaces participate in the network like other hosts on that network.
- Ethernet interfaces:
  - Have MAC Addresses
  - ARP Tables
  - Participate in the ARP Request and ARP Reply process like other hosts on that network.

Routers

- Make routing decisions based on Layer 3 information:
  - Destination IP address

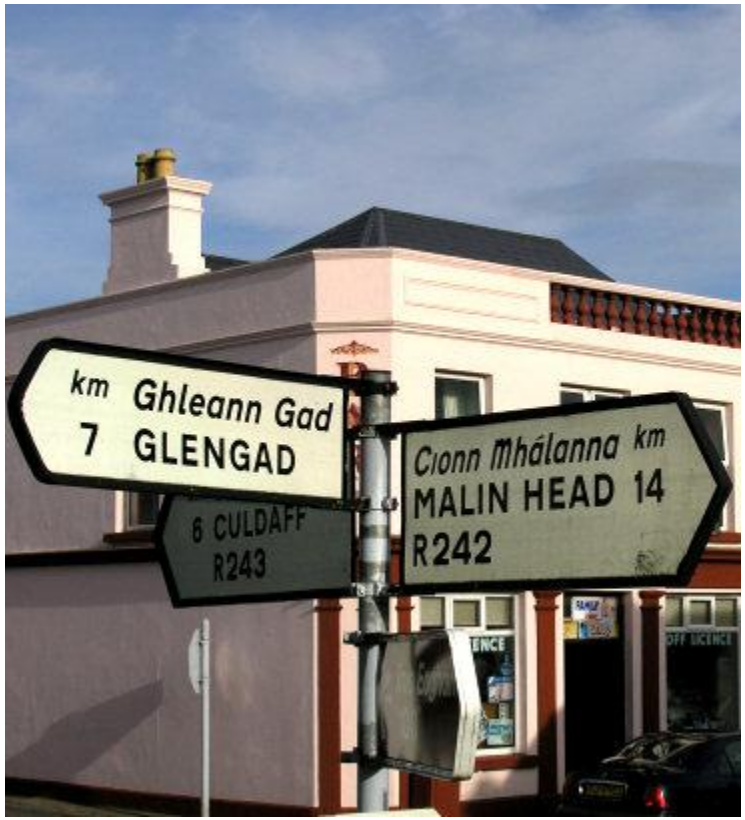




Routers make the routing decision by looking at the destination IP address and the routing table

Rick Graziani
graziani@cabrillo.edu

Routers and the Network Layer



- To get to GLENGAD **go left**
- To get to CULDAFF **go straight on**
- To get to MALIN HEAD **go back**

Using a routing table to make routing decisions is like using a signpost and deciding which direction to go

Note: if there is no sign for your destination you may be LOST!

Routers and the Network Layer

```
[root@lilly ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.10.15.48     0.0.0.0         255.255.255.240 U     0      0        0 eth1
172.30.1.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.1.1      0.0.0.0         UG    0      0        0 eth0
[root@lilly ~]#
```

this is a routing table

- To get to the 10.10.15.48/28 network **take the eth1 interface**
- To get to the 172.30.1.0/24 network **take the eth0 interface**
- To get to the 169.254.0.0/16 network **take the eth1 interface**
- For all other networks **take the eth0 interface till you get to the 172.30.1.1 router and get more instructions there**

Note: if there is no route to the destination the router will DROP the packet!

Routing Types

Two types of routes … static or dynamic

- A router must learn about non-directly connected networks either statically or dynamically.
- **Directly connected networks** are networks that the router is connected to, has an IP address/mask.
- **Non-directly connected networks** are remote networks connected to other routers.

Static

Uses a programmed route that a network administrator enters into the router

Dynamic

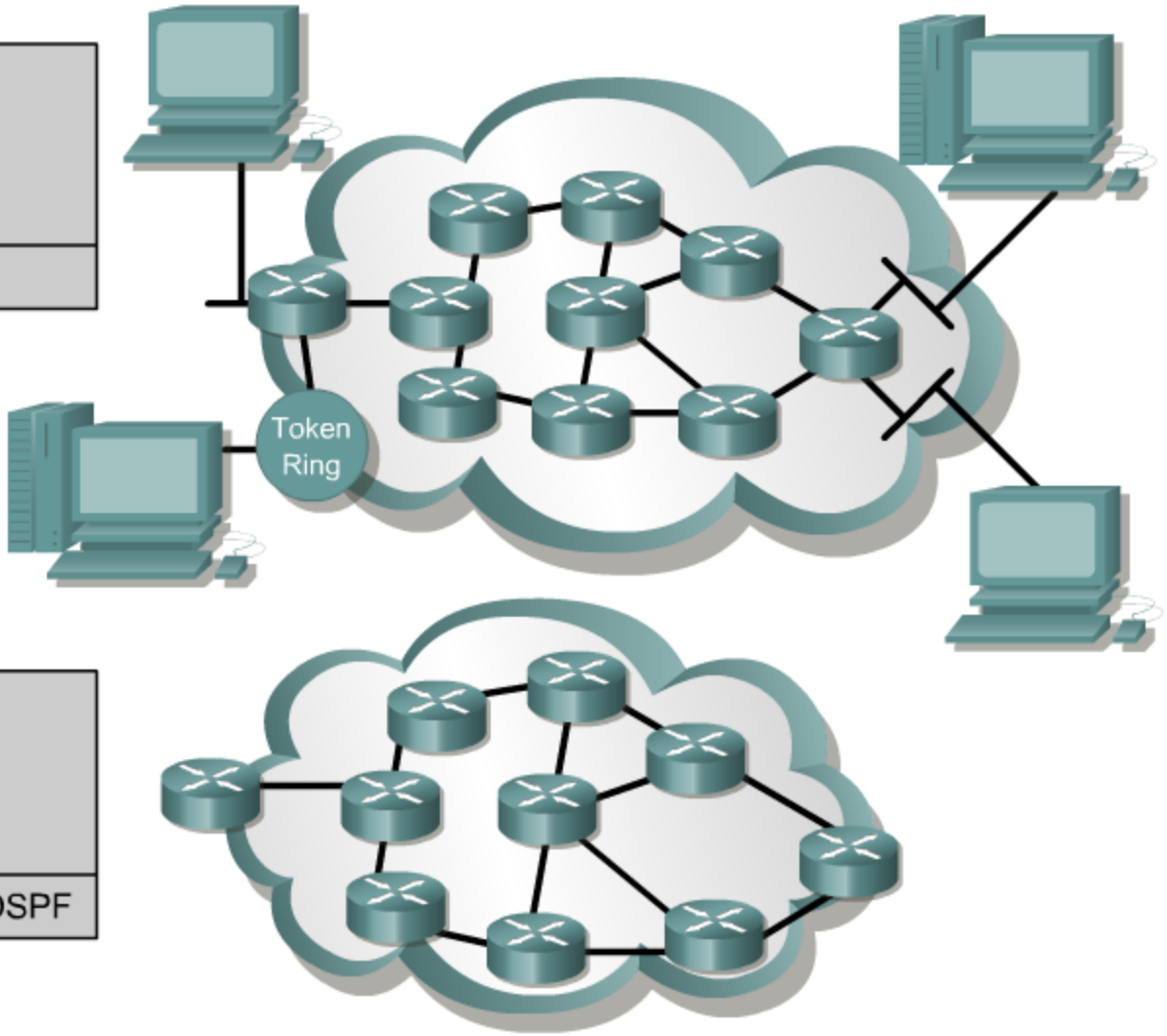
Uses a route that a routing protocol adjusts automatically for topology or traffic changes

Two types of destinations … directly connected or not-directly connected

Routed Protocols vs. Routing Protocols

Routed protocol used between routers to direct user traffic
Examples: IP and IPX

An important distinction!



Routing protocol used between routers to maintain tables
Examples: RIP, IGRP, OSPF

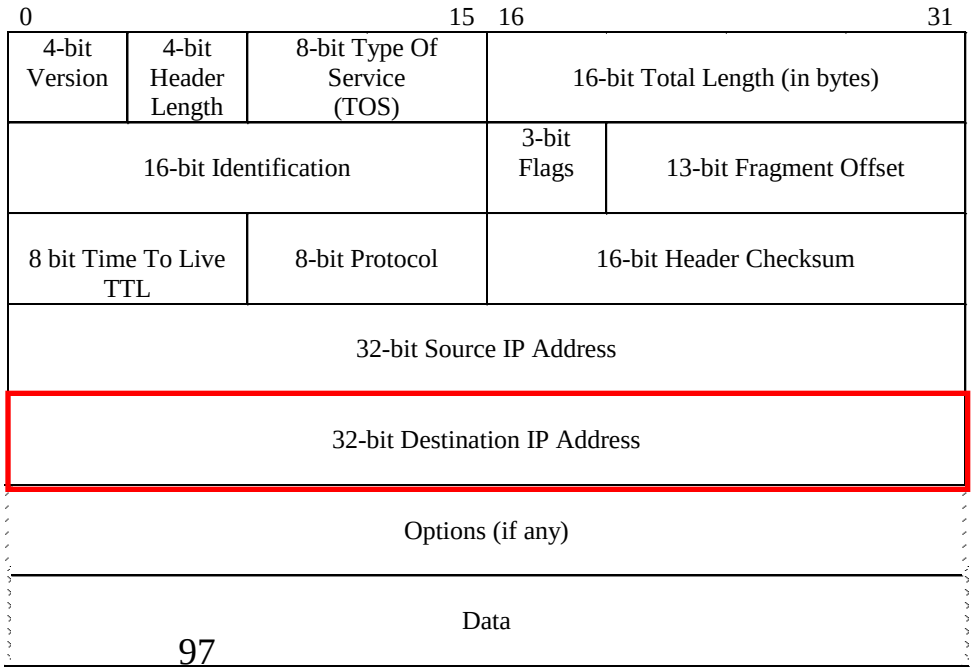
Routed Protocol

- IP is a routed protocol
- A routed protocol is a layer 3 protocol that contains network addressing information.
- This network addressing information is used by routers to determine the which interface, which next router, to forward this packet.

IP packets are routable because of the way IP address are organized and used around the world.

Ethernet frames are not routable. They would require massive routing tables and be an update nightmare!

IP Header



Rick Graziani
graziani@cabrillo.edu

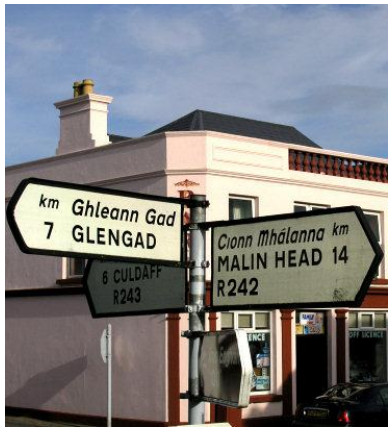
Routing Protocols

- Protocols used by routers to build routing tables.
- Routing tables are used by routers to forward packets.
  - **RIP**
  - **IGRP** and **EIGRP**
  - **OSPF**
  - **IS-IS**
  - **BGP**

This makes maintaining routing tables much more practical!

We will play with some of these next week!

Routing Summary



sign post

```
[root@lilly ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.10.15.48     0.0.0.0         255.255.255.240 U     0      0        0 eth1
172.30.1.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.1.1      0.0.0.0         UG    0      0        0 eth0
[root@lilly ~]#
```

routing table

- Routing is a making a decision on where to send a packet.
- The routing decision is based on the routing table.
- If there is no route, the packet is dropped

Packet Forwarding

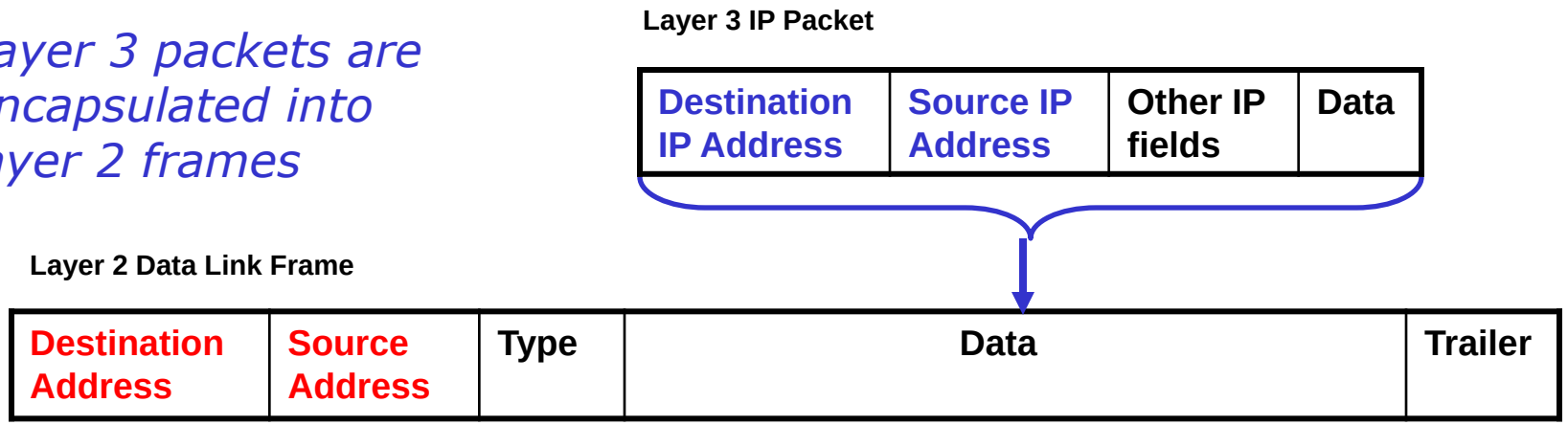
Packet Forwarding Summary

- Frame arrives.
- Layer 3 packet yanked (unencapsulated) from frame and the old frame is discarded.
- Routing decision is made using the destination IP address of the layer 3 packet and the routing table.
- A new layer 2 frame is created containing (encapsulating) the layer 3 packet.
- The new frame is sent out the interface determined by the routing decision.

Now lets looks at some of the details

Encapsulation

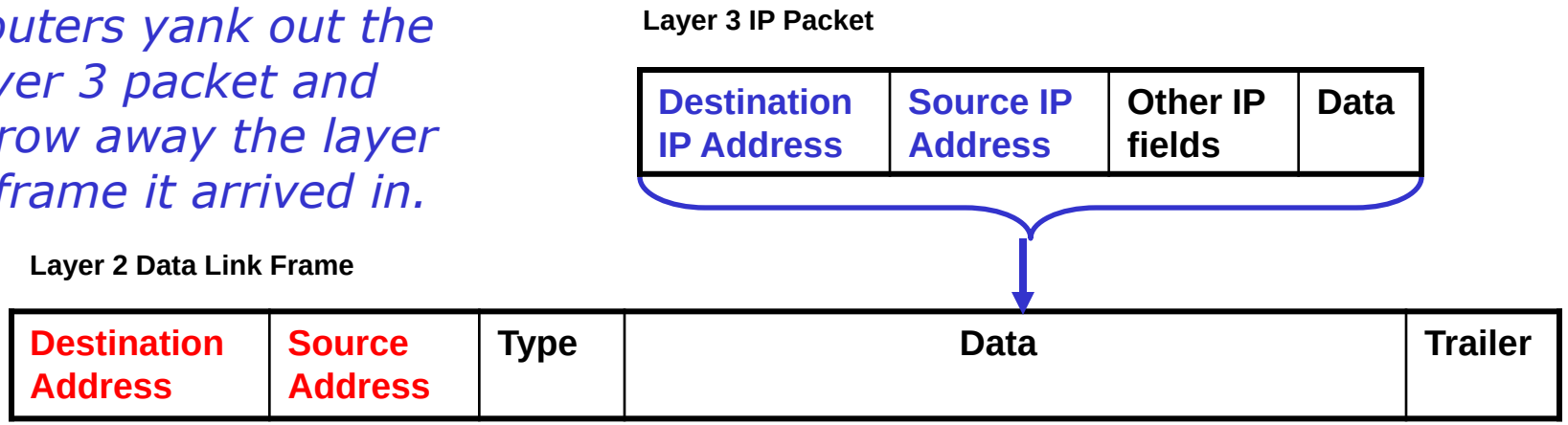
Layer 3 packets are encapsulated into layer 2 frames



- Layer 3 packets are encapsulated into Layer 2 frames by the host.
- **Hubs**: Only flood out the Layer 1 bits (repeater)
- **Switches**: Examine only Layer 2 information:
  1. Learn (Source MAC Address)
  2. Forward (Destination MAC Address)
- **Layer 2 frames** can be non-Ethernet frames, such as serial frames:
  - PPP, HDLC, Frame Relay, ATM, ISDN, etc.
  - Point-to-point serial frames (PPP, HDLC) are not multi-access networks and the Destination Address is many times just a layer 2 broadcast address.

Encapsulation

Routers yank out the layer 3 packet and throw away the layer 2 frame it arrived in.



- **Routers**:
  1. Un-encapsulate Layer 3 packet from Layer 2 frame.
  2. Lookup Layer 3 packet, Destination IP Address, in Routing Table.
  3. Encapsulate Layer 3 packet into new Layer 2 frame and forward out proper (exit) interface.

- **Note**: Destination IP Address and Source IP Address are not in their proper order.

Encapsulation

The layer 3 packet is then placed in a new layer 2 frame appropriate for the next network

These addresses do **not** change!

Layer 3 IP Packet

Destination IP Address	Source IP Address	Other IP fields	Data

These change from host to router, router to router, and router to host.

Layer 2 Data Link Frame

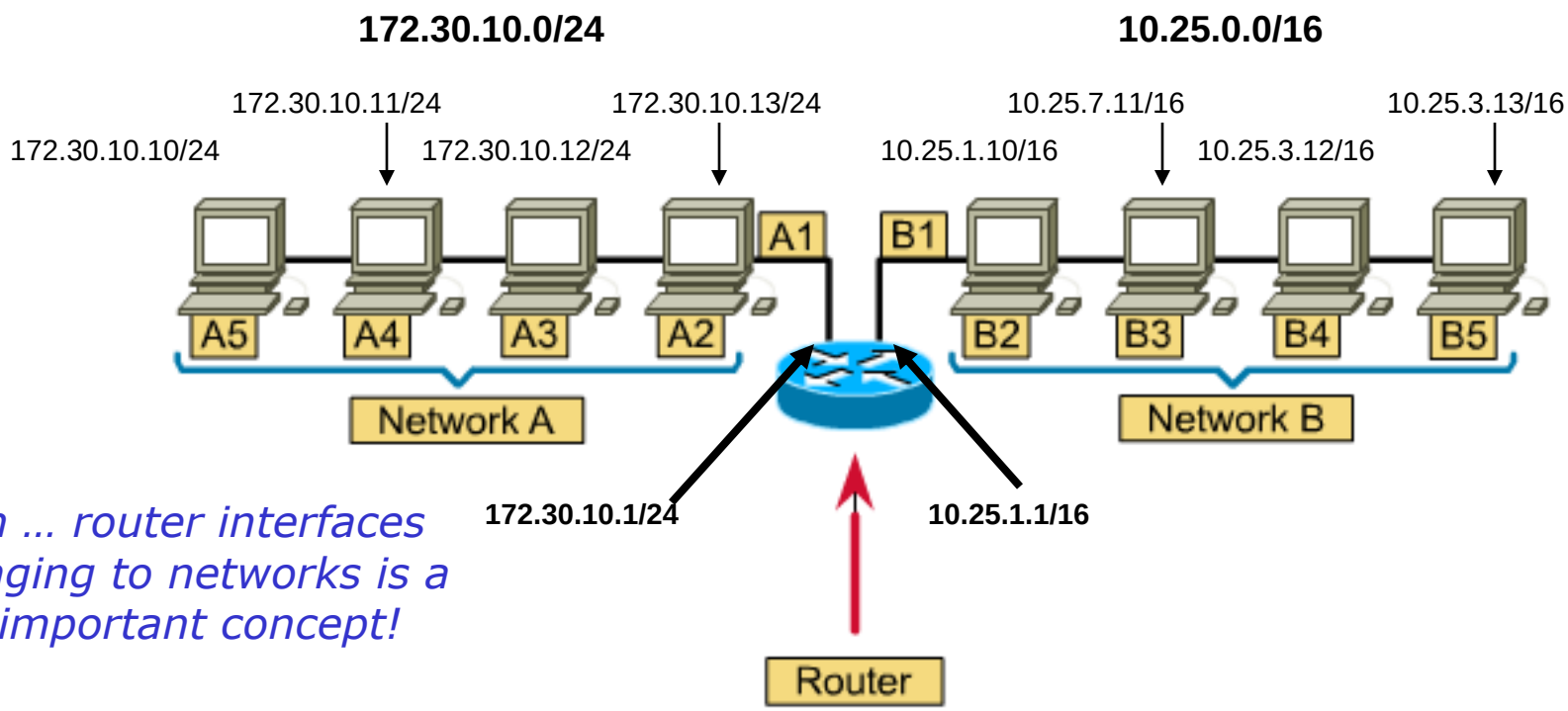
Destination Address	Source Address	Type	Data	Trailer

Next hop Data Link Address of Host or Router's interface

Current Data Link Address of Host or Router's exit interface

- **Note**: The only time Destination and Source IP Addresses change is with NAT/PAT. The only device that is aware of the change is the device doing the NAT, but for all intensive purposes the rule remains the same, IP Addresses do NOT change.

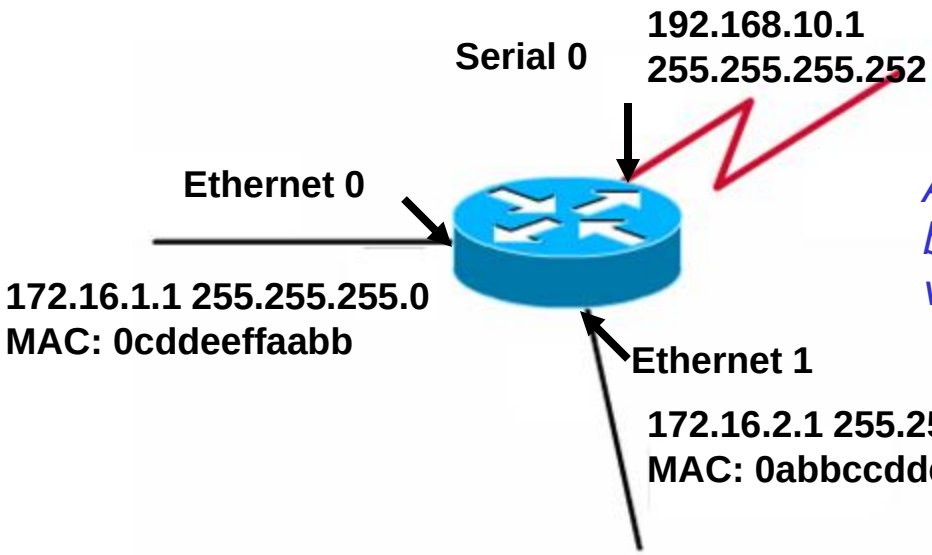
A router interface is a host on that network



Again … router interfaces belonging to networks is a very important concept!

- Since the interface where the router connects to a network is considered to be part of that network.
- Router interfaces have an IP Address and Subnet Mask which makes them a host on the network they are attached.
- Router interfaces must belong to separate networks!

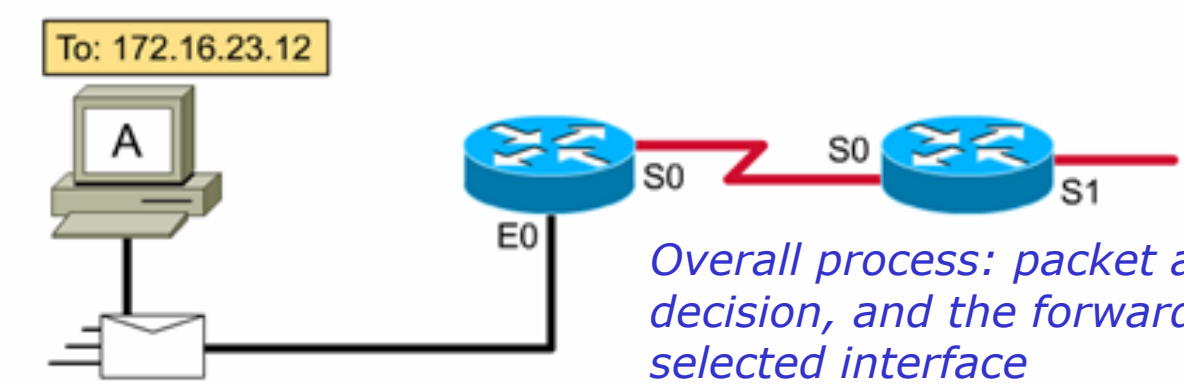
Routers belong to networks



Again … router interfaces belonging to networks is a very important concept!

- Router interfaces must be members of different networks.
- Router interfaces participate in the network like other hosts on that network.
- Ethernet interfaces:
  - Have MAC Addresses
  - ARP Tables
  - Participate in the ARP Request and ARP Reply process like other hosts on that network.

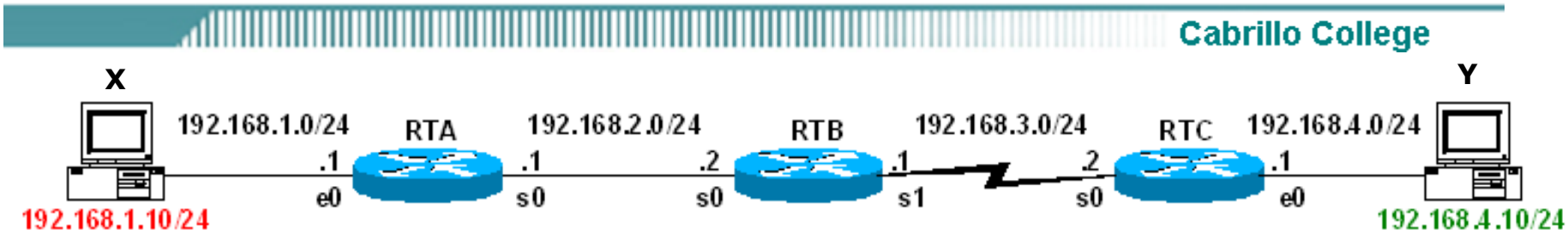
Router's Routing Table



Overall process: packet arrival, routing decision, and the forwarding out the selected interface

- The network layer provides best-effort end-to-end packet delivery across interconnected networks.
- Routers examine the Destination IP Address of a packet to determine where to send the packet next.
- After the router determines which path to use, it proceeds with forwarding the packet.
- It takes the packet that it accepted on one interface and forwards it to another interface or port that reflects the best path to the packet's destination.
- Much more information in the presentation on "The Routing Table Structure" (CIS 82 and CST 311)

Packet Forwarding



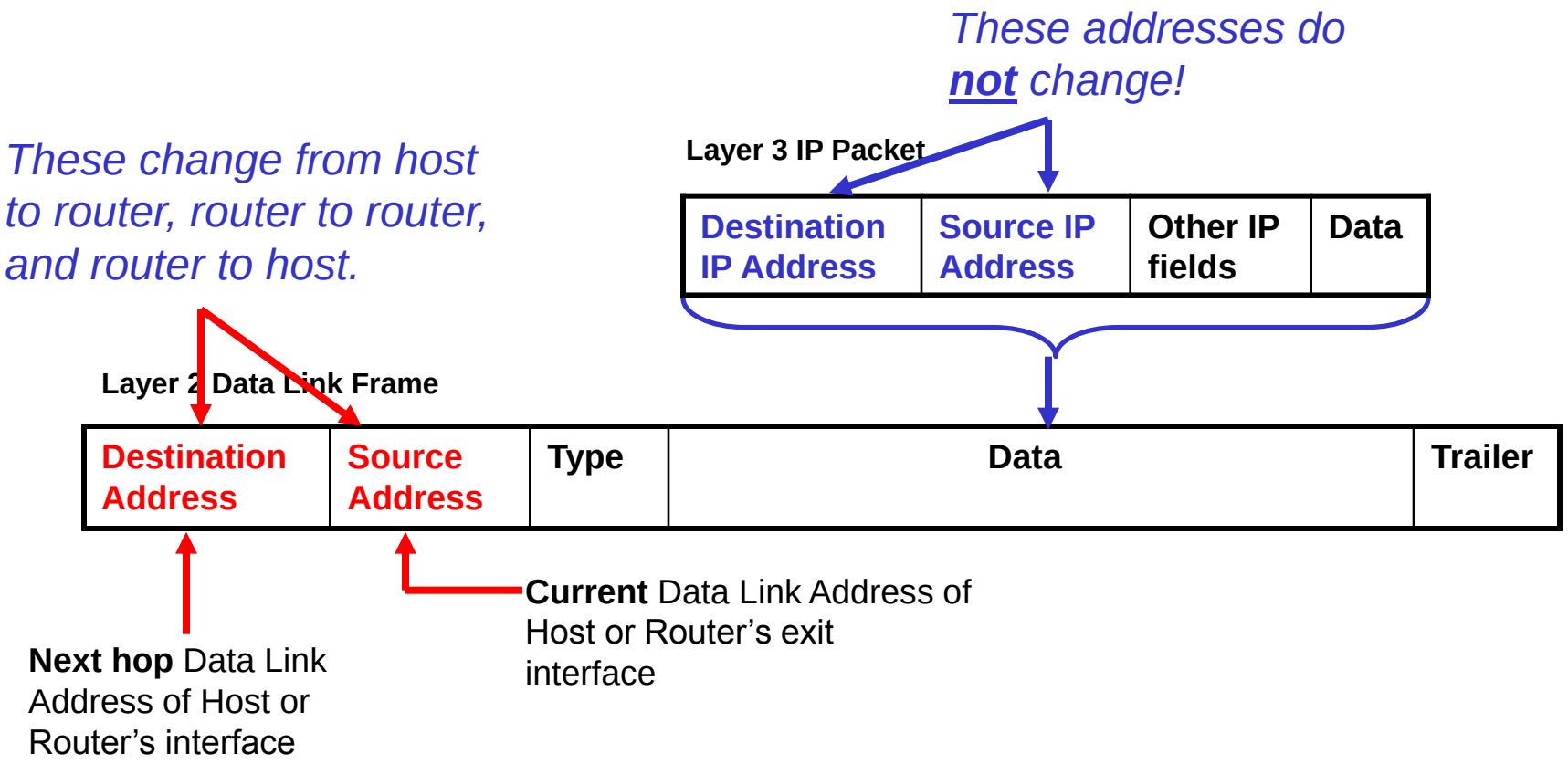
Let's do an example showing how a IP packet travels from network to network to get to its eventual destination.

Packet Forwarding

- Host X has a packet(s) to send to Host Y
- A router generally relays a packet from one data link to another, using two basic functions:
  1. a path determination function - **Routing**
  2. a switching function – **Packet Forwarding**

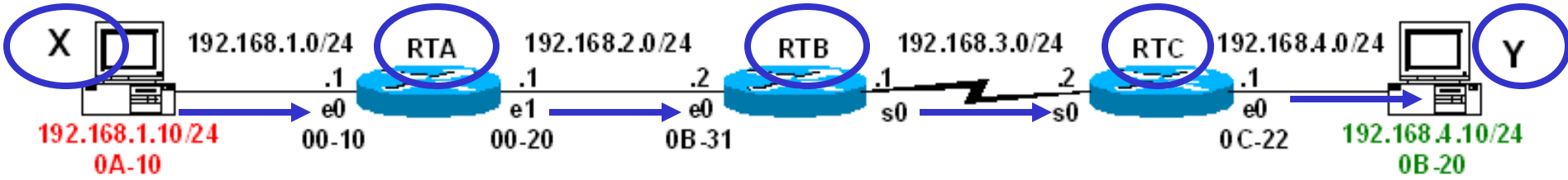
- Let's go through all of the stages these routers use to route and switch this packet.
- **Note**: Data link addresses have been abbreviated.

Remember: Encapsulation



- **Now, let's do an example…**

Remember: Encapsulation



Layer 2 Data Link Frame **Layer 3 IP Packet**

Dest. MAC 0B-20	Source MAC 0C-22	Type 800	Dest. IP 192.168.4.10	Source IP 192.168.1.10	IP fields	Data	Trailer

- This is just a summary.

Linux Routing and Packet Forwarding

- Linux has routing and packet forwarding already built in
- Routing tables are always maintained
- Packet forwarding needs to be enabled

Enable Packet Forwarding
(Red Hat Family)

Temporary

Copy a 1 into **/proc/sys/net/ipv4/ip_forward**

```
[root@elrond ~]# echo 1 > /proc/sys/net/ipv4/ip_forward
```

Or copy a 0 to disable packet forwarding

Enable Packet Forwarding
(Red Hat Family)

Permanent

Edit **/etc/sysctl.conf**

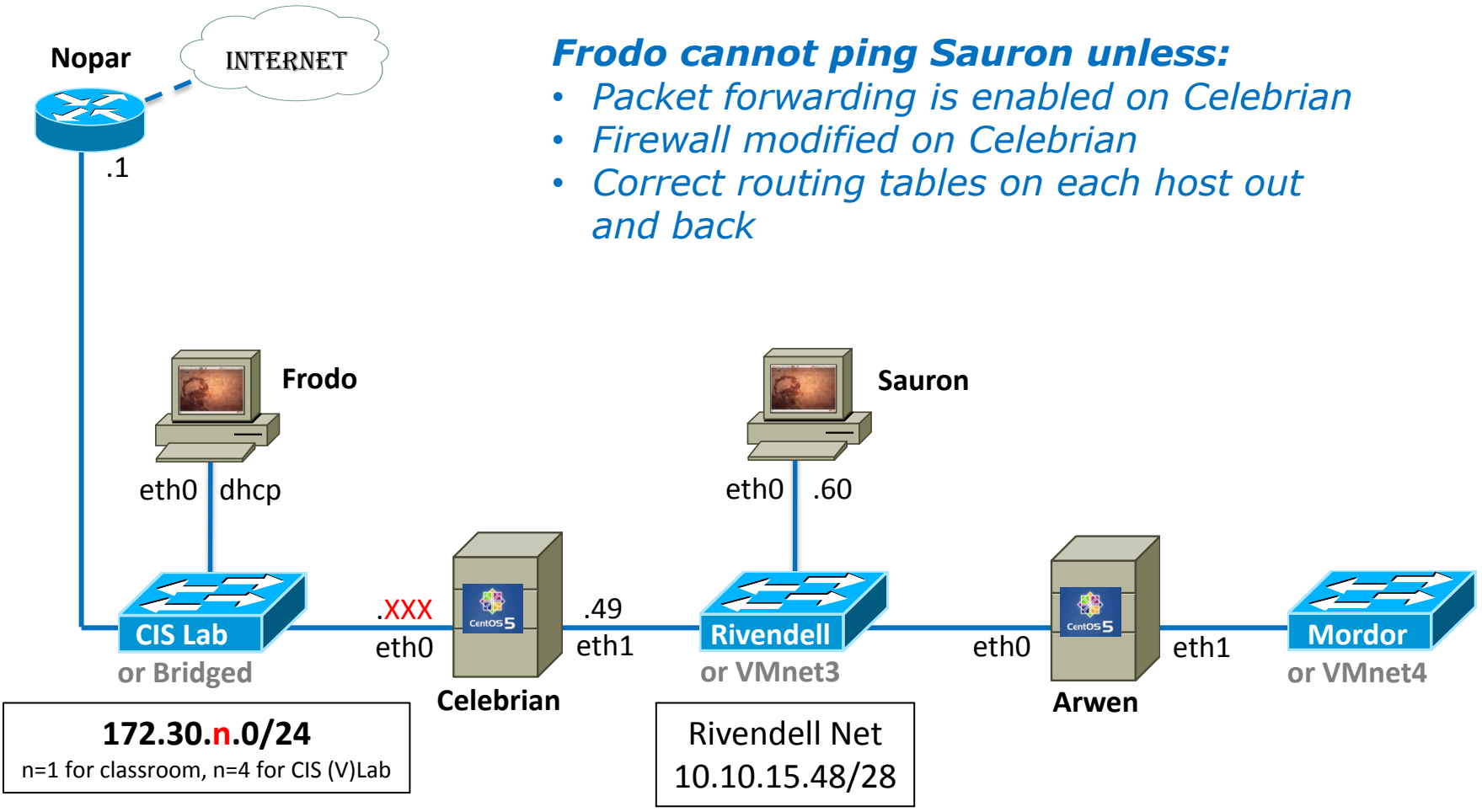
```
[root@elrond ~]# cat /etc/sysctl.conf
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

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

< snipped >

# Controls the maximum number of shared memory segments, in pages
kernel.shmall = 268435456
[root@elrond ~]#
```

enable packet forwarding

Or set a 0 to disable packet forwarding

Packet Forwarding Summary

- Frame arrives.
- Layer 3 packet yanked (unencapsulated) from frame and the old frame is discarded.
- Routing decision is made using the destination IP address of the layer 3 packet and the routing table.
- A new layer 2 frame is created containing (encapsulating) the layer 3 packet.
- The new frame is sent out the interface determined by the routing decision.



Frodo cannot ping Sauron unless:

- *Packet forwarding is enabled on Celebrian*
- *Firewall modified on Celebrian*
- *Correct routing tables on each host out and back*

To ping Sauron (10.10.15.60) from Frodo (172.30.1.193) we need:

- Packet forwarding enabled on Celebrian (Lilly)
- Correct routing tables out and back for the ping packets

Frodo has no route to 10.10.15.48/28 network, and default gateway will send packets in the wrong direction! ☹

```
root@frodo:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.1.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
0.0.0.0         172.30.1.1      0.0.0.0         UG    0      0        0 eth0
```

```
[root@lilly ~]# cat /proc/sys/net/ipv4/ip_forward
0
```

Celebrian packet forwarding is not enabled! ☹

Celebrian routing table has needed network information ☺

```
[root@lilly ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.10.15.48     0.0.0.0         255.255.255.240 U     0      0        0 eth1
172.30.1.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.1.1      0.0.0.0         UG    0      0        0 eth0
```

Sauron routing table does not have information on 172.30.1.0/24 network but default gateway will send packets in the right direction ☺

```
root@sauron:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.10.15.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 eth0
0.0.0.0         10.10.15.49     0.0.0.0         UG    100    0        0 eth0
```

The Firewall

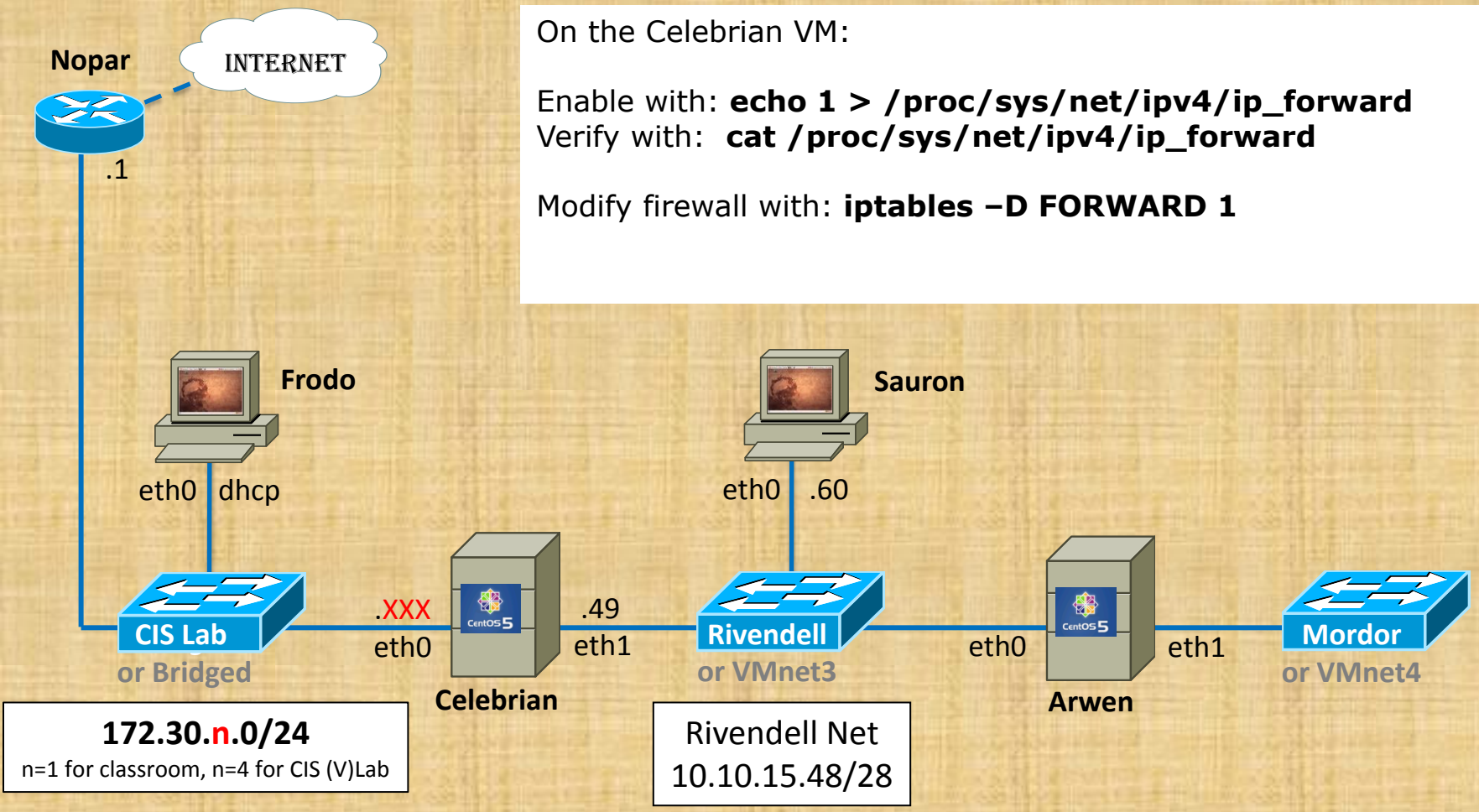
```
[root@lola ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp dpt:ssh
REJECT     all  --  anywhere             anywhere            reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  anywhere             anywhere            reject-with icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@lola ~]#
```

Default firewall rule for FORWARD chain

The Firewall

```
[root@lola ~]# iptables -D FORWARD 1

[root@lola ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp dpt:ssh
REJECT     all  --  anywhere             anywhere            reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@lola ~]#
```

Firewall modified to forward all packets



On the Celebrian VM:

Enable with: **echo 1 > /proc/sys/net/ipv4/ip_forward**
Verify with: **cat /proc/sys/net/ipv4/ip_forward**

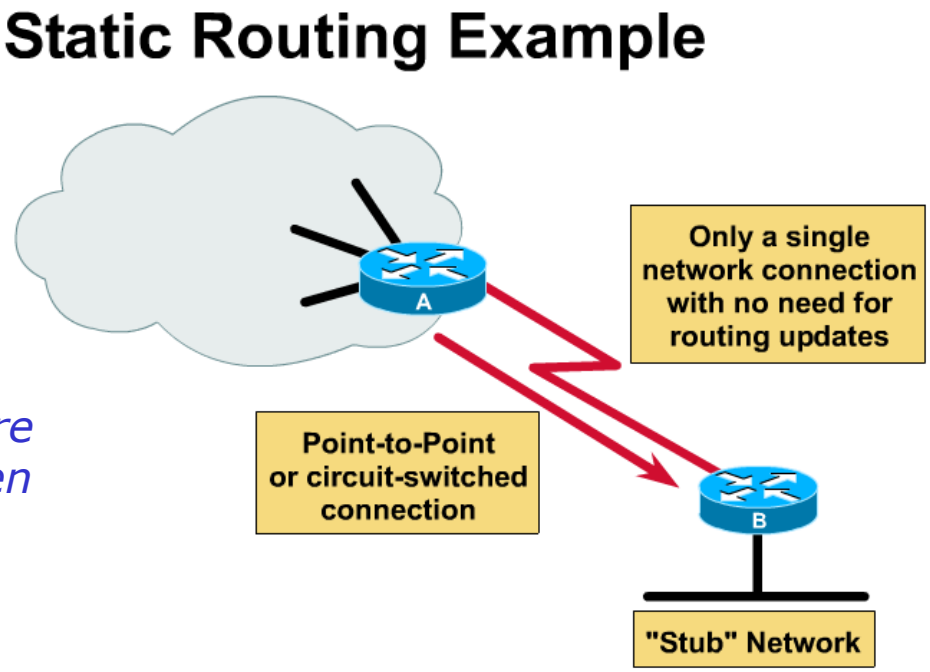
Modify firewall with: **iptables –D FORWARD 1**

Modify firewall and enable packet forwarding on Celebrian (Lilly)

Can Frodo ping Sauron?

Static Routes

Common uses for Static Routes

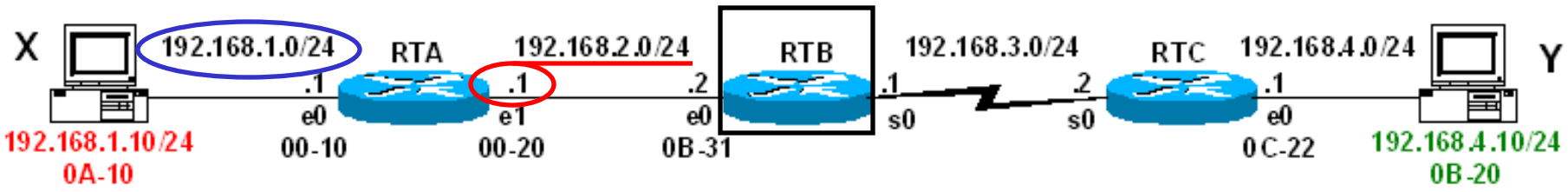


Static routes are very useful even in a dynamic routing world

Static routes in the real-world

- Soon we will learn about **dynamic routing protocols** (RIP, etc.), where routers can learn automatically about networks, without the manual configuration of static routes.
- **Does this mean that static routes are never used in the real-world?**
- **No!** Static routes are used in conjunction with dynamic routing protocols.
- It is common to use a static route where using a dynamic routing protocols would have disadvantages or where it just not needed.

Static Route Examples



- A router must learn about non-directly connected networks.
- To do this with static routes on a **Cisco router:**

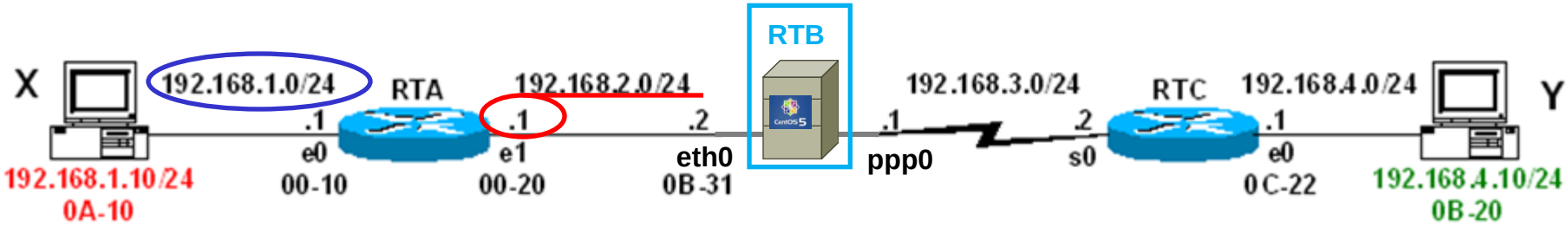
RTB(config)# ip route *network-address mask next-hop-ip-address*

To reach hosts like Host X in the 192.168.1.0/24 network:

RTB(config)# ip route 192.168.1.0 255.255.255.0 192.168.2.1

What would be the static route to reach hosts like Host Y in the 192.168.4.0/24 network?

Static Route Examples



- A router must learn about non-directly connected networks.
- To do this with static routes on a **Linux router** use:

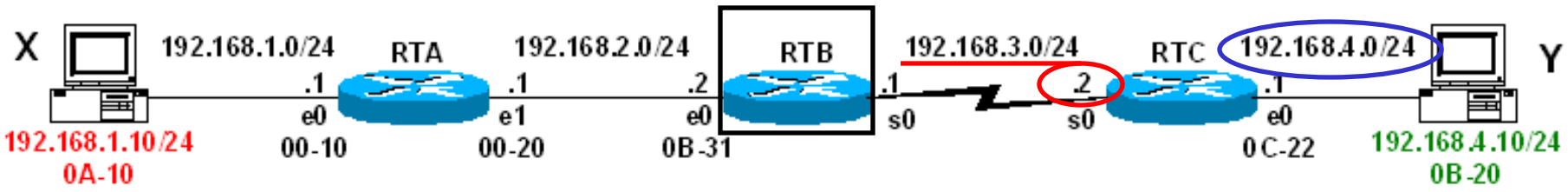
[root@RTB ~#] route add -net *network netmask mask gw next-hop*

To reach hosts like Host X in the 192.168.1.0/24 network:

[root@RTB ~#] route add –net 192.168.1.0 netmask 255.255.255.0 gw 192.168.2.1

What would be the static route to reach hosts like Host Y in the 192.168.4.0/24 network?

Static Route Examples

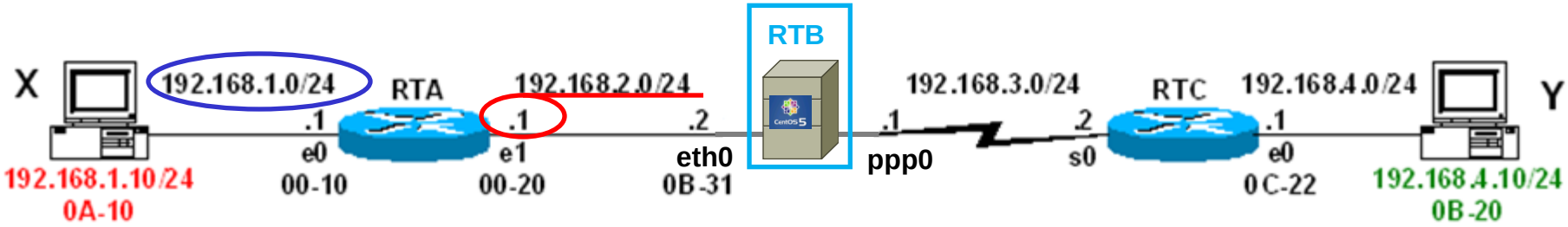


RTB(config)# ip route *network-address mask next-hop-ip-address*

To reach hosts like Host Y in the 192.168.4.0/24 network:

RTB(config)# ip route 192.168.4.0 255.255.255.0 192.168.3.2

Static Route Examples

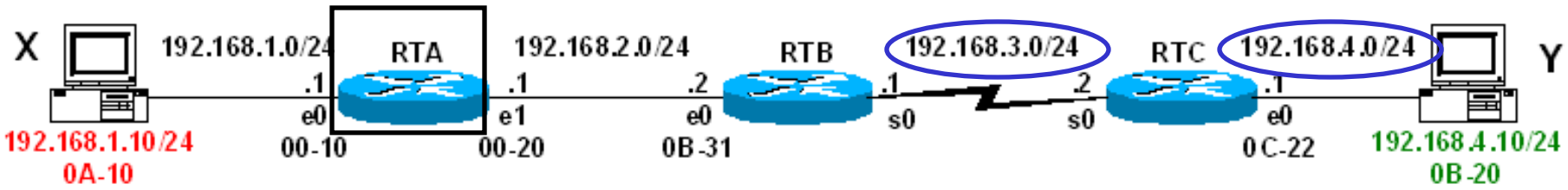


[root@RTB ~#] route add -net *network netmask mask gw next-hop*

To reach hosts like Host Y in the 192.168.4.0/24 network:

[root@RTB ~#] route add –net 192.168.4.0 netmask 255.255.255.0 gw 192.168.3.2

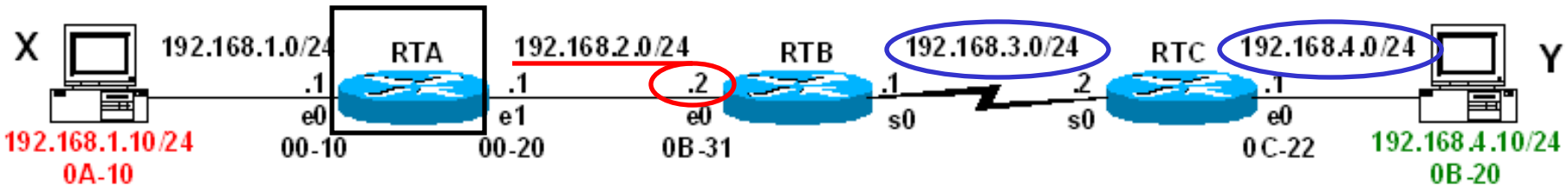
Static Route Examples



What would be the static routes for RTA to reach 192.168.3.0/24 and 192.168.4.0/24 networks?

RTA(config)# ip route *network-address mask next-hop-ip-address*

Static Route Examples



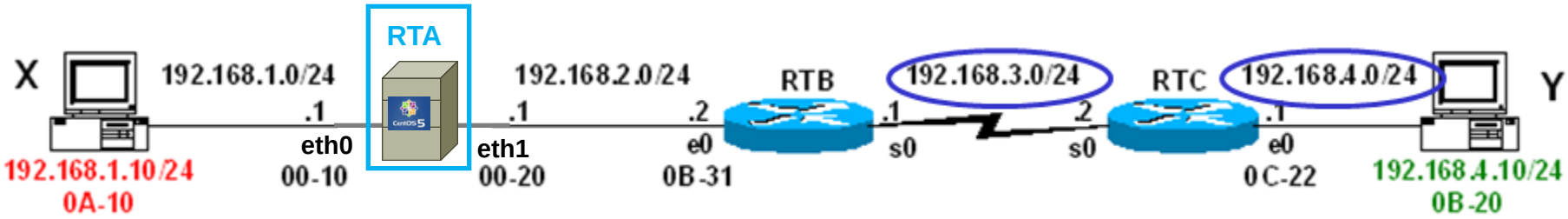
RTB(config)# ip route *network-address mask next-hop-ip-address*

The static routes for RTA to reach 192.168.3.0/24 and 192.168.4.0/24 networks:

RTA(config)# ip route 192.168.3.0 255.255.255.0 192.168.2.2

RTA(config)# ip route 192.168.4.0 255.255.255.0 192.168.2.2

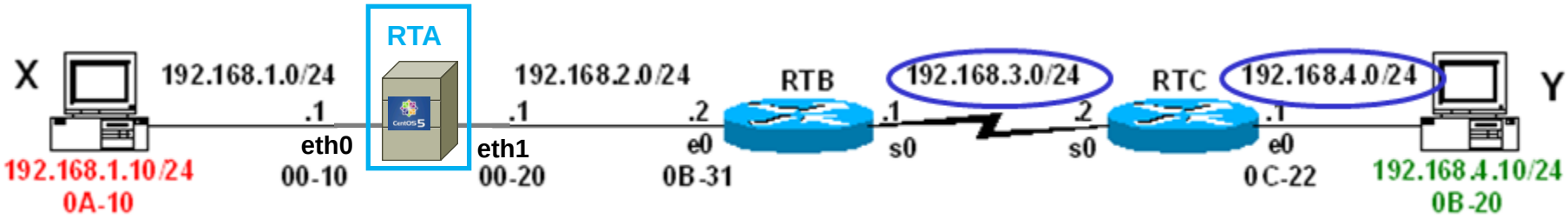
Static Route Examples



What would be the static routes for RTA to reach 192.168.3.0/24 and 192.168.4.0/24 networks?

[root@RTB ~#] route add -net *network netmask mask gw next-hop*

Static Route Examples



[root@RTA ~#] route add -net *network netmask mask gw next-hop*

The static routes for RTA to reach 192.168.3.0/24 and 192.168.4.0/24 networks:

[root@RTA ~#] route add –net 192.168.3.0 netmask 255.255.255.0 gw 192.168.2.2
[root@RTA ~#] route add –net 192.168.4.0 netmask 255.255.255.0 gw 192.168.2.2

Setting Static Routes
(Red Hat Family)

Temporary

- **route add -net *network netmask mask gw next-hop***
- **route del -net *network netmask mask gw next-hop***

```
[root@elrond ~]# route add -net 192.168.3.0 netmask 255.255.255.0 gw 192.168.2.123
[root@elrond ~]# route del -net 192.168.3.0 netmask 255.255.255.0 gw 192.168.2.123
```

- **route add -net *network/prefix gw next-hop***
- **route del -net *network/prefix gw next-hop***

alternate syntax

```
[root@elrond ~]# route add -net 192.168.3.0/24 gw 192.168.2.123
[root@elrond ~]# route del -net 192.168.3.0/24 gw 192.168.2.123
```

Permanent

- Edit **/etc/sysconfig/network-scripts/route-eth***

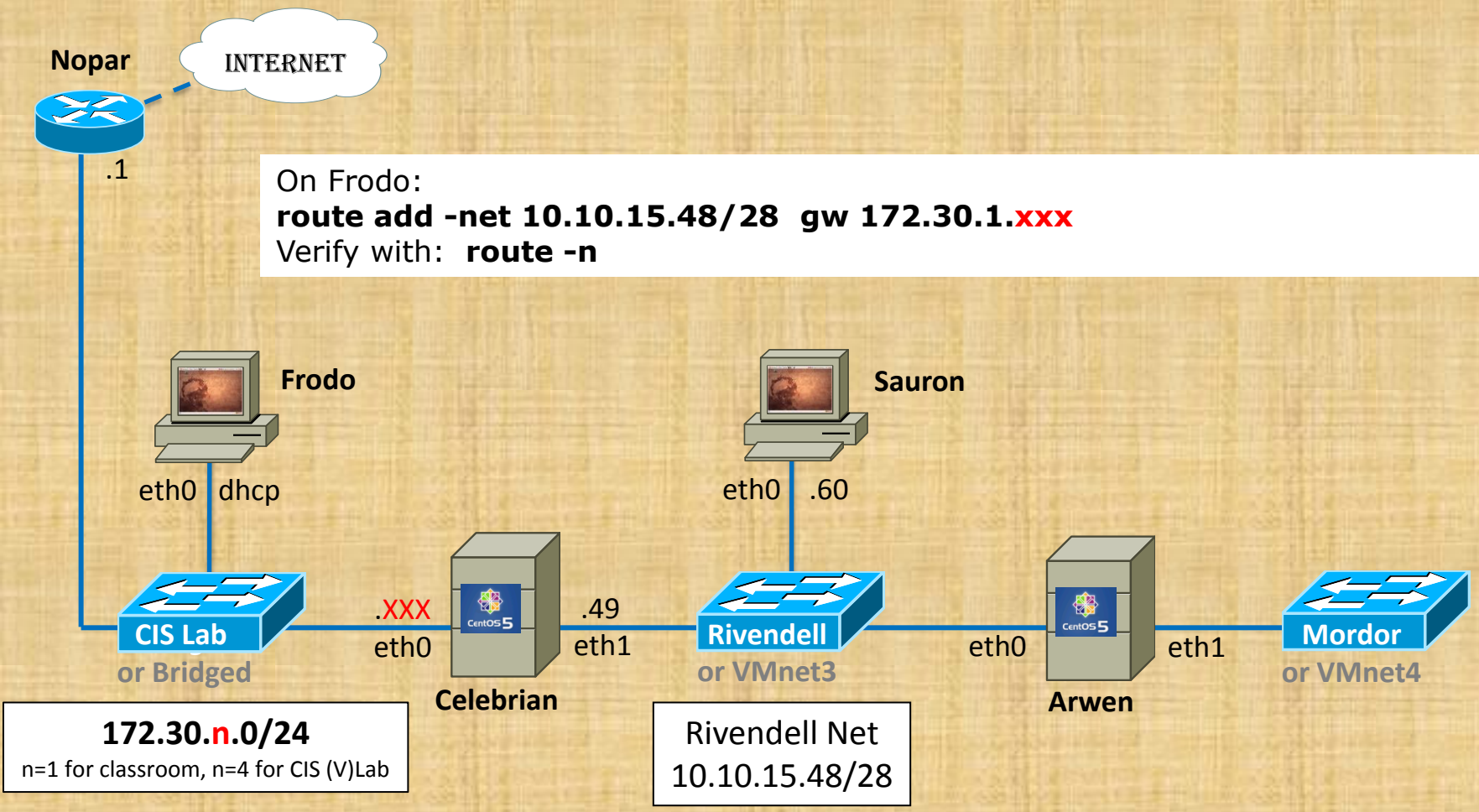
```
[root@elrond ~]# cat /etc/sysconfig/network-scripts/route-eth1
192.168.3.0/24 via 192.168.2.123
[root@elrond ~]# service network restart
```

Debian/Ubuntu
Permanent NIC Configuration

Static Routes

```
root@sun:~# cat /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
broadcast 172.30.4.255
network 172.30.4.0

gateway 172.30.4.1

up route add -net 192.168.2.0/24 gw 172.30.4.107
up route add -net 192.168.30.0/24 gw 172.30.4.107

root@sun:~#
```

Use up and down to configure what will happen when interface is brought up or down



Add a static route to Frodo's routing table for the 10.10.15.48/28 network

Routing Table

Routing Table Overview

- Directly connected networks are automatically added to the routing table.
- Static routes can be added using the route command.
- Default gateways can be added using the route command.
- Dynamic routing services that use routing protocols like RIP and OSPF can add additional routes to the routing table.

The Routing Table

Routing Table

-n shows IP addresses instead of names (faster)

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

Destination shows the networks that a route exists for.
The 0.0.0.0 network is used for the default route.

The Routing Table

Routing Table

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

*Gateway specifies the next-hop router or uses 0.0.0.0 for local **directly-connected** interfaces*

The Routing Table

Routing Table

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

Genmask is the mask applied to incoming destination IP addresses to determine if a route exists. These are sorted by longest (best match) to shortest prefix.

The Routing Table

Routing Table

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

Note the genmask of 0.0.0.0 is used for the default route. Applying this mask to any address will always result in a match.

The Routing Table

Routing Table

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

Possible flags include:
- *U (route is up)*
- *H (target is a host)*
- *G (use gateway)*

The Routing Table

Routing Table

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

Metric:
The distance to the target (usually counted in hops). It is not used by recent kernels, but may be needed by routing daemons.

The Routing Table

Routing Table

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

Ref:
Number of references to this route.
(Not used in the Linux kernel.)

The Routing Table

Routing Table

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

Use: Count of lookups for the route when using the –C option (show cache based information)

The Routing Table

Routing Table

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

Iface: Interface to which packets for this route will be sent.

The Routing Table
Supernetting

Routing Table

```
root@frodo:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.3.0     172.30.1.125    255.255.255.0   UG    0      0        0 eth0
172.30.1.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.2.0     172.30.1.125    255.255.255.0   UG    0      0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 eth0
0.0.0.0         172.30.1.1      0.0.0.0         UG    100    0        0 eth0
root@frodo:~#
```

*Note: these two routes could be replaced with a single route for **192.168.0.0 /16**. This is super-netting (the reverse of sub-netting)*

The Routing Algorithm

(How the decision is made)

Routing Algorithm

The purpose of the Routing Algorithm is to get the packet to its destination network.

- Compute the network number of the destination IP address
- Does the destination network match that on a local interface?
  If so, send it out that interface
- Does the destination network match one or more listed in the routing table?
  If so, send it using the best match (largest genmask) route
- Is there a default route listed in the routing table?
  If so, use that gateway
  Otherwise, drop the packet - "network is unreachable"

The Routing Algorithm

Network Number

Compute the network number

The network number is obtained by applying the genmask to the incoming IP destination address.

Example: 192.168.3.200 with genmask 255.255.255.0 is 192.168.3.0

- By hand


```
110000 10101000 00000011 11001000  192.168.3.200
111111 11111111 11111111 00000000  255.255.255.0
110000 10101000 00000011 00000000  192.168.3.0
```

- With ipcalc

```
[root@elrond ~]# ipcalc -n 192.168.3.200 255.255.255.0
NETWORK=192.168.3.0
```

The Routing Algorithm

Network Number

Compute the network number

The network number is obtained by applying the genmask to the incoming IP destination address.

Example: 192.168.30.100 with genmask 255.255.240.0 is 192.168.16.0

- By hand


```
110000 10101000 00011110 01100100  192.168.30.100
111111 11111111 11110000 00000000  255.255.240.0
110000 10101000 00010000 00000000  192.168.16.0
```

- With ipcalc

```
[root@elrond ~]# ipcalc -n 192.168.30.100 255.255.240.0
NETWORK=192.168.16.0
[root@elrond ~]#
```

route command –n option

show route table with names

```
[root@elrond ~]# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      *               255.255.255.0   U     0      0        0 eth0
192.168.3.0     legolas         255.255.255.0   UG    0      0        0 eth1
192.168.2.0     *               255.255.255.0   U     0      0        0 eth1
169.254.0.0     *               255.255.0.0     U     0      0        0 eth1
default         nosmo           0.0.0.0         UG    0      0        0 eth0
```

show route table with IP addresses

```
[root@elrond ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.30.4.0      0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.3.0     192.168.2.123   255.255.255.0   UG    0      0        0 eth1
192.168.2.0     0.0.0.0         255.255.255.0   U     0      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 eth1
0.0.0.0         172.30.4.1      0.0.0.0         UG    0      0        0 eth0
[root@elrond ~]#
```

route command for viewing cache

show route table cache with names

```
[root@elrond ~]# route -C
Kernel IP routing cache
Source          Destination     Gateway         Flags Metric Ref    Use Iface
192.168.2.125   sauron          legolas                0      0        0 eth1
172.30.4.125    nosmo           nosmo                  0      0        0 eth0
172.30.4.125    nosmo           nosmo                  0      0        6 eth0
sauron          192.168.2.125   192.168.2.125   l      0      0        1 lo
frodo           172.30.4.125    172.30.4.125    il     0      0        1 lo
172.30.4.108    172.30.4.255    172.30.4.255    ibl    0      0        0 lo
172.30.4.103    172.30.4.125    172.30.4.125    il     0      0      105 lo
nosmo           172.30.4.125    172.30.4.125    il     0      0        5 lo
172.30.4.125    172.30.4.103    172.30.4.103           0      1        0 eth0
legolas         192.168.2.125   192.168.2.125   il     0      0        0 lo
172.30.4.125    frodo           frodo                  0      0        0 eth0
172.30.4.125    frodo           frodo                  0      0        1 eth0
172.30.4.10     172.30.4.255    172.30.4.255    ibl    0      0       10 lo
192.168.2.125   sauron          legolas                0      0        2 eth1
172.30.4.12     255.255.255.255 255.255.255.255 ibl    0      0        3 lo
172.30.4.10     172.30.4.255    172.30.4.255    ibl    0      0       10 lo
192.168.2.125   sauron          legolas                0      0        2 eth1
172.30.4.12     255.255.255.255 255.255.255.255 ibl    0      0        3 lo
[root@elrond ~]#
```

route command for viewing cache

show route table cache with IP addresses

```
[root@elrond ~]# route -Cn
Kernel IP routing cache
Source          Destination     Gateway         Flags Metric Ref    Use Iface
192.168.2.125   192.168.3.200   192.168.2.123           0      0        0 eth1
172.30.4.125    172.30.4.1      172.30.4.1              0      0        0 eth0
172.30.4.125    172.30.4.1      172.30.4.1              0      0        6 eth0
192.168.3.200   192.168.2.125   192.168.2.125   l       0      0        1 lo
172.30.4.150    172.30.4.125    172.30.4.125    il      0      0        1 lo
172.30.4.108    172.30.4.255    172.30.4.255    ibl     0      0        0 lo
172.30.4.103    172.30.4.125    172.30.4.125    il      0      0      119 lo
172.30.4.125    207.62.187.53   172.30.4.1              0      0        7 eth0
172.30.4.1      172.30.4.125    172.30.4.125    il      0      0        5 lo
172.30.4.106    172.30.4.255    172.30.4.255    ibl     0      0        0 lo
172.30.4.110    172.30.4.255    172.30.4.255    ibl     0      0        0 lo
207.62.187.53   172.30.4.125    172.30.4.125    l       0      0        7 lo
172.30.4.125    172.30.4.103    172.30.4.103            0      1        0 eth0
192.168.2.123   192.168.2.125   192.168.2.125   il      0      0        0 lo
172.30.4.125    172.30.4.150    172.30.4.150            0      0        0 eth0
172.30.4.125    207.62.187.53   172.30.4.1              0      0        7 eth0
172.30.4.125    172.30.4.150    172.30.4.150            0      0        1 eth0
172.30.4.10     172.30.4.255    172.30.4.255    ibl     0      0       14 lo
192.168.2.125   192.168.3.200   192.168.2.123           0      0        2 eth1
172.30.4.12     255.255.255.255 255.255.255.255 ibl     0      0        5 lo
[root@elrond ~]#
```

route command
flushing the cache

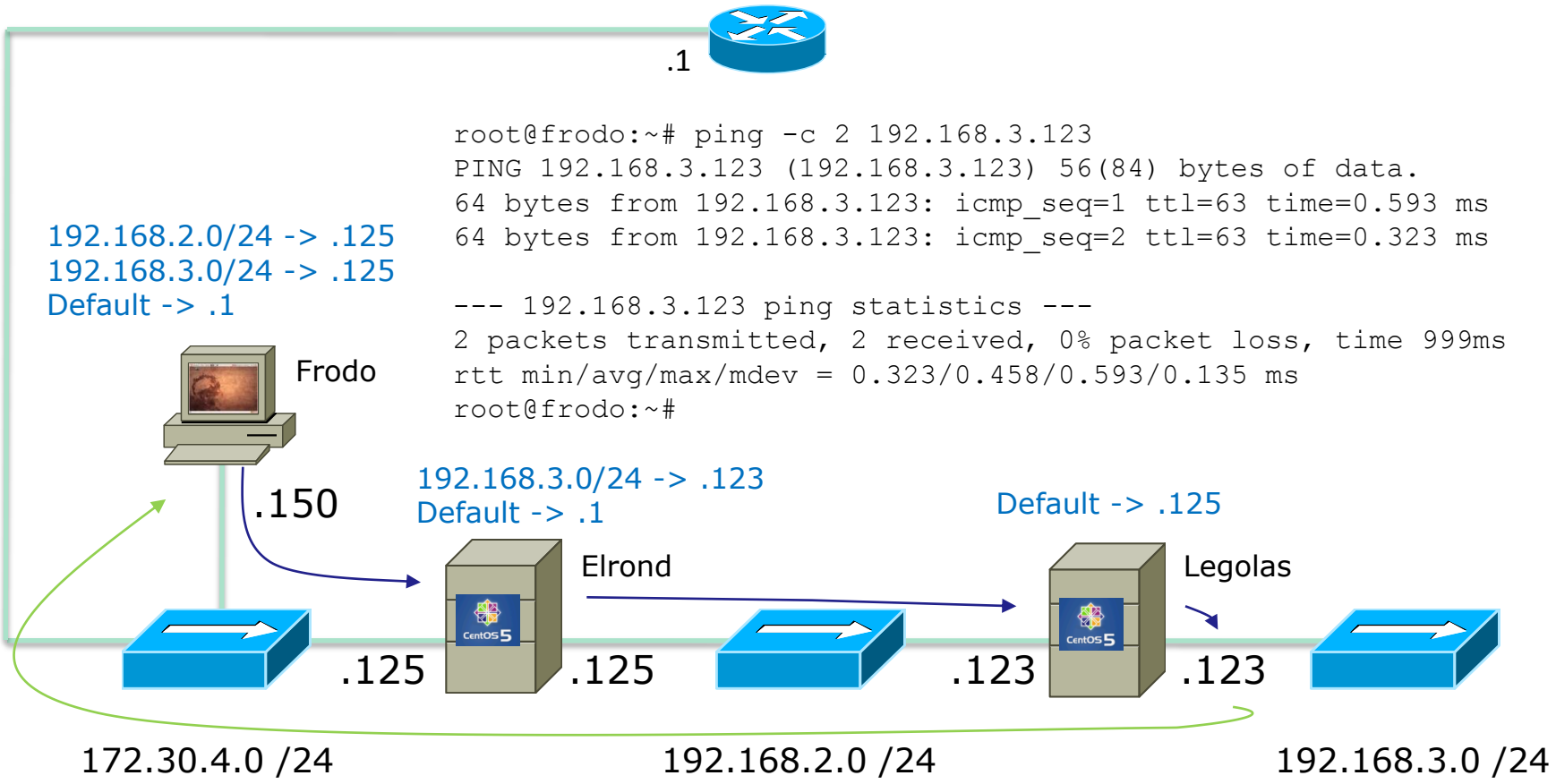
Flush the route cache

```
[root@elrond ~]# ip route flush cache
[root@elrond ~]# route -C
Kernel IP routing cache
Source          Destination     Gateway         Flags Metric Ref    Use Iface
172.30.4.103    172.30.4.125    172.30.4.125    il    0      0        3 lo
172.30.4.125    172.30.4.103    172.30.4.103          0      1        0 eth0
buttercup.cabri 172.30.4.125    172.30.4.125    l     0      0        1 lo
172.30.4.103    172.30.4.125    172.30.4.125    il    0      0        4 lo
172.30.4.125    172.30.4.103    172.30.4.103          0      1        0 eth0
[root@elrond ~]#
```

*Note: Use **route –CF** on Red Hat 9*

ICMP Redirect

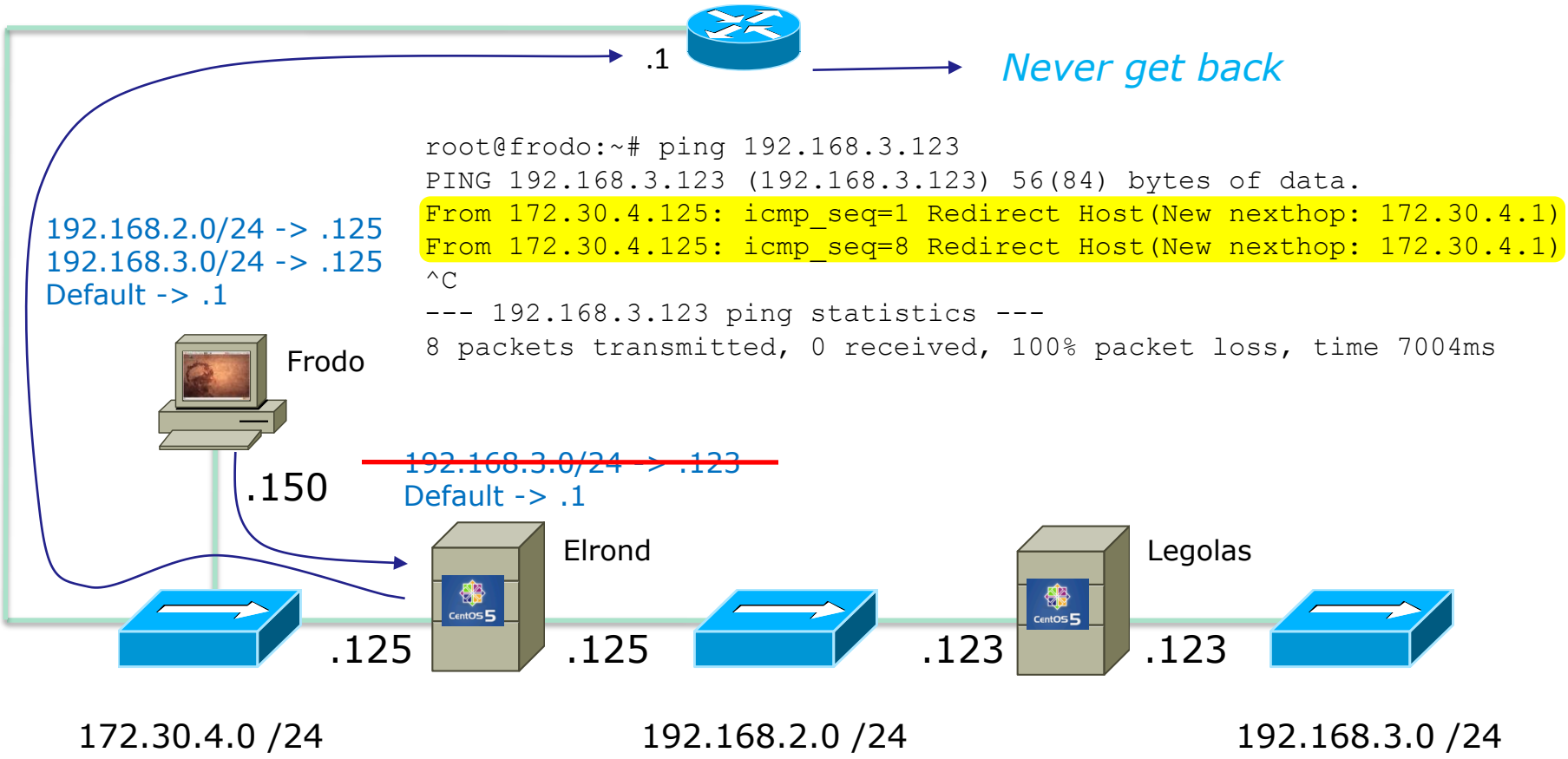
Routers will updates each others caches when they discover an inefficient route



When doing Lab 3, what happens if we left off the static route on Elrond?

ICMP Redirect

Routers will updates each others caches when they discover an inefficient route



Elrond tells Frodo there is a shorter route

Exercise

1. Explore the routing table and cache on Celebrian
   - route
   - route –n
   - route –C
   - route -Cn

2. Flush the route cache
   - ip route flush cache
   - route -Cn

3. Identify the directly connected, static and default routes.

Trouble shooting

ICMP Errors

Host Unreachable

```
root@frodo:~# ping 192.168.2.128
PING 192.168.2.128 (192.168.2.128) 56(84) bytes of data.
From 172.30.4.125 icmp_seq=3 Destination Host Unreachable
From 172.30.4.125 icmp_seq=4 Destination Host Unreachable
From 172.30.4.125 icmp_seq=5 Destination Host Unreachable
^C
--- 192.168.2.128 ping statistics ---
9 packets transmitted, 0 received, +3 errors, 100% packet loss, time 8019ms
, pipe 3
root@frodo:~#
```

When the packet arrives at the destination network there is no active host to receive the packet. The host is offline, does not exist, or is not cabled to the network. The ARP request for this host's MAC address is failing.

ICMP Errors

TTL exceeded

```
root@frodo:~# ping 192.168.5.200
PING 192.168.5.200 (192.168.5.200) 56(84) bytes of data.
From 192.168.2.123 icmp_seq=1 Time to live exceeded
From 192.168.2.123 icmp_seq=2 Time to live exceeded
From 192.168.2.123 icmp_seq=3 Time to live exceeded
From 192.168.2.123 icmp_seq=4 Time to live exceeded
From 192.168.2.123 icmp_seq=5 Time to live exceeded
From 192.168.2.123 icmp_seq=6 Time to live exceeded
^C
--- 192.168.5.200 ping statistics ---
6 packets transmitted, 0 received, +6 errors, 100% packet loss, time 5030ms

root@frodo:~#
```

One router is forwarding the packet to the next-hop router. The next-hop router has no specific route for this packet but does have a default route back to the previous router! Loops back and forth the until TTL count is 0 and then the packet is dropped.

ICMP Errors

Network Unreachable

```
[root@legolas ~]# ping 172.30.4.1
connect: Network is unreachable
[root@legolas ~]#
```

There is no matching route in the route table.
To fix, add a default gateway or a static route

ICMP Errors

Nothing

```
[root@legolas ~]# ping 10.240.1.2
PING 10.240.1.2 (10.240.1.2) 56(84) bytes of data.

--- 10.240.1.2 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7011ms

[root@legolas ~]#
```

No response! The ping is being sent out on a route where there is no route back!

Lab

Configuring a network (What you will be doing in Lab 3)

Overall steps:

- Cable systems
- Assign valid IP addresses to all hosts and routers
- Enable IP forwarding on all routers
- Open firewalls
- Configure the routing tables of all hosts and routers

*Tip: Use **ip route flush cache** when correcting any entries in the routing table*

Wrap

New commands, tools and services:

- service
- chkconfig
- dhclient
- ip
- route

New Files and Directories (Red Hat):

- /etc/sysconfig/network
- /etc/sysconfig/network-scripts/ifcfg-eth*
- /etc/sysconfig/network-scripts/route-eth*
- /proc/sys/net/ipv4/ip_forward
- /etc/sysctl.conf
- service network restart

New Files and Directories (Ubuntu):

- /etc/hostname
- /etc/network/interfaces
- /etc/init.d/networking restart

VMware:

Next Class

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

No Quiz for next class:

Test next week on lessons 1 through 3

- Open book, open notes, open VMs, during last hour of class
- 15 questions (2 points each)
- Practice test available
- Doing Lab 4 early would be good practice for test

Students may work together and use the forum to work out the answers on the practice test.

*The actual test will be **almost identical** to the practice test.*

For the actual test, students must work individually and neither ask nor give assistance to others.

Backup