Lesson Module Status

- Slides
- Whiteboard with 1st minute quiz

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

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

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

CIS 192 - Lesson 6

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

For tonight everyone join CCC Confer and power up Frodo, Elrond and Arwen

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

Course history and credits

Jim Griffin

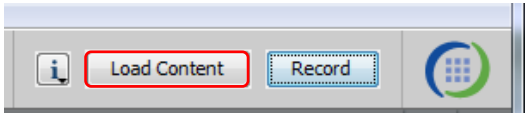
- Jim created the original version of this course
- Jim's site: http://cabrillo.edu/~jgriffin/

Rick Graziani

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

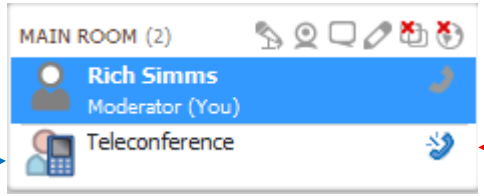
CCC Confer

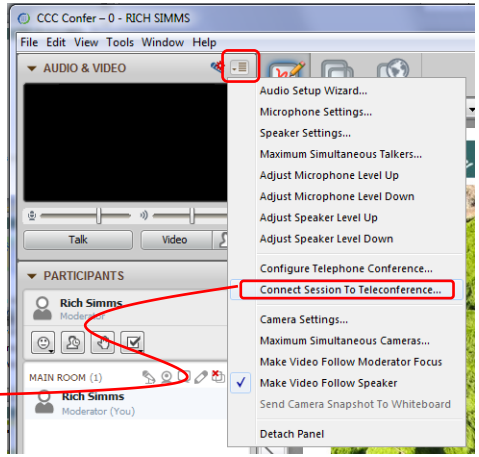
[ ] Preload White Board with *cis*lesson??*-WB*



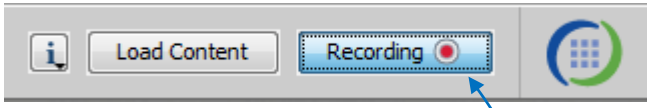
[ ] Connect session to Teleconference

Session now connected to teleconference





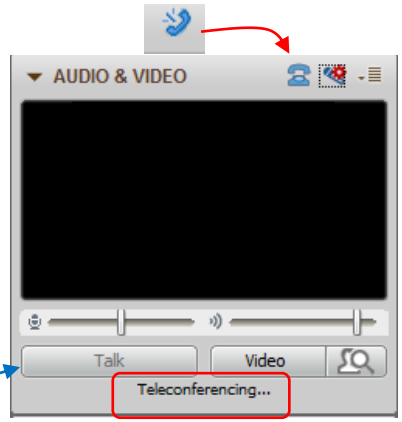
[ ] Is recording on?



Red dot means recording

[ ] Use teleconferencing, not mic

Should be greyed out

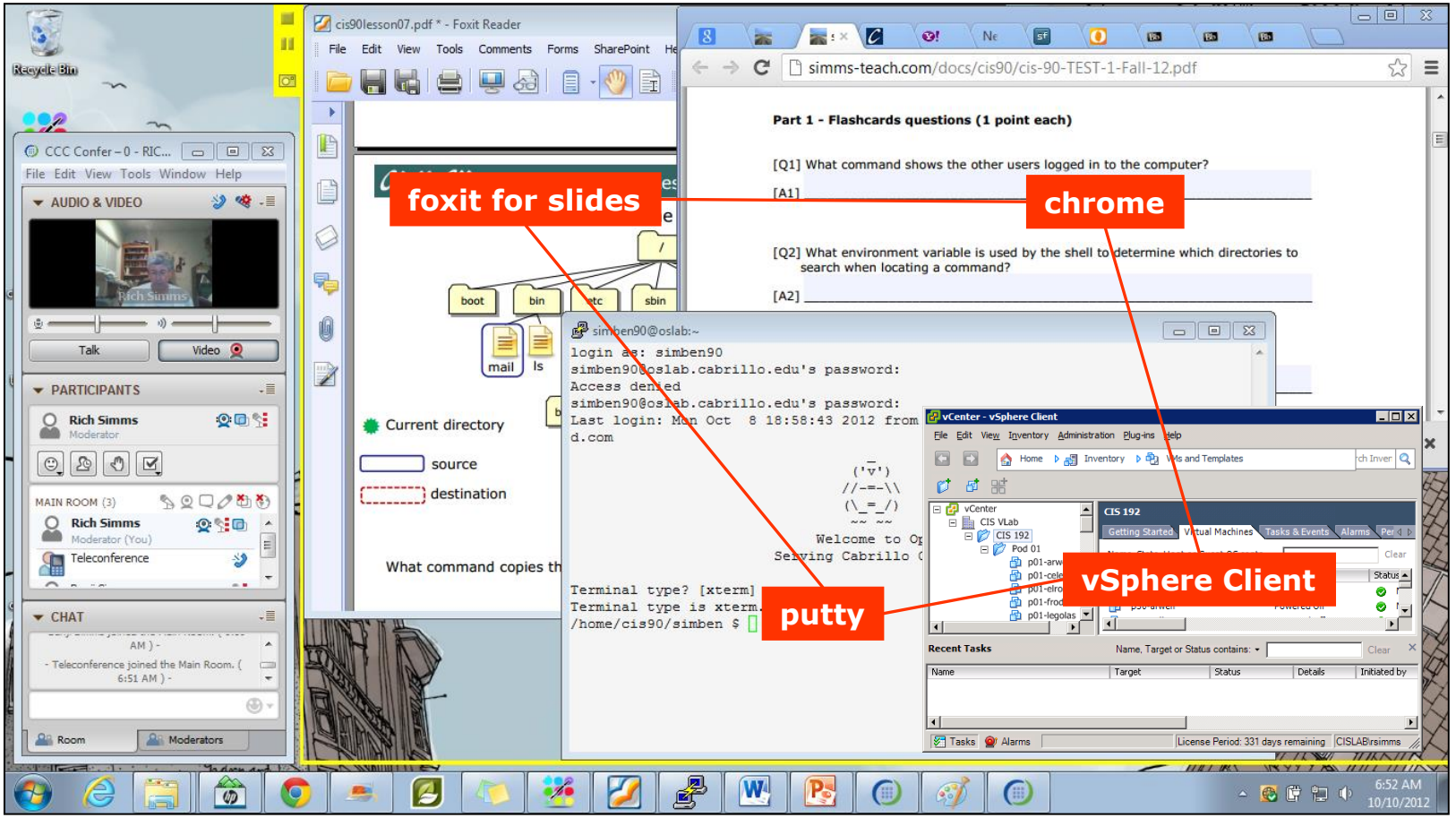


CCC Confer

Don't Forget

[ ] **Video (webcam) optional**

[ ] **layout and share apps**

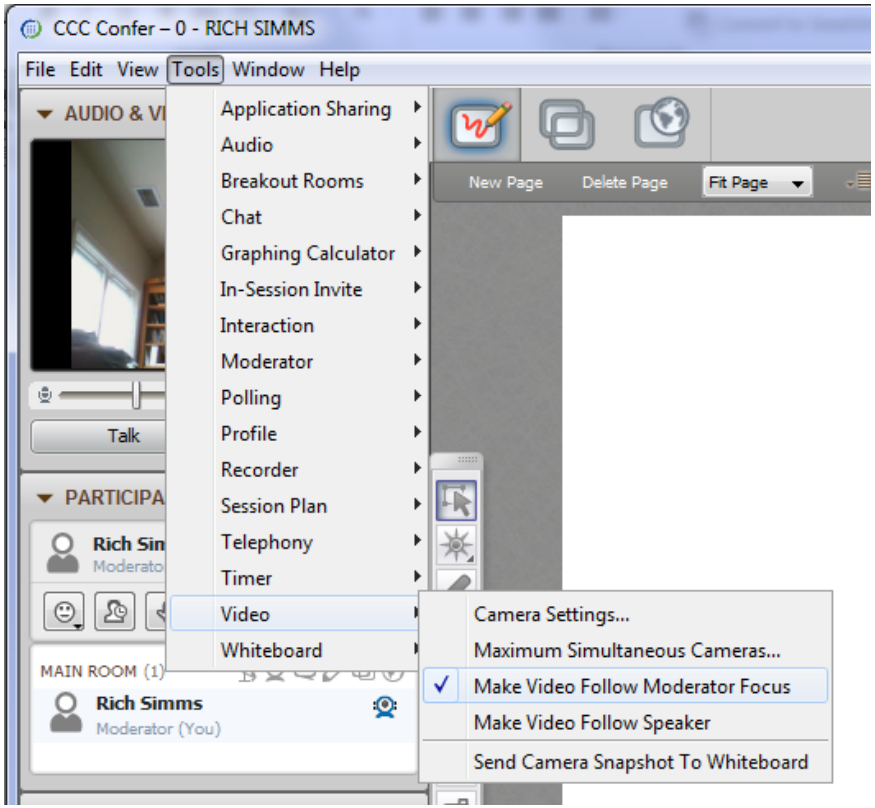


CCC Confer

[ ] Video (webcam) optional

[ ] Follow moderator

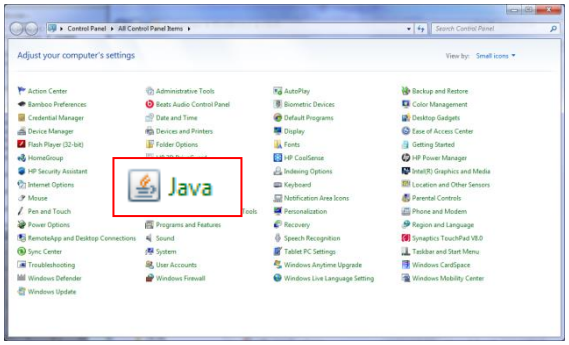
[ ] Double-click on postages stamps



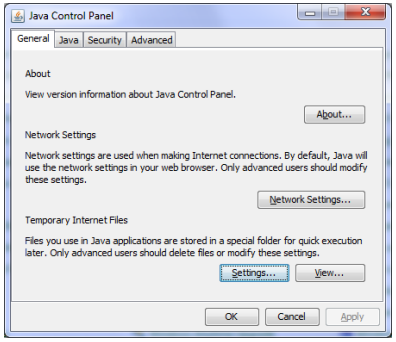
Universal Fix for CCC Confer:

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

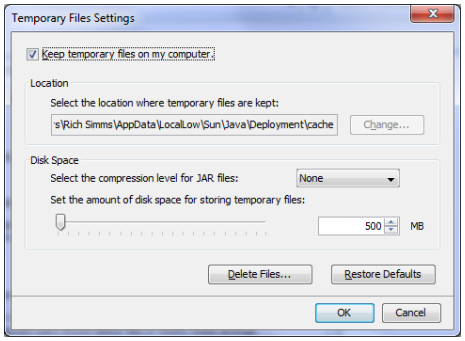
Control Panel (small icons)



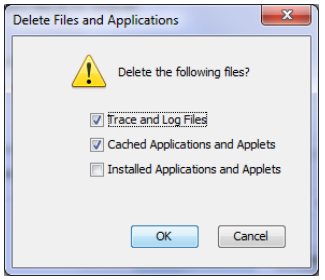
General Tab > Settings…



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

Firewalls and NAT

Objectives	Agenda
• Configure a network service with security restrictions for its use using either TCP Wrappers or a superdaemon. • Use iptables to build a permissive firewall by selectively filtering packets based on protocol type. • Create a secure tunnel between two hosts that allows port forwarding into a private network. • Use Network Address Translation (NAT) to allow hosts on a private network to access the Internet.	• Quiz • Questions • Housekeeping • Permanent configuration settings practice • SSH • SSH port forwarding • Netfilter • Example firewall and NAT • Lab 5 Prep • Wrap

Questions

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

- Lab 4 due 11:59PM tonight

- Send your Lab 4 map/crib sheets to risimms@cabrillo.edu (jpg, png or pdf please)

Perkins/VTEA Survey

Carl D. Perkins Career and Technical Education Act

POSTREPLY | Search this topic... | Search

Carl D. Perkins Career and Technical Education Act

by **Rich Simms** » Fri Mar 01, 2013 8:08 pm

The Carl D. Perkins Vocational and Technical Education Act was originally authorized by Congress in 1984. It was reauthorized in 1998 and again in 2006. This act provides federal funding for improving career technical education (CTE) within the United States in order to help the economy.

For Cabrillo College to receive a portion of this funding students in technical classes must fill out a survey. The more surveys completed the more funds the college will receive. The survey only needs to be completed once per term by each student.

This survey can be completed online using web advisor:

Log on to WEBADVISOR at https://wave.cabrillo.edu

Select "STUDENTS: Click Here" (navy blue bar)
- Under "Academic Profile" Click on "Student Update Form"
- Use drop down list under "Select the earliest term for which you are registered" and click on the current term.
- Select "SUBMIT"

Scroll down to the "Career Technical Information"
- Answer questions by clicking on the circle to the left of your "Yes" or "No" answers
- You can get details about a question by clicking on blue underlined phrase
- After answering all questions Select "SUBMIT"

Then "LOG OUT"

Thank you for taking a few minutes to help Cabrillo College CS/CIS programs!

- Rich

Deadline March 21

http://oslab.cabrillo.edu/forum/viewtopic.php?f=63&t=1883

This is an important source of funding for Cabrillo College.

*Send me an email that you completed this survey for **3 points extra credit!***

Help with labs

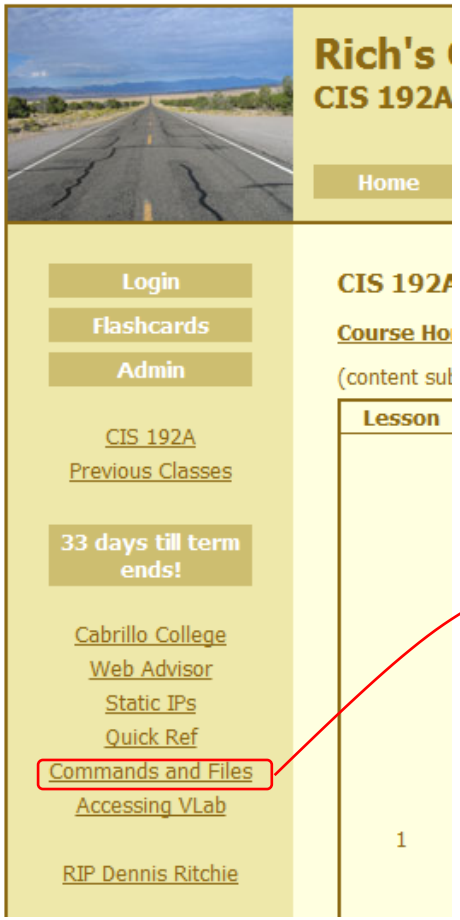
Like some help with labs?

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

Commands and Files
Quick Reference and Examples



Linux Network Commands & Files

Click on the link in the table below to see commands, configuration files and examples.

General Linux commands - root & shutdown General Linux commands - basic inventory Installing more commands	Network Testing
IP Addressing	
Interfaces Interfaces - DHCP client (temporary) Interfaces - Static IP (temporary) Interfaces - Red Hat family (permanent) Interfaces - Debian family (permanent)	
Name resolution	
ARP commands	
Linux hardware and driver commands	

Network configuration - Debian family (permanent)

~~Edit /etc/network/interfaces~~

Use this "deprecated" script to restart network services:

/etc/init.d/networking restart

It seems this script in now deprecated and each interface must be manually shut down then brought back up!

See: http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=565187

Edit this file to permanently configure networking on Debian and Ubuntu systems.

EXAMPLE - DHCP:

/etc/network/interfaces
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

EXAMPLE - static IP:

/etc/network/interfaces
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 172.30.4.222
netmask 255.255.255.0

gateway 172.30.4.1

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											20				30	30	23									3		
Bilbo	Grade	3	3	3											20				29	28	29									11		
Denethor	P/NP	3	3	3											16				8	13	26									6		
Dwalin	Grade		3	3											20					29	30											
Elrohir	Grade	3	3	3											20				30	30	30									33		
Elrond	Grade	3		3											20				30	30	30									12		
Faramir	Grade	3	3	3											20				30	30	28									16		
Frodo	Grade	3	3	3											20				29	30	30									8		
Gwaihir	Grade		3	3											20				30	27	30											
Ioreth	Grade	3	3	3											0				30	30	30											
Legolas	Grade	3		3											20				30	29	29											
Nazgul	Grade	3	3	2											20				30	30	30											
Pippin	Grade	3	3	3											20				30	30	30											
Samwise	Grade	3	3	2											20				30	30	12											
Saruman	Grade	3	3												20				30	30	30											
Strider	Grade	3	3	2											20				29	30												
Theoden	Grade	3	3	3											20				30	29	27											
Treebeard	Grade																															

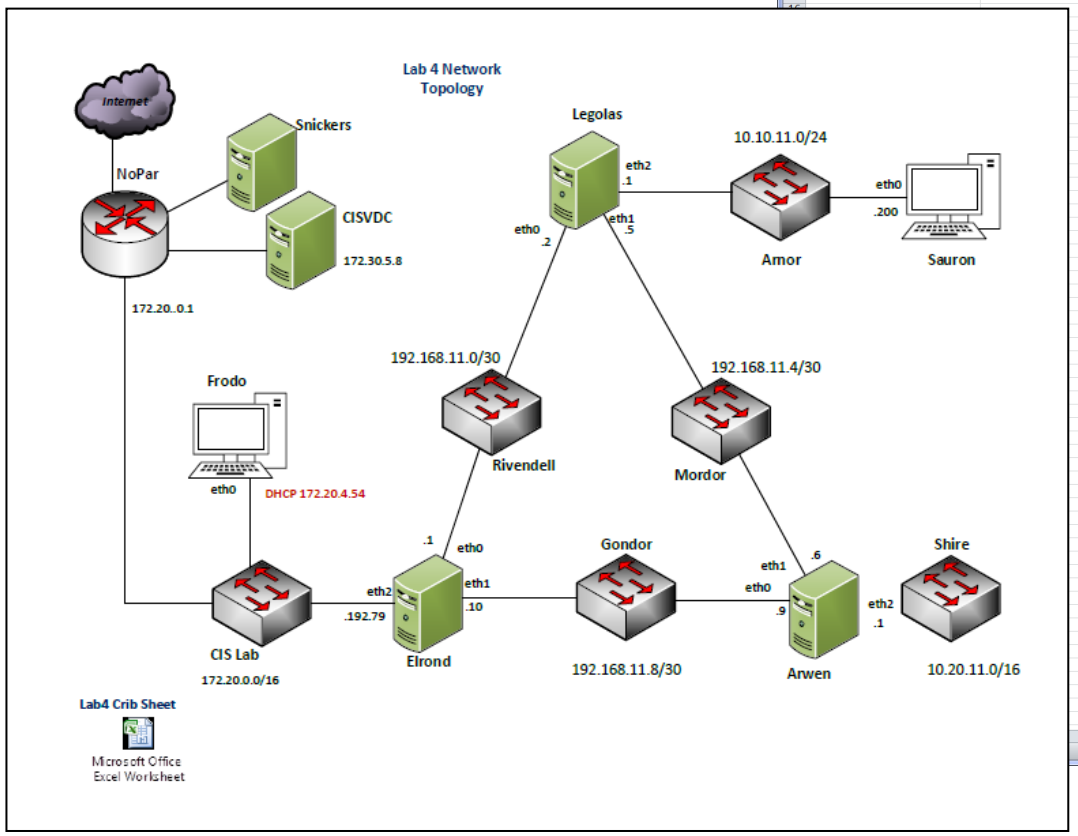
Please check your:
- Grading Choice
- Quiz points
- Lab points
- Extra Credit points

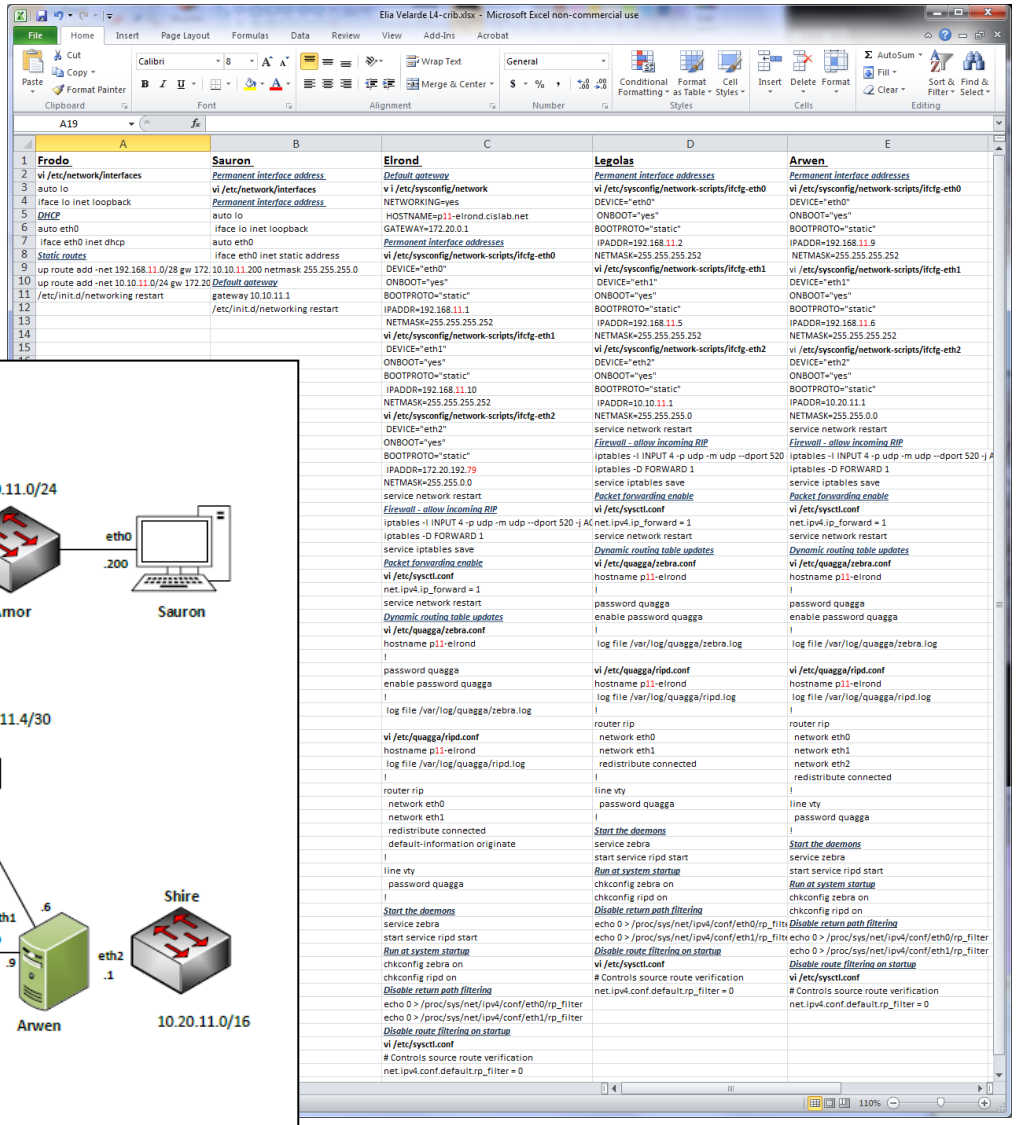
Don't know you secret LOR code name?

… then email me your student survey to get it!

Anonymous Lab 4 Map/Crib Gallery

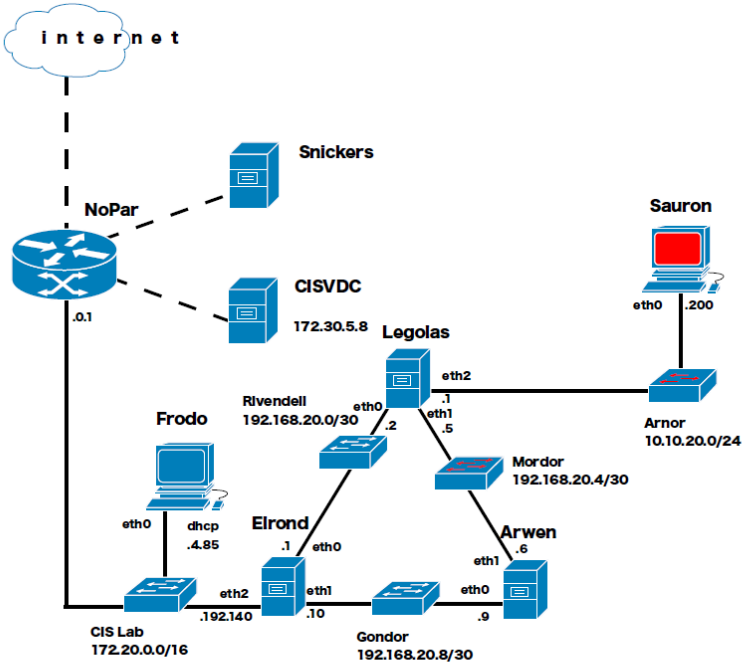
CIS 192 - Lesson 6





Frodo	Sauron	Elrond	Legolas	Arwen
vi /etc/network/interfaces	Permanent interface address	Default gateway	Permanent interface addresses	Permanent interface addresses
auto lo	vi /etc/network/interfaces	v i /etc/sysconfig/network	vi /etc/sysconfig/network-scripts/ifcfg-eth0	vi /etc/sysconfig/network-scripts/ifcfg-eth0
iface lo inet loopback	Permanent interface address	NETWORKING=yes	DEVICE="eth0"	DEVICE="eth0"
DHCP	auto lo	HOSTNAME=p11-elrond.cislab.net	ONBOOT="yes"	ONBOOT="yes"
auto eth0	iface lo inet loopback	GATEWAY=172.20.0.1	BOOTPROTO="static"	BOOTPROTO="static"
iface eth0 inet dhcp	auto eth0	Permanent interface addresses	IPADDR=192.168.11.2	IPADDR=192.168.11.9
Static routes	iface eth0 inet static address	vi /etc/sysconfig/network-scripts/ifcfg-eth0	NETMASK=255.255.255.252	NETMASK=255.255.255.252
up route add -net 192.168.11.0/28 gw 172.	10.10.11.200 netmask 255.255.255.0	DEVICE="eth0"	vi /etc/sysconfig/network-scripts/ifcfg-eth1	vi /etc/sysconfig/network-scripts/ifcfg-eth1
up route add -net 10.10.11.0/24 gw 172.20	Default gateway	ONBOOT="yes"	DEVICE="eth1"	DEVICE="eth1"
/etc/init.d/networking restart	gateway 10.10.11.1	BOOTPROTO="static"	ONBOOT="yes"	ONBOOT="yes"
	/etc/init.d/networking restart	IPADDR=192.168.11.1	BOOTPROTO="static"	BOOTPROTO="static"
		NETMASK=255.255.255.252	IPADDR=192.168.11.5	IPADDR=192.168.11.6
		vi /etc/sysconfig/network-scripts/ifcfg-eth1	NETMASK=255.255.255.252	NETMASK=255.255.255.252
		DEVICE="eth1"	vi /etc/sysconfig/network-scripts/ifcfg-eth2	vi /etc/sysconfig/network-scripts/ifcfg-eth2
		ONBOOT="yes"	DEVICE="eth2"	DEVICE="eth2"
		BOOTPROTO="static"	ONBOOT="yes"	ONBOOT="yes"
		IPADDR=192.168.11.10	BOOTPROTO="static"	BOOTPROTO="static"
		NETMASK=255.255.255.252	IPADDR=10.10.11.1	IPADDR=10.20.11.1
		vi /etc/sysconfig/network-scripts/ifcfg-eth2	NETMASK=255.255.255.0	NETMASK=255.255.0.0
		DEVICE="eth2"	service network restart	service network restart
		ONBOOT="yes"	Firewall - allow incoming RIP	Firewall - allow incoming RIP
		BOOTPROTO="static"	iptables -I INPUT 4 -p udp -m udp --dport 520	iptables -I INPUT 4 -p udp -m udp --dport 520 -j A
		IPADDR=172.20.192.79	iptables -D FORWARD 1	iptables -D FORWARD 1
		NETMASK=255.255.0.0	service iptables save	service iptables save
		service network restart	Packet forwarding enable	Packet forwarding enable
		Firewall - allow incoming RIP	vi /etc/sysctl.conf	vi /etc/sysctl.conf
		iptables -I INPUT 4 -p udp -m udp --dport 520 -j AC	net.ipv4.ip_forward = 1	net.ipv4.ip_forward = 1
		iptables -D FORWARD 1	service network restart	service network restart
		service iptables save	Dynamic routing table updates	Dynamic routing table updates
		Packet forwarding enable	vi /etc/quagga/zebra.conf	vi /etc/quagga/zebra.conf
		vi /etc/sysctl.conf	hostname p11-elrond	hostname p11-elrond
		net.ipv4.ip_forward = 1	!	!
		service network restart	password quagga	password quagga
		Dynamic routing table updates	enable password quagga	enable password quagga
		vi /etc/quagga/zebra.conf	!	!
		hostname p11-elrond	log file /var/log/quagga/zebra.log	log file /var/log/quagga/zebra.log
		!	!	!
		password quagga	vi /etc/quagga/ripd.conf	vi /etc/quagga/ripd.conf
		enable password quagga	hostname p11-elrond	hostname p11-elrond
		!	log file /var/log/quagga/ripd.log	log file /var/log/quagga/ripd.log
		log file /var/log/quagga/zebra.log	!	!
			router rip	router rip
		vi /etc/quagga/ripd.conf	network eth0	network eth0
		hostname p11-elrond	network eth1	network eth1
		log file /var/log/quagga/ripd.log	redistribute connected	network eth2
		!	!	redistribute connected
		router rip	line vty	!
		network eth0	password quagga	line vty
		network eth1	!	password quagga
		redistribute connected	Start the daemons	!
		default-information originate	service zebra	Start the daemons
		!	start service ripd start	service zebra
		line vty	Run at system startup	start service ripd start
		password quagga	chkconfig zebra on	Run at system startup
		!	chkconfig ripd on	chkconfig zebra on
		Start the daemons	Disable return path filtering	chkconfig ripd on
		service zebra	echo 0 > /proc/sys/net/ipv4/conf/eth0/rp_filte	Disable return path filtering
		start service ripd start	echo 0 > /proc/sys/net/ipv4/conf/eth1/rp_filte	echo 0 > /proc/sys/net/ipv4/conf/eth0/rp_filter
		Run at system startup	Disable route filtering on startup	echo 0 > /proc/sys/net/ipv4/conf/eth1/rp_filter
		chkconfig zebra on	vi /etc/sysctl.conf	Disable route filtering on startup
		chkconfig ripd on	# Controls source route verification	vi /etc/sysctl.conf
		Disable return path filtering	net.ipv4.conf.default.rp_filter = 0	# Controls source route verification
		echo 0 > /proc/sys/net/ipv4/conf/eth0/rp_filter		net.ipv4.conf.default.rp_filter = 0
		echo 0 > /proc/sys/net/ipv4/conf/eth1/rp_filter		
		Disable route filtering on startup		
		vi /etc/sysctl.conf		
		# Controls source route verification		
		net.ipv4.conf.default.rp_filter = 0		

rip sheet.



all routes 75¢

CRIB SHEET

Arwen

```
iptables -I INPUT 4 -p udp -m udp --dport 520 -j
ACCEPT
iptables -D FORWARD 1
service iptables save

/etc/sysctl.conf
net.ipv4.ip_forward = 1
```

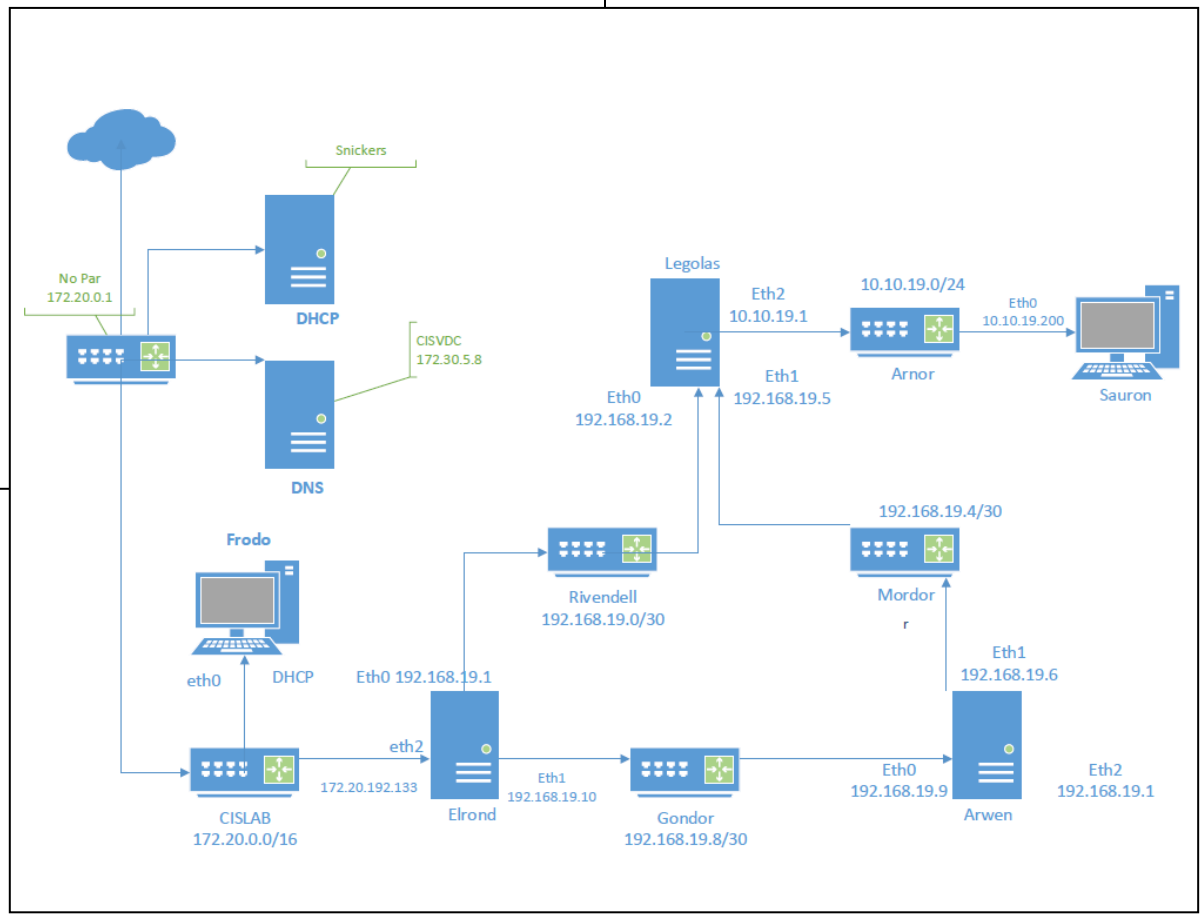
Sauron

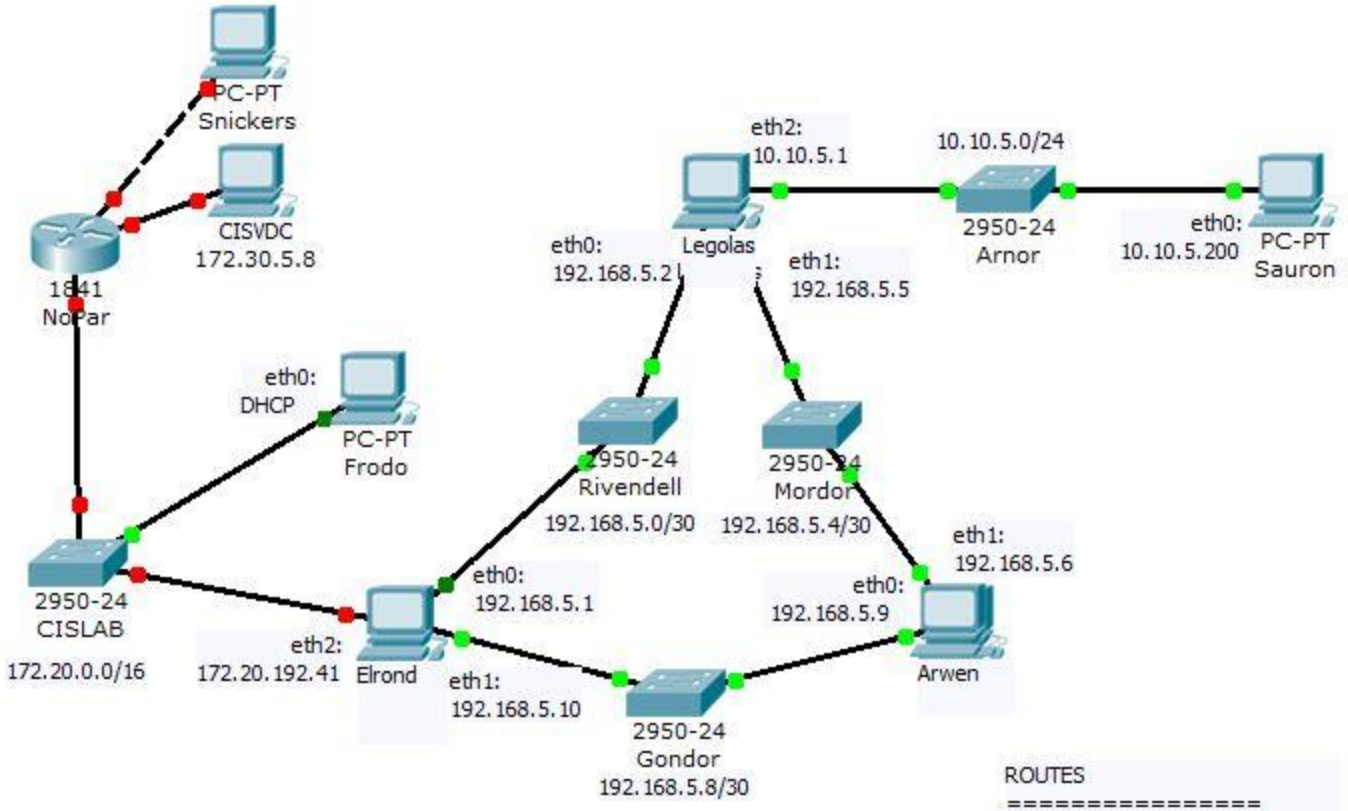
Elrond

```
iptables -I INPUT 4 -p udp -m udp --dport 520 -j
ACCEPT
iptables -D FORWARD 1
service iptables save

/etc/sysctl.conf
net.ipv4.ip_forward = 1
```

Legolas

```
iptables -I INPUT 4 -p udp -m udp --dport 520 -j
ACCEPT
iptables -D FORWARD 1
service iptables save

/etc/sysctl.conf
net.ipv4.ip_forward = 1
```





ROUTES
==========
ELROND:
route add default gw 172.20.0.1

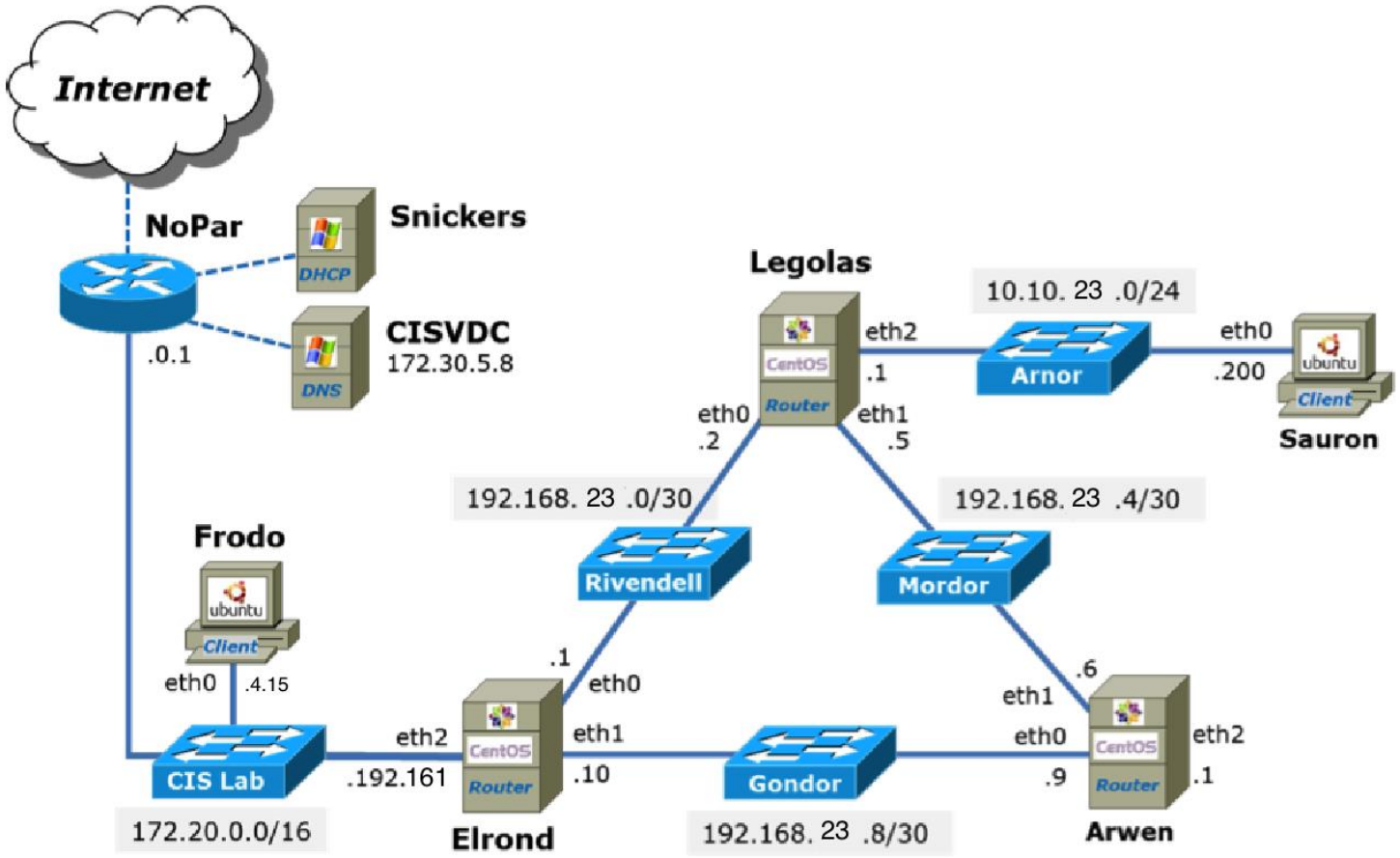
SAURON:
route add default gw 10.10.5.1

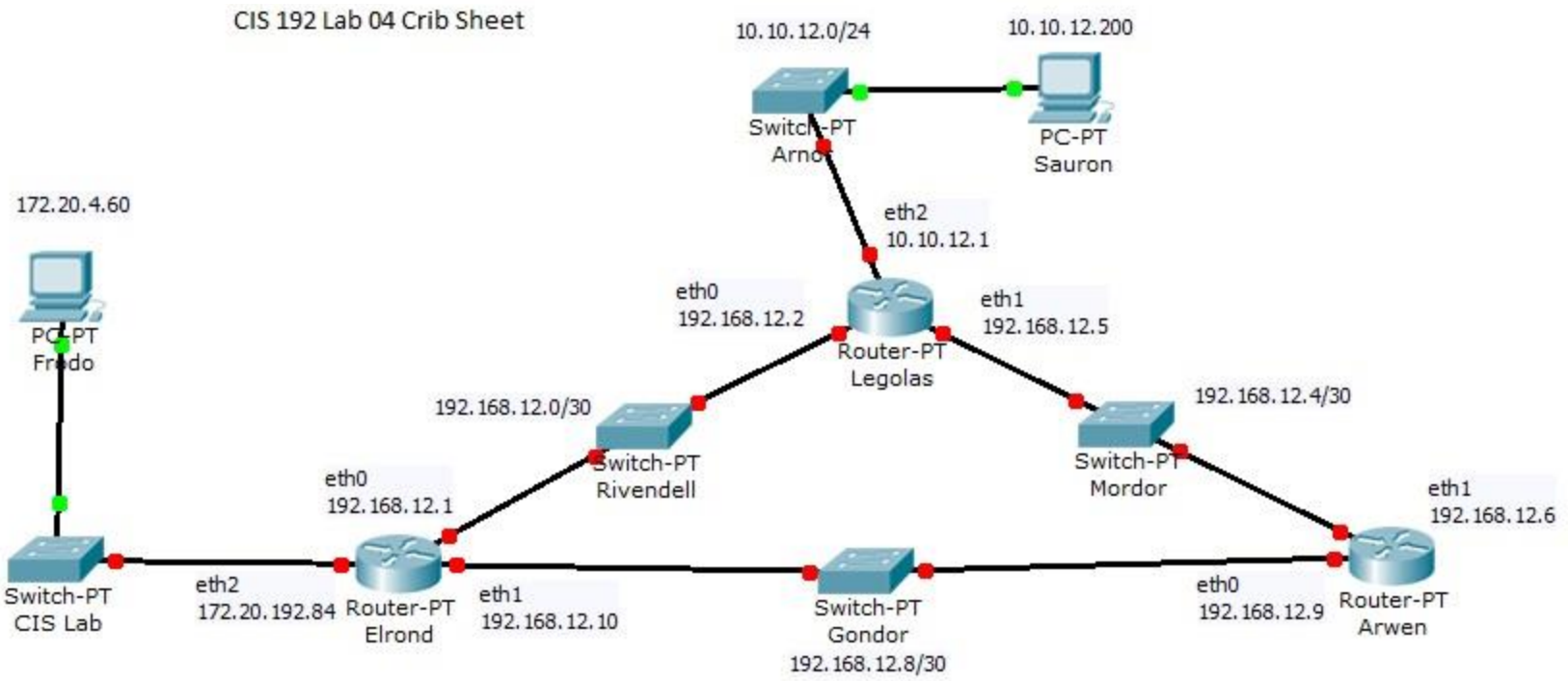
ELROND
---
eth1:
ifconfig eth1 192.168.5.1/30
eth2:
ifconfig eth2 172.20.192.41/16

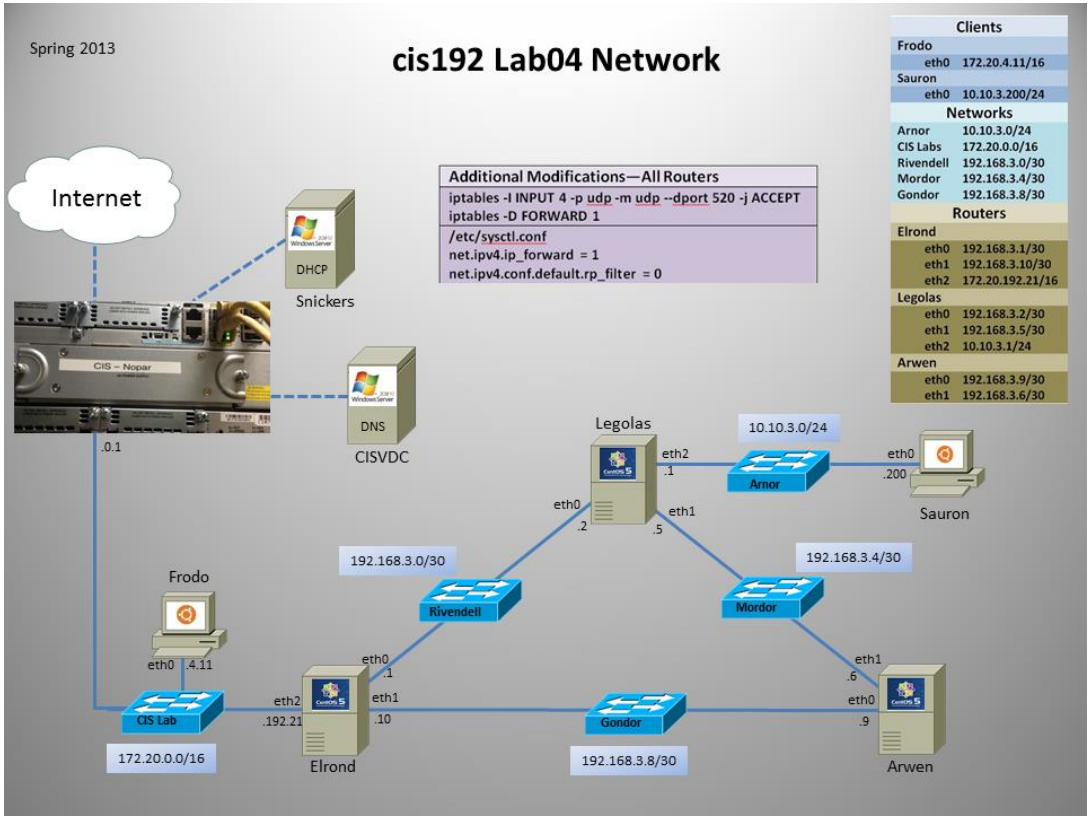
ARWEN:
---
eth0:
ifconfig eth0 192.168.5.9/30
eth1:
ifconfig eth1 192.168.5.6/30

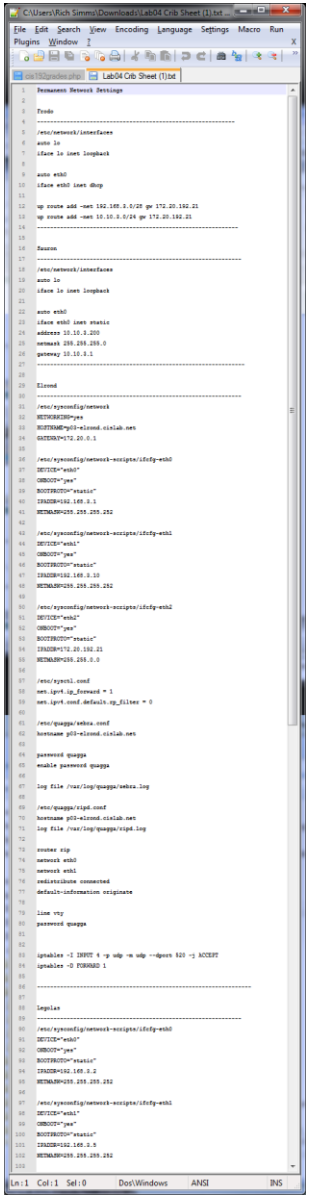
SAURON
---
eth0:
ifconfig eth0 10.10.5.200/24

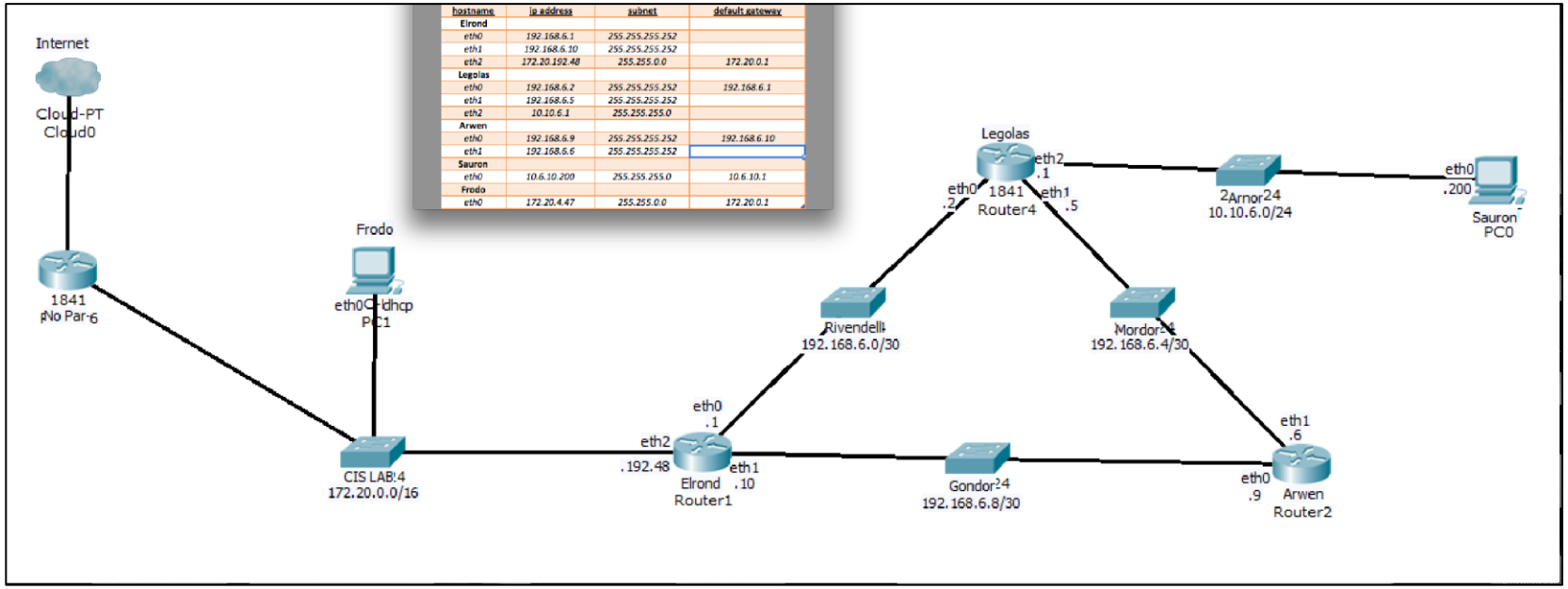
LEGOLAS
---
eth0:
ifconfig eth0 192.168.5.2/30
eth1:
ifconfig eth1 192.168.5.5/30
eth2:
ifconfig eth2 10.10.5.1/24


Internet
NoPar
Snickers
DHCP
CISVDC
172.30.5.8
DNS
.0.1
Legolas
10.10. 23 .0/24
eth2
.1
CentOS
Router
eth0
.2
eth1
.5
Arnor
eth0
.200
ubuntu
Client
Sauron
192.168. 23 .0/30
192.168. 23 .4/30
Rivendell
Mordor
Frodo
ubuntu
Client
eth0
.4.15
.1
eth0
.6
eth1
eth0
eth2
CentOS
Router
eth2
.192.161
eth1
.10
CIS Lab
Gondor
.9
.1
172.20.0.0/16
Elrond
192.168. 23 .8/30
Arwen


CIS 192 Lab 04 Crib Sheet
10.10.12.0/24
10.10.12.200
Switch-PT
Arno
PC-PT
Sauron
172.20.4.60
eth2
10.10.12.1
eth0
192.168.12.2
eth1
192.168.12.5
PC-PT
Frodo
Router-PT
Legolas
192.168.12.0/30
192.168.12.4/30
Switch-PT
Rivendell
Switch-PT
Mordor
eth0
192.168.12.1
eth1
192.168.12.6
Switch-PT
CIS Lab
eth2
172.20.192.84
Router-PT
Elrond
eth1
192.168.12.10
Switch-PT
Gondor
192.168.12.8/30
eth0
192.168.12.9
Router-PT
Arwen


Spring 2013
cis192 Lab04 Network
Clients
Frodo
eth0 172.20.4.11/16
Sauron
eth0 10.10.3.200/24
Networks
Arnor 10.10.3.0/24
CIS Labs 172.20.0.0/16
Rivendell 192.168.3.0/30
Mordor 192.168.3.4/30
Gondor 192.168.3.8/30
Routers
Elrond
eth0 192.168.3.1/30
eth1 192.168.3.10/30
eth2 172.20.192.21/16
Legolas
eth0 192.168.3.2/30
eth1 192.168.3.5/30
eth2 10.10.3.1/24
Arwen
eth0 192.168.3.9/30
eth1 192.168.3.6/30
Additional Modifications—All Routers
iptables -I INPUT 4 -p udp -m udp --dport 520 -j ACCEPT
iptables -D FORWARD 1
/etc/sysctl.conf
net.ipv4.ip_forward = 1
net.ipv4.conf.default.rp_filter = 0
Internet
DHCP
Snickers
CIS – Nopar
.0.1
DNS
CISVDC
Legolas
eth2
.1
10.10.3.0/24
Arnor
eth0
.200
Sauron
eth0
.2
eth1
.5
192.168.3.0/30
Rivendell
192.168.3.4/30
Mordor
Frodo
eth0 .4.11
CIS Lab
172.20.0.0/16
eth0
.1
eth2
.192.21
eth1
.10
Elrond
Gondor
192.168.3.8/30
eth1
.6
eth0
.9
Arwen





hostname	ip address	subnet	default gateway
Elrond			
eth0	192.168.6.1	255.255.255.252	
eth1	192.168.6.10	255.255.255.252	
eth2	172.20.192.48	255.255.0.0	172.20.0.1
Legolas			
eth0	192.168.6.2	255.255.255.252	192.168.6.1
eth1	192.168.6.5	255.255.255.252	
eth2	10.10.6.1	255.255.255.0	
Arwen			
eth0	192.168.6.9	255.255.255.252	192.168.6.10
eth1	192.168.6.6	255.255.255.252	
Sauron			
eth0	10.6.10.200	255.255.255.0	10.6.10.1
Frodo			
eth0	172.20.4.47	255.255.0.0	172.20.0.1

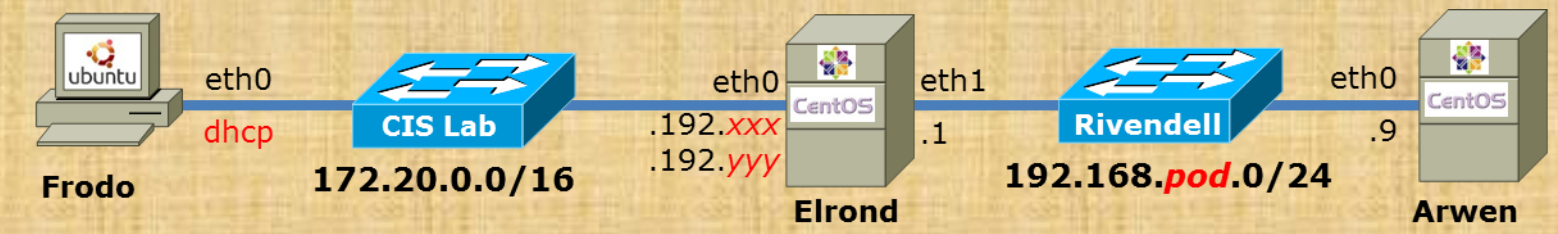
Permanent Network Settings Practice

Build Lesson 6 Network (prep for Lab 5)

1. Connect Arwen eth0 temporarily to the CIS Lab network, get an IP address using **dhclient -v eth0**

2. Install telnet and telnet-server using **yum install telnet telnet-server**

3. Release IP address with **dhclient -v -r eth0**

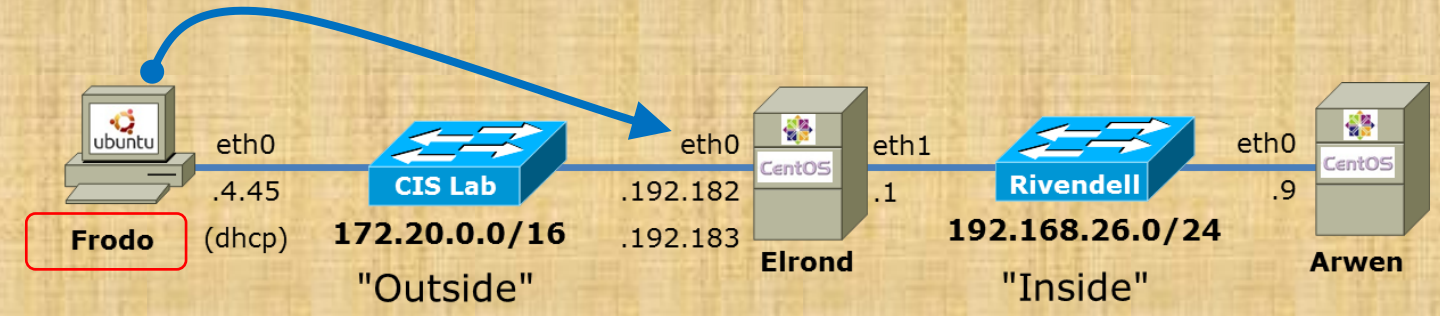


Client *Gateway* *Telnet Server*

Cable these systems in your pod

Telnet install group debug if needed

Static route to Rivendell via Elrond



cis192@p26-frodo:~$ **cat /etc/network/interfaces**
auto lo
iface lo inet loopback

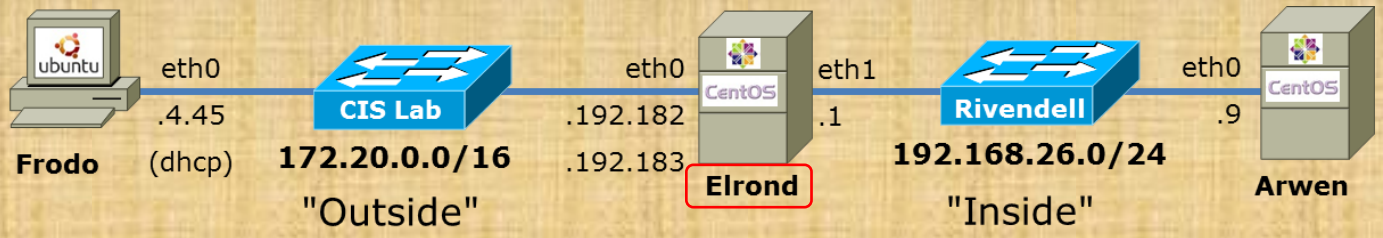
auto eth0
iface eth0 inet dhcp

up route add -net 192.168.26.0/24 gw 172.20.192.182

service network-manager stop
/etc/init.d/networking restart

Please use IP addresses assigned to your own pod

Frodo Group Debug if needed

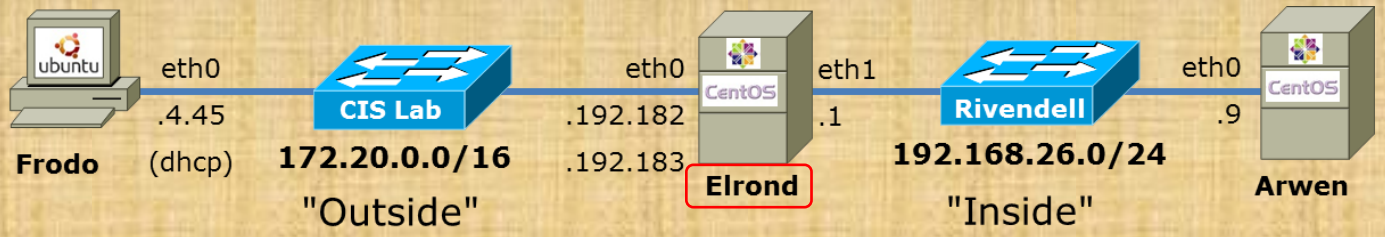


[cis192@p26-elrond ~]$ **cat /etc/sysconfig/network-scripts/ifcfg-eth0**
DEVICE="eth0"
BOOTPROTO="static"
NM_CONTROLLED="no"
ONBOOT="yes"
TYPE="Ethernet"
IPADDR=172.20.192.182
NETMASK=255.255.0.0

Please use IP addresses assigned to your own pod

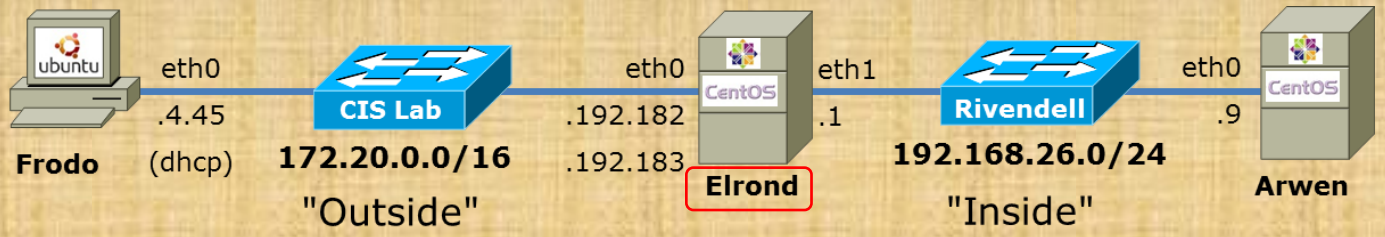
[cis192@p26-elrond ~]$ **cat /etc/sysconfig/network-scripts/ifcfg-eth0:1**
DEVICE="eth0:1"
BOOTPROTO="static"
NM_CONTROLLED="no"
ONBOOT="yes"
TYPE="Ethernet"
IPADDR=172.20.192.183
NETMASK=255.255.0.0

Please use IP addresses assigned to your own pod




```
[cis192@p26-elrond ~]$ cat /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE="eth1"
BOOTPROTO="static"
NM_CONTROLLED="no"
ONBOOT="yes"
TYPE="Ethernet"
IPADDR=192.168.26.1
NETMASK=255.255.255.0
```

Please use IP addresses assigned to your own pod

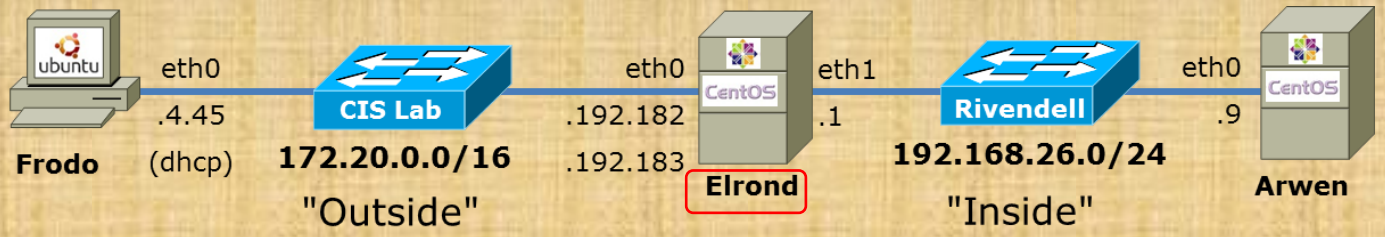


[root@p26-elrond ~]# **cat /etc/sysconfig/network**
NETWORKING=yes
HOSTNAME=p26-elrond.rivendell
GATEWAY=172.20.0.1

[root@p26-elrond ~]# **cat /etc/resolv.conf**
search cislab.net
nameserver 172.30.5.8

service network restart

Elrond should have Internet access now

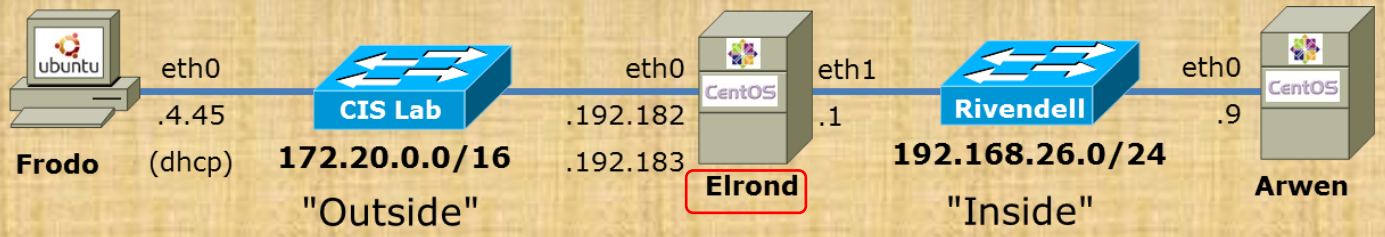


[root@p26-elrond ~]# **cat /etc/sysctl.conf**
Kernel sysctl configuration file for Red Hat Linux
#
For binary values, 0 is disabled, 1 is enabled. See sysctl(8) and
sysctl.conf(5) for more details.

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

< snipped >

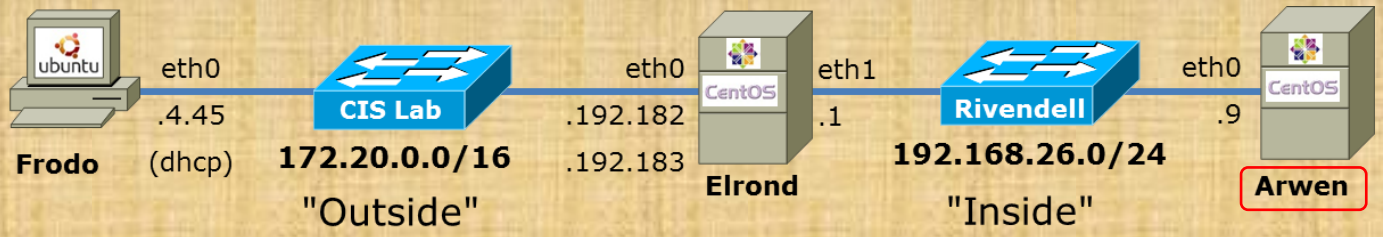
sysctl -p



[root@p26-elrond ~]# **iptables -F**

Flush all rules from firewall

Elrond Group Debug if needed



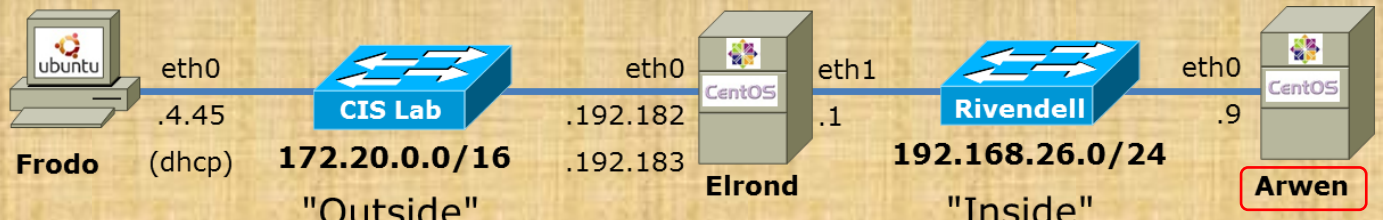
[root@p26-arwen ~]# **cat /etc/sysconfig/network-scripts/ifcfg-eth0**
DEVICE="eth0"
BOOTPROTO="static"
NM_CONTROLLED="no"
ONBOOT="yes"
TYPE="Ethernet"
IPADDR=192.168.26.9
NETMASK=255.255.255.0

Please use IP addresses assigned to your own pod

[root@p26-arwen ~]# **cat /etc/sysconfig/network**
NETWORKING=yes
HOSTNAME=p26-arwen.rivendell
GATEWAY=192.168.26.1

[root@p26-arwen ~]# **cat /etc/resolv.conf**
search cislab.net
nameserver 172.30.5.8

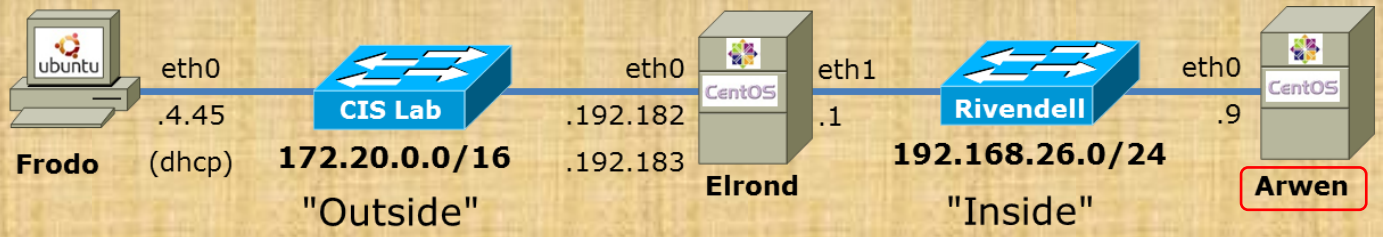
service network restart




```
[root@p26-arwen ~]# cat /etc/xinetd.d/telnet
# default: on
# description: The telnet server serves telnet sessions; it uses \
#       unencrypted username/password pairs for authentication.
service telnet
{
        flags           = REUSE
        socket_type     = stream
        wait            = no
        user            = root
        server          = /usr/sbin/in.telnetd
        log_on_failure  += USERID
        disable         = no
}

[root@p26-arwen ~]#
```

Install and configure Telnet server

service xinetd start
chkconfig xinetd on



iptables -I INPUT 5 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT

Allow Telnet connections

Arwen Group Debug if needed

More FTP (module)

FTP (more)

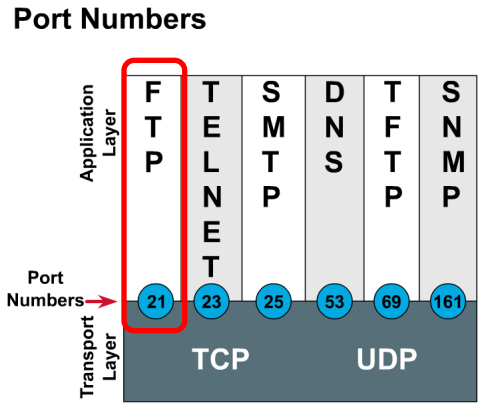
Installing and Configuring FTP

FTP

- File transfer protocol
- Client-server model
- Uses port 20 (for data) and 21 (for commands)
- Not secure, uses clear text over the network that can be sniffed

FTP uses ports 20 and 21

```
[root@elrond bin]# cat /etc/services
< snipped >
ftp-data        20/tcp
ftp-data        20/udp
# 21 is registered to ftp, but also used by fsp
ftp             21/tcp
ftp             21/udp          fsp fspd
< snipped >
[root@elrond bin]#
```



vsftp

Step 10 *Additional security*

This is why root cannot login for ftp access

```
[root@legolas ~]# cat /etc/vsftpd/user_list
# vsftpd userlist
# If userlist_deny=NO, only allow users in this file
# If userlist_deny=YES (default), never allow users in this file, and
# do not even prompt for a password.
# Note that the default vsftpd pam config also checks /etc/vsftpd/ftpusers
# for users that are denied.
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
news
uucp
operator
games
nobody
[root@legolas ~]#
```

```
[root@legolas ~]# cat /etc/vsftpd/ftpusers
# Users that are not allowed to login via ftp
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
news
uucp
operator
games
nobody
[root@legolas ~]#
```

FTP

Two sockets are used

- Commands (requests and responses)
- Data transfer

Active mode

- Server initiates new connection for data transfer
- Client firewall must allow incoming connection

Passive mode

- Client initiates new connection for data transfer
- Server firewall must allow incoming connection

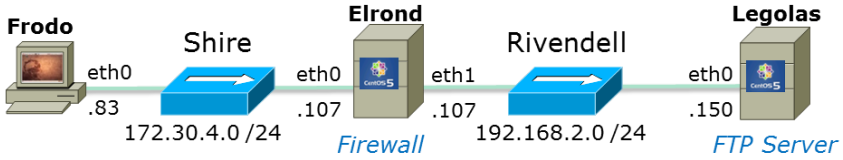
FTP

Active mode

- Client sends PORT command to indicate the port it will listen on
- Server initiates new connection to that port for data transfer

Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21



SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75

PORT 172, 30,4, 83, 166, 75
166 decimal = A6 hex
75 decimal = 4b hex
A64B hex = 42571 (decimal)

Socket for data transfer

Client	Server
172.30.4.83	192.168.2.150
42571	20

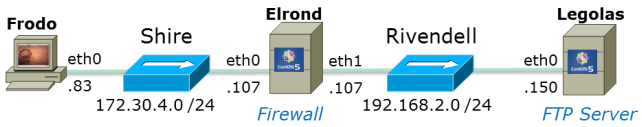
FTP

Socket for data transfer

Client	Server
172.30.4.83	192.168.2.150
42571	20

Active mode

- Client sends PORT command to indicate port it will listen on
- Server initiates new connection for data transfer to that port



PORT command to listen on port 166, 75
166 decimal = A6 hex
75 decimal = 4b hex
A64B hex = 42571 (decimal)

SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Wi
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Se
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, AC
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

Retrieve legolas file

3 way handshake initiated by server

File transfer

4 way handshake to close connection

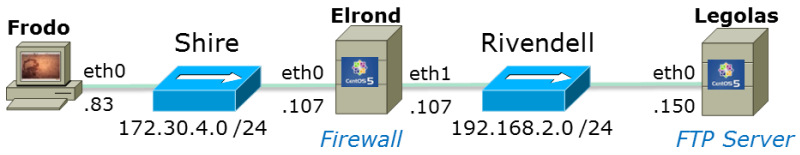
FTP

Passive mode

- Client send PASV request
- Server replies with port it will listen on
- Client initiates new connection to that port for data transfer

Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21



SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PASV
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 227 Entering Passive Mode (192,168,2,150,200,83

Response 192, 168, 2, 150, 200, 83
200 decimal = C8 hex
83 decimal = 53 hex
C853 hex = 51283 (decimal)

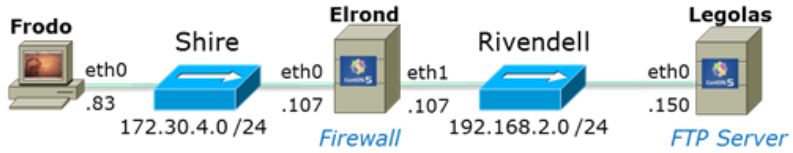
Socket for data transfer

Client	Server
172.30.4.83	192.168.2.150
41025	51283

FTP

Passive mode

- Client send PASV request
- Server replies with port it will listen on
- Client initiates new connection to that port for data transfer



Client sends passive request

SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PASV
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 227 Entering Passive Mode (192,168,2,150,200,83
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=88 Ack=313 Win=5856 Len=0
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [SYN] Seq=0 Win=
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [SYN, ACK] Seq=0
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	51283	172.30.4.83	41025	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [FIN, ACK] Seq=19 Ac
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=19 W
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=378
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [FIN, ACK] Seq=1 Ack
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=397 Win=5856 Len=0

Server to listen on 200, 83 = C853 = 51283

3 way handshake initiated by client

Retrieve legolas file

File transfer

4 way handshake to close connection

Example FTP Session

```
root@frodo:~# ftp legolas
Connected to legolas.
220 (vsFTPd 2.0.5)
Name (legolas:cis192): cis192
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> get legolas
local: legolas remote: legolas
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.04 secs (0.5 kB/s)
ftp> passive
Passive mode on.
ftp> get legolas
local: legolas remote: legolas
227 Entering Passive Mode (192,168,2,150,200,83)
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.00 secs (5.1 kB/s)
ftp> passive
Passive mode off.
ftp> get legolas
local: legolas remote: legolas
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.00 secs (23.8 kB/s)
ftp> bye
221 Goodbye.
root@frodo:~#
```

Connect to server

Login

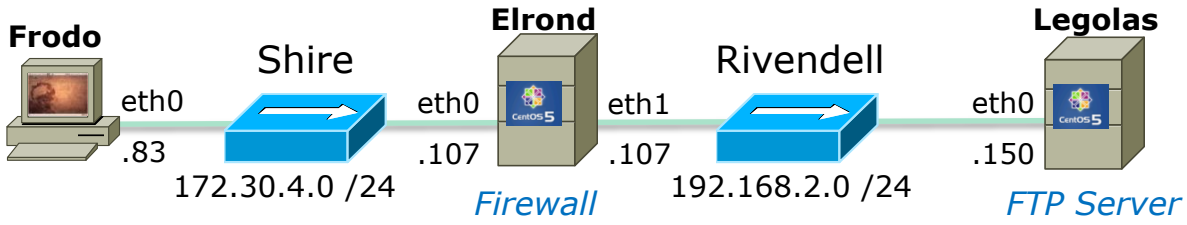
Initialize

*Get legolas file using **active** mode*

Get legolas file using passive mode

*Get legolas file using **active** mode*

End




```
root@frodo:~# ftp legolas
Connected to legolas.
220 (vsFTPd 2.0.5)
```

Frodo FTP's into Legolas

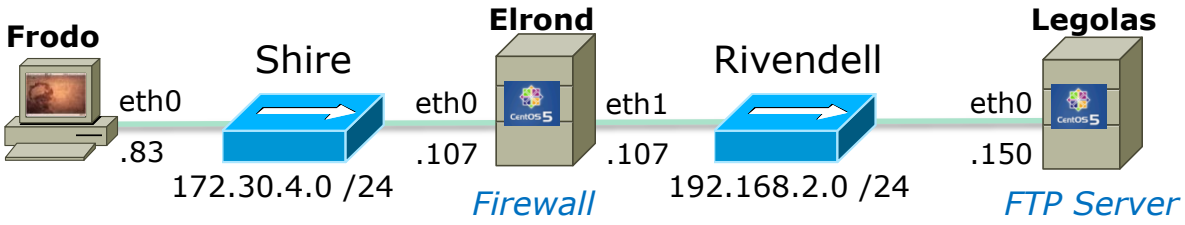
SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [SYN] Seq=0 Win=58
192.168.2.150	21	172.30.4.83	42855	TCP	ftp > 42855 [SYN, ACK] Seq=0 A
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=1 Ack=1
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 220 (vsFTPd 2.0.5)
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=1 Ack=21 Win=5856 Len=0

3 way handshake initiated by client

- *3 way handshake*
- *New connection initiated by client*

Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21




```
Name (legolas:cis192): cis192
331 Please specify the password.
Password:
230 Login successful.
```

Note the login happens over the wire in clear "sniffable" text

SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: USER cis192
192.168.2.150	21	172.30.4.83	42855	TCP	ftp > 42855 [ACK] Seq=21 Ack=14 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 331 Please specify the password.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=14 Ack=55 Win=5856 Len=0
Vmware_4e:21::		Vmware_7c:18:f5		ARP	Who has 192.168.2.150? Tell 192.168.2.107
Vmware_7c:18:'		Vmware_4e:21:a5		ARP	192.168.2.150 is at 00:0c:29:7c:18:f5
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PASS Cabrillo
192.168.2.150	52916	207.62.187.54	53	DNS	Standard query PTR 83.4.30.172.in-addr.arpa
207.62.187.54	53	192.168.2.150	52916	DNS	Standard query response, No such name
192.168.2.150	21	172.30.4.83	42855	TCP	ftp > 42855 [ACK] Seq=55 Ack=29 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 230 Login successful.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=29 Ack=78 Win=5856 Len=0

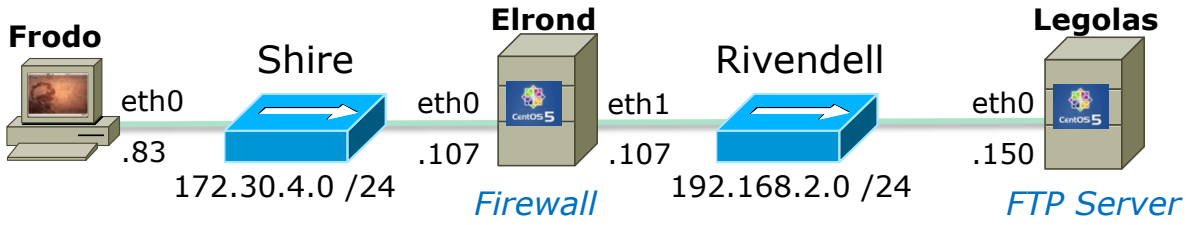
username

password

Login with username and password. Note the reverse DNS lookup attempt by the FTP server

Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21



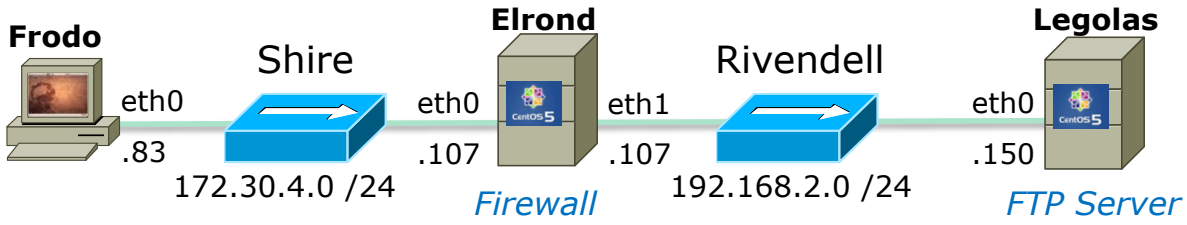

```
Remote system type is UNIX.
Using binary mode to transfer files.
```

- *Client requests system type and server replies UNIX.*
- *Client requests binary mode (Type I) transfers and server changes to binary mode*

SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: SYST
192.168.2.150	21	172.30.4.83	42855	TCP	ftp > 42855 [ACK] Seq=78 Ack=35 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 215 UNIX Type: L8
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=35 Ack=97 Win=5856 Len=0
172.30.4.83	42855	192.168.2.150	21	FTP	Request: TYPE I
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 Switching to Binary mode.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=43 Ack=128 Win=5856 Len=0

Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21



Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21

Socket for data transfer

Client	Server
172.30.4.83	192.168.2.150
42571	20

***Active Mode** is when server initiates new connection for data transfer*

```
ftp> get legolas
local: legolas remote: legolas
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.04 secs (0.5 kB/s)
```

SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Wi
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Se
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, AC
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

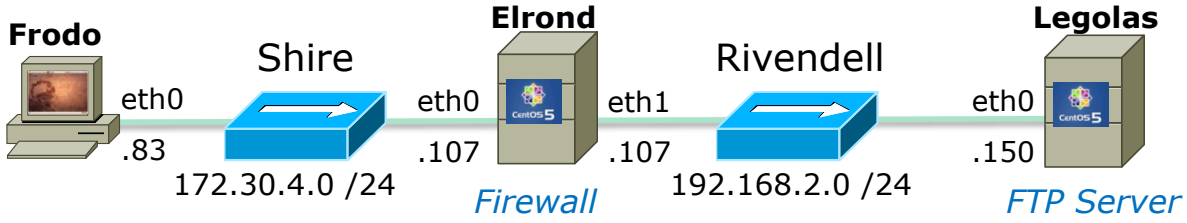
PORT command to listen on 166, 75 = A64B = 42571

Retrieve legolas file

3 way handshake initiated by server

File transfer

4 way handshake to close connection



Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21

Socket for data transfer

Client	Server
172.30.4.83	192.168.2.150
41025	51283

***Passive Mode** is when client initiates new connection for data transfer*

```
ftp> passive
Passive mode on.
ftp> get legolas
local: legolas remote: legolas
227 Entering Passive Mode (192,168,2,150,200,83)
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.00 secs (5.1 kB/s)
```

SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PASV
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 227 Entering Passive Mode (192,168,2,150,200,83
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=88 Ack=313 Win=5856 Len=0
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [SYN] Seq=0 Win=
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [SYN, ACK] Seq=0
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	51283	172.30.4.83	41025	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [FIN, ACK] Seq=19 Ac
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [ACK] Seq=1 Ack=19 W
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=378
172.30.4.83	41025	192.168.2.150	51283	TCP	41025 > 51283 [FIN, ACK] Seq=1 Ack
192.168.2.150	51283	172.30.4.83	41025	TCP	51283 > 41025 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=102 Ack=397 Win=5856 Len=0

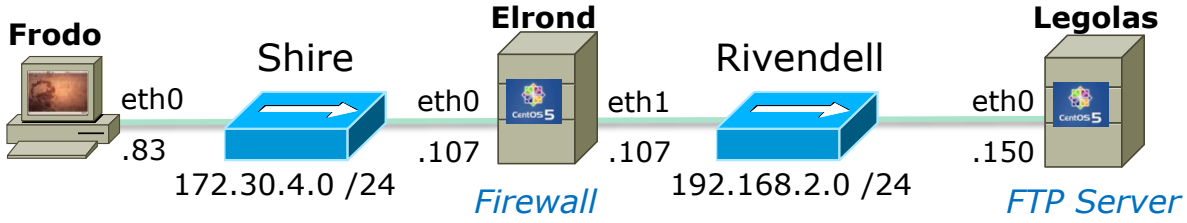
Passive reply to listen on 200, 83 = C853 = 51283

3 way handshake initiated by client

Retrieve legolas file

File transfer

4 way handshake to close connection



Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21

Socket for data transfer

Client	Server
172.30.4.83	192.168.2.150
34098	20

***Active Mode** is when server initiates new connection for data transfer*

```
ftp> passive
Passive mode off.
ftp> get legolas
local: legolas remote: legolas
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.00 secs (23.8 kB/s)
```

SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,133,50
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=127 Ack=448 Win=5856 Len=0
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	34098	TCP	ftp-data > 34098 [SYN] Seq=0 Wi
172.30.4.83	34098	192.168.2.150	20	TCP	34098 > ftp-data [SYN, ACK] Seq
192.168.2.150	20	172.30.4.83	34098	TCP	ftp-data > 34098 [ACK] Seq=1 Ac
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	34098	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	34098	TCP	ftp-data > 34098 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	34098	192.168.2.150	20	TCP	34098 > ftp-data [ACK] Seq=1
172.30.4.83	34098	192.168.2.150	20	TCP	34098 > ftp-data [ACK] Seq=1
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=141 Ack=513 Win=5856 Len=0
172.30.4.83	34098	192.168.2.150	20	TCP	34098 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
192.168.2.150	20	172.30.4.83	34098	TCP	ftp-data > 34098 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=141 Ack=532 Win=5856 Len=0

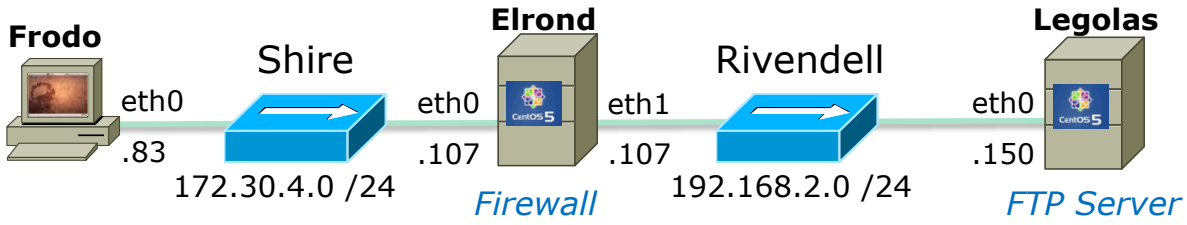
PORT command to listen on 133, 50 = 8532 = 34098

Retrieve legolas file

3 way handshake initiated by server

File transfer

4 way handshake to close connection




```
ftp> bye
221 Goodbye.
```

SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: QUIT
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 221 Goodbye.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=147 Ack=546
192.168.2.150	21	172.30.4.83	42855	TCP	ftp > 42855 [FIN, ACK] Seq=546 Ac
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [FIN, ACK] Seq=147 Ac
192.168.2.150	21	172.30.4.83	42855	TCP	ftp > 42855 [ACK] Seq=547 Ack=148

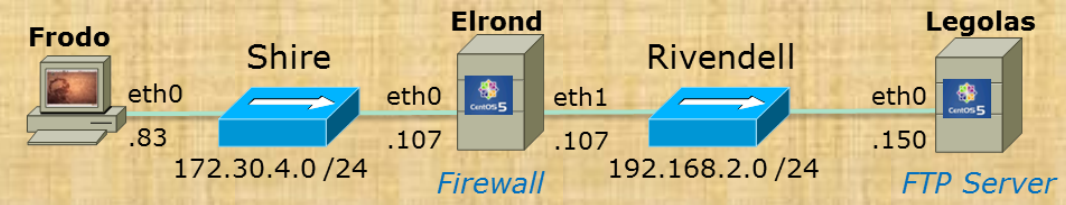
4 way handshake to close connection

Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21

Practice

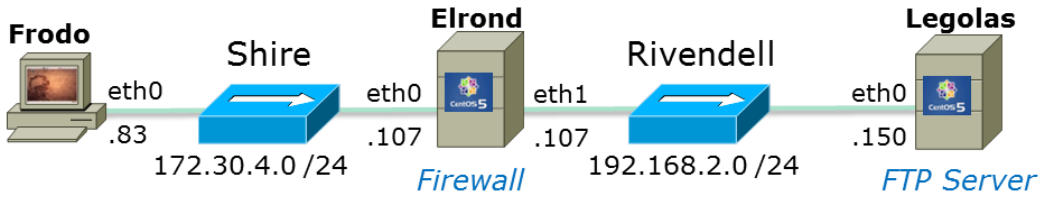
FTP



SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MS
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack=1 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

Was this FTP file transfer done in active or passive mode

FTP

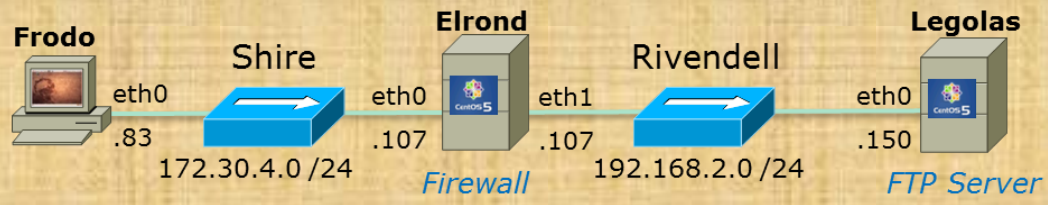


SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq MS
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ac
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

3 way handshake initiated by server

Active: server initiated the connection for file transfer

FTP

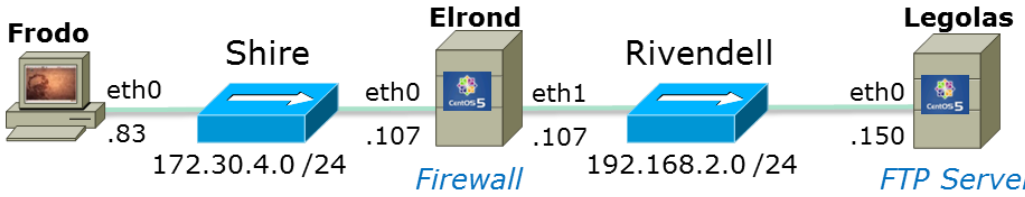


SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MS
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack=1 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

What socket was created for the data transfer?

Client	Server
IP:	IP:
Port:	Port:

FTP



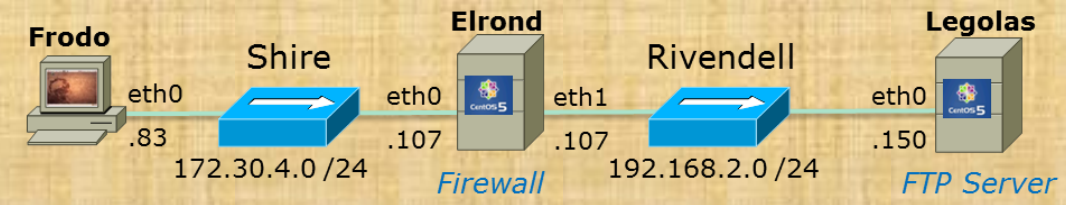
SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 W
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Se
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 A
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

3 way handshake initiated by server

Socket for data transfer

Client	Server
172.30.4.83	192.168.2.150
42571	20

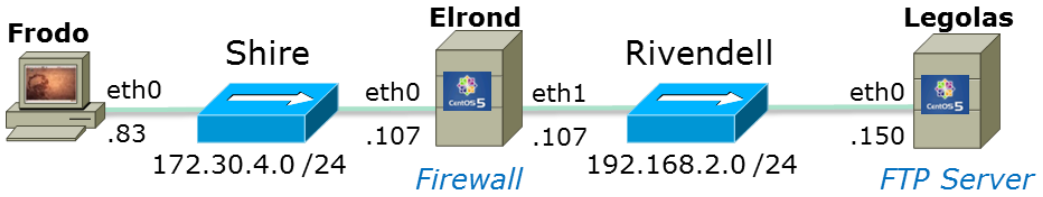
FTP



SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MS
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack=1 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

Is the same socket being used for both commands and data?

FTP



SIP	SP	DIP	DP	Protocol	Info
172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MS
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack=1 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 150 Opening BINARY mode data connection for leg
192.168.2.150	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

No, one socket for commands and another for data transfer

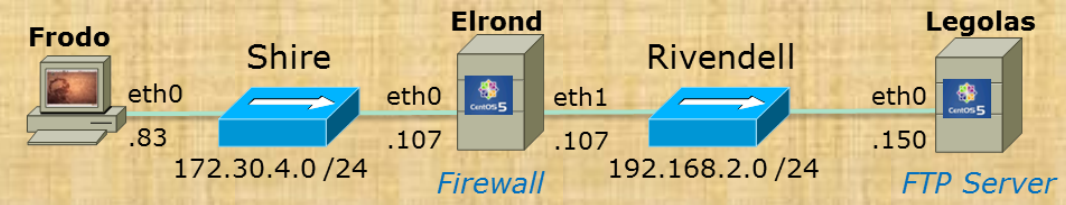
Socket for commands

Client	Server
172.30.4.83	192.168.2.150
42855	21

Socket for data transfer

Client	Server
172.30.4.83	192.168.2.150
42571	20

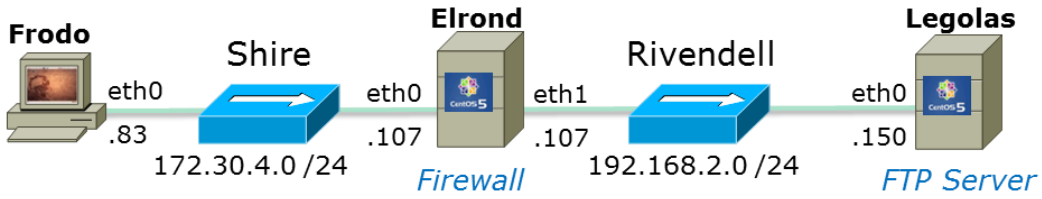
FTP



	SIP	SP	DIP	DP	Protocol	Info
1	172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
2	192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
3	172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
4	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
5	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MS
6	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack=1 Win=5888 Len=0
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	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
9	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
10	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
11	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
12	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
13	192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
14	172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

After which packet was the connection made for data transfer considered to be ESTABLISHED?

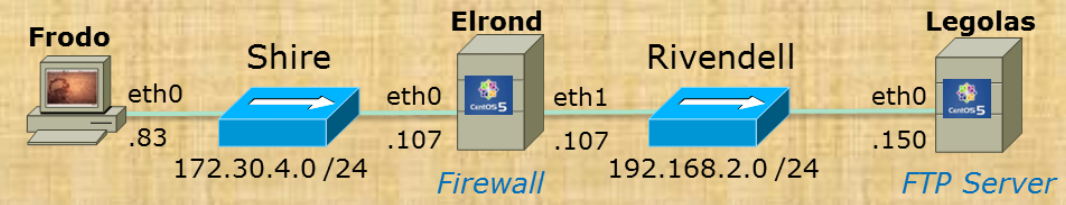
FTP



	SIP	SP	DIP	DP	Protocol	Info
1	172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
2	192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
3	172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
4	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
5	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MS
6	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack=1 Win=5888 Len=0
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	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
9	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
10	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
11	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
12	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
13	192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
14	172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

After packet 6 was sent, That is when the three way handshake was completed.

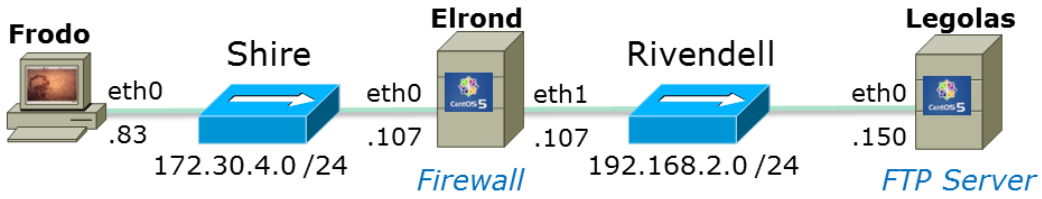
FTP



	SIP	SP	DIP	DP	Protocol	Info
1	172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
2	192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
3	172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
4	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
5	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MS
6	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack=1 Win=5888 Len=0
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	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
9	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
10	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
11	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
12	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
13	192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
14	172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

Which packet or packets would be considered RELATED to the connection used for sending packet 3 above

FTP



	SIP	SP	DIP	DP	Protocol	Info
1	172.30.4.83	42855	192.168.2.150	21	FTP	Request: PORT 172,30,4,83,166,75
2	192.168.2.150	21	172.30.4.83	42855	FTP	Response: 200 PORT command successful. Consider using PAS
3	172.30.4.83	42855	192.168.2.150	21	FTP	Request: RETR legolas
4	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TSV=
5	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MS
6	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=1 Ack=1 Win=5888 Len=0
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	20	172.30.4.83	42571	FTP-DATA	FTP Data: 18 bytes
9	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [FIN, ACK] Seq=19 Ack=1 Win=5888 Len=0
10	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [ACK] Seq=1 Ack=19 Win=5856 Len=0
11	172.30.4.83	42571	192.168.2.150	20	TCP	42571 > ftp-data [FIN, ACK] Seq=1 Ack=20 Win=5856 Len=0
12	192.168.2.150	20	172.30.4.83	42571	TCP	ftp-data > 42571 [ACK] Seq=20 Ack=2 Win=5888 Len=0
13	192.168.2.150	21	172.30.4.83	42855	FTP	Response: 226 File send OK.
14	172.30.4.83	42855	192.168.2.150	21	TCP	42855 > ftp [ACK] Seq=82 Ack=263 Win=5856 Len=0

Any of the packets used for the data transfer connection would be considered RELATED to the connection used for commands.

Firewalls and FTP

Firewall - FTP Command port

```
[root@elrond pub]# iptables -I RH-Firewall-1-INPUT 9 -p tcp -m state --state NEW -m tcp --dport 21 -j ACCEPT

[root@elrond pub]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0           0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0           0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:21
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-prohibited

[root@elrond pub]# iptables-save > /etc/sysconfig/iptables
[root@elrond pub]#
```

Open TCP port 21 for FTP

Save to make changes persist across restarts

FTP Connection Tracking
(for kernel versions after 2.6.19)

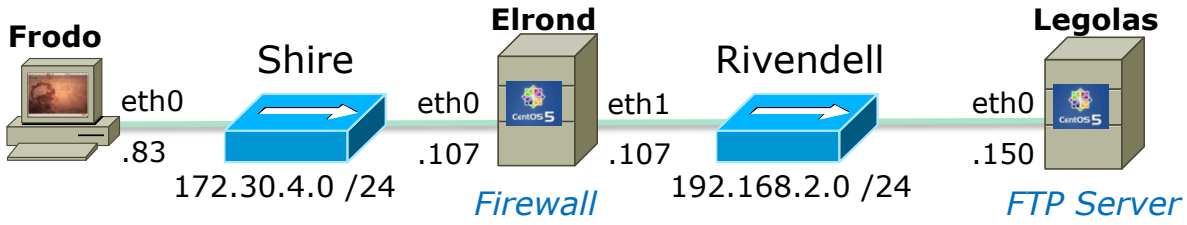
nf_conntrack_ftp and **nf_nat_ftp** are kernel modules. They are used to track related FTP connections so they can get through the firewall.

From the command line (temporary)

```
[root@celebrian ~]# modprobe nf_conntrack_ftp
[root@celebrian ~]# modprobe nf_nat_ftp

[root@beast pub]# lsmod | grep ftp
nf_nat_ftp              2544  0
nf_nat                 18618  1 nf_nat_ftp
nf_conntrack_ftp       10449  1 nf_nat_ftp
nf_conntrack           66010  6
nf_nat_ftp,nf_nat,nf_conntrack_ftp,nf_conntrack_ipv4,nf_conntrack_ipv6,xt_state
```

To load at system boot (permanent), edit this file to include:

```
[root@celebrian ~]# cat /etc/sysconfig/iptables-config
# Load additional iptables modules (nat helpers)
#   Default: -none-
# Space separated list of nat helpers (e.g. 'ip_nat_ftp ip_nat_irc'), which
# are loaded after the firewall rules are applied. Options for the helpers are
# stored in /etc/modprobe.conf.
IPTABLES_MODULES="nf_conntrack_ftp nf_nat_ftp"
< snipped >
```

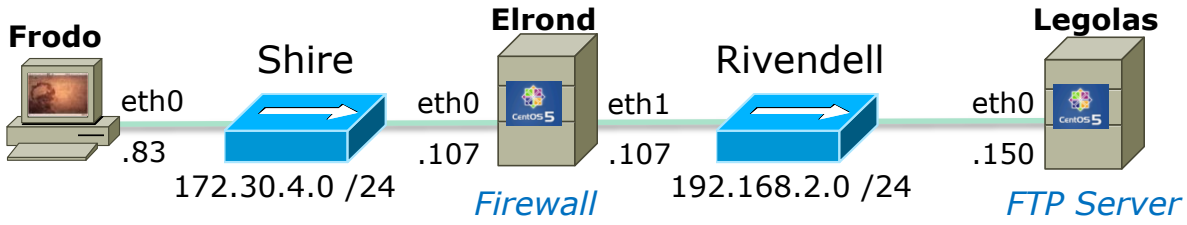



```
[root@elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:53
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:21

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

For DNS lookups by FTP server

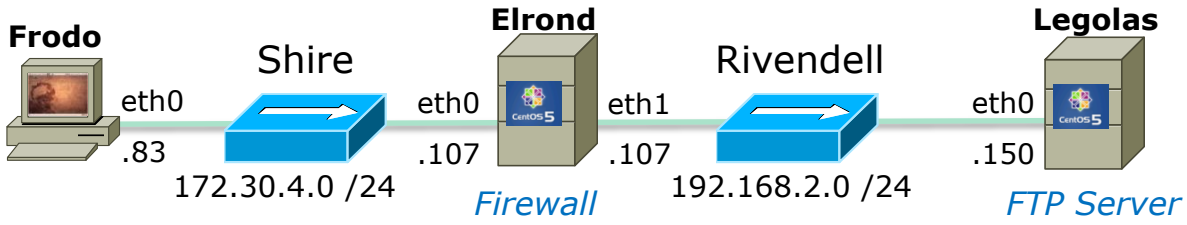
This firewall setting allows external clients (Frodo) to access the FTP server (Legolas)

Note: The FTP data port 20 is not specified




```
root@frodo:~# ftp legolas
Connected to legolas.
220 (vsFTPd 2.0.5)
Name (legolas:cis192): cis192
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> get legolas
local: legolas remote: legolas
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.00 secs (15.1 kB/s)
ftp> passive
Passive mode on.
ftp> get legolas
local: legolas remote: legolas
227 Entering Passive Mode (192,168,2,150,224,164)
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.00 secs (8.6 kB/s)
ftp> bye
221 Goodbye.
root@frodo:~#
```

Successful downloads using both active and passive mode using the firewall settings in previous slide

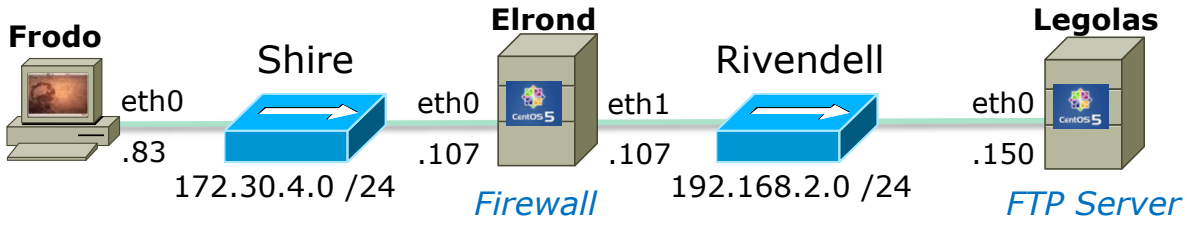


What If? *We remove the firewall opening for the DNS lookups sent by the FTP server*

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
~~ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:53~~
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:21

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

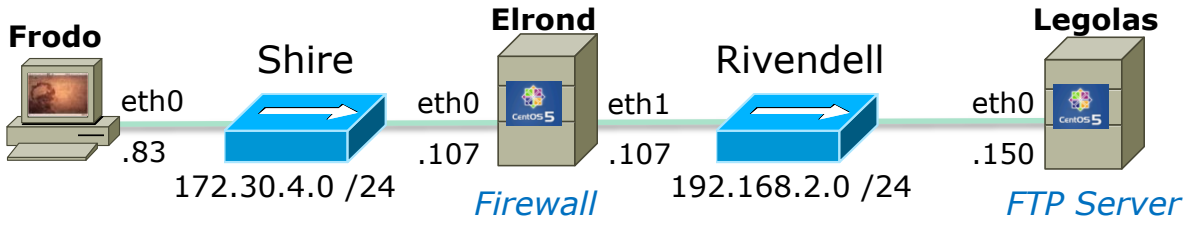
```
[root@elrond ~]# iptables -D FORWARD 1
```

Now DNS lookups are blocked




```
root@frodo:~# ftp legolas
Connected to legolas.
220 (vsFTPd 2.0.5)
Name (legolas:cis192): cis192
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> get legolas
local: legolas remote: legolas
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.00 secs (15.1 kB/s)
ftp> passive
Passive mode on.
ftp> get legolas
local: legolas remote: legolas
227 Entering Passive Mode (192,168,2,150,224,164)
150 Opening BINARY mode data connection for legolas (18 bytes).
226 File send OK.
18 bytes received in 0.00 secs (8.6 kB/s)
ftp> bye
221 Goodbye.
root@frodo:~#
```

Result: Instead of a fast login, now there is a delay of about 15 seconds before the successful login messages and ftp prompt are displayed

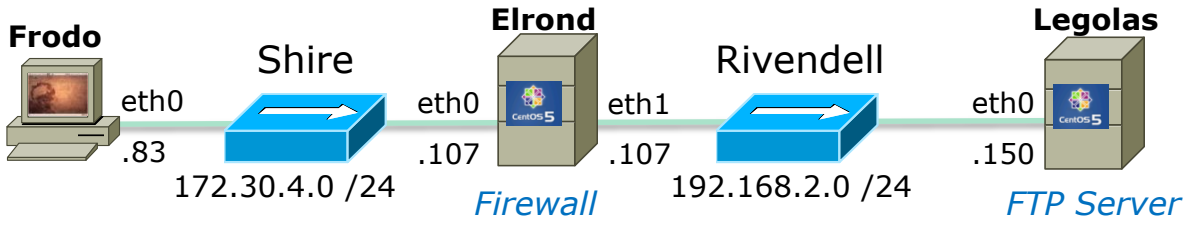



```
root@frodo:~# ftp legolas
Connected to legolas.
220 (vsFTPd 2.0.5)
Name (legolas:cis192): cis192
331 Please specify the password.
Password:
230 Login successful.
```

Delay encountered (~15 seconds) here after dropping DNS lookups in firewall

SIP	SP	DIP	DP	Protocol	Info	No.	Time
172.30.4.195	40823	192.168.2.150	21	FTP	Request: PASS Cabrillo	12	8.920738
192.168.2.150	58200	207.62.187.54	53	DNS	Standard query PTR 195.4.30.172.in-addr.arp	13	8.938715
192.168.2.150	21	172.30.4.195	40823	TCP	ftp > 40823 [ACK] Seq=55 Ack=29 Win=5888 Le	14	8.951876
192.168.2.150	58200	207.62.187.54	53	DNS	Standard query PTR 195.4.30.172.in-addr.arp	15	16.612474
192.168.2.150	21	172.30.4.195	40823	FTP	Response: 230 Login successful.	16	24.336986

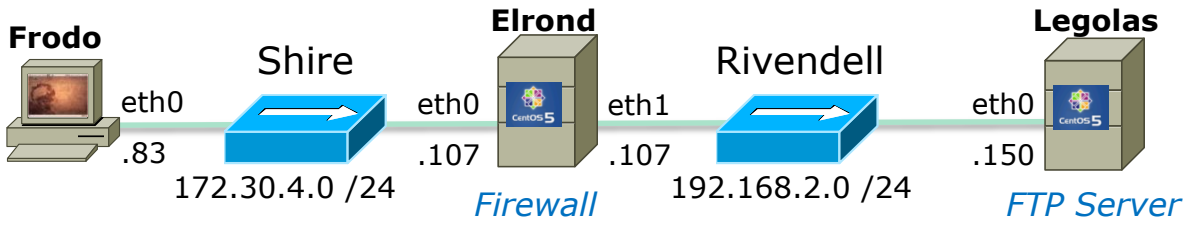
The login is delayed while the two DNS requests time-out.



What If? *We next remove the related state condition from the firewall?*

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:21

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

```
[root@elrond ~]# iptables -D FORWARD 1
[root@elrond ~]# iptables -I FORWARD 1 -m state --state ESTABLISHED -j ACCEPT
```




```
root@frodo:~# ftp legolas
Connected to legolas.
220 (vsFTPd 2.0.5)
Name (legolas:cis192): cis192
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> get legolas
local: legolas remote: legolas
200 PORT command successful. Consider using PASV.
425 Failed to establish connection.
ftp>
```

Hangs up here, because the related connection for the data transfer is now blocked by the firewall.

Gives up after 5 tries of attempting to do a 3-way handshake

SIP	SP	DIP	DP	Protocol	Info	No.	Time
172.30.4.195	59956	192.168.2.150	21	FTP	Request: RETR legolas	123	383.241428
192.168.2.150	20	172.30.4.195	58333	TCP	ftp-data > 58333 [SYN] Seq=0 Win=5840 Len=(	124	383.242944
192.168.2.150	21	172.30.4.195	59956	TCP	ftp > 59956 [ACK] Seq=179 Ack=84 Win=5888 l	125	383.316282
192.168.2.150	20	172.30.4.195	58333	TCP	ftp-data > 58333 [SYN] Seq=0 Win=5840 Len=(	129	388.071827
192.168.2.150	20	172.30.4.195	58333	TCP	ftp-data > 58333 [SYN] Seq=0 Win=5840 Len=(	134	397.449484
192.168.2.150	20	172.30.4.195	58333	TCP	ftp-data > 58333 [SYN] Seq=0 Win=5840 Len=(	143	416.129995
Vmware_7c:18:		Vmware_4e:21:a5		ARP	Who has 192.168.2.107? Tell 192.168.2.150	154	443.727874
Vmware_4e:21:		Vmware_7c:18:f5		ARP	192.168.2.107 is at 00:0c:29:4e:21:a5	155	443.727967
192.168.2.150	20	172.30.4.195	58333	TCP	ftp-data > 58333 [SYN] Seq=0 Win=5840 Len=(	159	453.553314
192.168.2.150	21	172.30.4.195	59956	FTP	Response: 425 Failed to establish connectic	167	476.875137
172.30.4.195	59956	192.168.2.150	21	TCP	59956 > ftp [ACK] Seq=84 Ack=216 Win=5856 l	168	476.916311

Practice

Activity

Your friend says his new vsftp server transfers file correctly however his users are experiencing very slow authentications

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:21

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

What is your advice?

Activity

Your friend says his new vsftp server transfers file correctly however his users are experiencing a slow logins

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:53
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:21

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

Open port 53 for DNS traffic

iptables -I INPUT n -p udp --dport 53 -j ACCEPT

Activity

Your friend says her new vsftp server only works in active mode. It hangs when they enter passive mode!

What is your advice?

1) Make sure the connection tracking module for FTP has been loaded into the kernel

2) Make sure RELATED connections are being accepted by the firewall

sshd

sshd

sshd

The SSH server

- openssh-server package

- Red Hat Family
  - Installed by default
  - Use **rpm –qa | grep openssh-server** to check if installed
- Ubuntu
  - Not installed by default
  - Use **dpkg –l | grep openssh-server** to check if installed

sshd

Installation on Ubuntu

```
[root@sauron ~]# apt-get update
[root@sauron ~]# apt-get install openssh-server
```

Install using aptitude or apt-get

sshd

Installation on Ubuntu

```
root@sauron:~# apt-get update
Get:1 http://security.ubuntu.com intrepid-security Release.gpg [189B]
Ign http://security.ubuntu.com intrepid-security/main Translation-en_US
Hit http://us.archive.ubuntu.com intrepid Release.gpg
Ign http://us.archive.ubuntu.com intrepid/main Translation-en_US
Ign http://security.ubuntu.com intrepid-security/restricted Translation-en_US
Ign http://security.ubuntu.com intrepid-security/universe Translation-en_US
Ign http://security.ubuntu.com intrepid-security/multiverse Translation-en_US
Get:2 http://security.ubuntu.com intrepid-security Release [51.2kB]
Ign http://us.archive.ubuntu.com intrepid/restricted Translation-en_US
Ign http://us.archive.ubuntu.com intrepid/universe Translation-en_US

< snipped >

Get:20 http://us.archive.ubuntu.com intrepid-updates/multiverse Sources [4118B]
Fetched 784kB in 8s (93.5kB/s)
Reading package lists... Done

Current status: 270 updates [+55], 24979 new [+12].
root@sauron:~#
```

sshd

Installation on Ubuntu

```
root@sauron:~# aptitude install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
Reading extended state information
Initializing package states... Done
The following NEW packages will be installed:
  openssh-server
0 packages upgraded, 1 newly installed, 0 to remove and 270 not upgraded.
Need to get 285kB of archives. After unpacking 782kB will be used.
Writing extended state information... Done
Get:1 http://us.archive.ubuntu.com intrepid/main openssh-server 1:5.1p1-3ubuntu1 [285kB]
Fetched 285kB in 2s (99.3kB/s)
Preconfiguring packages ...
Selecting previously deselected package openssh-server.
(Reading database ... 102936 files and directories currently installed.)
Unpacking openssh-server (from .../openssh-server_1%3a5.1p1-3ubuntu1_i386.deb) ...
Processing triggers for ufw ...
Processing triggers for man-db ...
Setting up openssh-server (1:5.1p1-3ubuntu1) ...
 * Restarting OpenBSD Secure Shell server sshd                          [ OK ]

Reading package lists... Done
Building dependency tree
Reading state information... Done
Reading extended state information
Initializing package states... Done
Writing extended state information... Done

root@sauron:~#
```

sshd

Daemon control on Ubuntu

```
root@sauron:~# /etc/init.d/ssh status
 * sshd is running.

root@sauron:~# /etc/init.d/ssh stop
 * Stopping OpenBSD Secure Shell server sshd                    [ OK ]

root@sauron:~# /etc/init.d/ssh start
 * Starting OpenBSD Secure Shell server sshd                    [ OK ]
```

sshd

Daemon control on Red Hat family

```
[root@arwen ~]# service sshd status
sshd (pid 4805) is running...


[root@arwen ~]# service sshd stop
Stopping sshd:                                             [   OK   ]

[root@arwen ~]# service sshd start
Starting sshd:                                             [   OK   ]
```

Firewall for sshd

CentOS Modified

```
[root@legolas ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.3.5 on Thu Feb 26 04:33:47 2009
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [2883:272960]
:RH-Firewall-1-INPUT - [0:0]
-A INPUT -j RH-Firewall-1-INPUT
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type any -j ACCEPT
-A RH-Firewall-1-INPUT -p esp -j ACCEPT
-A RH-Firewall-1-INPUT -p ah -j ACCEPT
-A RH-Firewall-1-INPUT -d 224.0.0.251 -p udp -m udp --dport 5353 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m udp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m state --state NEW -m udp --dport 520 -j ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
# Completed on Thu Feb 26 04:33:47 2009
[root@legolas ~]#
```

New connections for the SSH port are allowed

sshd

Using netstat to view listening ssh ports

```
root@sauron:~# netstat -tln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN

root@sauron:~# netstat -tl
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 *:ssh                   *:*                     LISTEN
tcp        0      0 localhost:ipp           *:*                     LISTEN
tcp6       0      0 [::]:ssh                [::]:*                  LISTEN
root@sauron:~#
```

sshd

One SSH daemon per session

```
root@sauron:~# ps -ef | grep ssh
root      7601     1  0 13:59 ?        00:00:00 /usr/sbin/sshd
root      7607  7601  1 14:11 ?        00:00:00 sshd: root@pts/2
root      7632  7601  1 14:11 ?        00:00:00 sshd: root@pts/3
root      7658  7280  0 14:12 pts/1    00:00:00 grep ssh


root@sauron:~# who
root     tty2         2009-03-13 14:32
cis192   tty7         2009-03-15 13:16 (:0)
cis192   pts/0        2009-03-15 13:19 (:0.0)
cis192   pts/1        2009-03-15 13:19 (:0.0)
root     pts/2        2009-03-15 14:11 (legolas)
root     pts/3        2009-03-15 14:11 (arwen)
root@sauron:~#
```

sshd

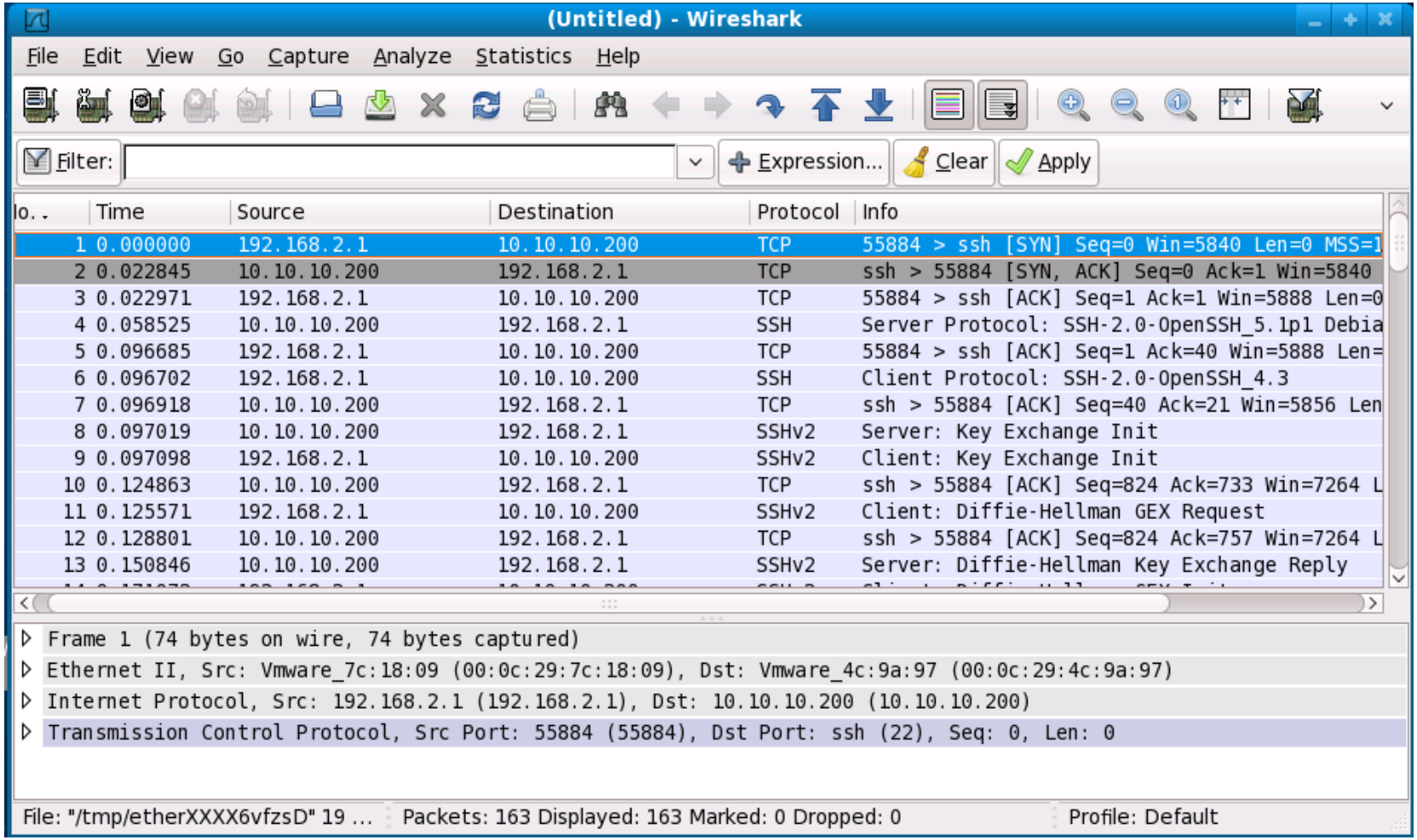
Sample session

```
[root@elrond ~]# ssh cis192@sauron
The authenticity of host 'sauron (10.10.10.200)' can't be established.
RSA key fingerprint is 61:f3:89:a3:b5:a3:2a:b9:6e:f0:9b:59:f5:93:14:b8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'sauron,10.10.10.200' (RSA) to the list of known
hosts.
cis192@sauron's password:
Linux sauron 2.6.27-7-generic #1 SMP Fri Oct 24 06:42:44 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
cis192@sauron:~$ echo This is a secret!
This is a secret!
cis192@sauron:~$ exit
logout
Connection to sauron closed.
[root@elrond ~]#
```

ssh fingerprints

```
[root@p26-elrond ~]# ssh-keygen -l -f /etc/ssh/ssh_host_rsa_key.pub
2048 81:46:a3:17:7a:4b:91:c9:24:96:f3:ac:05:5a:c4:29
/etc/ssh/ssh_host_rsa_key.pub (RSA)


root@p26-frodo:~# > .ssh/known_hosts
root@p26-frodo:~# ssh elrond
The authenticity of host 'elrond (172.20.192.182)' can't be
established.
RSA key fingerprint is
81:46:a3:17:7a:4b:91:c9:24:96:f3:ac:05:5a:c4:29.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'elrond,172.20.192.182' (RSA) to the list
of known hosts.
root@elrond's password:
Last login: Tue Mar 19 12:52:13 2013 from frodo
[root@p26-elrond ~]#
```

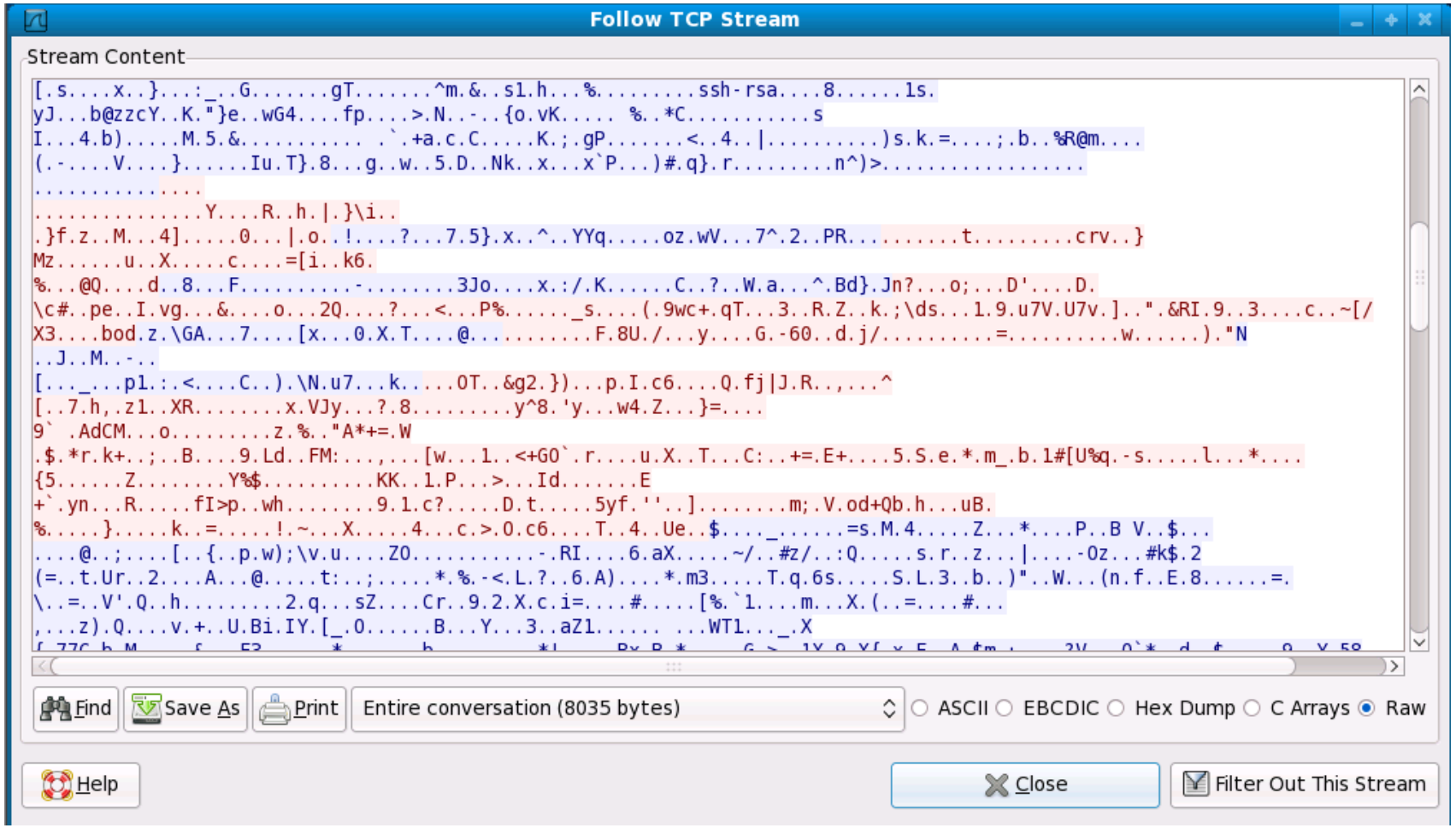
sshd



3 Way hand shake

sshd

The session is encrypted



sshd

TCP Wrappers and sshd

- sshd is compiled with TCP wrappers

```
[root@arwen ~]# type sshd
sshd is /usr/sbin/sshd
[root@arwen ~]# ldd /usr/sbin/sshd
        linux-gate.so.1 =>  (0x00146000)
        libwrap.so.0 => /usr/lib/libwrap.so.0 (0x00fb8000)
        < snipped >
        libpthread.so.0 => /lib/libpthread.so.0 (0x00185000)
[root@arwen ~]#
```

- /etc/hosts.allow – for permitted hosts
- /etc/hosts.deny – to ban hosts

sshd

TCP Wrappers and sshd example

Arwen


```
[root@arwen ~]# cat /etc/hosts.allow
sshd: frodo 192.168. 10.0.0.0/255.0.0.0
in.telnetd:  192.168.2.10  127.0.0.1
vsvtpd: frodo arwen sauron
```

For sshd, Frodo, all 192.168.x.x and all 10.x.x.x hosts are allowed

Sauron at 10.10.10.200 is included. Nosmo at 172.30.1.1 is NOT included

```
[root@arwen ~]# cat /etc/hosts.deny
ALL: ALL
```

Everyone else is denied (this includes Nosmo)

sshd

TCP Wrappers and sshd example

Arwen


```
[root@arwen ~]# cat /etc/hosts.allow
sshd: frodo 192.168. 10.0.0.0/255.0.0.0
in.telnetd:  192.168.2.10  127.0.0.1
vsftpd: frodo arwen sauron

[root@arwen ~]# cat /etc/hosts.deny
ALL: ALL
```

Sauron

```
root@sauron:~# ssh arwen
root@arwen's password:
Last login: Sun Mar 15 20:11:31 2009 from frodo
[root@arwen ~]#
```

Access permitted

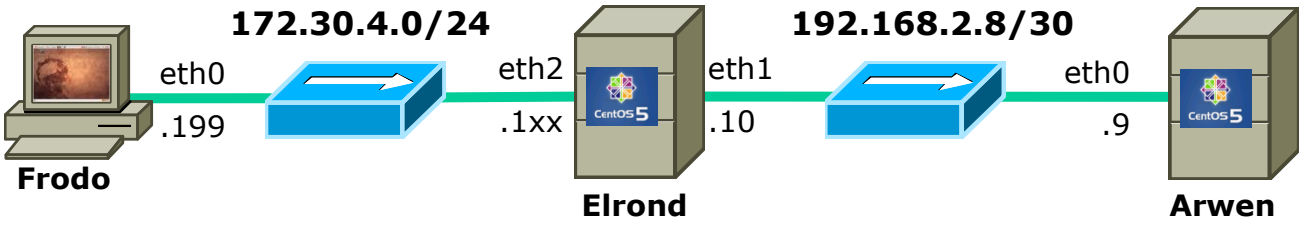
Nosmo

```
[root@nosmo root]# ssh 192.168.2.9
ssh_exchange_identification: Connection closed by remote host
[root@nosmo root]#
```

Access denied

SSH tunneling and port forwarding

SSH Port Forwarding



Public Client

Gateway and Firewall

Private Network Server

Is there a way we can tunnel an insecure protocol, like Telnet, through an SSH connection to reach a private server on our home or business network?

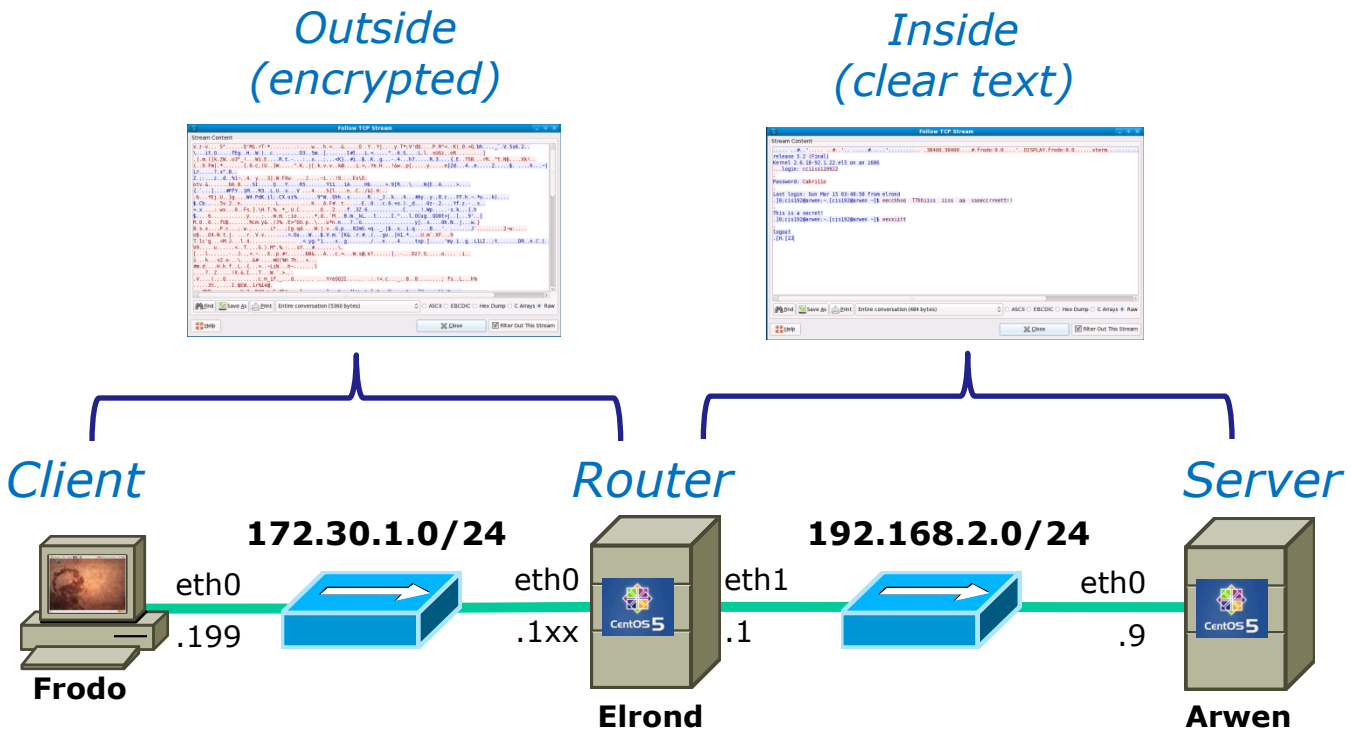
SSH Port Forwarding

-L [bind_address:]port:host:hostport

Specifies that the given port on the local (client) host is to be forwarded to the given host and port on the remote side. This works by allocating a socket to listen to port on the local side, optionally bound to the specified bind_address. Whenever a connection is made to this port, the connection is forwarded over the secure channel, and a connection is made to host port hostport from the remote machine. Port forwardings can also be specified in the configuration file. IPv6 addresses can be specified with an alternative syntax: [bind_address/]port/host/hostport or by enclosing the address in square brackets. Only the superuser can forward privileged ports. By default, the local port is bound in accordance with the GatewayPorts setting. However, an explicit bind_address may be used to bind the connection to a specific address. The bind_address of \u201clocalhost\u201d indicates that the listening port be bound for local use only, while an empty address or \u2018*\u2019 indicates that the port should be available from all inter\u2010 faces.

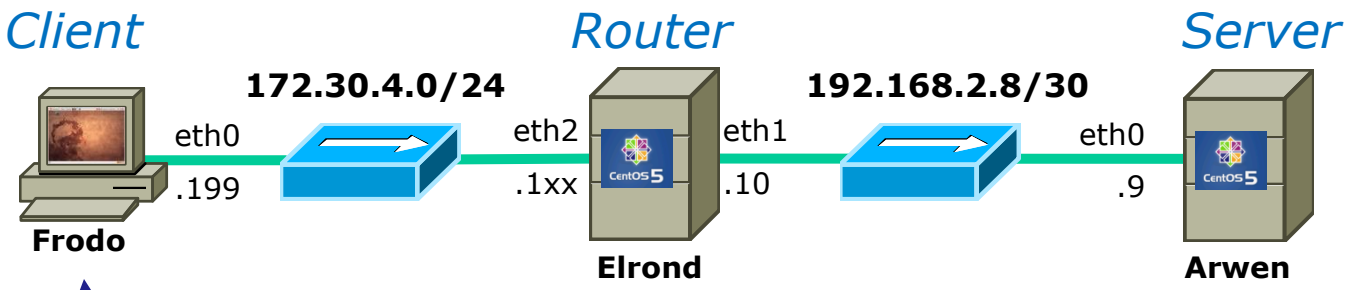
From the man page on ssh ... is that enough documentation for you?

SSH Port Forwarding



In this example we will tunnel a telnet session through an encrypted SSH connection.

SSH Port Forwarding

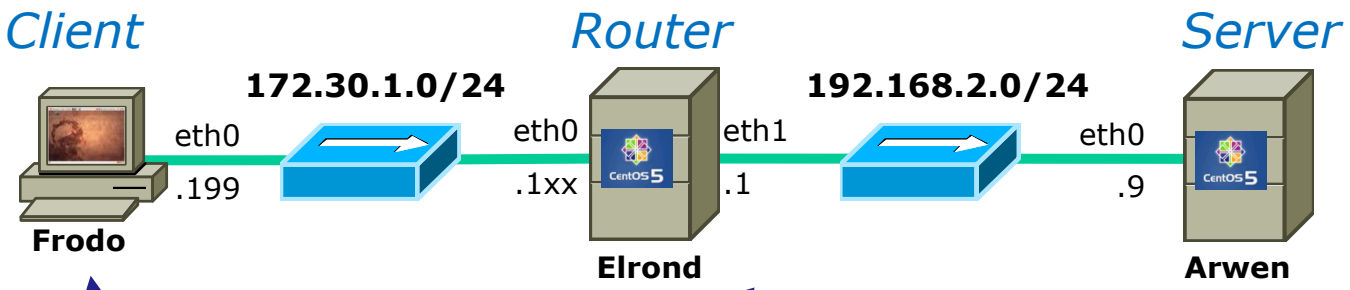



```
cis192@frodo:~$ ssh -L 8000:arwen:23  cis192@elrond
```

Any connection made to port 8000 on Frodo will get forwarded to port 23 on Arwen via Elrond.

The portion of the connection between Frodo and Elrond will be encrypted

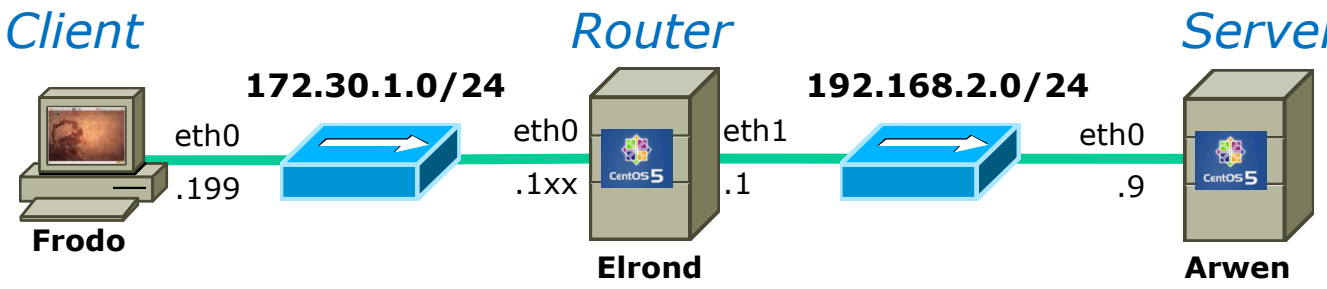
SSH Port Forwarding

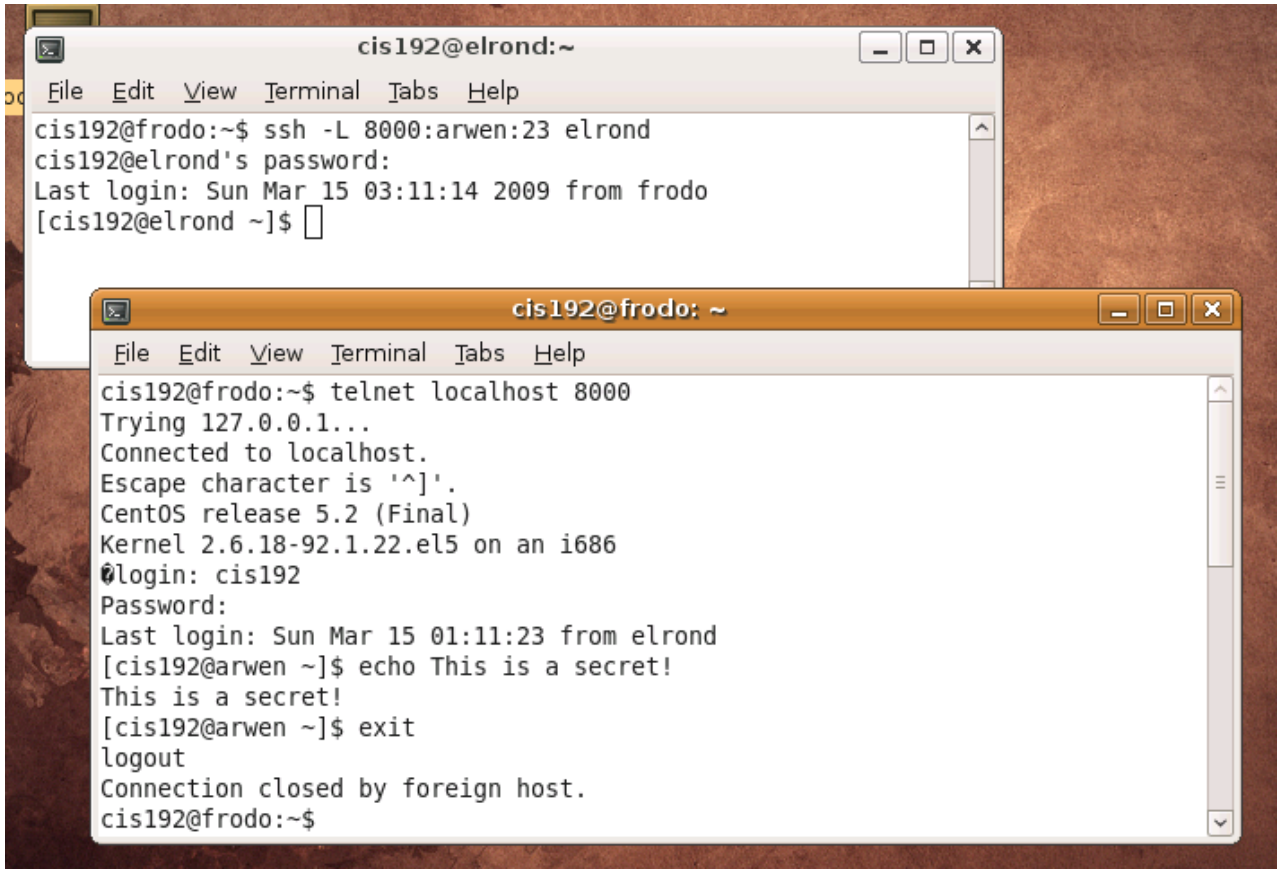



```
cis192@frodo:~$ ssh -L 8000:192.168.2.9:23   cis192@172.30.1.107
```

Same as before just using IP addresses instead of names in /etc/hosts.

SSH Port Forwarding

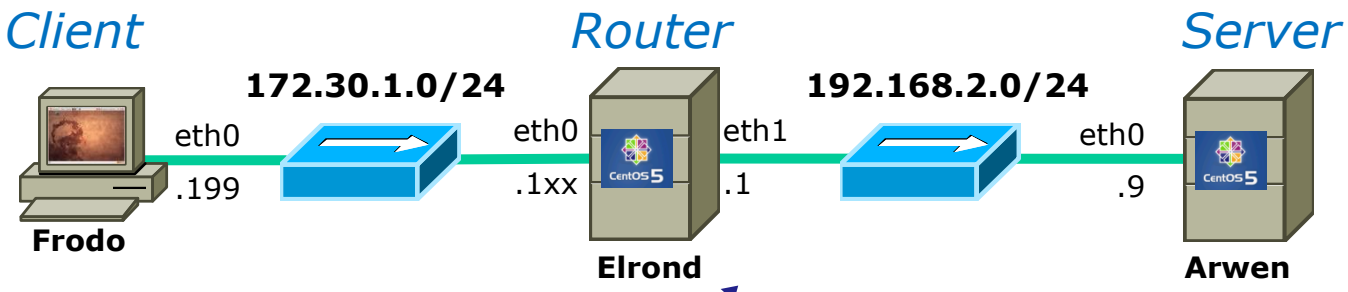




Requires one Frodo terminal to setup SSH port forwarding

And another Frodo terminal to make the Telnet connection

SSH Port Forwarding

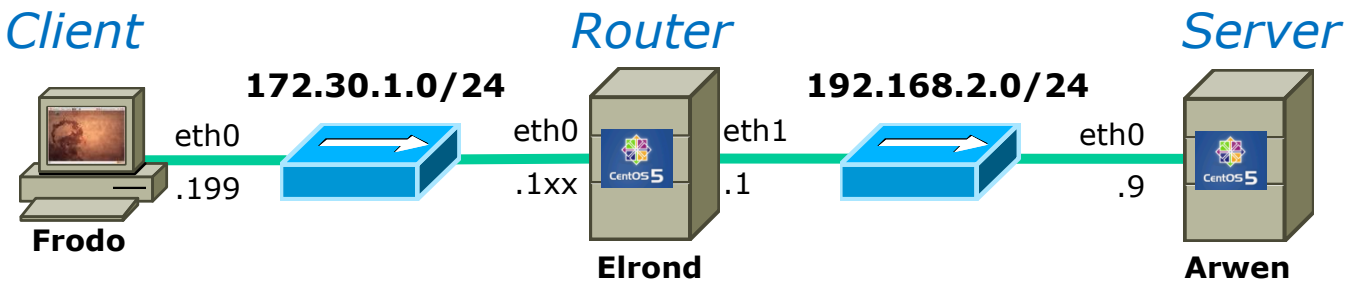



```
cis192@frodo:~$ ssh -L 8000:arwen:23 elrond
cis192@elrond's password:
Last login: Sun Mar 15 03:11:14 2009 from frodo
[cis192@elrond ~]$
[cis192@elrond ~]$
[cis192@elrond ~]$ exit
logout

Connection to elrond closed.

cis192@frodo:~$
```

Port forwarding enabled

Port forwarding disabled

SSH Port Forwarding




```
cis192@frodo:~$ telnet localhost 8000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
CentOS release 5.2 (Final)
Kernel 2.6.18-92.1.22.el5 on an i686
login: cis192
Password:
Last login: Sun Mar 15 03:48:58 from elrond
[cis192@arwen ~]$ echo This is a secret!
This is a secret!
[cis192@arwen ~]$ exit
logout


Connection closed by foreign host.
cis192@frodo:~$
```

On a different terminal on Frodo:

Telnet "to yourself" at port 8000 and notice you end up on Arwen!

SSH Port Forwarding

Frodo

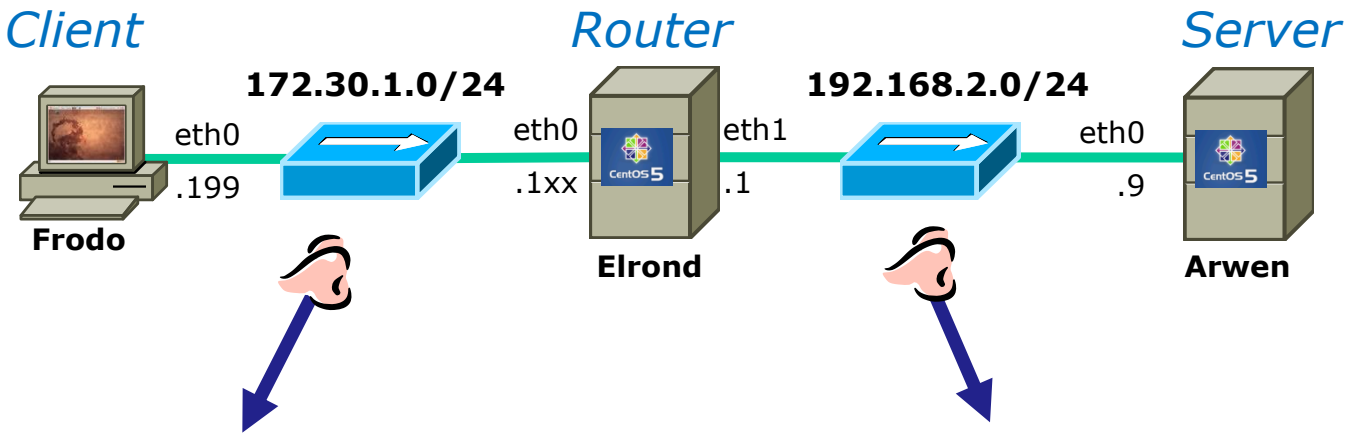
Enable port forwarding in first terminal

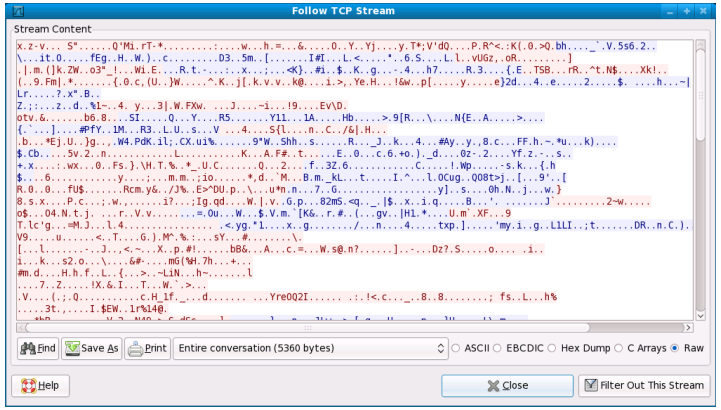
```
cis192@elrond:~
File  Edit  View  Terminal  Tabs  Help
cis192@frodo:~$ ssh -L 8000:arwen:23 elrond
cis192@elrond's password:
Last login: Sun Mar 15 03:11:14 2009 from frodo
[cis192@elrond ~]$
```

Use port forwarding in second terminal

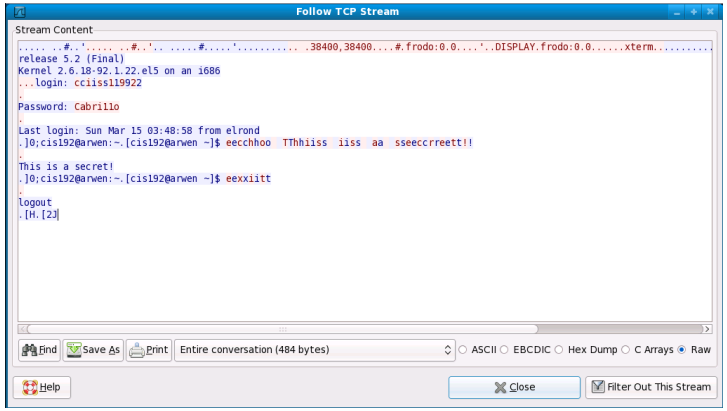
```
cis192@frodo: ~
File  Edit  View  Terminal  Tabs  Help
cis192@frodo:~$ telnet localhost 8000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
CentOS release 5.2 (Final)
Kernel 2.6.18-92.1.22.el5 on an i686
login: cis192
Password:
Last login: Sun Mar 15 03:48:58 from elrond
[cis192@arwen ~]$ echo This is a secret!
This is a secret!
[cis192@arwen ~]$ exit
logout


Connection closed by foreign host.
cis192@frodo:~$
```

SSH Port Forwarding

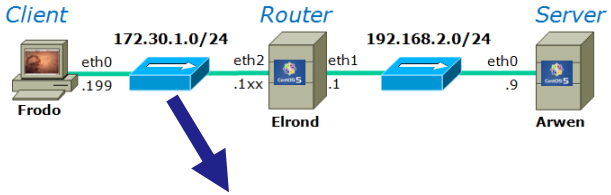




This portion is encrypted



This portion is in clear text



SSH Port Forwarding

Encrypted portion of the connection

(Untitled) - Wireshark

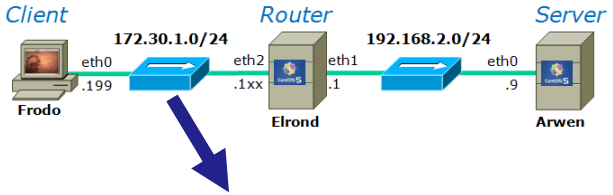
File Edit View Go Capture Analyze Statistics Help

Filter: (ip.addr eq 172.30.4.107 and ip.addr eq 172.30 Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
30	4.479350	172.30.4.199	172.30.4.107	TCP	44022 > ssh [ACK] Seq=561 Ack=625 Win=316 Le
31	4.662263	172.30.4.199	172.30.4.107	SSH	Encrypted request packet len=48
32	4.662313	172.30.4.107	172.30.4.199	SSH	Encrypted response packet len=48
33	4.662325	172.30.4.199	172.30.4.107	TCP	44022 > ssh [ACK] Seq=609 Ack=673 Win=316 Le
34	4.830786	172.30.4.199	172.30.4.107	SSH	Encrypted request packet len=48
35	4.834560	172.30.4.107	172.30.4.199	SSH	Encrypted response packet len=48
36	4.834600	172.30.4.199	172.30.4.107	TCP	44022 > ssh [ACK] Seq=657 Ack=721 Win=316 Le
37	5.581184	172.30.4.199	172.30.4.107	SSH	Encrypted request packet len=48
38	5.586744	172.30.4.107	172.30.4.199	SSH	Encrypted response packet len=48
39	5.588110	172.30.4.199	172.30.4.107	TCP	44022 > ssh [ACK] Seq=705 Ack=769 Win=316 Le
40	5.588788	172.30.4.107	172.30.4.199	SSH	Encrypted response packet len=48
41	5.589934	172.30.4.199	172.30.4.107	TCP	44022 > ssh [ACK] Seq=705 Ack=817 Win=316 Le
42	7.824815	172.30.4.199	172.30.4.107	SSH	Encrypted request packet len=48

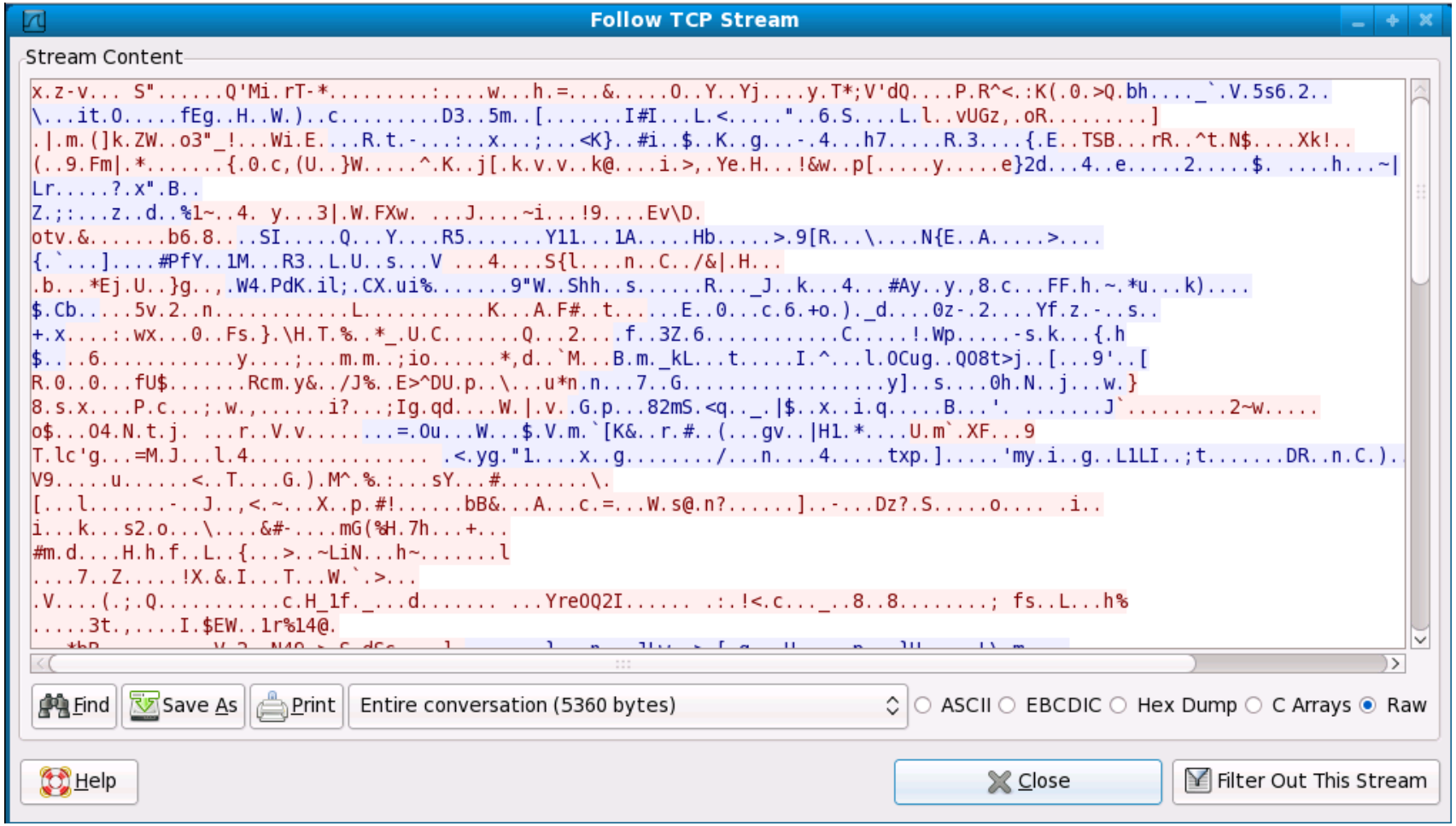
Frame 10 (118 bytes on wire, 118 bytes captured)
Ethernet II, Src: Vmware_4e:21:af (00:0c:29:4e:21:af), Dst: Vmware_6f:53:d9 (00:0c:29:6f:53:d9)
Internet Protocol, Src: 172.30.4.107 (172.30.4.107), Dst: 172.30.4.199 (172.30.4.199)
Transmission Control Protocol, Src Port: ssh (22), Dst Port: 44022 (44022), Seq: 161, Ack: 257, Len: 64
SSH Protocol

Frame (frame), 118 bytes | Packets: 168 Displayed: 168 Marked: 0 Dropped: 0 | Profile: Default

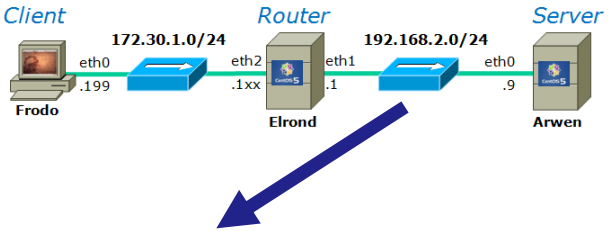


SSH Port Forwarding

Encrypted portion of the connection



SSH Port Forwarding



Clear text portion of the connection

(Untitled) - Wireshark

File Edit View Go Capture Analyze Statistics Help

Filter: Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
6	10.945158	192.168.2.10	192.168.2.9	TCP	35155 > telnet [SYN] Seq=0 Win=5840 Len=0 MS
7	10.945253	192.168.2.9	192.168.2.10	TCP	telnet > 35155 [SYN, ACK] Seq=0 Ack=1 Win=57
8	10.946441	192.168.2.10	192.168.2.9	TCP	35155 > telnet [ACK] Seq=1 Ack=1 Win=5888 Le
9	10.973505	192.168.2.9	192.168.2.10	TELNET	Telnet Data ...
10	10.974504	192.168.2.10	192.168.2.9	TCP	35155 > telnet [ACK] Seq=1 Ack=13 Win=5888 L
11	10.985690	192.168.2.10	192.168.2.9	TELNET	Telnet Data ...
12	10.993869	192.168.2.9	192.168.2.10	TCP	telnet > 35155 [ACK] Seq=13 Ack=13 Win=5824
13	10.994944	192.168.2.9	192.168.2.10	TELNET	Telnet Data ...
14	11.001281	192.168.2.10	192.168.2.9	TELNET	Telnet Data ...
15	11.051578	192.168.2.9	192.168.2.10	TELNET	Telnet Data ...
16	11.055691	192.168.2.10	192.168.2.9	TELNET	Telnet Data ...
17	11.083456	192.168.2.9	192.168.2.10	TELNET	Telnet Data ...
18	11.083690	192.168.2.10	192.168.2.9	TELNET	Telnet Data ...

Internet Protocol, Src: 192.168.2.9 (192.168.2.9), Dst: 192.168.2.10 (192.168.2.10)

Transmission Control Protocol, Src Port: telnet (23), Dst Port: 35155 (35155), Seq: 52, Ack: 104, Len: 69

Telnet

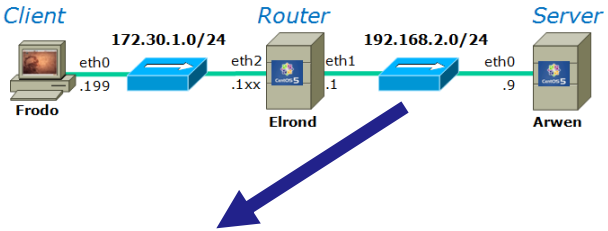
Command: Will Echo

Data: CentOS release 5.2 (Final)\r\n

Data: Kernel 2.6.18-92.1.22.el5 on an i686\r\n

File: "/tmp/etherXXXXruBIW6" 14 ... Packets: 168 Displayed: 168 Marked: 0 Dropped: 0 Profile: Default

SSH Port Forwarding



Clear text portion of the connection

Follow TCP Stream

Stream Content

```
..... ..#..'..... ..#..'.. .....#.....'.......... .38400,38400....#.frodo:0.0....'..DISPLAY.frodo:0.0......xterm..........
release 5.2 (Final)
Kernel 2.6.18-92.1.22.el5 on an i686
...login: cciiss119922
.
Password: Cabrillo
.
Last login: Sun Mar 15 03:48:58 from elrond
.]0;cis192@arwen:~.[cis192@arwen ~]$ eecchhoo  TThhiiss  iiss  aa  sseeccrreett!!
.
This is a secret!
.]0;cis192@arwen:~.[cis192@arwen ~]$ eexxiitt
.
logout
.[H.[2J
```

Find | Save As | Print | Entire conversation (484 bytes) | ASCII | EBCDIC | Hex Dump | C Arrays | Raw

Help | Close | Filter Out This Stream

Netfilter

Using iptables for firewalls and NAT

Examples

Netfilter

(iptables)

Netfilter

Netfilter

- Packet filtering (firewall)
- Port and Address translation (NAT*)
- Logging
- Other types of packet mangling
- Implemented by the iptables utility
- Replaces ipchains in older kernels (2.2 and earlier)

*Note, the term NAT can mean different things. Linux really does PAT which includes both address and port translation. This allows multiple private address to be concurrently translated to a single public IP address.

Netfilter

Firewalls and Access Control Lists
A Firewall is a system that prevents unauthorized network communications to it, from it, and through it.

Netfilter

iptables - chains are grouped into tables:

Filter table:
- Input chain
- Output chain
- Forward chain

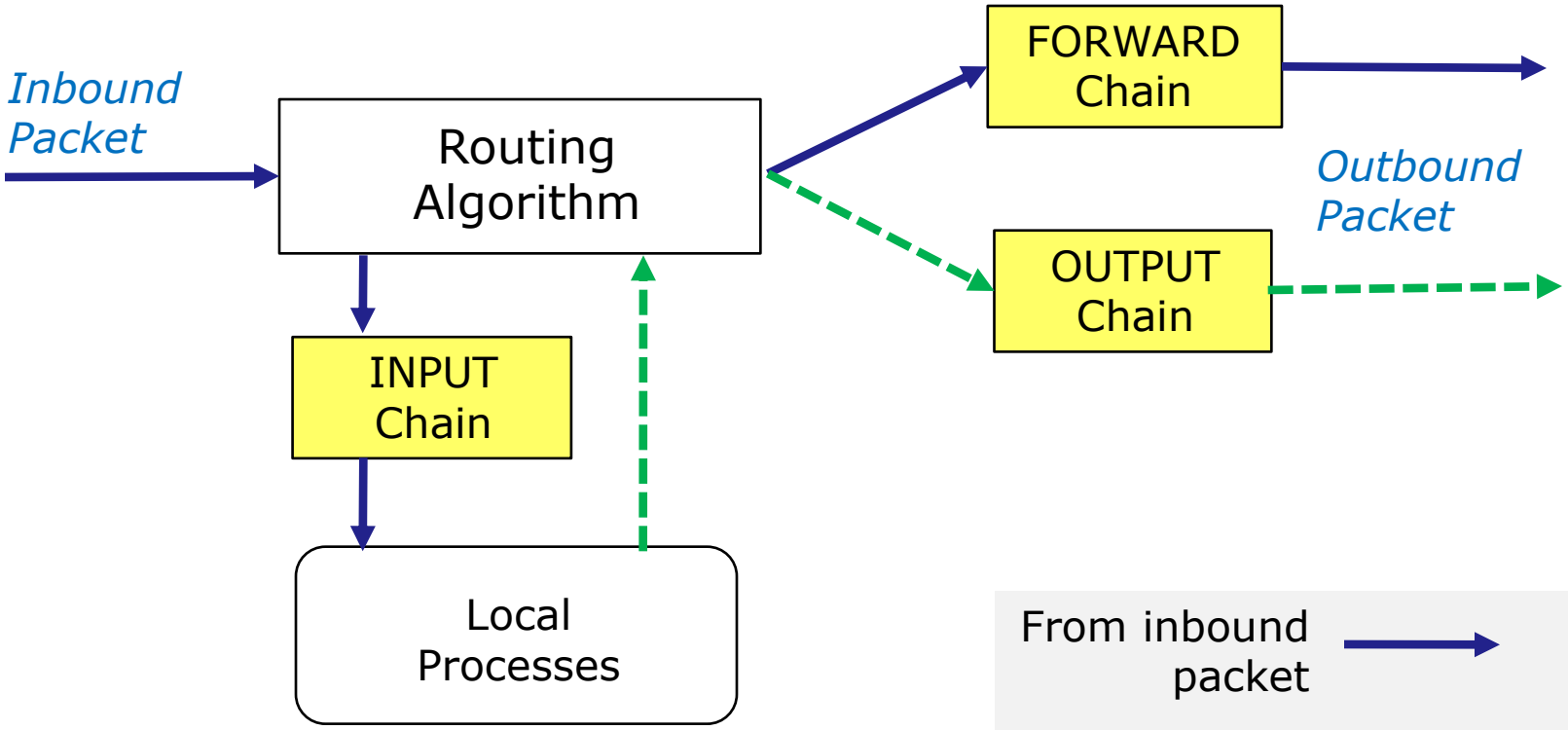
NAT table:
- Prerouting chain
- Output chain
- Postrouting chain

Mangle table:
- Prerouting chain
- Output chain

Each chain can be customized with a set of rules and a default policy to use if none of the rules apply.

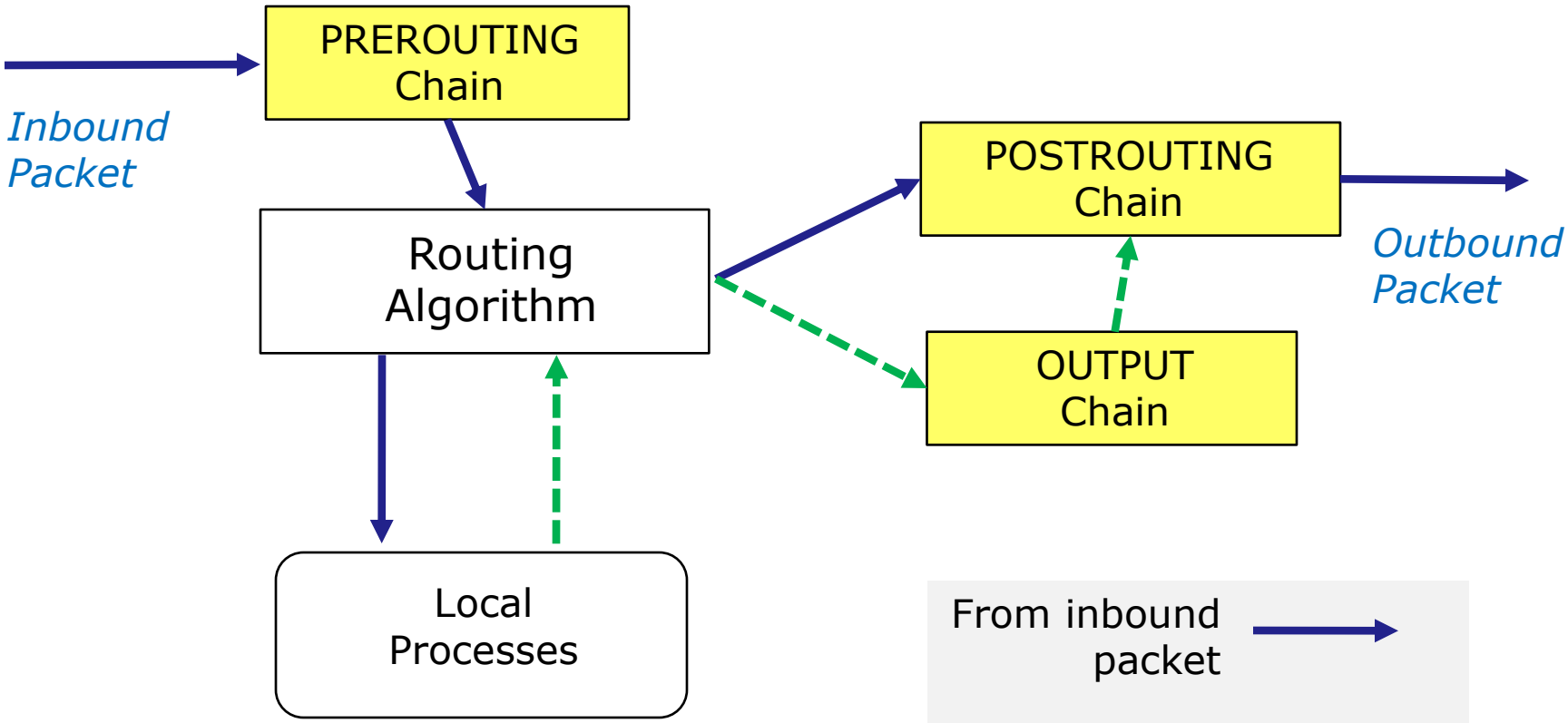
filter table

used to filter incoming, outgoing and forwarded packets



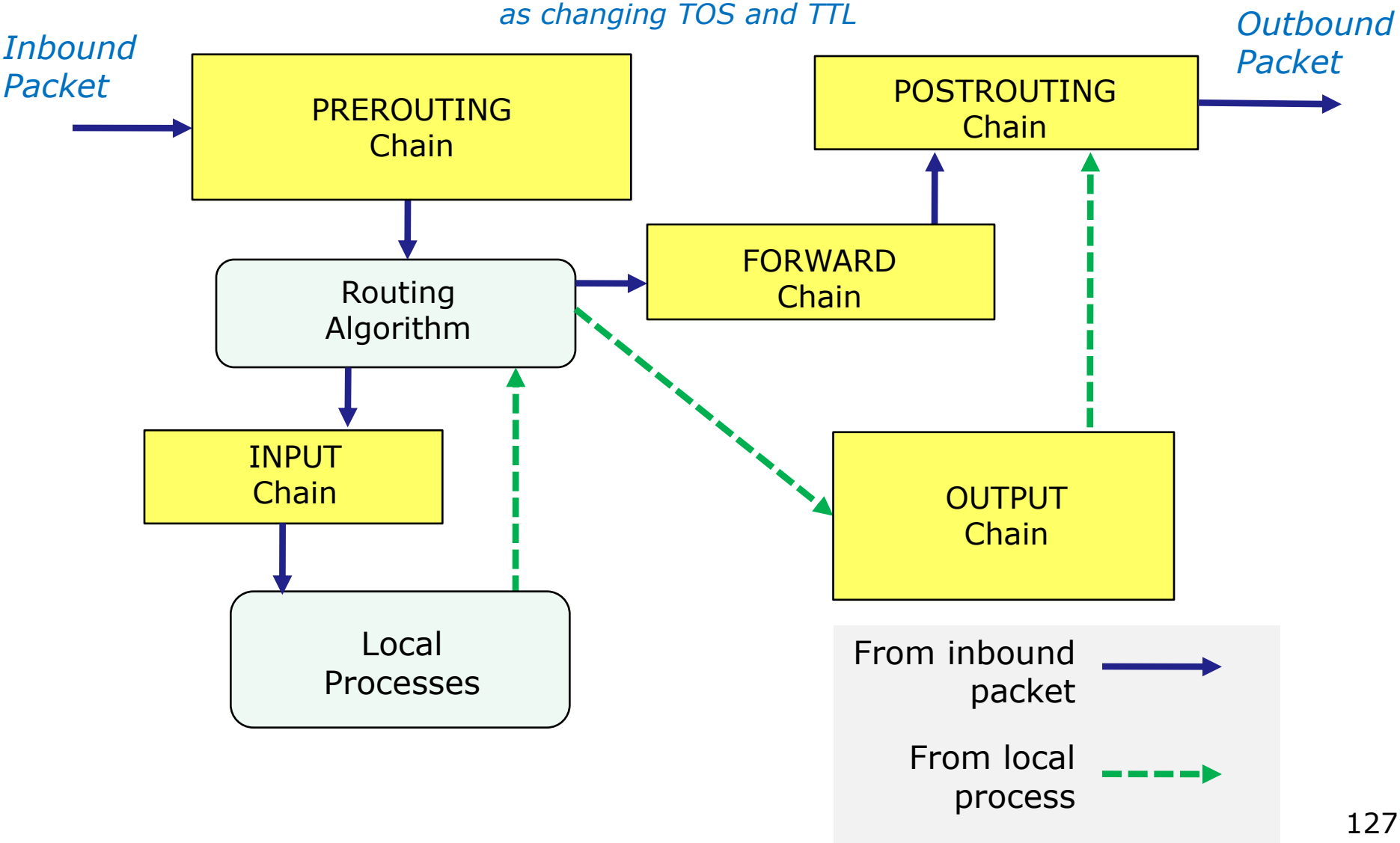
nat table

used to translate source and destination addresses

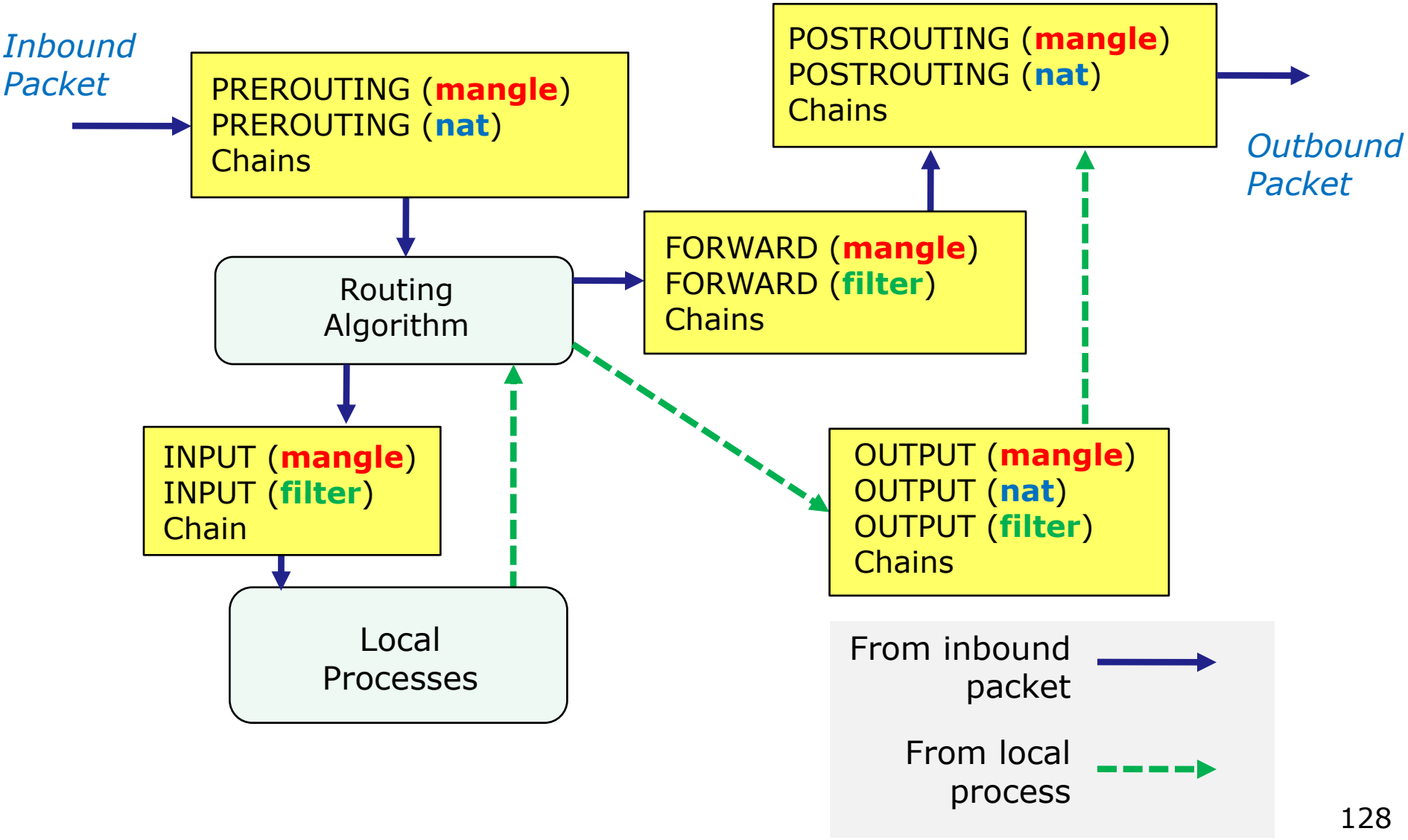


mangle table

specialized packet alteration such as changing TOS and TTL



Netfilter – all tables and chains



Netfilter

iptables command syntax

iptables [-flags] [chain] [options [extensions]] [action]

Netfilter

Flags

-t	specify table (default is filter)
-A	append a rule
-D	delete a rule
-F	flush all rules for a specified chain
-X	delete custom chain
-I	insert a rule at the specified position
-L	list all rules
-P	policy - the default chain rule
-R	replace a rule at specified position

Netfilter

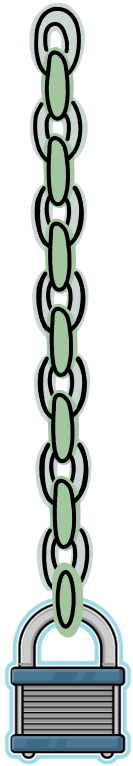
Options

-d destination IP address (accepts CIDR and 0/0 as "all")
-s source IP address (accepts CIDR and 0/0 as "all")
-p protocol - any name listed in */etc/protocols*
-i the inbound interface
-o the outbound interface
-j the target action
-m extended matching module - has many extensions e.g.
-m state --state RELATED,ESTABLISHED

Actions

ACCEPT	*accept packet*
DROP	*drop packet with no error returned to sender*
REJECT	*drop packet with error returned to sender*
LOG	*log packet*
DNAT	*Destination NAT (Network Address Translation)*
SNAT	*Source NAT (Network Address Translation)*

Netfilter – chains



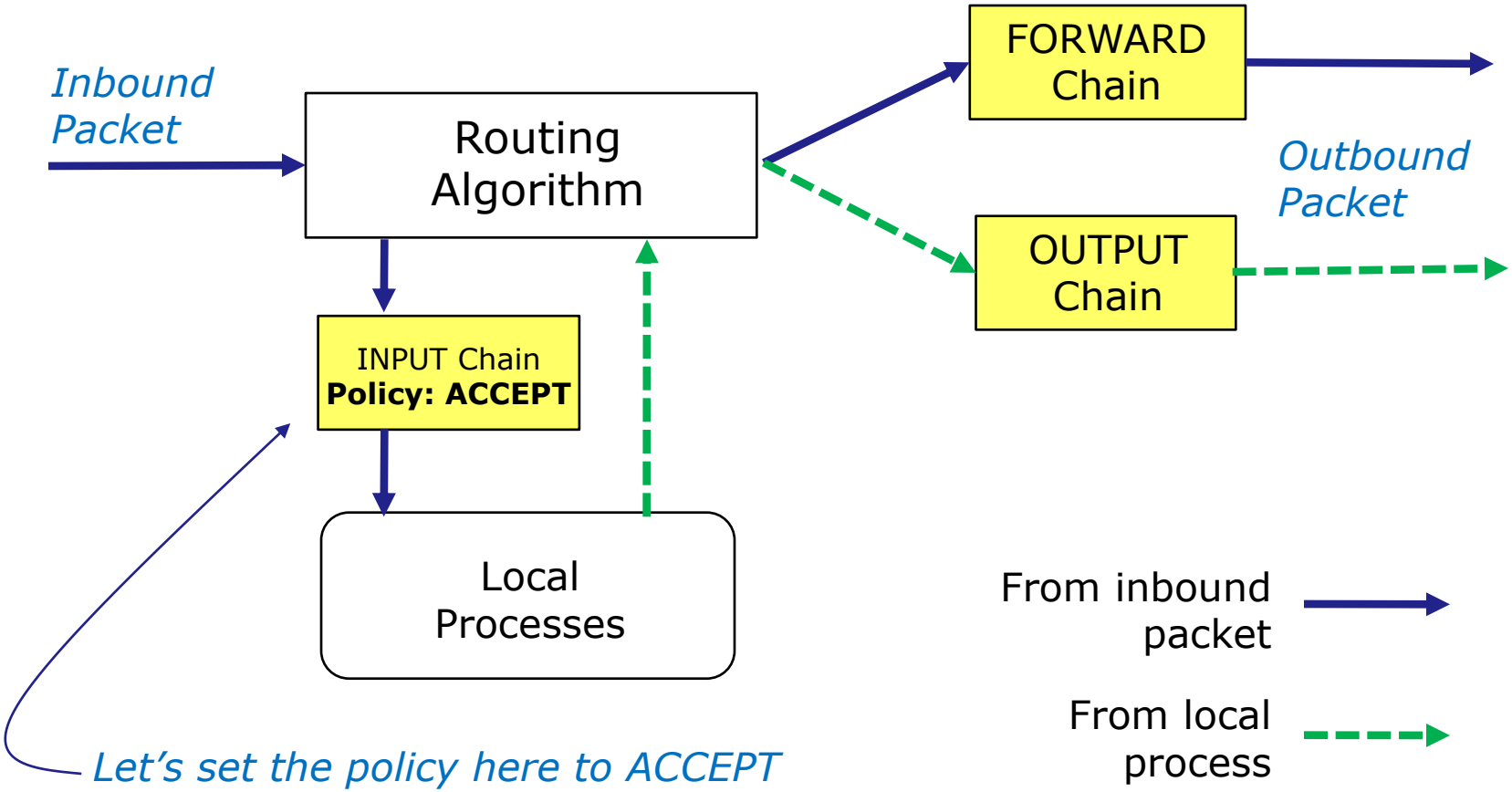
Rules

Policy – the action to take if you get through all the rules on the chain

Table: filter
Chain: INPUT
Policy: ACCEPT

Netfilter – examples

Filter table on Elrond



Netfilter – examples

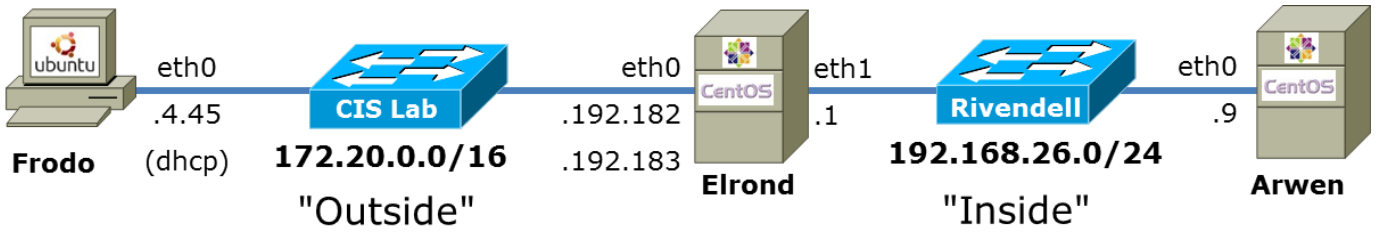
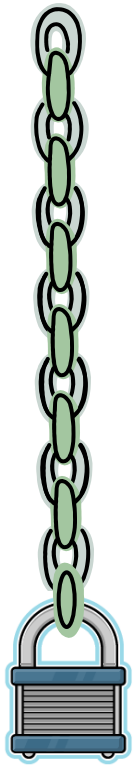


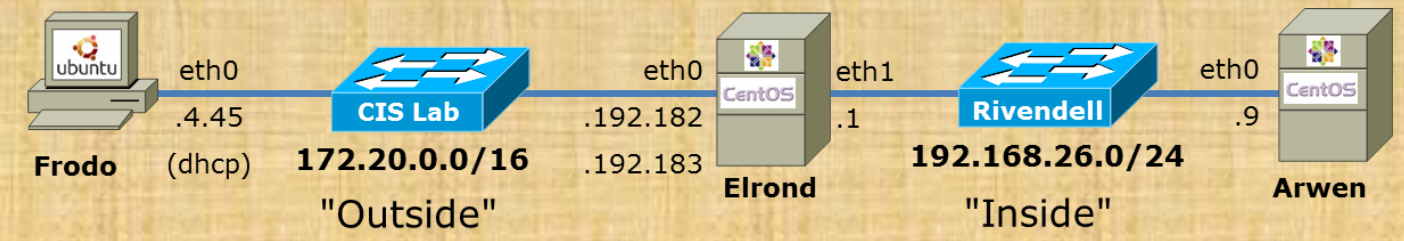
Table: filter
Chain: INPUT



No Rules

Chain Policy: ACCEPT

Netfilter – examples




```
[root@p26-elrond ~]# iptables -F
[root@p26-elrond ~]# iptables -X
[root@p26-elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@p26-elrond ~]# _
```

Flush filter chain rules and delete any custom chains.

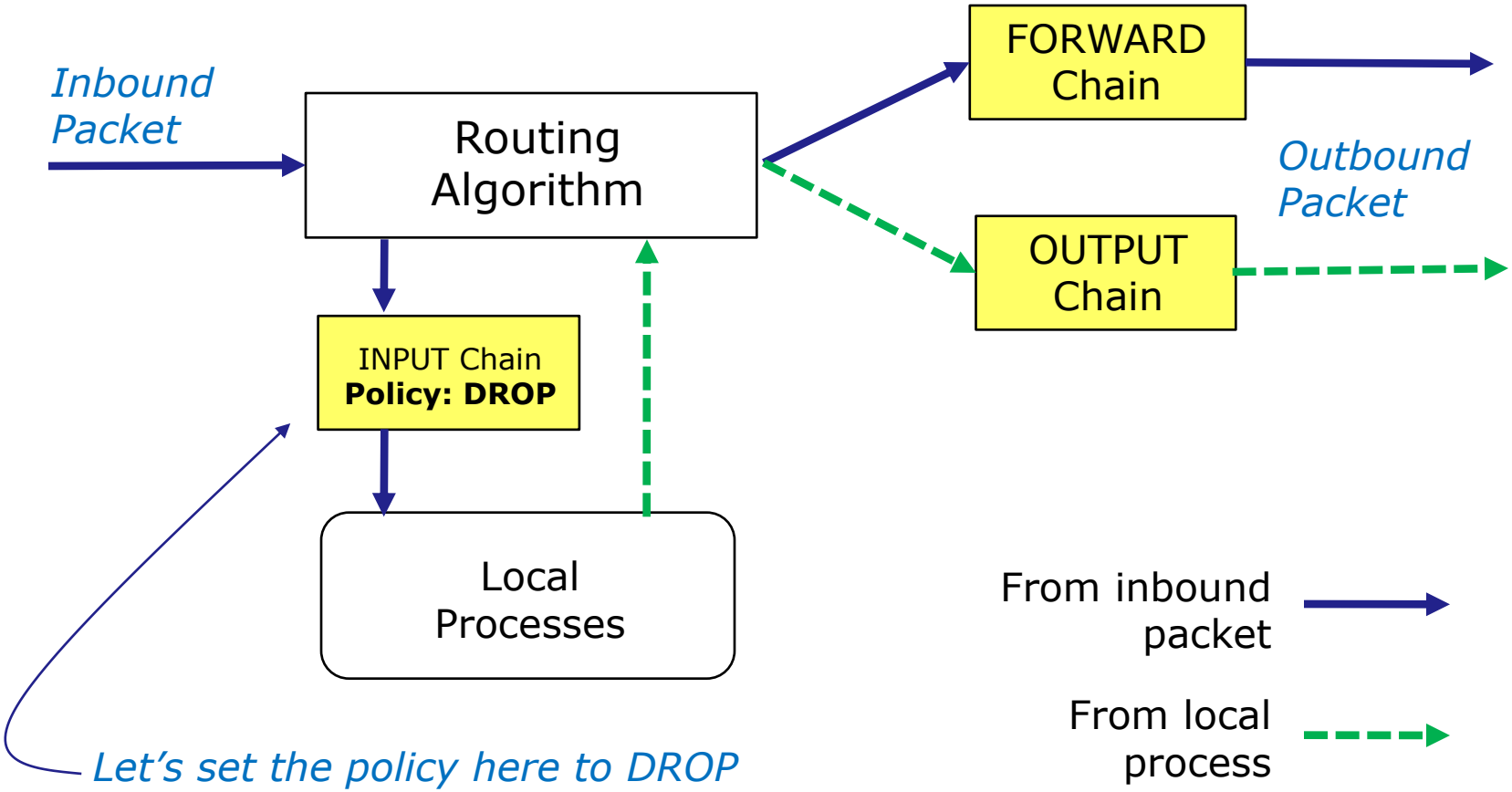
Frodo can ping Elrond

```
cis192@p26-frodo:~$ ping -c1 elrond
PING elrond (172.20.192.182) 56(84) bytes of data.
64 bytes from celebrian (172.20.192.182): icmp_req=1 ttl=64 time=0.709 ms

--- elrond ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
```

Table: filter
Chain: INPUT
Policy: DROP

Netfilter – examples

Filter table on Elrond



Netfilter – examples

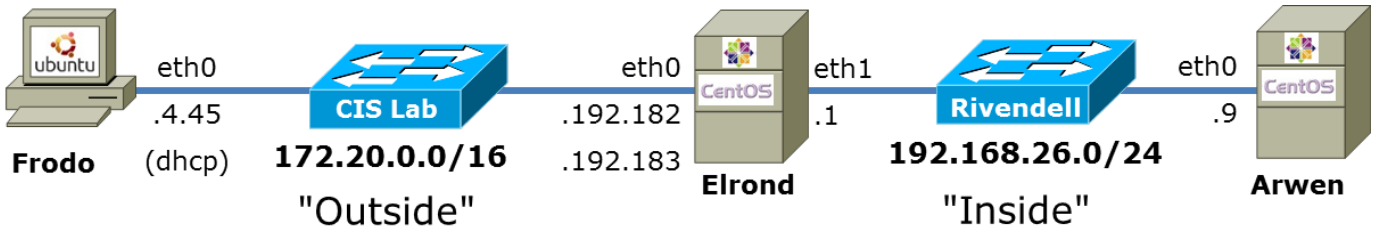
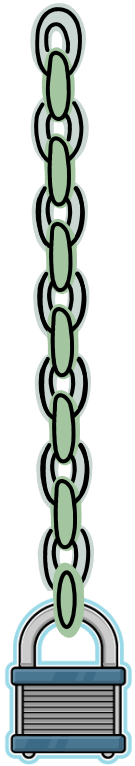


Table: filter
Chain: INPUT

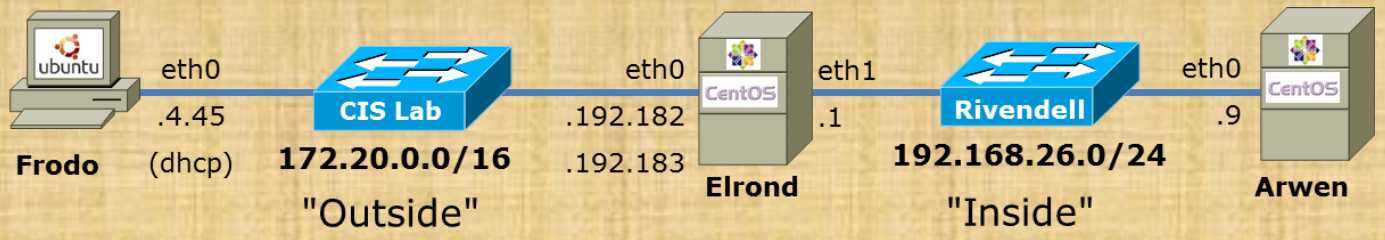


No Rules

Chain Policy: DROP

DROP silently

Netfilter – examples

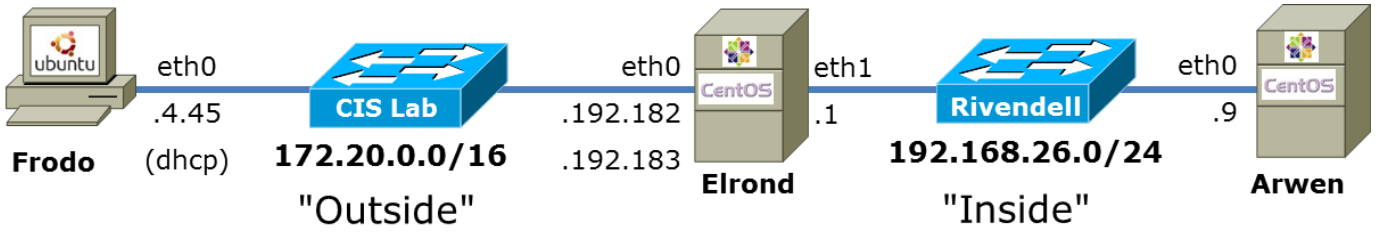


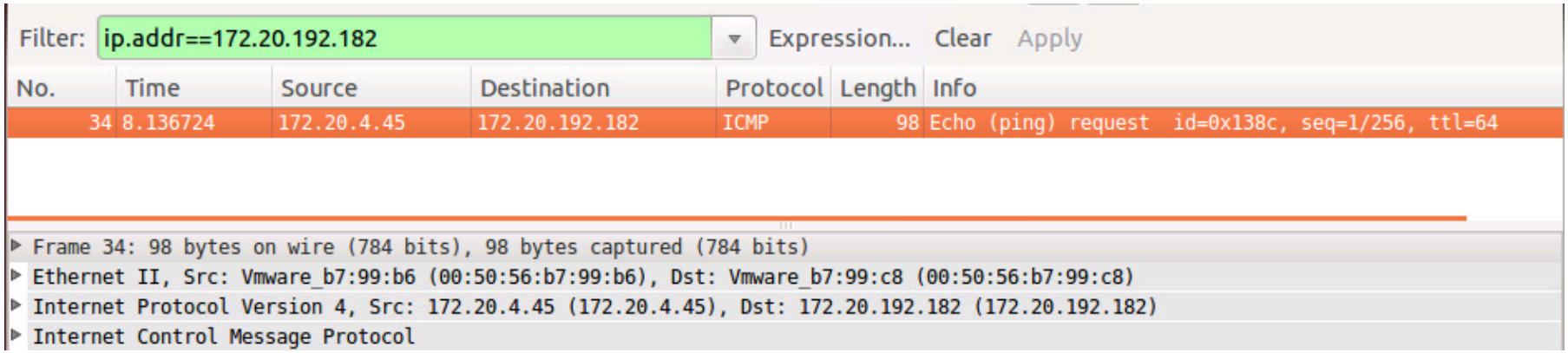

```
[root@p26-elrond ~]# iptables -P INPUT DROP
[root@p26-elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@p26-elrond ~]# _
```

Frodo cannot ping Elrond

```
cis192@p26-frodo:~$ ping -c1 elrond
PING elrond (172.20.192.182) 56(84) bytes of data.

--- elrond ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

Netfilter – examples





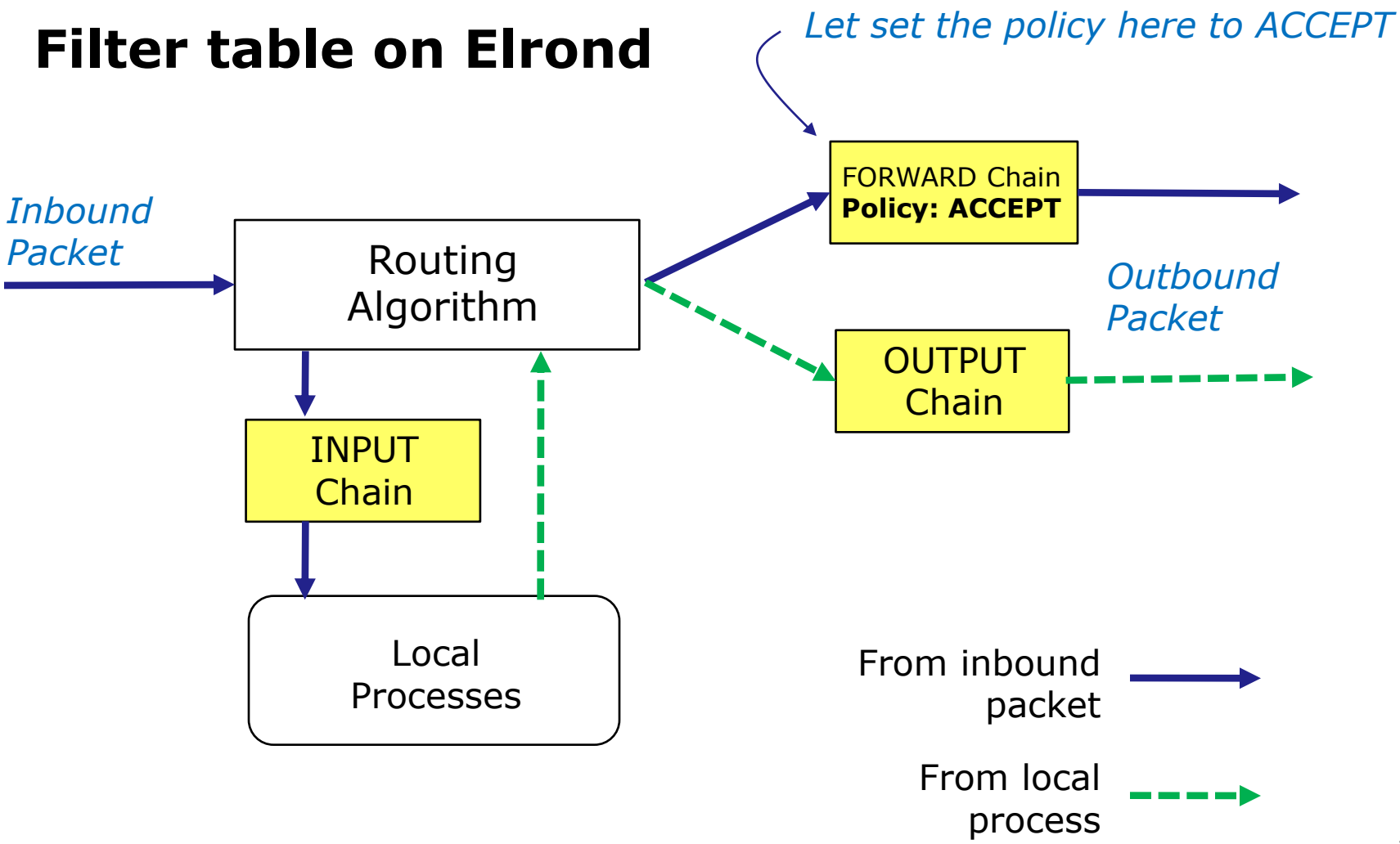
Frodo cannot ping Elrond now

```
cis192@p26-frodo:~$ ping -c1 elrond
PING elrond (172.20.192.182) 56(84) bytes of data.

--- elrond ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

Table: filter
Chain: FORWARD
Policy: ACCEPT

Netfilter – examples

Filter table on Elrond



Netfilter – examples

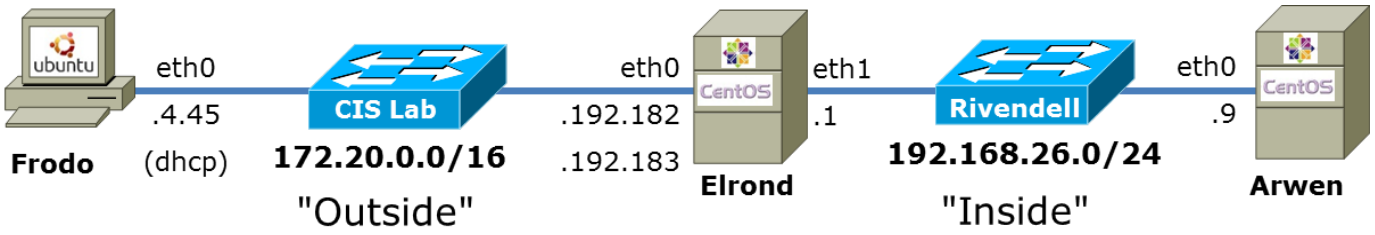
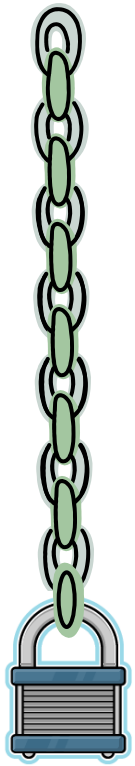


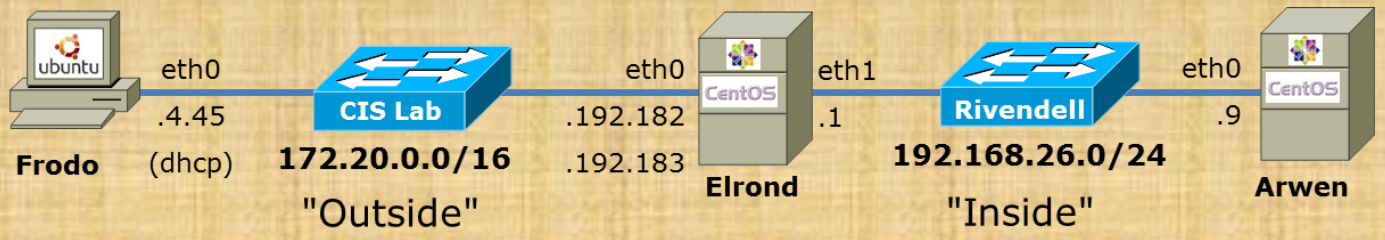
Table: filter
Chain: FORWARD



No Rules

Chain Policy: ACCEPT

Netfilter – examples



Frodo has static route to 192.168.26.0/24 network

```
[root@p26-elrond ~]# iptables -P FORWARD ACCEPT
[root@p26-elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@p26-elrond ~]# _
```

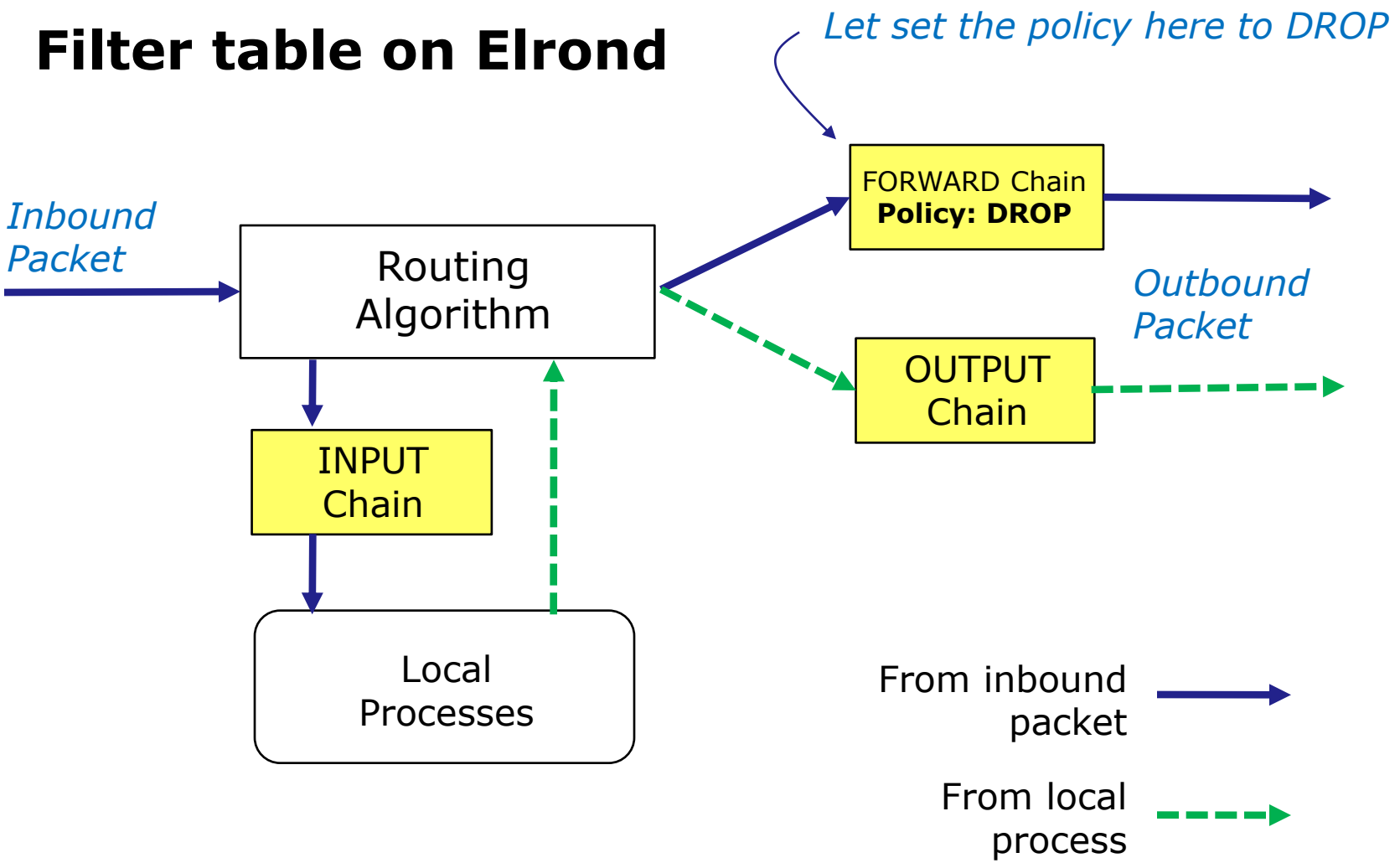
Frodo can ping Arwen via Elrond

```
cis192@p26-frodo:~$ ping arwen -c1
PING arwen (192.168.26.9) 56(84) bytes of data.
64 bytes from arwen (192.168.26.9): icmp_req=1 ttl=63 time=0.785 ms

--- arwen ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
```

Table: filter
Chain: FORWARD
Policy: DROP

Netfilter – examples

Filter table on Elrond



Netfilter – examples

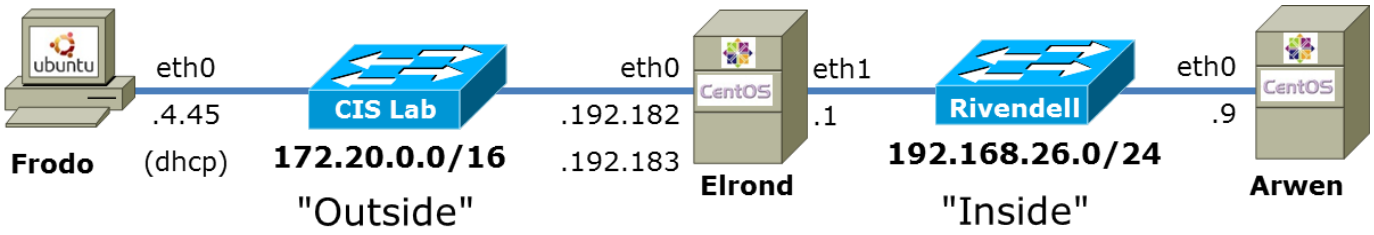
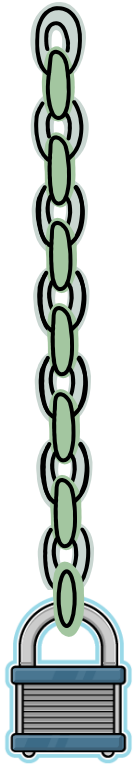


Table: filter
Chain: FORWARD

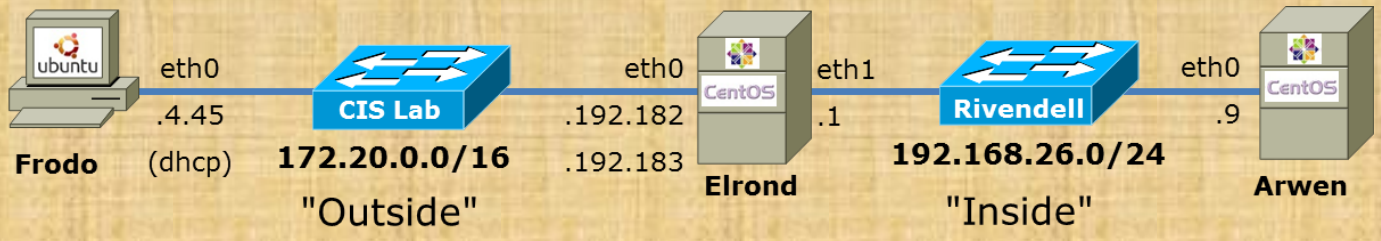


No Rules

Chain Policy: DROP

DROP everything else

Netfilter – examples



Frodo has static route to 192.168.26.0/24 network

```
[root@p26-elrond ~]# iptables -P FORWARD DROP
[root@p26-elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@p26-elrond ~]# _
```

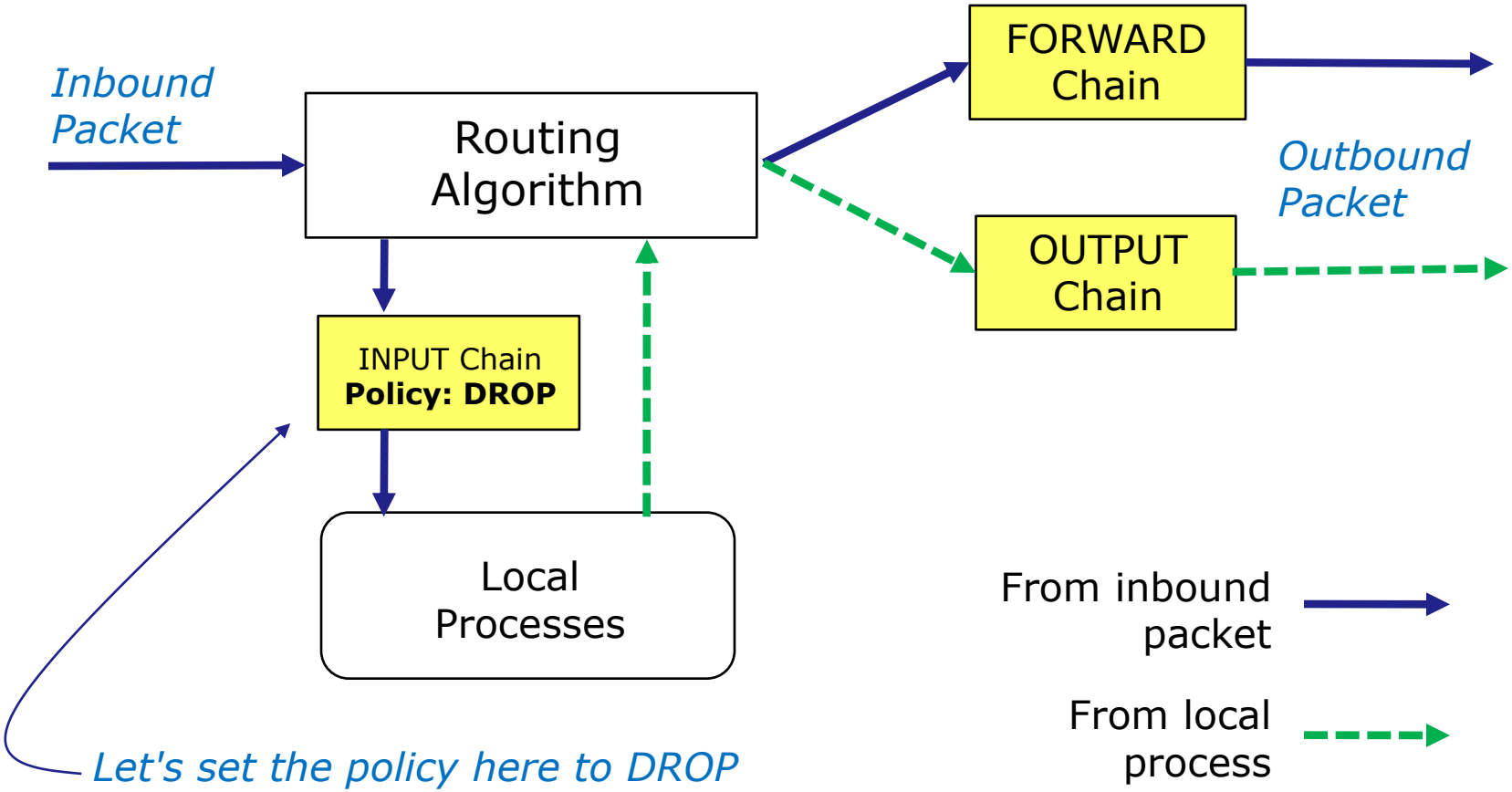
Frodo cannot ping Arwen via Elrond

```
cis192@p26-frodo:~$ ping arwen -c1
PING arwen (192.168.26.9) 56(84) bytes of data.

--- arwen ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

cis192@p26-frodo:~$
```

Table: filter
Chain: INPUT
IP address rules

Netfilter – examples

Filter table on Elrond



Netfilter – examples

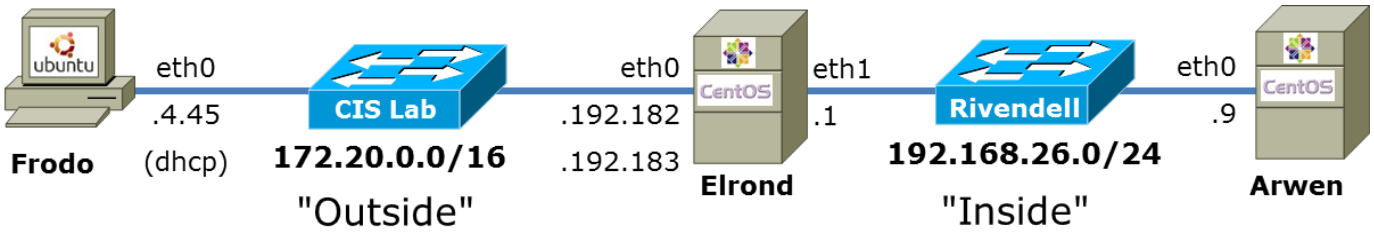
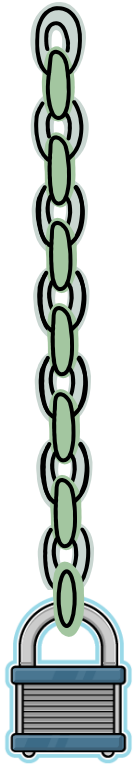


Table: filter
Chain: INPUT



Chain Rules:

-s 172.20.4.48/32 -j REJECT

Reject anything from p27-frodo

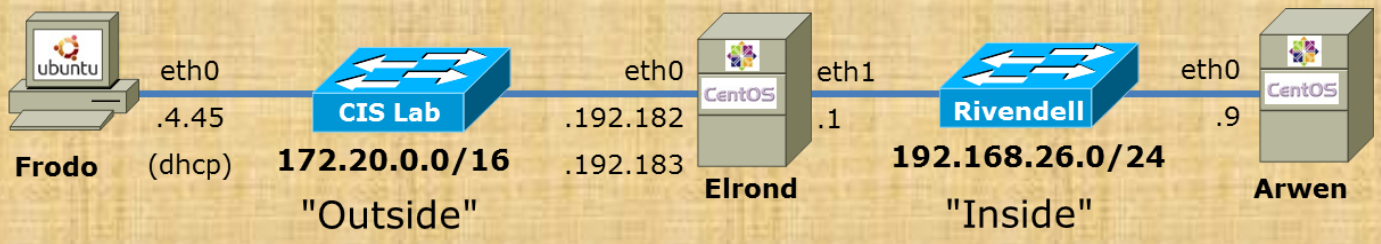
-s 172.20.0.0/16 –j ACCEPT

Accept packets from all other CIS Lab hosts

Chain Policy: DROP

DROP everything else

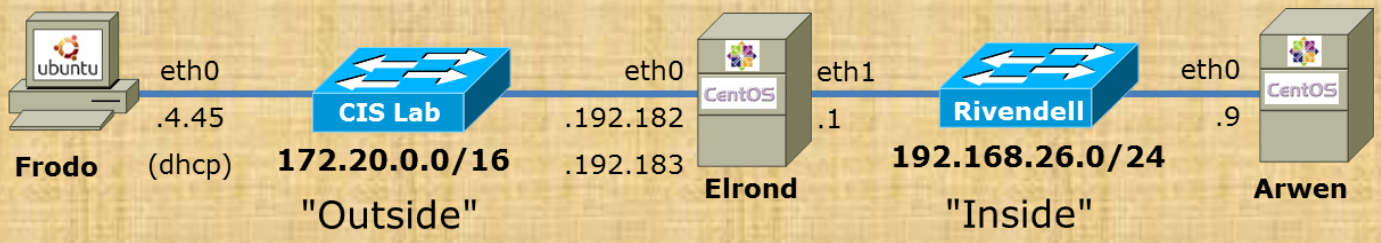
Netfilter – examples




```
[root@p26-elrond ~]# iptables -F
[root@p26-elrond ~]# iptables -A INPUT -s 172.20.4.48/32 -j REJECT
[root@p26-elrond ~]# iptables -A INPUT -s 172.20.0.0/16 -j ACCEPT
[root@p26-elrond ~]# iptables -nL
```

```
[root@p26-elrond ~]# iptables -A INPUT -s 172.20.4.48/32 -j REJECT
[root@p26-elrond ~]# iptables -A INPUT -s 172.20.0.0/16 -j ACCEPT
[root@p26-elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination
REJECT     all  --  172.20.4.48          0.0.0.0/0           reject-with icmp-po
rt-unreachable
ACCEPT     all  --  172.20.0.0/16        0.0.0.0/0

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Netfilter – examples

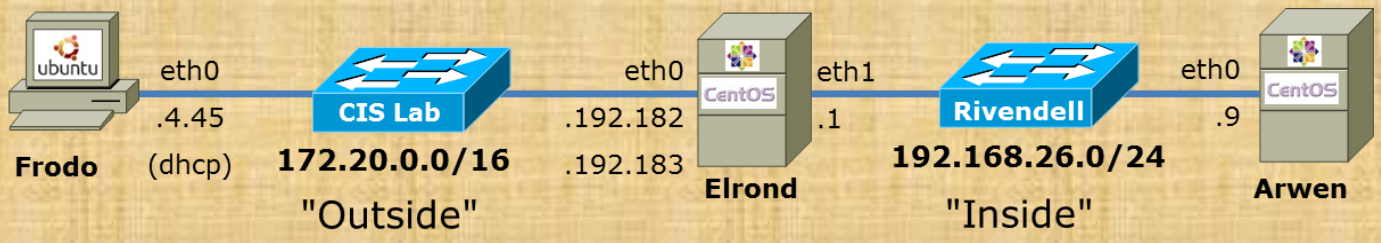



```
Chain INPUT (policy DROP)
target     prot opt source               destination
REJECT     all  --  172.20.4.48          0.0.0.0/0           reject-with icmp-po
rt-unreachable
ACCEPT     all  --  172.20.0.0/16        0.0.0.0/0
```

```
cis192@p26-frodo:~$ ping -c1 172.20.192.182
PING 172.20.192.182 (172.20.192.182) 56(84) bytes of data.
64 bytes from 172.20.192.182: icmp_req=1 ttl=64 time=0.393 ms

--- 172.20.192.182 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.393/0.393/0.393/0.000 ms
cis192@p26-frodo:~$
```

Pings from p26-frodo (172.20.4.45) to Elrond are successful

Netfilter – examples

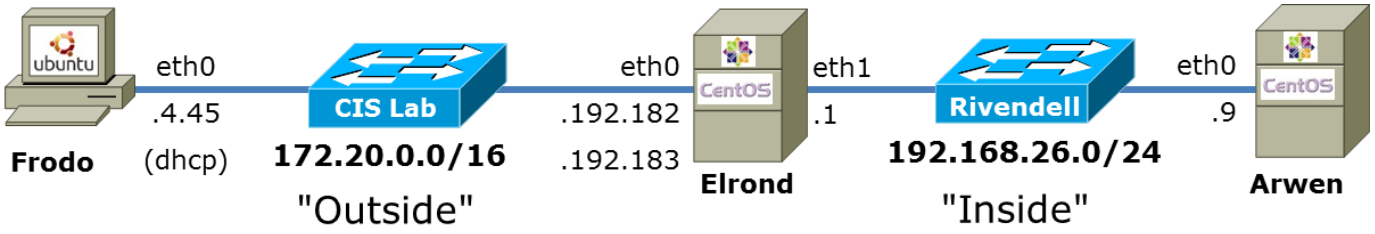



```
Chain INPUT (policy DROP)
target     prot opt source               destination
REJECT     all  --  172.20.4.48          0.0.0.0/0           reject-with icmp-po
rt-unreachable
ACCEPT     all  --  172.20.0.0/16        0.0.0.0/0
```

```
cis192@p27-frodo:~$ ping -c1 172.20.192.182
PING 172.20.192.182 (172.20.192.182) 56(84) bytes of data.
From 172.20.192.182 icmp_seq=1 Destination Port Unreachable

--- 172.20.192.182 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

cis192@p27-frodo:~$
```

Pings from p27-frodo (172.20.4.48) to Elrond are rejected

Netfilter – examples




```
Chain INPUT (policy DROP)
target     prot opt source               destination
REJECT     all  --  172.20.4.48          0.0.0.0/0           reject-with icmp-po
rt-unreachable
ACCEPT     all  --  172.20.0.0/16        0.0.0.0/0
```

```
cis192@p27-frodo:~$ ping -c1 172.20.192.182
PING 172.20.192.182 (172.20.192.182) 56(84) bytes of data.
From 172.20.192.182 icmp_seq=1 Destination Port Unreachable

--- 172.20.192.182 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

cis192@p27-frodo:~$
```

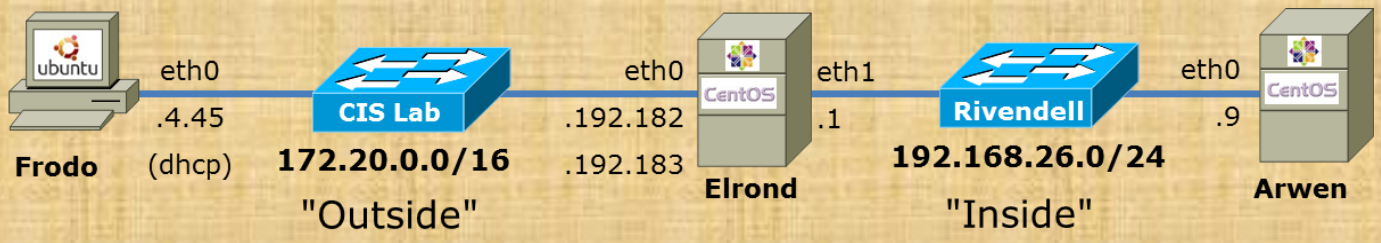
Filter: ip.addr==172.20.192.182 Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
52	18.340526	172.20.4.48	172.20.192.182	ICMP	98	Echo (ping) request id=0x14c2, seq=1/256, ttl=64
53	18.340871	172.20.192.182	172.20.4.48	ICMP	126	Destination unreachable (Port unreachable)

Pings from p27-frodo (172.20.4.48) to Elrond are rejected resulting in an ICMP error packet sent back to Frodo

Some Fun Stuff

Netfilter – examples

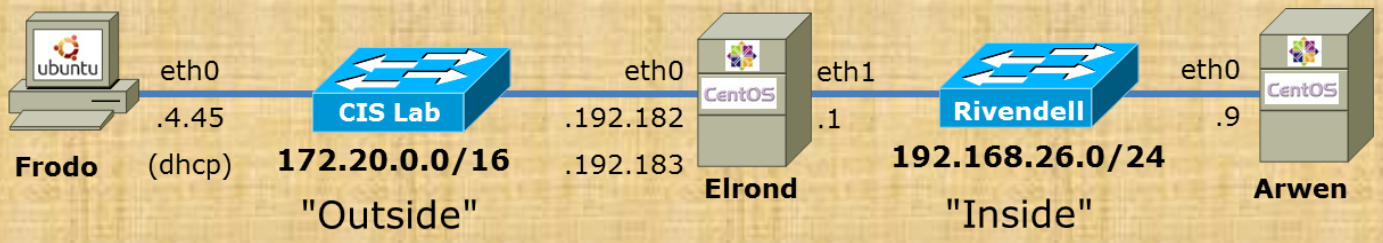


iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A FORWARD -s 192.168.26.0/24 -m state --state NEW -j ACCEPT

```
[root@p26-elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTAB
LISHED
ACCEPT     all  --  192.168.26.0/24      0.0.0.0/0           state NEW

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

What does this FORWARD chain allow?

Netfilter – examples



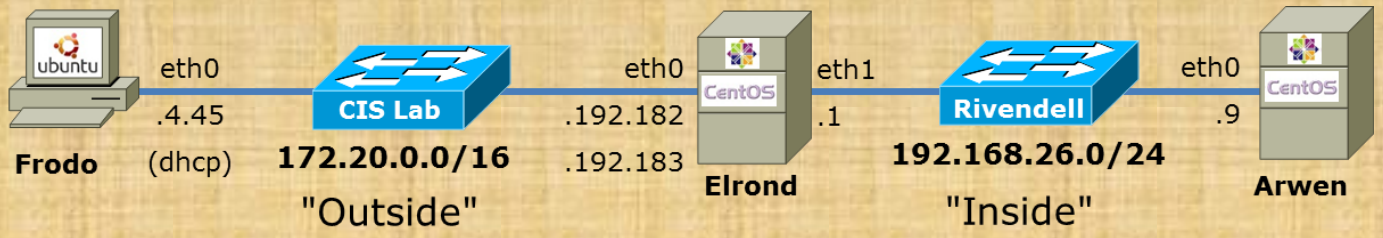

```
[root@p26-elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTAB
LISHED
ACCEPT     all  --  192.168.26.0/24      0.0.0.0/0           state NEW

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Arwen can ping Frodo but not the other way around

```
cis192@p26-frodo:~$ ping arwen -c2
PING arwen (192.168.26.9) 56(84)
bytes of data.

--- arwen ping statistics ---
2 packets transmitted, 0 received,
100% packet loss, time 1008ms
```

```
[cis192@p26-arwen ~]$ ping frodo -c1
PING frodo (172.20.4.45) 56(84) bytes of data.
64 bytes from frodo (172.20.4.45): icmp_seq=1
ttl=63 time=0.455 ms

--- frodo ping statistics ---
1 packets transmitted, 1 received, 0% packet loss,
time 0ms
rtt min/avg/max/mdev = 0.455/0.455/0.455/0.000 ms
```

Netfilter – examples



iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A FORWARD -s 192.168.26.0/24 -m state --state NEW -j ACCEPT

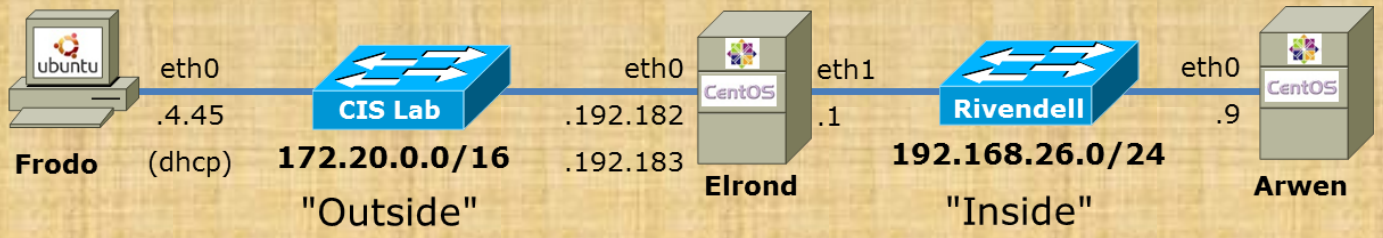
```
[root@p26-elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTAB
LISHED
ACCEPT     all  --  192.168.26.0/24      0.0.0.0/0           state NEW

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

iptables -t nat -A POSTROUTING -s 192.168.26.0/24 -o eth0 -j SNAT --to-source 172.20.192.182

```
[root@p26-elrond ~]# iptables -nL -t nat
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
SNAT       all  --  192.168.26.0/24      0.0.0.0/0           to:172.20.192.182

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

What does this allow Rivendell hosts to do?

Netfilter – examples




```
iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A FORWARD -s 192.168.26.0/24 -m state --state NEW -j ACCEPT
iptables -t nat -A POSTROUTING -s 192.168.26.0/24 -o eth0 -j SNAT --to-source 172.20.192.182
```

```
[root@p26-arwen ~]# ping google.com
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_seq=1 ttl=54 time=6.03 ms
64 bytes from nuq04s09-in-f7.1e100.net (74.125.224.135): icmp_seq=2 ttl=54 time=5.82 ms
64 bytes from nuq04s09-in-f7.1e100.net (74.125.224.135): icmp_seq=3 ttl=54 time=5.79 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2435ms
rtt min/avg/max/mdev = 5.792/5.885/6.036/0.124 ms
```

It gives Rivendell hosts Internet access via NAT

Firewall operations

Managing Red Hat Firewall

Show what is currently in memory

iptables -L -t filter (or just **iptables -L**)
iptables -L -t nat
iptables -L -t mangle

Show the permanent settings which will be used at next system boot

cat /etc/sysconfig/iptables

Managing Red Hat Firewall

Backup the permanent settings

```
[root@elrond ~]# cp /etc/sysconfig/iptables /etc/sysconfig/iptables.bak
```

Save the current firewall and NAT settings for use at next reboot

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

Start using the rules saved in /etc/sysconfig/iptables

```
[root@elrond ~]# service iptables restart
iptables: Flushing firewall rules:                         [  OK  ]
iptables: Setting chains to policy ACCEPT: nat filter      [  OK  ]
iptables: Unloading modules:                               [  OK  ]
iptables: Applying firewall rules:                         [  OK  ]
[root@elrond ~]#
```

Just like IP addresses and static routes we can set firewall and NAT rules temporarily (in memory) or permanently (in a file on the hard drive).

Managing Red Hat Firewall

Save "active" rules in memory to file

```
[root@p26-elrond ~]# iptables-save > cmds
[root@p26-elrond ~]# cat cmds
# Generated by iptables-save v1.4.7 on Tue Mar 19 08:35:55 2013
*nat
:PREROUTING ACCEPT [35:5932]
:POSTROUTING ACCEPT [4:312]
:OUTPUT ACCEPT [1:60]
COMMIT
# Completed on Tue Mar 19 08:35:55 2013
# Generated by iptables-save v1.4.7 on Tue Mar 19 08:35:55 2013
*filter
:INPUT ACCEPT [5379:6293851]
:FORWARD DROP [9:756]
:OUTPUT ACCEPT [3177:215732]
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.26.0/24 -m state --state NEW -j ACCEPT
COMMIT
# Completed on Tue Mar 19 08:35:55 2013
```

Managing Red Hat Firewall

Restore rules in file to memory and make active

```
[root@p26-elrond ~]# iptables-restore < cmds

[root@p26-elrond ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state
RELATED,ESTABLISHED
ACCEPT     all  --  192.168.26.0/24      anywhere            state NEW

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@p26-elrond ~]#
```

"previous" Red Hat default

Default "previous" Red Hat Firewall

Current settings

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

Default "previous" Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

The three standard filter chains and one custom chain

Default "previous" Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

The policy on the three filter chains is ACCEPT.

The policy is the final rule in the chain and is used when no other rules in the chain apply.

Default "previous" Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

The INPUT and FORWARD filter chains have no rules of their own, they will use the rules in same custom chain named RH-Firewall-1-INPUT

Default "previous" Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

Accept all traffic that arrives on the loopback interface.

Its not obvious from this output but the details can be seen in /etc/sysconfig/iptables

Default Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

All ICMP protocol traffic (of any type) is allowed.

Default Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

All ESP and AH protocol traffic is allowed.

ESP (Encapsulating Security Payload) and AH (Authentication Header) are used for IPsec.

Default "previous" Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

All multicast DNS traffic to port 5353 is allowed.

This is used with zeroconf (Zero configuration networking) to locate DNS services on small LANs .

Default "previous" Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

All UDP and TCP protocol traffic to port 631 is allowed.

This allows CUPS to listen for IPP (Internet Printing Protocol) requests.

Default Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

Any traffic whose connection was locally originated or related to that connection is allowed

Default Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

Any new incoming connections to port 22 (ssh) are allowed

Default Red Hat Firewall

```
[root@elrond ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RH-Firewall-1-INPUT  all  --  0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0           icmp type 255
ACCEPT     esp  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     ah   --  0.0.0.0/0            0.0.0.0/0
ACCEPT     udp  --  0.0.0.0/0            224.0.0.251         udp dpt:5353
ACCEPT     udp  --  0.0.0.0/0            0.0.0.0/0           udp dpt:631
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           tcp dpt:631
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited
[root@elrond ~]#
```

If any of the previous rules did not apply, then send an error back using ICMP

Default "previous" Red Hat Firewall

```
[root@elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.3.5 on Wed Mar 17 12:04:26 2010
*nat
:PREROUTING ACCEPT [1:94]
:POSTROUTING ACCEPT [6:994]
:OUTPUT ACCEPT [6:994]
COMMIT
# Completed on Wed Mar 17 12:04:26 2010
# Generated by iptables-save v1.3.5 on Wed Mar 17 12:04:26 2010
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [34:7149]
:RH-Firewall-1-INPUT - [0:0]
-A INPUT -j RH-Firewall-1-INPUT
-A FORWARD -j RH-Firewall-1-INPUT
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type any -j ACCEPT
-A RH-Firewall-1-INPUT -p esp -j ACCEPT
-A RH-Firewall-1-INPUT -p ah -j ACCEPT
-A RH-Firewall-1-INPUT -d 224.0.0.251 -p udp -m udp --dport 5353 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m udp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
# Completed on Wed Mar 17 12:04:26 2010
[root@elrond ~]#
```

Permanent settings to be used at next system boot or when restarting the iptables service

Shows the actual iptables commands used to create the firewall

"new" Red Hat default

Default "new" Red Hat Firewall

Current settings

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

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

Default "new" Red Hat Firewall

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

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

- *Much simpler than version on older version of Red Hat.*
- *The custom RH-Firewall-1-INPUT chain is no longer used.*
- *The policy is still set to ACCEPT on all three filter table chains.*

Default "new" Red Hat Firewall

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

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

Any traffic related to connections that originated on this system are accepted.

/etc/sysconfig/iptables
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT

Default "new" Red Hat Firewall

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

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

Accept any pings

/etc/sysconfig/iptables
-A INPUT -p icmp -j ACCEPT

Default "new" Red Hat Firewall

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

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

Accept all loopback traffic.

/etc/sysconfig/iptables
-A INPUT -i lo -j ACCEPT

Default "new" Red Hat Firewall

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

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

Accept all new ssh connections

/etc/sysconfig/iptables
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT

Default "new" Red Hat Firewall

```
[root@elrond ~]# iptables -nL
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp dpt:22
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with icmp-host-
prohibited

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

Reject with a prohibited error anything else

/etc/sysconfig/iptables
-A FORWARD -j REJECT --reject-with icmp-host-prohibited

Nosmo RH9 VM

Nosmo RH9 Firewall

```
[root@nosmo root]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.2.7a on Mon Jan 11 12:14:00 2010
*filter
:INPUT ACCEPT [4229:434875]
:FORWARD ACCEPT [1481:444016]
:OUTPUT ACCEPT [3340:350240]
COMMIT
# Completed on Mon Jan 11 12:14:00 2010
# Generated by iptables-save v1.2.7a on Mon Jan 11 12:14:00 2010
*nat
:PREROUTING ACCEPT [8414:1265541]
:POSTROUTING ACCEPT [226:15381]
:OUTPUT ACCEPT [95:7826]
-A PREROUTING -d 207.62.187.53 -j DNAT --to-destination 192.168.0.1
-A POSTROUTING -o eth0 -j MASQUERADE
COMMIT
# Completed on Mon Jan 11 12:14:00 2010
[root@nosmo root]#
```

I used to use this VM at home to simulate the lab router

Nosmo RH9 Firewall

```
[root@nosmo root]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.2.7a on Mon Jan 11 12:14:00 2010
*filter
:INPUT ACCEPT [4229:434875]
:FORWARD ACCEPT [1481:444016]
:OUTPUT ACCEPT [3340:350240]
COMMIT
# Completed on Mon Jan 11 12:14:00 2010
# Generated by iptables-save v1.2.7a on Mon Jan 11 12:14:00 2010
*nat
:PREROUTING ACCEPT [8414:1265541]
:POSTROUTING ACCEPT [226:15381]
:OUTPUT ACCEPT [95:7826]
-A PREROUTING -d 207.62.187.53 -j DNAT --to-destination 192.168.0.1
-A POSTROUTING -o eth0 -j MASQUERADE
COMMIT
# Completed on Mon Jan 11 12:14:00 2010
[root@nosmo root]#
```

This is the DNS server IP address I use at home which goes to my Netgear router

Forward DNS traffic intended for Bubbles (Cabrillo DNS server) to my DNS server used at home

Nosmo RH9 Firewall

```
[root@nosmo root]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.2.7a on Mon Jan 11 12:14:00 2010
*filter
:INPUT ACCEPT [4229:434875]
:FORWARD ACCEPT [1481:444016]
:OUTPUT ACCEPT [3340:350240]
COMMIT
# Completed on Mon Jan 11 12:14:00 2010
# Generated by iptables-save v1.2.7a on Mon Jan 11 12:14:00 2010
*nat
:PREROUTING ACCEPT [8414:1265541]
:POSTROUTING ACCEPT [226:15381]
:OUTPUT ACCEPT [95:7826]
-A PREROUTING -d 207.62.187.53 -j DNAT --to-destination 192.168.0.1
-A POSTROUTING -o eth0 -j MASQUERADE
COMMIT
# Completed on Mon Jan 11 12:14:00 2010
[root@nosmo root]#
```

NAT all outgoing traffic to the public IP address on Nosmo. This gives CIS Lab hosts Internet access

Opus Firewall Brute force attacks

/var/log/wtmp and var/log/btmp

```
[root@opus ~]# lastb | grep "cool.nju.edu.cn" | head
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)
bind     ssh:notty    cool.nju.edu.cn  Sun Nov 30 06:35 - 06:35  (00:00)

[root@opus ~]# lastb | grep "cool.nju.edu.cn" | wc -l
3104
[root@opus ~]#
```

Shows break in attempt on 11/30/2008

/var/log/wtmp and var/log/btmp

```
[root@opus ~]# lastb | grep "Nov  2 17:45"
webadmin ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
webadmin ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
retsu    ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
retsu    ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
sbear    ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
sbear    ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
sky      ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
sky      ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
harvey   ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
harvey   ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
harvey   ssh:notty    211.96.97.179    Sun Nov  2 17:45 - 17:45  (00:00)
[root@opus ~]#

[root@opus ~]# lastb -i | grep "211.96.97.179" | wc -l
598
[root@opus ~]#
```

Shows break in attempt by 211.96.97.179 on 11/2/2008

/var/log/wtmp and var/log/btmp

```
[root@opus log]# lastb | sort | cut -f1 -d' ' | grep -v ^$ | uniq -c > bad
[root@opus log]# sort -g bad > bad.sort
[root@opus log]# cat bad.sort | tail -50
    471 ftp
    472 public
    490 test
    490 tomcat
    498 user
    506 service
    508 mike
    508 username
    524 cyrus
    530 pgsql
    532 test1
    544 master
    554 linux
    554 toor
    576 paul
    584 support
    590 testuser
    604 irc
    610 test
    656 noc
    686 www
    690 postfix
    723 john
    734 testing
    738 adam
    746 alex
    754 info
    798 tester
    832 library
    935 guest
    990 admin
   1002 office
   1022 temp
   1070 ftpuser
   1138 webadmin
   1298 nagios
   1332 web
   1374 a
   1384 student
   1416 postgres
   1690 user
   1858 oracle
   1944 mysql
   2086 webmaste
   5324 test
  10803 root
  10824 admin
  18679 root
  24064 root
[root@opus log]#
```

Top 50 usernames used by the bad guys in 2008

/var/log/wtmp and var/log/btmp

22128 usernames used and failed

```
[root@opus log]# lastb | sort | cut -f1 -d' ' | grep -v ^$| uniq -c | wc -l
22128
[root@opus log]#
```

53117 failed root logins

```
[root@opus log]# lastb | grep root | wc -l
  54117
[root@opus log]#
```

Now you know why you need a strong password!

Impeding brute force attacks

Adds the current IP address to the recent list using the recent module

```
[rsimms@opus ~]$ cat /etc/sysconfig/iptables
< snipped >
# Impede brute force SSH dictionary attacks using the recent module (Rule added by RJS)
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --set --
name SSHBF --rsource
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --update -
-seconds 60 --hitcount 5 --rttl --name SSHBF --rsource -j LOG --log-prefix "iptables
brute force block: " --log-level 6
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --update -
-seconds 60 --hitcount 5 --rttl --name SSHBF --rsource -j DROP < snipped >
[rsimms@opus ~]$
```

http://kevin.vanzonneveld.net/techblog/article/block_brute_force_attacks_with_iptables/

Impeding brute force attacks

If five packets were sent from the same IP address in the last 60 seconds then log the packet.

```
[rsimms@opus ~]$ cat /etc/sysconfig/iptables
< snipped >
# Impede brute force SSH dictionary attacks using the recent module (Rule added by RJS)
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --set --
name SSHBF --rsource
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --update -
-seconds 60 --hitcount 5 --rttl --name SSHBF --rsource -j LOG --log-prefix "iptables
brute force block: " --log-level 6
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --update -
-seconds 60 --hitcount 5 --rttl --name SSHBF --rsource -j DROP < snipped >
[rsimms@opus ~]$
```

Impeding brute force attacks

If five packets were sent from the same IP address in the last 60 seconds then drop the packet.

```
[rsimms@opus ~]$ cat /etc/sysconfig/iptables
< snipped >
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --set --
name SSHBF --rsource
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --update -
-seconds 60 --hitcount 5 --rttl --name SSHBF --rsource -j LOG --log-prefix "iptables
brute force block: " --log-level 6
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 22 -m state --state NEW -m recent --update -
-seconds 60 --hitcount 5 --rttl --name SSHBF --rsource -j DROP
< snipped >
[rsimms@opus ~]$
```

Impeding brute force attacks

```
[root@opus ~]# cat /var/log/messages | grep brute
< snipped >
Mar 14 11:32:56 Opus kernel: iptables brute force block: IN=eth0 OUT=
MAC=00:50:56:90:7f:d8:00:22:55:97:10:0f:08:00 SRC=202.113.16.118 DST=207.62.186.9 LEN=60
TOS=0x00 PREC=0x00 TTL=49 ID=16335 DF PROTO=TCP SPT=34937 DPT=22 WINDOW=5840 RES=0x00 SYN
URGP=0
Mar 14 11:32:59 Opus kernel: iptables brute force block: IN=eth0 OUT=
MAC=00:50:56:90:7f:d8:00:22:55:97:10:0f:08:00 SRC=202.113.16.118 DST=207.62.186.9 LEN=60
TOS=0x00 PREC=0x00 TTL=49 ID=16336 DF PROTO=TCP SPT=34937 DPT=22 WINDOW=5840 RES=0x00 SYN
URGP=0
Mar 14 11:33:05 Opus kernel: iptables brute force block: IN=eth0 OUT=
MAC=00:50:56:90:7f:d8:00:22:55:97:10:0f:08:00 SRC=202.113.16.118 DST=207.62.186.9 LEN=60
TOS=0x00 PREC=0x00 TTL=49 ID=16337 DF PROTO=TCP SPT=34937 DPT=22 WINDOW=5840 RES=0x00 SYN
URGP=0
Mar 14 13:00:42 Opus kernel: iptables brute force block: IN=eth0 OUT=
MAC=00:50:56:90:7f:d8:00:22:55:97:10:0f:08:00 SRC=121.11.66.70 DST=207.62.186.9 LEN=60
TOS=0x00 PREC=0x00 TTL=50 ID=18877 DF PROTO=TCP SPT=14752 DPT=22 WINDOW=5792 RES=0x00 SYN
URGP=0
Mar 14 13:00:45 Opus kernel: iptables brute force block: IN=eth0 OUT=
MAC=00:50:56:90:7f:d8:00:22:55:97:10:0f:08:00 SRC=121.11.66.70 DST=207.62.186.9 LEN=60
TOS=0x00 PREC=0x00 TTL=50 ID=18879 DF PROTO=TCP SPT=14752 DPT=22 WINDOW=5792 RES=0x00 SYN
URGP=0
Mar 14 13:00:51 Opus kernel: iptables brute force block: IN=eth0 OUT=
MAC=00:50:56:90:7f:d8:00:22:55:97:10:0f:08:00 SRC=121.11.66.70 DST=207.62.186.9 LEN=60
TOS=0x00 PREC=0x00 TTL=50 ID=18881 DF PROTO=TCP SPT=14752 DPT=22 WINDOW=5792 RES=0x00 SYN
URGP=0
Mar 14 16:25:58 Opus kernel: iptables brute force block: IN=eth0 OUT=
< snipped >
```

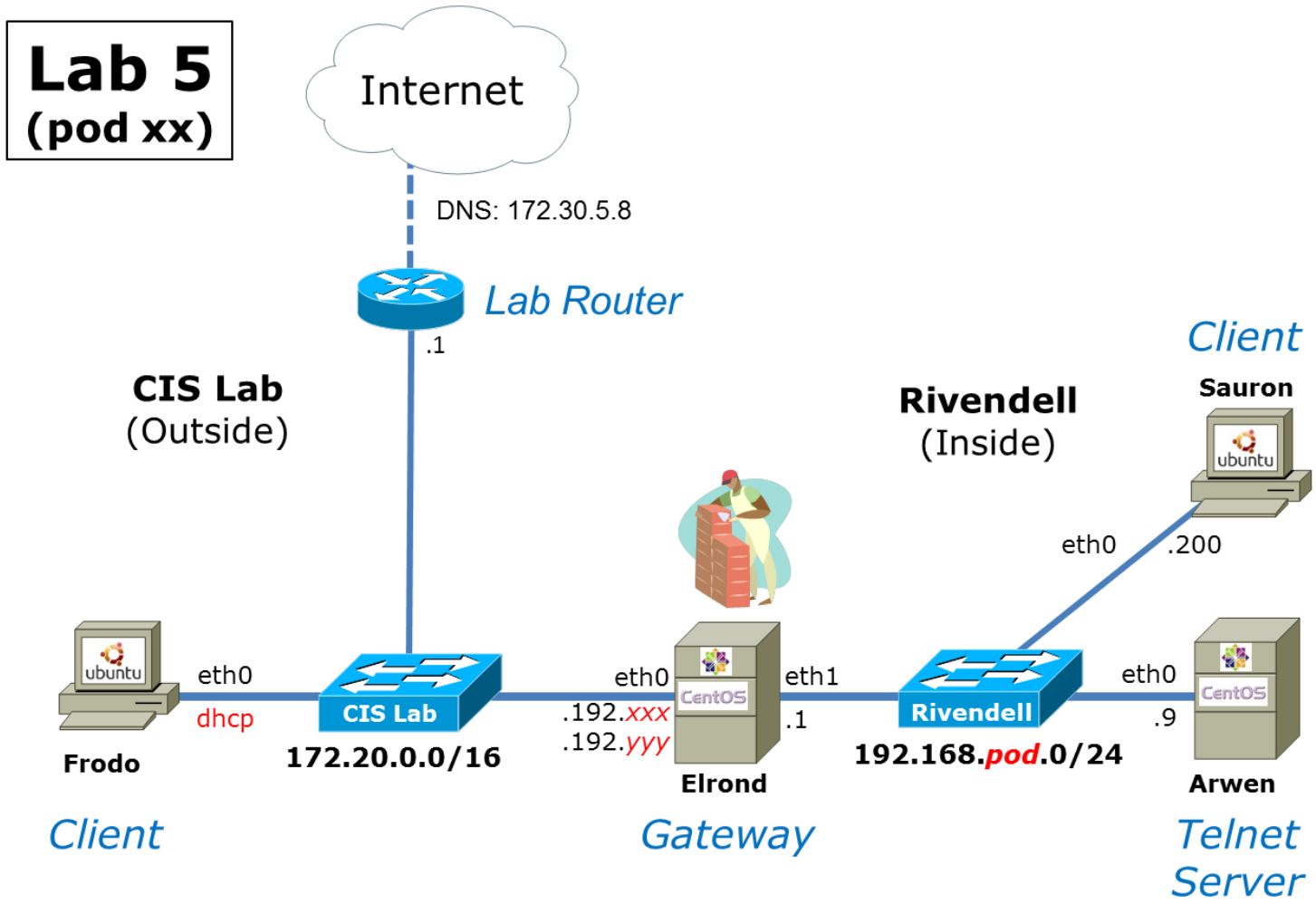
Impeding brute force attacks

Recent dictionary attacks

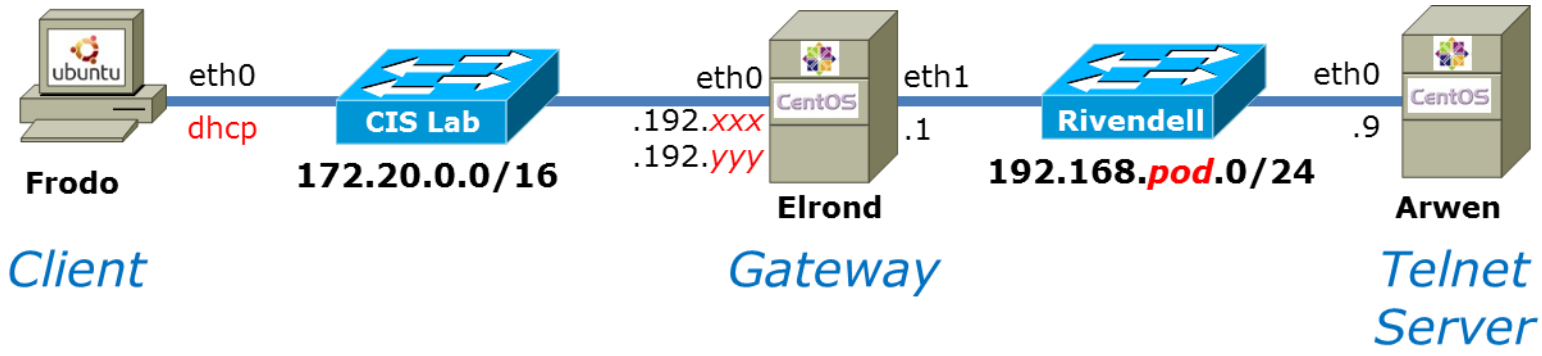
From logwatch report:

```
Failed logins from:
    10.64.25.2: 1 time
    71.198.220.114 (c-71-198-220-114.hsd1.ca.comcast.net): 1 time
    74.220.66.39 (dsl-74-220-66-39.dhcp.cruzio.com): 1 time
    81.93.193.216 (credinfo.hu): 2 times
    95.18.14.156 (156.14.18.95.dynamic.jazztel.es): 1 time
    169.233.218.248 (dhcp-218-248.cruznet.ucsc.edu): 1 time
    180.153.127.111: 2 times

[root@opus ~]# lastb | grep 180.153.127.111
root     ssh:notty    180.153.127.111  Fri Nov 18 11:35 - 11:35  (00:00)
db2inst1 ssh:notty    180.153.127.111  Fri Nov 18 11:35 - 11:35  (00:00)
db2inst1 ssh:notty    180.153.127.111  Fri Nov 18 11:35 - 11:35  (00:00)
root     ssh:notty    180.153.127.111  Fri Nov 18 11:34 - 11:34  (00:00)
root     ssh:notty    180.153.127.111  Fri Oct  7 13:24 - 13:24  (00:00)
db2inst1 ssh:notty    180.153.127.111  Fri Oct  7 13:24 - 13:24  (00:00)
db2inst1 ssh:notty    180.153.127.111  Fri Oct  7 13:24 - 13:24  (00:00)
root     ssh:notty    180.153.127.111  Fri Oct  7 13:24 - 13:24  (00:00)
[root@opus ~]#
```

Lab 5

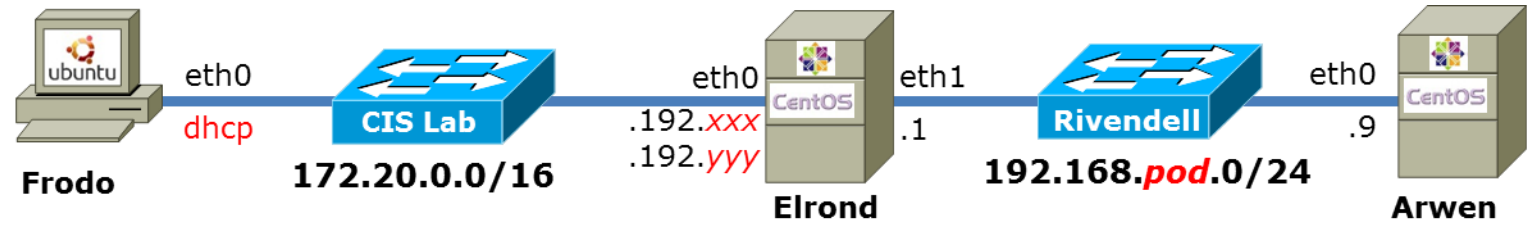


Elrond is the gateway which provides firewall and NAT services for the Rivendell network



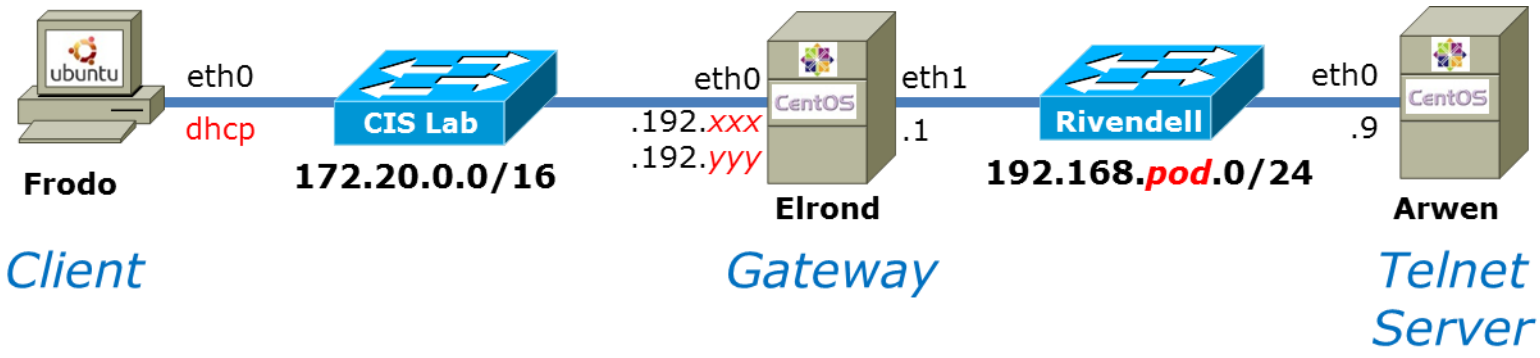

```
[cis192@p24-arwen ~]$ ping -c2 google.com
PING google.com (74.125.224.128) 56(84) bytes of data.
64 bytes from nuq04s09-in-f0.1e100.net (74.125.224.128): icmp_seq=1 ttl=54 time=5.97 ms
64 bytes from nuq04s09-in-f0.1e100.net (74.125.224.128): icmp_seq=2 ttl=54 time=5.85 ms

--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1007ms
rtt min/avg/max/mdev = 5.851/5.914/5.978/0.099 ms
[cis192@p24-arwen ~]$
```

All Rivendell hosts have Internet access with NAT on Elrond




```
cis192@p27-frodo:~$ telnet 172.20.192.171
Trying 172.20.192.171...
Connected to 172.20.192.171.
Escape character is '^]'.
CentOS release 6.3 (Final)
Kernel 2.6.32-279.el6.x86_64 on an x86_64
login: cis192
Password:
Last login: Mon Mar 18 21:09:54 from frodo
[cis192@p24-arwen ~]$ exit
                          logout
Connection closed by foreign host.
cis192@p27-frodo:~$
```

Outsiders can access the internal Telnet server on Arwen using the public IP address .192.yyy on Elrond

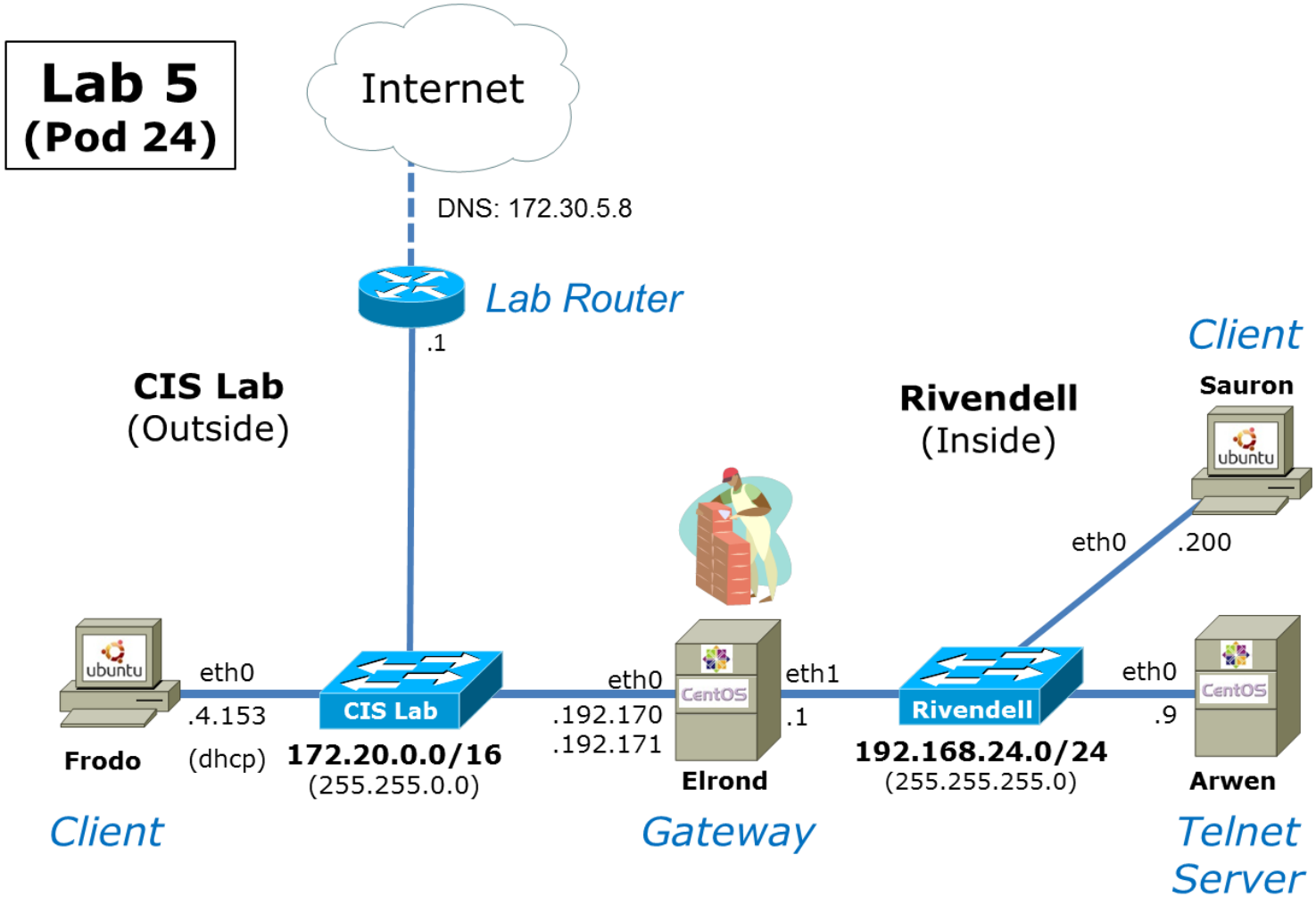



```
cis192@p27-frodo:~$ ping 172.20.192.171
PING 172.20.192.171 (172.20.192.171) 56(84) bytes of data.
^C
--- 172.20.192.171 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2015ms
```

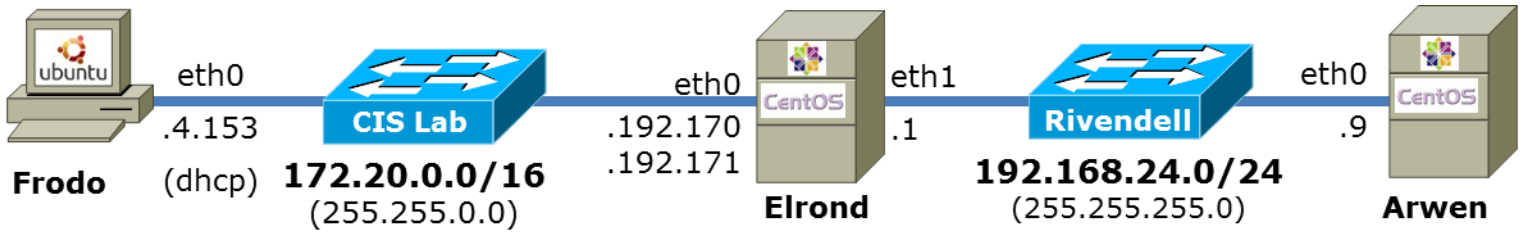
Except for Telnet access via .192.yyy all other incoming traffic towards Rivendell is blocked

Lab 5 firewall and nat configuration

Firewall and NAT settings for Lab 5

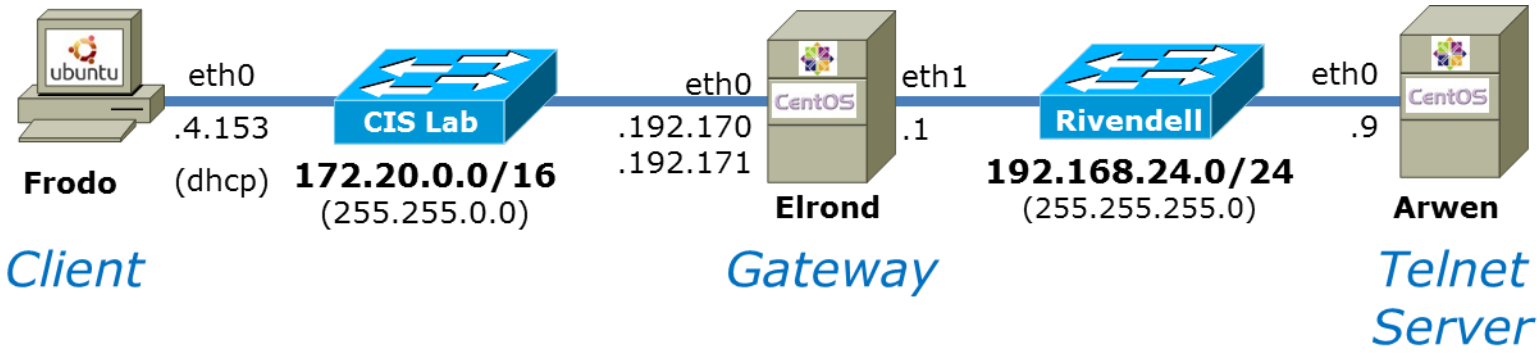


Firewall and NAT settings for Lab 5



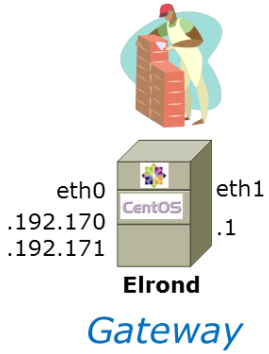

```
[root@p24-elrond ~]# iptables -nL
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     all  --  192.168.24.0/24      192.168.24.1        state NEW
LOG        all  --  0.0.0.0/0            0.0.0.0/0           LOG flags 0 level 6 prefix
`iptables INPUT:'

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,ESTABLISHED
ACCEPT     all  --  192.168.24.0/24      0.0.0.0/0           state NEW
ACCEPT     tcp  --  0.0.0.0/0            192.168.24.9        state NEW tcp dpt:23
LOG        all  --  0.0.0.0/0            0.0.0.0/0           LOG flags 0 level 6 prefix
`iptables FORWARD:'

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state NEW,RELATED,ESTABLISHED
[root@p24-elrond ~]#
```

Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# iptables -nL -t nat
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination
DNAT       all  --  0.0.0.0/0            172.20.192.171      to:192.168.24.9

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
SNAT       all  --  192.168.24.9         0.0.0.0/0           to:172.20.192.171
SNAT       all  --  192.168.24.0/24      0.0.0.0/0           to:172.20.192.170

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

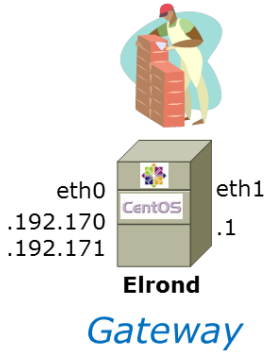
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Shows actual iptables commands used to build firewall and configure NAT

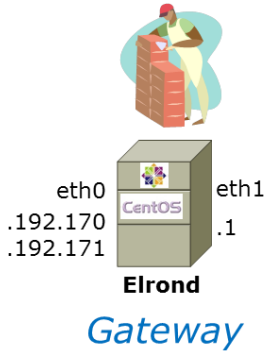
Firewall and NAT settings for Lab 5



Standard NAT chains

```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

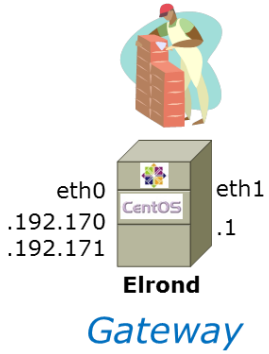
Firewall and NAT settings for Lab 5



Standard filter chains

```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

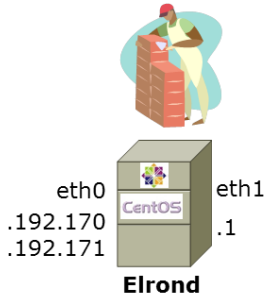
Firewall and NAT settings for Lab 5



Policy settings which are used if no rules on the chain apply

```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

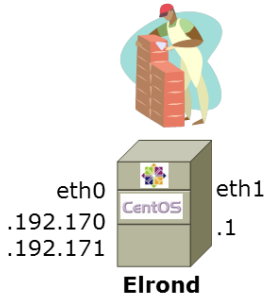
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

For arriving packets to 172.20.192.171 translate destination address to 192.168.24.9 (NAT)

Firewall and NAT settings for Lab 5

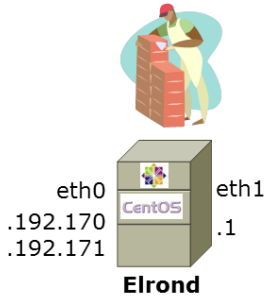


Gateway

```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Make outgoing packets from 192.168.24.9 appear as if they came from 172.20.192.171 (NAT)

Firewall and NAT settings for Lab 5

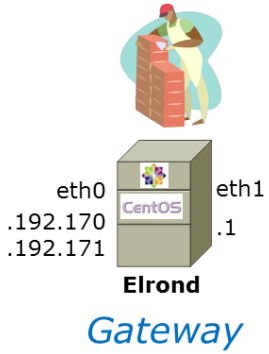


Gateway

```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Allows all Rivendell hosts to have Internet access (NAT)

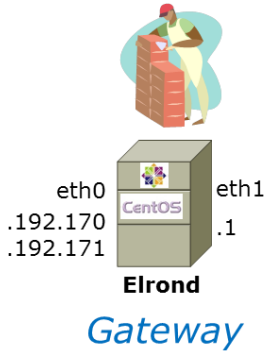
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Allow ongoing traffic based on existing or related to existing connections

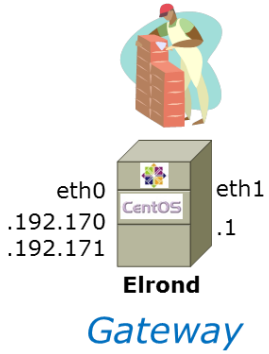
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Allow new incoming connections from any "inside" Rivendell host

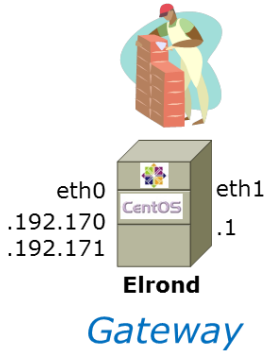
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Log incoming packet that is about to be dropped

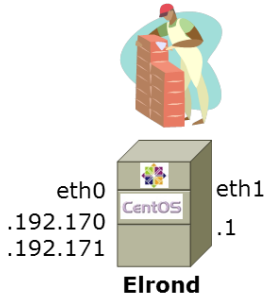
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Forward ongoing traffic based on established or related connections

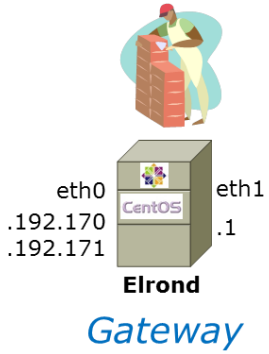
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Allow all new outgoing connections from hosts on the "inside" Rivendell network to be forwarded

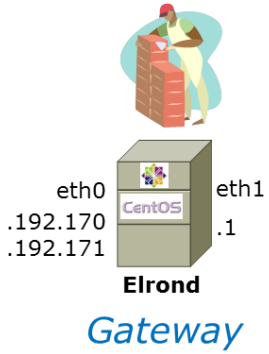
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Allow incoming new telnet connections going to the Telnet server

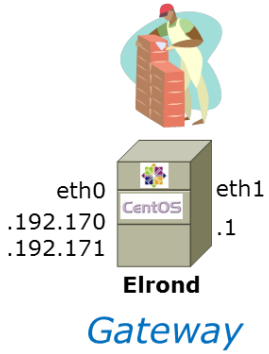
Firewall and NAT settings for Lab 5




```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Log any traffic that is about to be dropped (this is the last rule on the chain before the DROP policy is applied)

Firewall and NAT settings for Lab 5



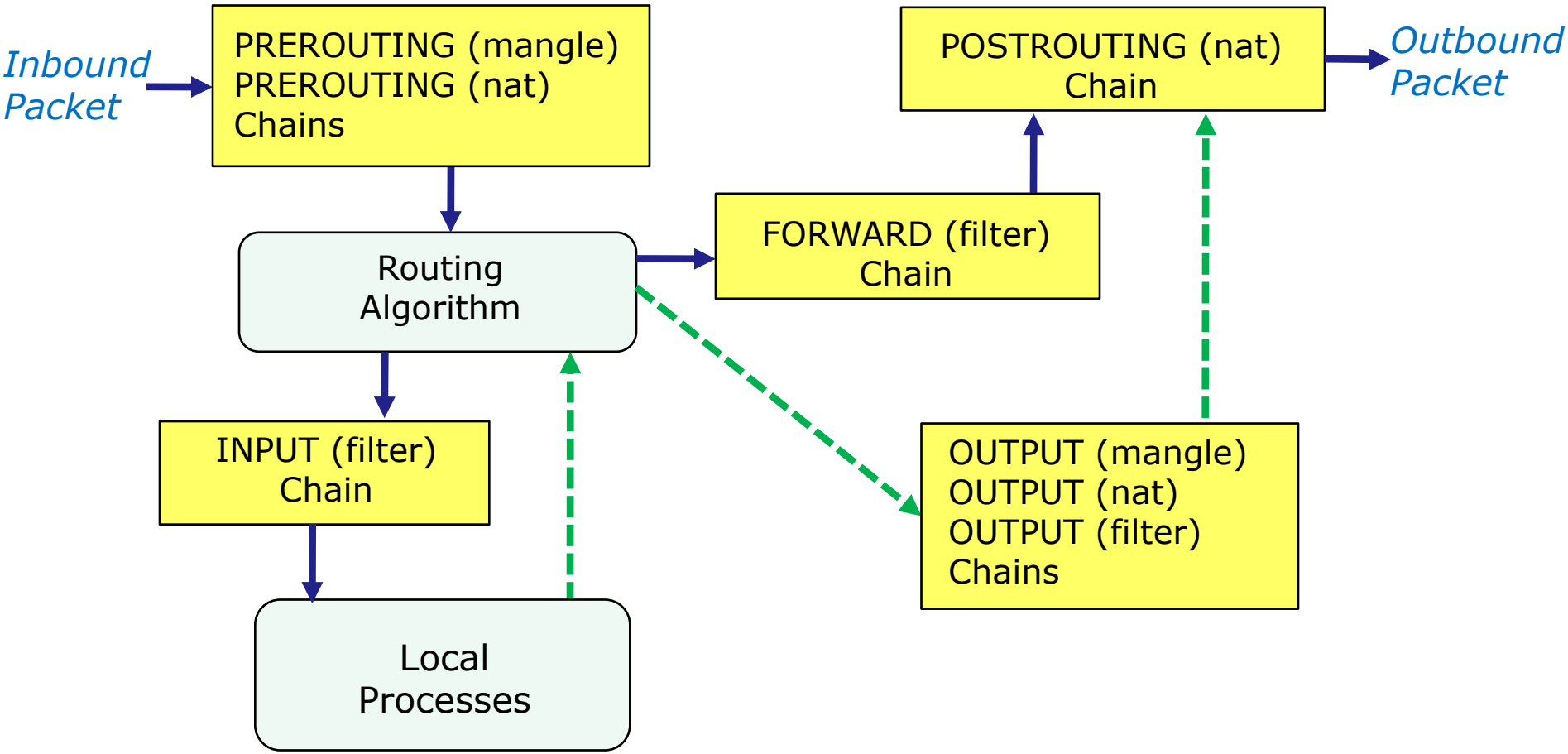

```
[root@p24-elrond ~]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*nat
:PREROUTING ACCEPT [274:29705]
:POSTROUTING ACCEPT [17:1421]
:OUTPUT ACCEPT [15:1301]
-A PREROUTING -d 172.20.192.171/32 -i eth0 -j DNAT --to-destination 192.168.24.9
-A POSTROUTING -s 192.168.24.9/32 -o eth0 -j SNAT --to-source 172.20.192.171
-A POSTROUTING -s 192.168.24.0/24 -o eth0 -j SNAT --to-source 172.20.192.170
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
# Generated by iptables-save v1.4.7 on Sun Mar 17 13:38:54 2013
*filter
:INPUT DROP [10:985]
:FORWARD DROP [9:756]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.24.0/24 -d 192.168.24.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.24.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.24.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sun Mar 17 13:38:54 2013
[root@p24-elrond ~]#
```

Allow all outgoing traffic

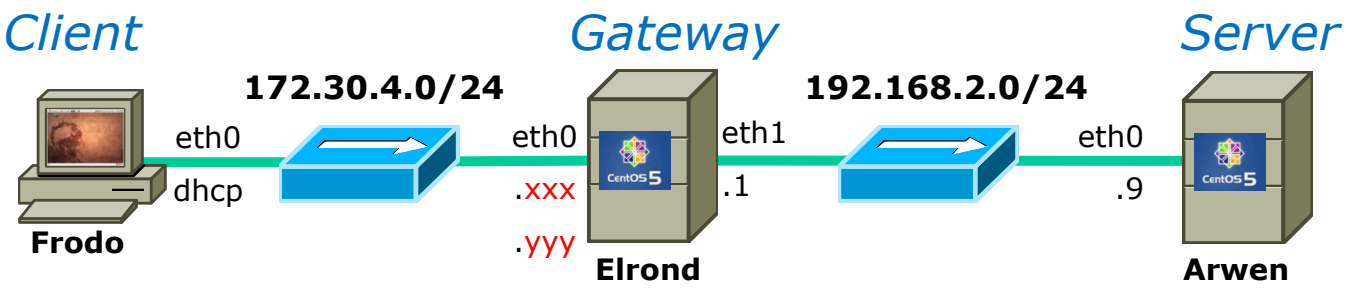
NAT port forwarding

NAT port forwarding

Netfilter – all tables and chains



Use the PREROUTING NAT table chain for port forwarding

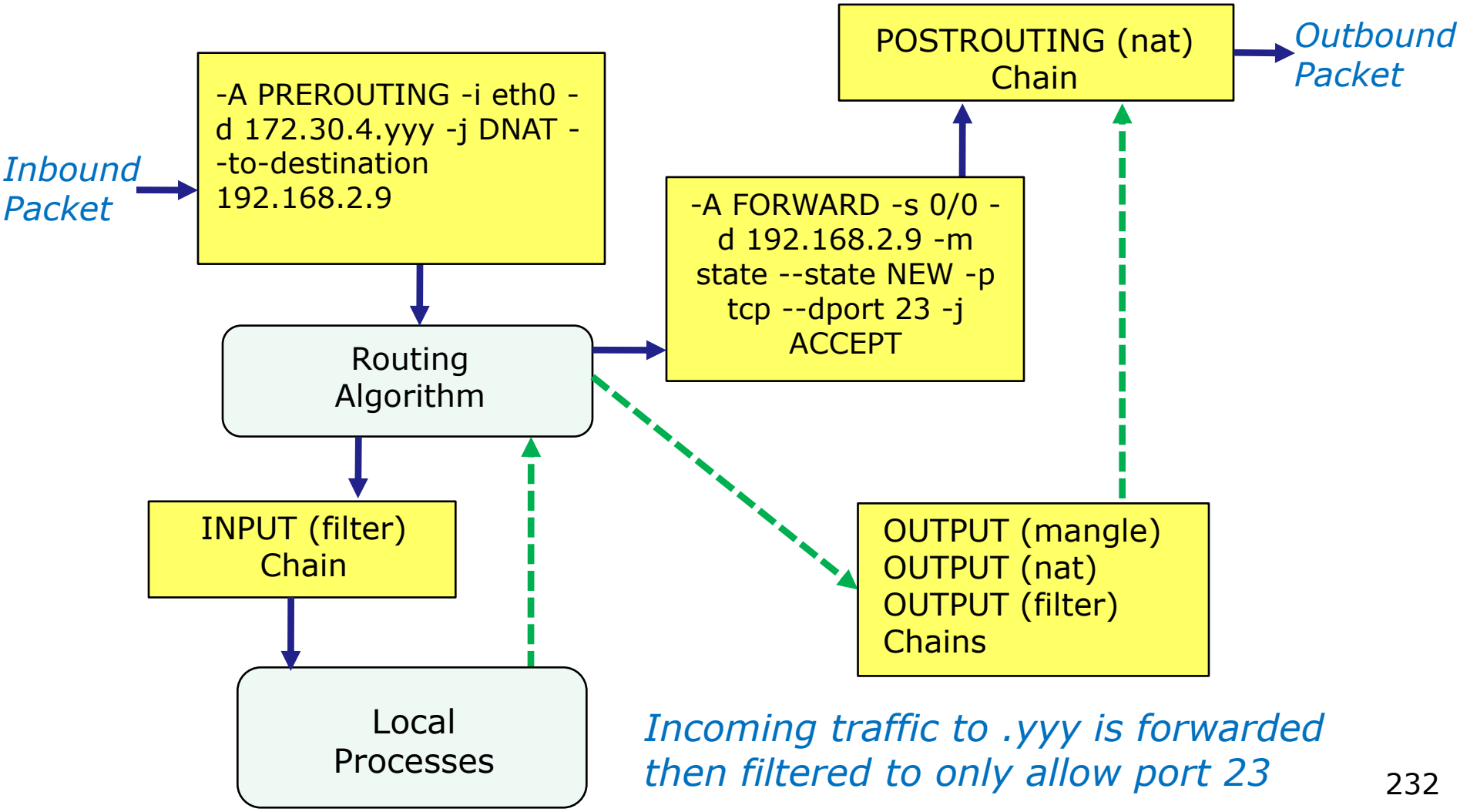


In Lab 5, all incoming traffic to .yyy is forwarded to Arwen

```
iptables -t nat -A PREROUTING -i eth0 -d 172.30.4.yyy -j DNAT --to-destination 192.168.2.9

iptables -A FORWARD -s 192.168.2.0/24 -d 0/0 -m state --state NEW -j ACCEPT
iptables -A FORWARD -s 0/0 -d 192.168.2.9 -m state --state NEW -p tcp --dport 23 -j ACCEPT
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -j LOG --log-level info --log-prefix "iptables FORWARD: "
iptables -P FORWARD DROP

iptables -A INPUT -i eth1 -s 192.168.2.0/24 -d 192.168.2.1 -m state --state NEW -j ACCEPT
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -j LOG --log-level info --log-prefix "iptables INPUT: “
iptables -P INPUT DROP

iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
iptables -P OUTPUT DROP

iptables -t nat -A POSTROUTING -o eth0 -s 192.168.2.9 -j SNAT --to-source 172.30.4.yyy
iptables -t nat -A POSTROUTING -o eth0 -s 192.168.2.0/24 -j SNAT --to-source 172.30.4.xxx
```

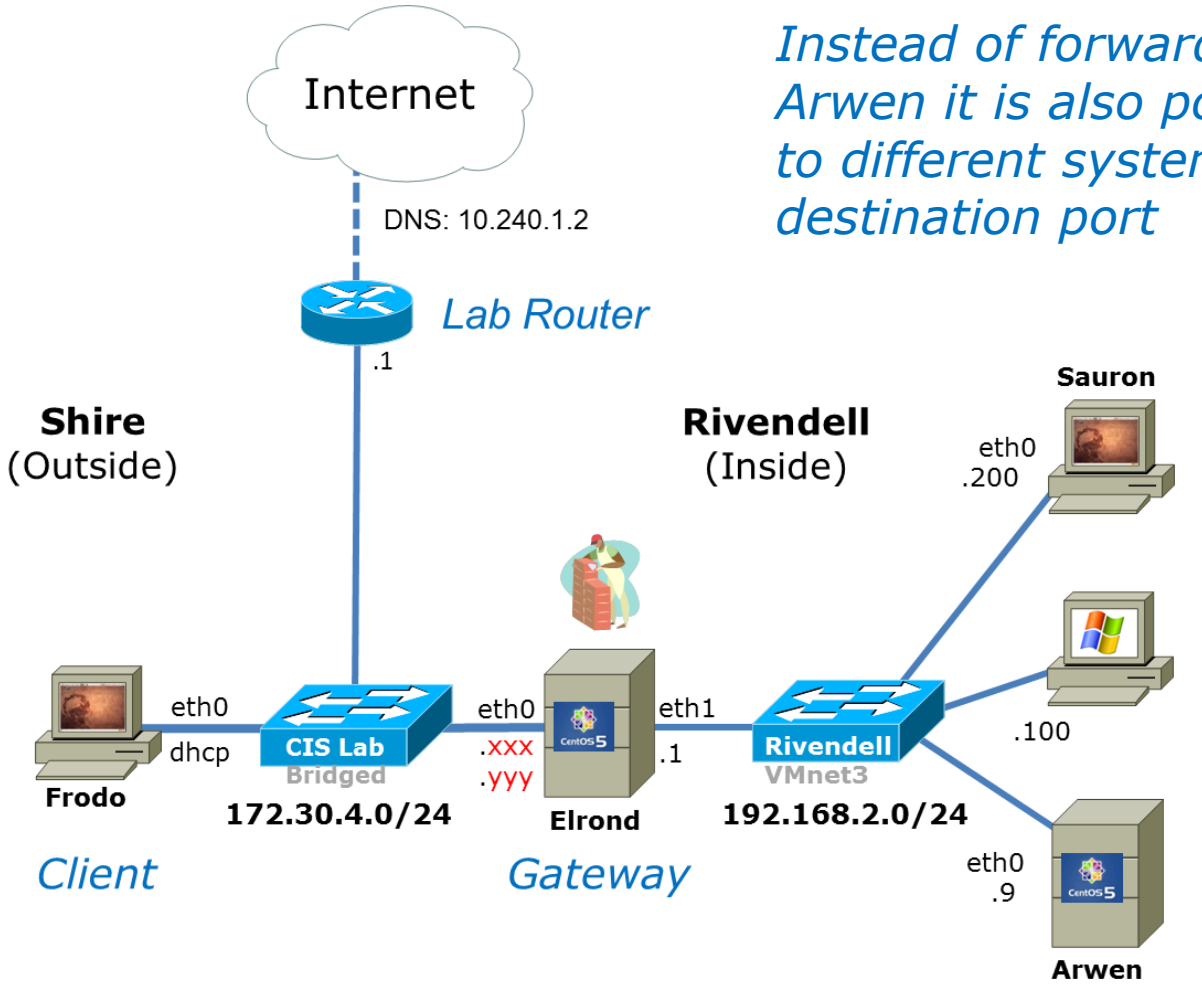
Lab 5

Incoming traffic is forwarded to Arwen



Incoming traffic to .yyy is forwarded then filtered to only allow port 23

Lab 5 modification



Instead of forwarding all traffic to Arwen it is also possible to forward to different systems based on destination port

All SSH traffic goes to Sauron

All Remote Desktop traffic goes to William

All Telnet traffic still goes to Arwen

Lab 5 modification

Let xxx=252
and yyy=253

~~iptables -t nat -A PREROUTING -i eth0 -d 172.30.4.yyy -j DNAT --to-destination 192.168.2.9~~
iptables -t nat -A PREROUTING -i eth0 -d 172.30.4.253/32 -p tcp -m tcp --dport 22 -j DNAT --to-destination 192.168.2.200
iptables -t nat -A PREROUTING -i eth0 -d 172.30.4.253/32 -p tcp -m tcp --dport 23 -j DNAT --to-destination 192.168.2.9
iptables -t nat -A PREROUTING -i eth0 -d 172.30.4.253/32 -p tcp -m tcp --dport 3389 -j DNAT --to-destination 192.168.2.100

Forward to different systems based on destination port

iptables -A FORWARD -s 192.168.2.0/24 -d 0/0 -m state --state NEW -j ACCEPT
~~iptables -A FORWARD -s 0/0 -d 192.168.2.9 -m state --state NEW -p tcp --dport 23 -j ACCEPT~~
iptables -A FORWARD -d 192.168.2.200/32 -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
iptables -A FORWARD -d 192.168.2.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
iptables -A FORWARD -d 192.168.2.100/32 -p tcp -m state --state NEW -m tcp --dport 3389 -j ACCEPT

Open the firewall to allow the selected destination port traffic to be forwarded

iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -j LOG --log-level info --log-prefix "iptables FORWARD: "
iptables -P FORWARD DROP

iptables -A INPUT -i eth1 -s 192.168.2.0/24 -d 192.168.2.1 -m state --state NEW -j ACCEPT
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -j LOG --log-level info --log-prefix "iptables INPUT: "
iptables -P INPUT DROP

iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
iptables -P OUTPUT DROP

172.30.4.253
iptables -t nat -A POSTROUTING -o eth0 -s 192.168.2.9 -j SNAT --to-source ~~172.30.4.yyy~~
iptables -t nat -A POSTROUTING -o eth0 -s 192.168.2.0/24 -j SNAT --to-source ~~172.30.4.xxx~~
172.30.4.252

Lab 5 modification

```
[root@elrond sysconfig]# cat iptables
# Generated by iptables-save v1.4.7 on Sat Nov 19 08:25:01 2011
*nat
:PREROUTING ACCEPT [1216:196031]
:POSTROUTING ACCEPT [8:510]
:OUTPUT ACCEPT [3:210]
# Redirect incoming public IP traffic based on destination port
-A PREROUTING -i eth0 -d 172.30.4.253/32 -p tcp -m tcp --dport 22 -j DNAT --to-destination 192.168.2.200
-A PREROUTING -i eth0 -d 172.30.4.253/32 -p tcp -m tcp --dport 23 -j DNAT --to-destination 192.168.2.9
-A PREROUTING -i eth0 -d 172.30.4.253/32 -p tcp -m tcp --dport 3389 -j DNAT --to-destination 192.168.2.100
# Internet for Rivendell hosts using NAT
-A POSTROUTING -s 192.168.2.9/32 -o eth0 -j SNAT --to-source 172.30.4.253
-A POSTROUTING -s 192.168.2.0/24 -o eth0 -j SNAT --to-source 172.30.4.252
COMMIT
# Completed on Sat Nov 19 08:25:01 2011
# Generated by iptables-save v1.4.7 on Sat Nov 19 08:25:01 2011
*filter
:INPUT DROP [894:156935]
:FORWARD DROP [7:668]
:OUTPUT DROP [0:0]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.2.0/24 -d 192.168.2.1/32 -i eth1 -m state --state NEW -j ACCEPT
-A INPUT -j LOG --log-prefix "iptables INPUT:" --log-level 6
-A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -s 192.168.2.0/24 -m state --state NEW -j ACCEPT
-A FORWARD -d 192.168.2.200/32 -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A FORWARD -d 192.168.2.9/32 -p tcp -m state --state NEW -m tcp --dport 23 -j ACCEPT
-A FORWARD -d 192.168.2.100/32 -p tcp -m state --state NEW -m tcp --dport 3389 -j ACCEPT
-A FORWARD -j LOG --log-prefix "iptables FORWARD:" --log-level 6
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Sat Nov 19 08:25:01 2011
[root@elrond sysconfig]#
```

Lab 5 modified to support port forwarding

Wrap

New commands, daemons and files:

- iptables
- netstat
- service
- yum

Daemons and related configuration files

tcpd	/etc/hosts.allow,hosts.deny

Next Class

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

Lab 5 due

Quiz questions for next class:

- How do you display the current filter table chains?
- How do you display the current nat table chains?
- How do set the FORWARD chain policy to ACCEPT?

Backup