Lesson Module Status

- Slides –
- Properties - done
- Flashcards -
- 1st minute quiz –
- Web Calendar summary –
- Web book pages –
- Commands – done
- Howtos –
- Skills pacing -
- Lab – done
- Depot (VMs) – na
- Tests graded
- Tests histogram
- Printer, cable, cord, dhcp reservation
- Pizza

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/

Joe P.

Joe A.

Chuck

Kay

Lieven

Joe B.

Chris H.

Edwin

Julio

Rich

Jesus

Josh

Casady

Jack

Drew

Edgar

Ryan

Aaron

Chris B.

John

Junious

Robert

Brynden

www.cccconfer.org
dial-in: 888-886-3951
passcode: 439080

VMs for tonight
(**Revert**, **384MB** RAM and **power up**)
Arwen Celebrian Sniffer

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

Quiz

Please take out a blank piece of paper, switch off your monitor, close your books, put away your notes and answer these questions:

- What two packages must be installed to setup a name server with caching?
- How does the serial number effect zone transfers?
- What is the purpose of a PTR record?

Network File System and Printing

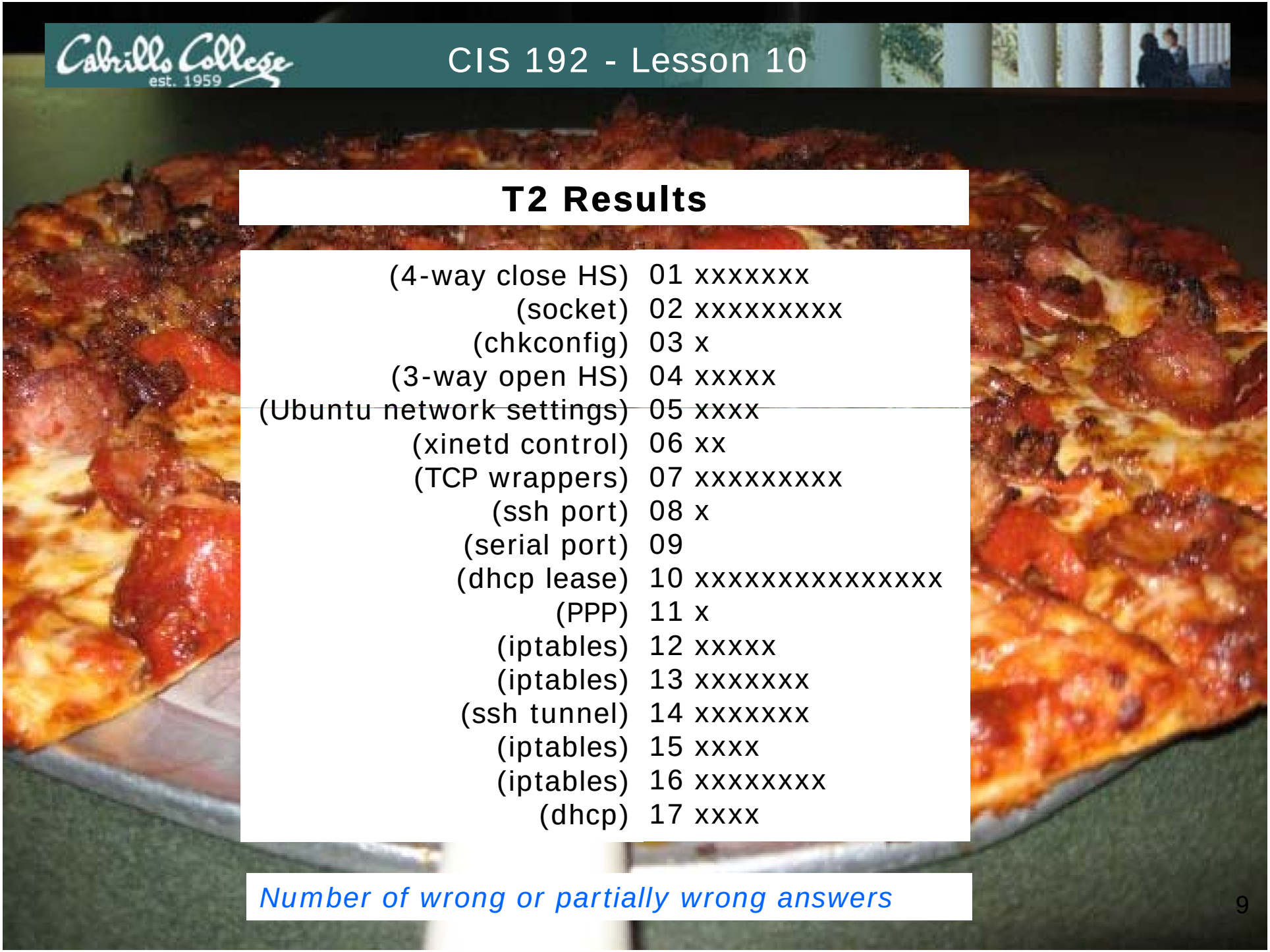
Objectives	Agenda
• Use NFS to share a directory of files on one machine with the other hosts on the same network.	• Quiz • Questions on previous material • Test 2 Results • Housekeeping • Mounting • LVM sidetrack • RPC and Port Mapper • NFS • Printing • Lab X3 (NFS) • Wrap

Questions on previous material

Questions?

- Previous lesson material
- Lab assignments
- Tests

Test 2 Results



T2 Results

(4-way close HS)	01 xxxxxxx
(socket)	02 xxxxxxxxx
(chkconfig)	03 x
(3-way open HS)	04 xxxxx
(Ubuntu network settings)	05 xxxx
(xinetd control)	06 xx
(TCP wrappers)	07 xxxxxxxxx
(ssh port)	08 x
(serial port)	09
(dhcp lease)	10 xxxxxxxxxxxxxx
(PPP)	11 x
(iptables)	12 xxxxx
(iptables)	13 xxxxxxx
(ssh tunnel)	14 xxxxxxx
(iptables)	15 xxxx
(iptables)	16 xxxxxxxx
(dhcp)	17 xxxx

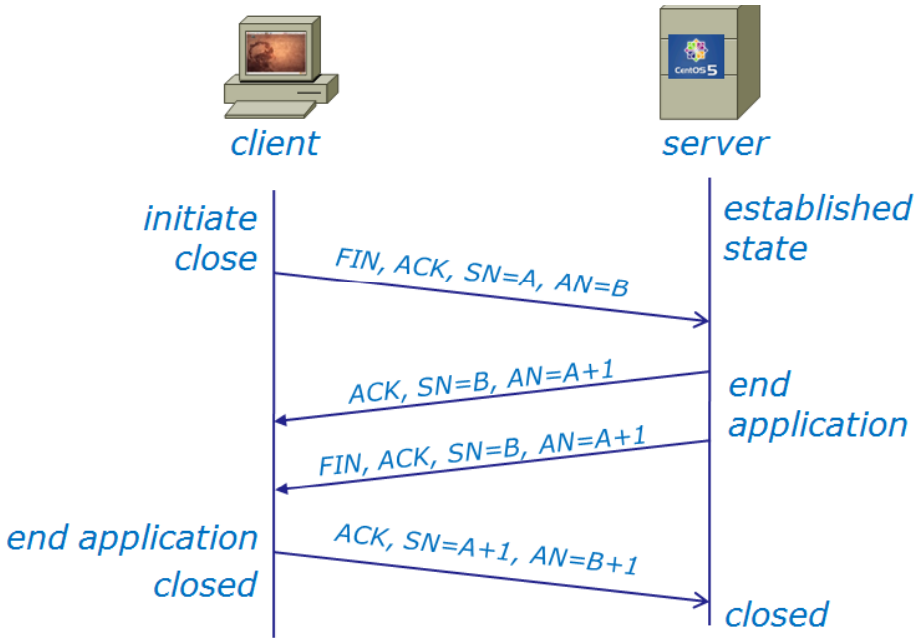
Number of wrong or partially wrong answers

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

*3 way handshake to **Open***

*Data **transfer***

*4 way handshake to **Close***



Q1 Referring to figure 1 above and using the packet numbers on the right, which packet marks the point after which the connection used for the data transfer is closed on the server? **14**

XXXXXXX

[L6:20-21]

Client		Server	
IP:	172.30.4.83	IP:	192.168.2.150
Port:	41025	Port:	51283

XXXXXXXXX
[L6:21]

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

*3 way handshake to **Open***

*Data **transfer***

*4 way handshake to **Close***

Q2 Referring to figure 1 above, what socket is used for the FTP data transfer? (To answer, fill in the table)

Note: FTP uses one socket for commands and another for data transfers

*The FTP server is in **Passive mode** so the client initiates the data connection (3 way handshake) with the server.*

Q3 What command on Red Hat family systems would configure the vsftpd service to startup automatically when powering up?

chkconfig vsftpd on x**[L6:99]**

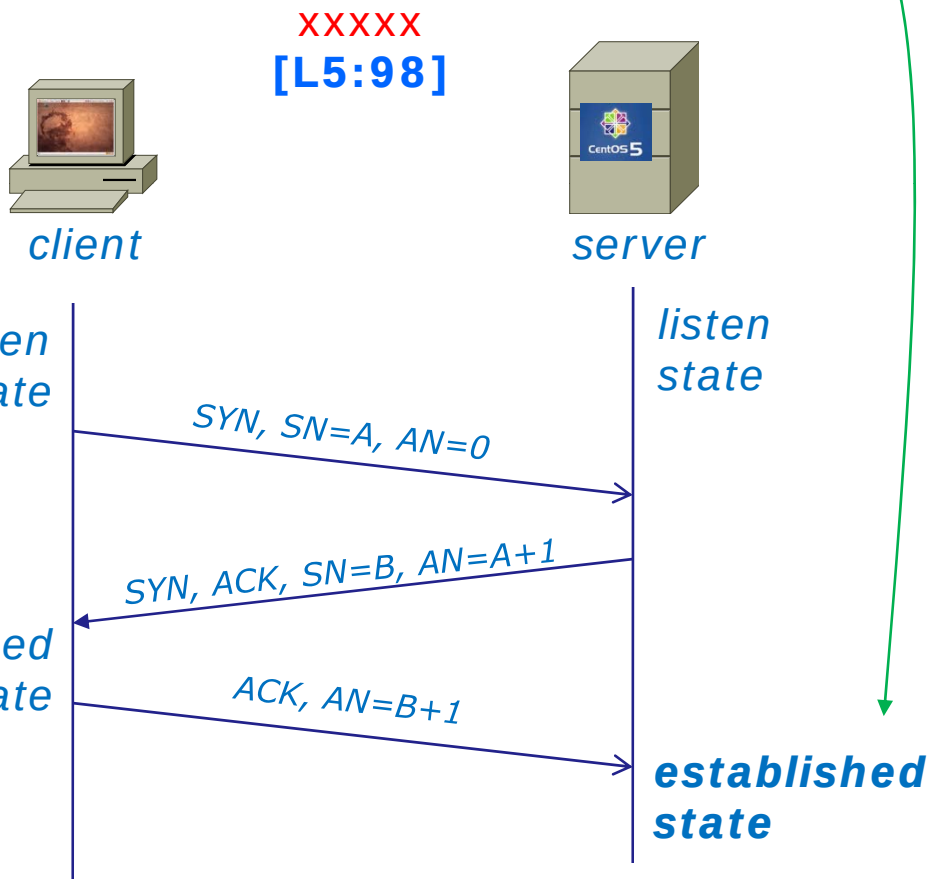
Q4 For firewall purposes when is a TCP stream considered to be *established* on the server side?

After client sends an ACK to finish the 3-way handshake

Initial Connection

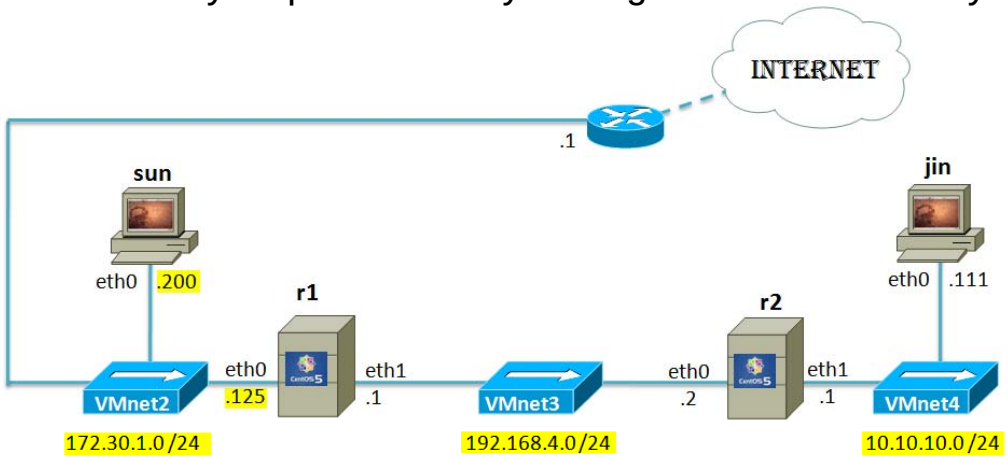
Three-Way Handshake

1. SYN
2. SYN-ACK
3. ACK



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

Q5 How would you permanently configure the Ubuntu system named **sun** below



with a static IP, default gateway, and all necessary routes to reach the other two private networks?

Configuration file to edit on Sun: **/etc/network/interfaces**

Fill in the blanks below for Sun's configuration file:

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet static
address 172.30.1.200
netmask 255.255.255.0
gateway 172.30.1.1
up route add -net 192.168.4.0/24 gw 172.30.1.125
up route add -net 10.10.10.0.0/24 gw 172.30.1.125
```

*Note: static route gateways, like default gateways, are **next hops** (on a directly connected network)*

XXXX

[L3:57-59,109 or L8:75]

Q6 What are **two** different commands on Red Hat family systems that would cause the xinetd daemon to reread its configuration files?

service xinetd restart
killall -1 xinetd

alternate answer:
ps -ef | grep xinetd then **kill -1** ***pid-of-xinetd***

xx
[L6:60]

Q7 How would you configure TCP wrappers to only allow incoming SSH connections from hosts in our classroom (room 2501) network? (Answer by writing the lines you would add to the two files below)

/etc/hosts.allow: **sshd: 172.30.1.**
/etc/hosts.deny: **ALL: ALL**

OR

/etc/hosts.allow: **sshd: 172.30.1.0/255.255.255.0**
/etc/hosts.deny: **ALL: ALL**

xxxxxxxxx
[L1:54 or ifconfig output on any bridged VM]
[L6:161-163]

Q8 What port number is used by the DDH service? **22** x **[L6:134]**

Use this when configuring firewall rules to allow SSH access

Q9 In the DOS world the first serial port is called COM 1, what Linux device is used to reference this same port? **/dev/ttyS0**

[L8:133]

Q10 A DHCP service is running on Elrond using the file below.

```
[root@elrond ~]# cat /etc/dhcpd.conf
ddns-update-style interim;
ignore client-updates;
option time-offset                -25200;
subnet 192.168.2.0 netmask 255.255.255.0 {
         option routers                     192.168.2.1;
         option subnet-mask                 255.255.255.0;
         option domain-name                 "rivendell";
         option domain-name-servers         207.62.187.53;
         range dynamic-bootp                192.168.2.100
192.168.2.254;
         default-lease-time                 14400;
         max-lease-time                     36000;
}
[root@elrond ~]#
```

Clients will attempt to renew their leases when 50% of the lease time has passed

For Rivendell clients that get their IP address from Elrond how long will they wait before attempting to renew their leases?
Assume they did not specify a lease time on their original request.

2 hours (7200 seconds) xxxxxxxxxxxxxxx**[L7:61]**

Q11 Regarding the command below:

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

a) What does this command do?

Makes a PPP connection using serial port

b) What are the arguments assword: and secret used for?

when logging in, to automatically answer the Password prompt with “secret”

x

[L8:149]

[L8:152]

Q12 What **five** complete iptables commands would
a) flush all the rules from the current filter chains,
b) delete any custom chains and
c) set the policy to ACCEPT on the INPUT, FORWARD and OUTPUT chains?

iptables -F
iptables -X
iptables -P FORWARD ACCEPT
iptables -P INPUT ACCEPT
iptables -P OUTPUT ACCEPT

xxxxx
[L6:224,234]

This completely removes any firewall

Q13 Given the following default firewall on a CentOS (Red Hat) system:

```
[root@arwen ~]# iptables -nL RH-Firewall-1-INPUT --line-numbers
Chain RH-Firewall-1-INPUT (2 references)
num  target     prot opt source         destination
1    ACCEPT     all  --  0.0.0.0/0      0.0.0.0/0
2    ACCEPT     icmp --  0.0.0.0/0      0.0.0.0/0      icmp type 255
3    ACCEPT     esp  --  0.0.0.0/0      0.0.0.0/0
4    ACCEPT     ah   --  0.0.0.0/0      0.0.0.0/0
5    ACCEPT     udp  --  0.0.0.0/0      224.0.0.251    udp dpt:5353
6    ACCEPT     udp  --  0.0.0.0/0      0.0.0.0/0      udp dpt:631
7    ACCEPT     tcp  --  0.0.0.0/0      0.0.0.0/0      tcp dpt:631
8    ACCEPT     all  --  0.0.0.0/0      0.0.0.0/0      state RELATED,ESTABLISHED
9    ACCEPT     tcp  --  0.0.0.0/0      0.0.0.0/0      state NEW tcp dpt:22
10   REJECT     all  --  0.0.0.0/0      0.0.0.0/0      reject-with icmp-host-
prohibited
[root@arwen ~]#
```

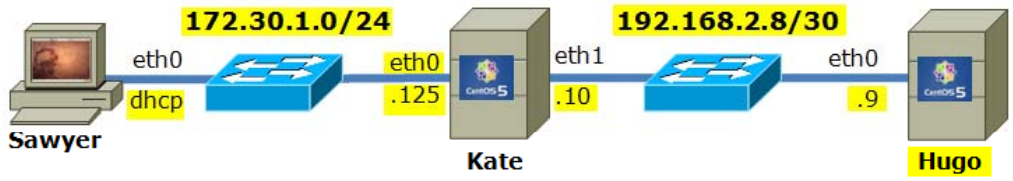
What complete iptables command would insert a rule to enable new incoming FTP (command) connections? (You can ignore handling port used for FTP data transfers)

iptables -I RH-Firewall-1-INPUT 9 -m state --state NEW -m tcp -p tcp --dport 21 -j ACCEPT

XXXXXXX

[L6:93]

Q14 Refer to the diagram below. Kate's firewall allows incoming new and established SSH connections from the outside. All other new connection attempts from the outside are blocked. A Telnet server is running on Hugo that can be accessed from all "inside" systems including Kate.



a) **ssh –L 8000:192.168.2.9:23 172.30.1.125**
b) **telnet localhost 8000**

Port 23 is Telnet

xxxxxxx

[L6:149,152]

a) What command would set up SSH port forwarding so that Sawyer could use its own port 8000 to access the Telnet server on Hugo? **and** b) once the port forwarding had been set up what second command on Sawyer would be used to make the actual connection to the Telnet server?

Q15 A Linux system named Rascal has the following firewall configured:

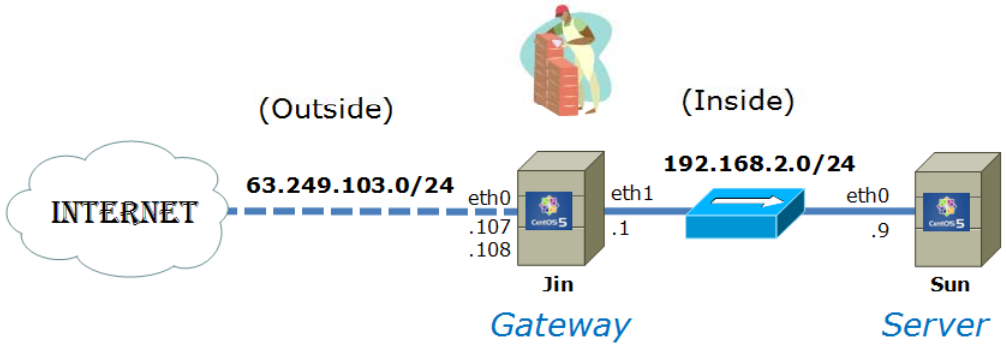
```
[root@rascal ~]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[root@rascal ~]#
```

Rascal is getting bombarded with malicious login attempts from a host with an IP address of 63.13.102.84. What single iptables command would drop (without any error feedback) all packets coming from this malicious system yet allow in everything else?

iptables –A INPUT –s 63.13.102.84 –j DROP xxxx**[L6:240,251]**

Q16 A network address translation service is set up on Jin for hosts on the private inside network, including Sun, using:

```
iptables -t nat -A PREROUTING -i eth0 -d 63.249.103.108 -j DNAT --to-destination 192.168.2.9
iptables -t nat -A POSTROUTING -o eth0 -s 192.168.2.9 -j SNAT --to-source 63.249.103.108
iptables -t nat -A POSTROUTING -o eth0 -s 192.168.2.0/24 -j SNAT --to-source 63.249.103.107
```



Imagine that Sun has made an ssh connection to a system, opus.cabrillo.edu, on the Internet. If you were to sniff the packets that Opus receives from Sun, what would the source and destination IP addresses be?

SIP: **63.249.103.108**
DIP: **207.62.186.9** xxxxxxxxx**[L6:254, ping opus.cabrillo.edu]**

Q17 Elrond has been configured to provide DHCP services.

```
[root@elrond ~]# cat /var/lib/dhcpd/dhcpd.leases
< snipped >
lease 192.168.3.99 {
  starts 4 2010/03/25 22:55:56;
  ends 5 2010/03/26 04:55:56;
  binding state active;
  next binding state free;
  hardware ethernet 08:00:27:ad:6f:50;
  client-hostname "sauron";
}
[root@elrond ~]#

[root@elrond ~]# cat /etc/dhcpd.conf
< snipped >
subnet 192.168.3.0 netmask 255.255.255.0 {
        option routers                  192.168.3.150; # Default GW
        option subnet-mask              255.255.255.0;
        option domain-name              "mordor";
        option domain-name-servers      207.62.187.53;

        range dynamic-bootp             192.168.3.50 192.168.3.99;
        default-lease-time              21600; # 6 hours
        max-lease-time                  43200; # 12 hours
```

This scope definition in dhcpd.conf was used to assign the 192.168.3.99 address from its pool

Using the information above, what IP address, netmask and default gateway were leased to Sauron?

IP: **192.168.3.99**

Netmask: **255.255.255.0**

Default gateway: **192.168.3.150**

xxxxxx**[L7:88-89,124]**

Housekeeping

- Lab 7 (DNS) due today
- VirtualBox on Stations 21-24, CIS-Lab-05 (GAH free zones I hope!)
- Cabrillo College application for certificates and degrees are due April 26th. Use Web Advisor to check eligibility.
- Fine print in some of the slides tonight so best download PDF to see details
- Extra credit labs available:
  - X1 Permanent NIC configuration (30 points)
  - X2 PPP (30 points)
  - X3 NFS (30 points)
  - Original NIC lab (20 points)
  - Original routing lab (20 points)
  - Original port forwarding lab (20 points)
  - Original firewall lab (20 points)

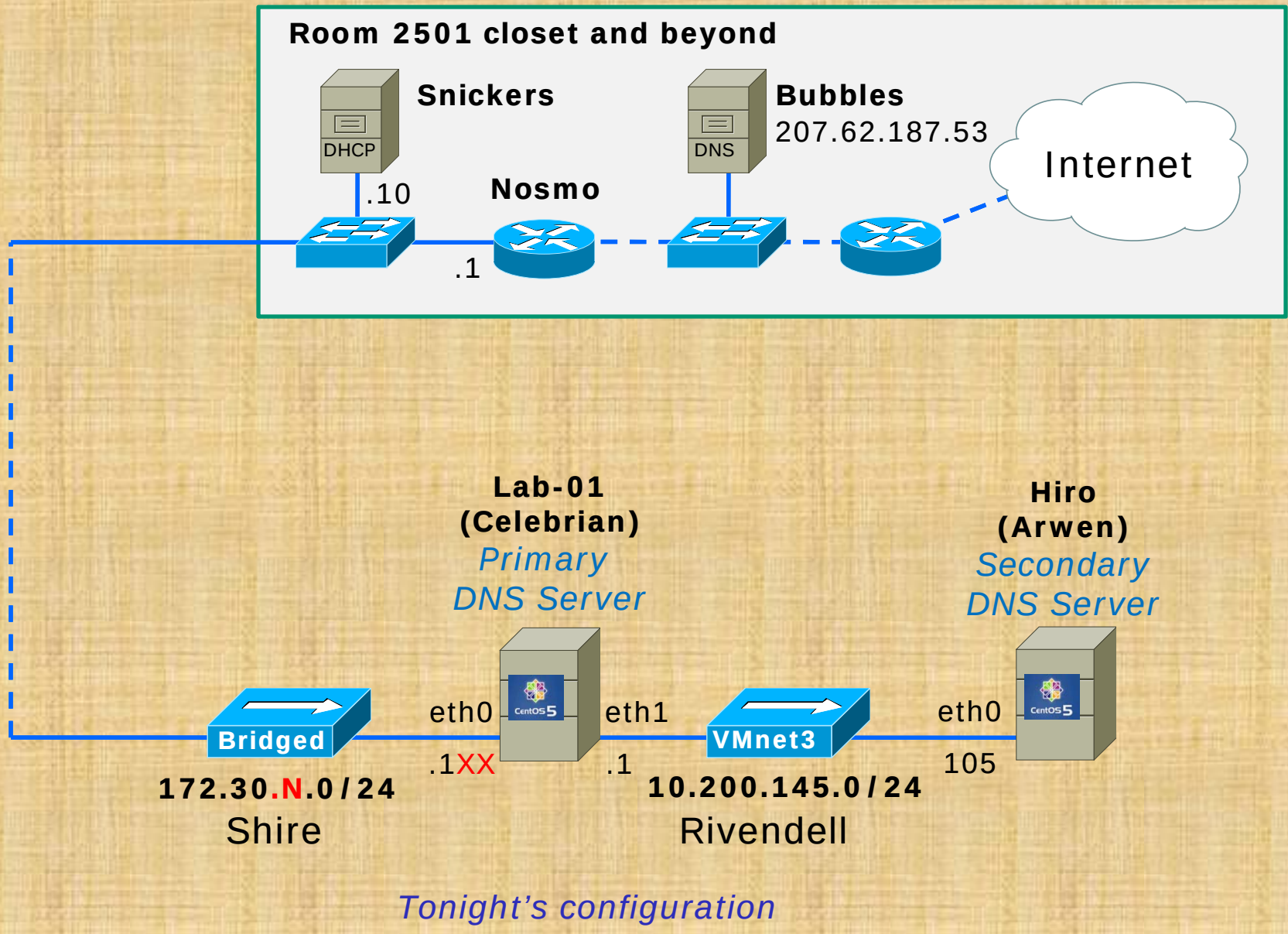
Warm-up

For next activity

VMs for tonight
(**Revert**, **384MB** RAM and **power up**)
Arwen Celebrian Sniffer

If at first you don't succeed, try, try again!

- *Update script prompts for Opus logname*



Tonight's configuration

Activity – Download Celebrian scripts

1. Cable Celebrian's eth0 to the Shire network and connect with: **dhclient eth0**
2. Change to root's bin directory if not there already with: **cd /root/bin**
3. Pull down Celebrian scripts with:

   scp logname@opus.cabrillo.edu:/home/cis192/scripts/*celebrian /root/bin
4. Set execute permission with **chmod 700 /root/bin/***
5. Run script with: **./update-scripts-celebrian** (Enter y for all ?'s)
6. Set execute permission on all new scripts with **chmod 700 /root/bin/***
7. Release IP address with: **dhclient –r**
8. Verify files:

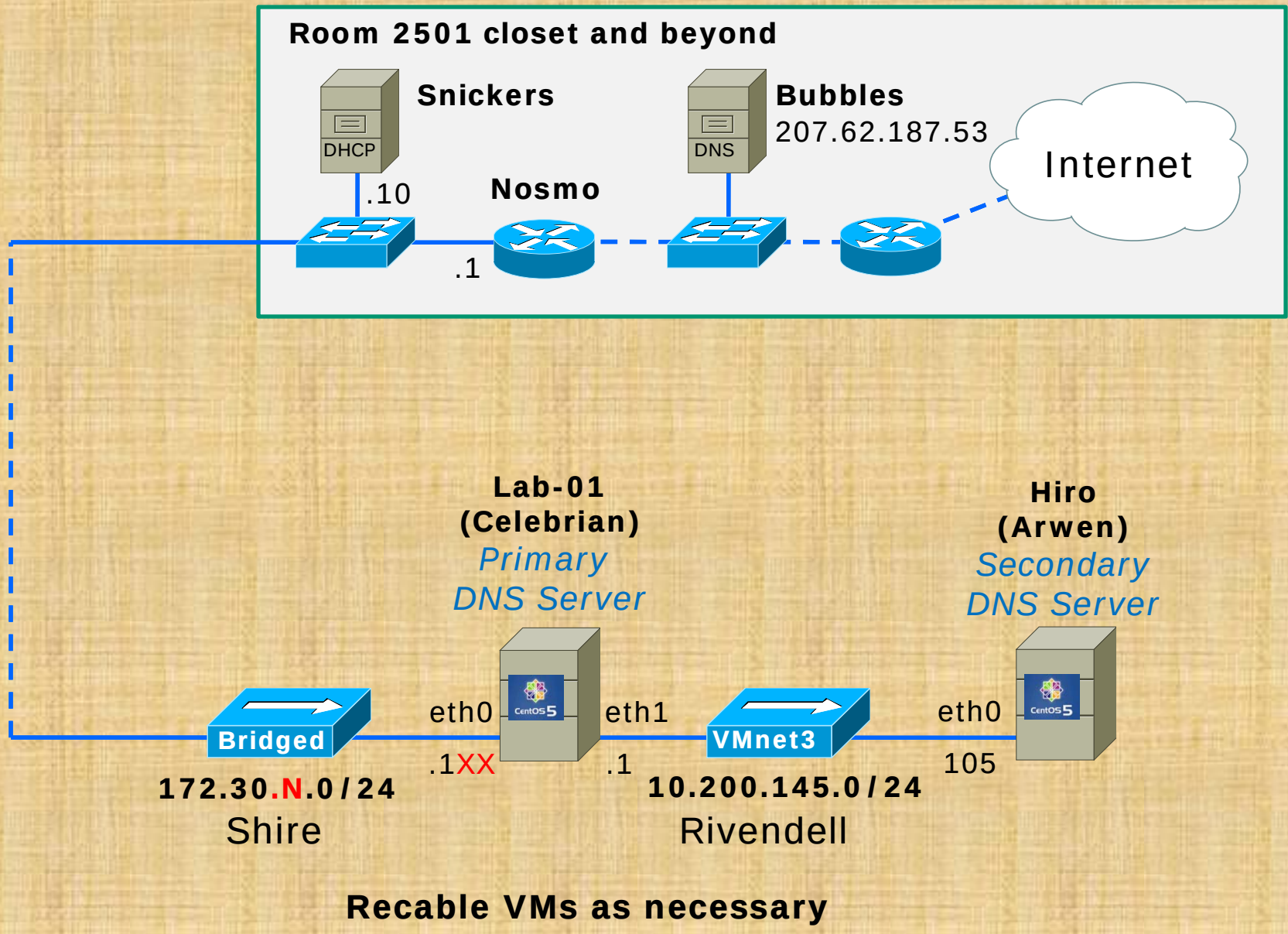
```
[root@celebrian bin]# ls /root/bin
do-act8A-celebrian      set-dns-centos         set-interface-centos
do-act9A-celebrian      set-forwarding-centos  set-route-centos
init-network-centos     set-gateway-centos     show-network-centos
restart-network-centos  set-hostname-centos    update-scripts-celebrian
[root@celebrian bin]#

[root@celebrian bin]# ls /root/packages/{bind*,caching*}
/root/packages/bind-9.3.6-4.P1.el5_4.2.i386.rpm
/root/packages/bind-libs-9.3.6-4.P1.el5_4.2.i386.rpm
/root/packages/bind-utils-9.3.6-4.P1.el5_4.2.i386.rpm
/root/packages/caching-nameserver-9.3.6-4.P1.el5_4.2.i386.rpm
[root@celebrian bin]#
```

Activity – Download Arwen scripts

1. Cable Celebrian's eth0 to the Shire network and connect with: **dhclient eth0**
2. Change to root's bin directory if not there already with: **cd /root/bin**
3. Pull down Celebrian scripts with:

   scp logname@opus.cabrillo.edu:/home/cis192/scripts/*arwen /root/bin
4. Set execute permission with **chmod 700 /root/bin/***
5. Run script with: **./update-scripts-arwen** (Enter y for all ?'s)
6. Set execute permission on all new scripts with **chmod 700 /root/bin/***
7. Release IP address with: **dhclient –r**
8. Verify files:

```
[root@arwen bin]# ls
do-act8A-arwen           set-dns-centos         set-interface-centos
do-act9A-arwen           set-forwarding-centos  set-route-centos
init-network-centos      set-gateway-centos     show-network-centos
restart-network-centos   set-hostname-centos    update-scripts-arwen
[root@arwen bin]#

[root@arwen bin]# ls /root/packages/{bind*,caching*}
/root/packages/bind-9.3.6-4.P1.el5_4.2.i386.rpm
/root/packages/bind-libs-9.3.6-4.P1.el5_4.2.i386.rpm
/root/packages/bind-utils-9.3.6-4.P1.el5_4.2.i386.rpm
/root/packages/caching-nameserver-9.3.6-4.P1.el5_4.2.i386.rpm
[root@arwen bin]#
```

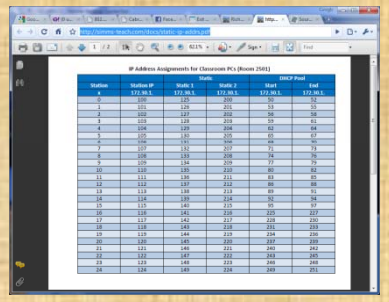


Recable VMs as necessary

Customize do-act9A-celebrian script

```
[root@celebrian bin]# head -15 do-act9A-celebrian
#!/bin/bash
#
# Do Activity 9A on Celebrian
#

# Modify the following lines for static IP your workstation
# using http://simms-teach.com/docs/static-ip-addrs.pdf

# Station-00 in classroom
static1=172.30.1.1XX
router=172.30.1.1
# CIS-Lab-06 in lab
#static1=172.30.4.131
#router=172.30.4.1

[root@celebrian bin]#
```

Modify to your unique static IP address from



http://simms-teach.com/docs/static-ip-addrs.pdf

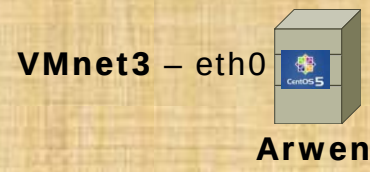
Activity – Peer Walkthrough

The power of a second set of eyes is invaluable!

1. Pair up with another student

2. Verify **Celebrian** and **Arwen** VMs:
   - ❑ Logged on as root
   - ❑ Scripts are in root's bin directory
   - ❑ RPMs are in root's packages directory
   - ❑ The "do-*" scripts match the VM's name
   - ❑ The other scripts match VM's distro (CentOS)
   - ❑ Execute permission has been set on all scripts
   - ❑ Cabling is correct

3. Verify the do-act9A-celebrian script on **Celebrian** has the correct eth0 IP address



Activity 9A

1. On Celebrian, in /root/bin, use:

   ./do-act9A-celebrian

Use Enter key to confirm each step and continue

2. On Arwen, in /root/bin, use:

   ./do-act9A-arwen

If all worked as planned:

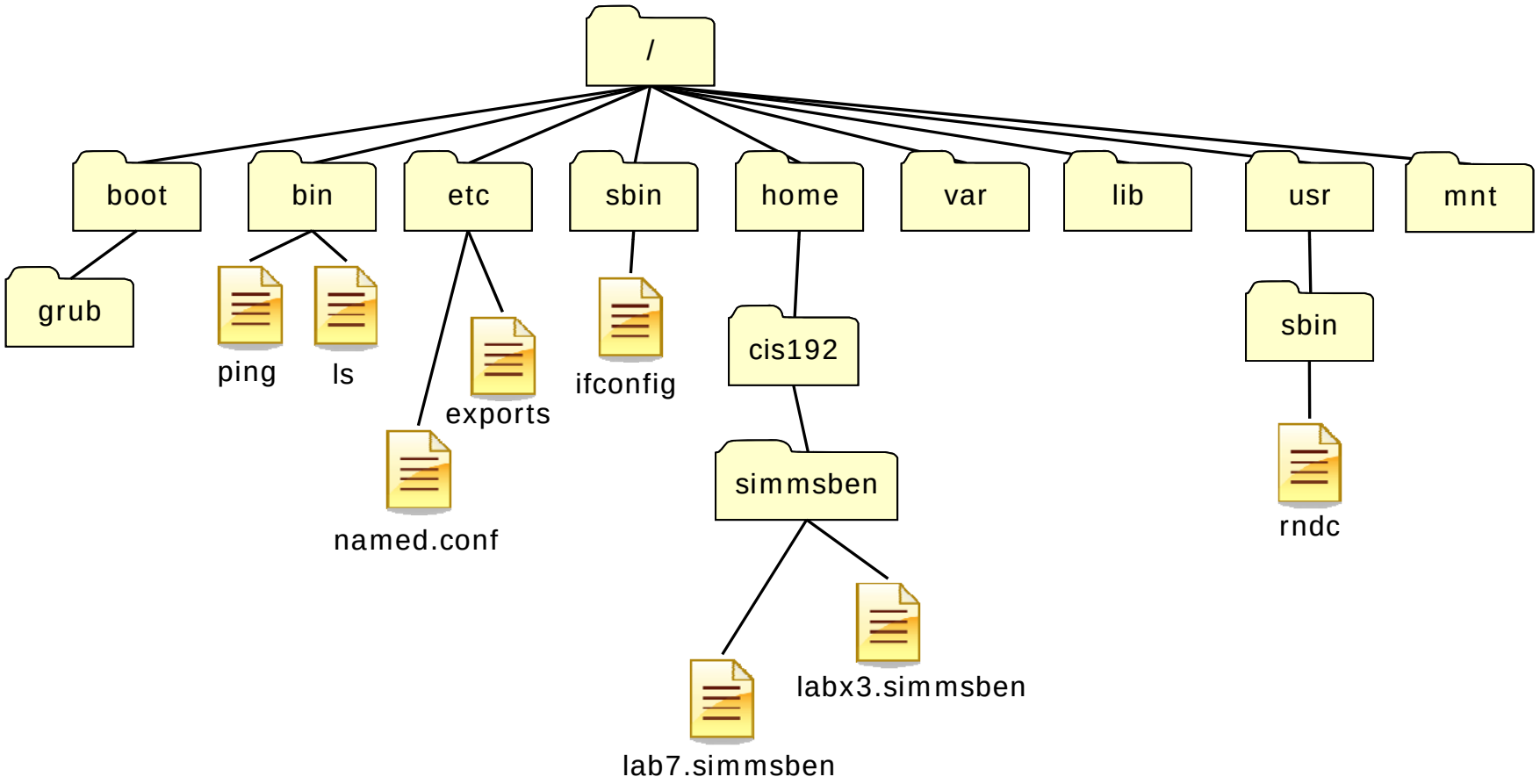
- You should be able to ping Hiro and Lab-01 from either system
- Both systems should have Internet access
- Updated prompt string using new hostnames after logging out and back in again

Mounting File Systems

Mounting File Systems

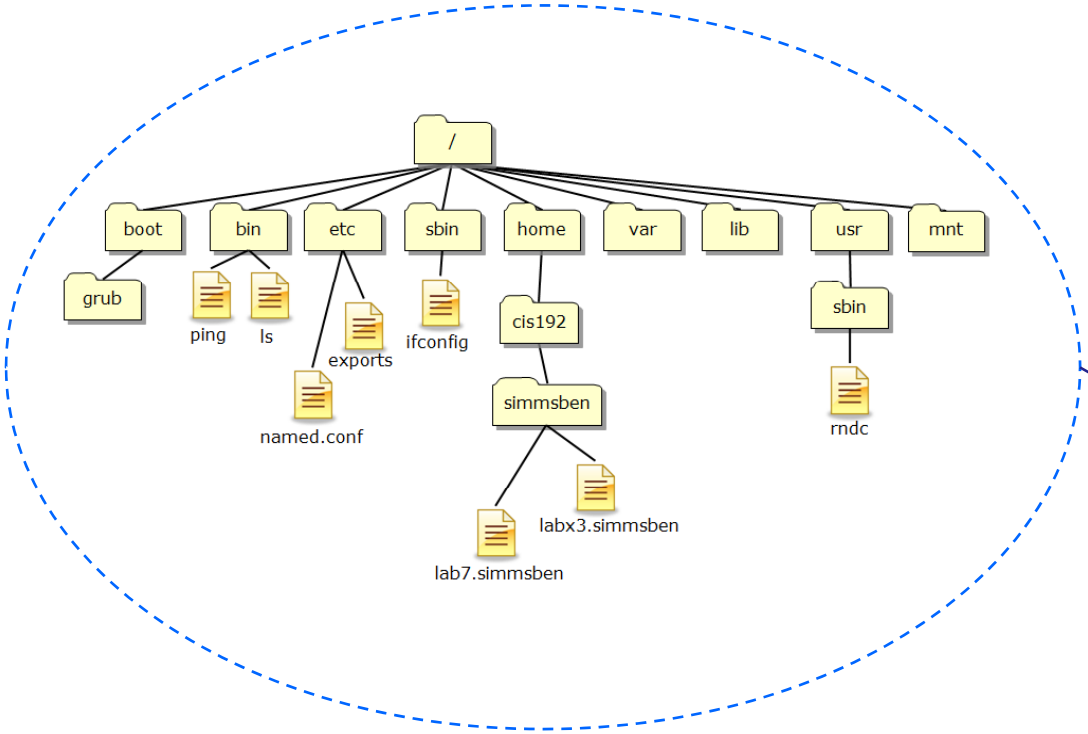
- The UNIX file tree offers a lot of flexibility in how it can be mapped to storage devices
- The tree may span multiple partitions, kernel memory locations, multiple hard drives, optical drives, flash drives and even directories on other computers

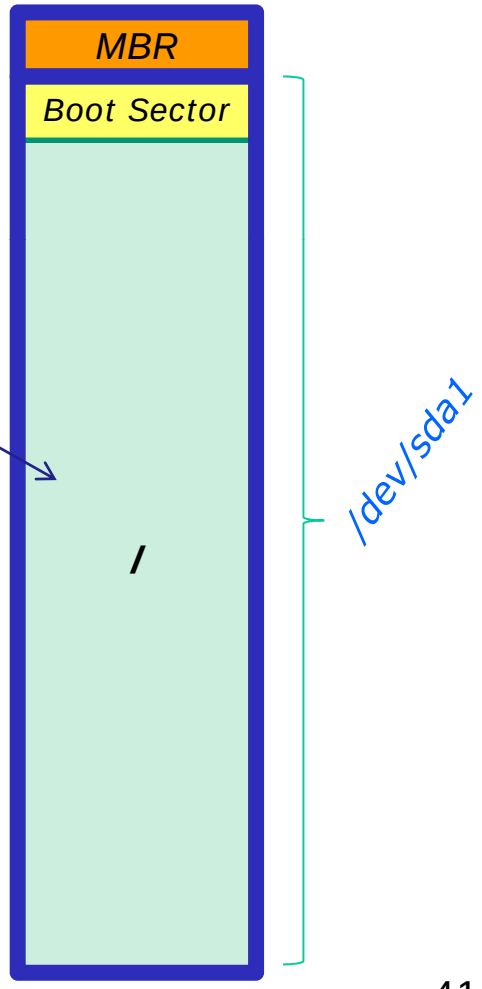
Mounting File Systems



A example UNIX file tree

Mounting File Systems

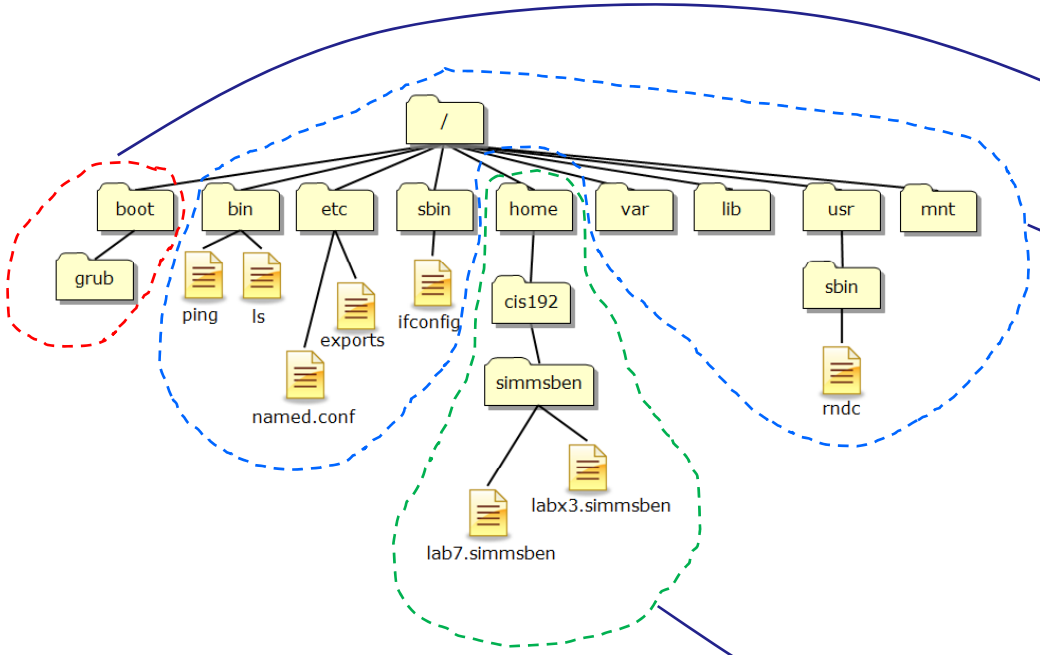


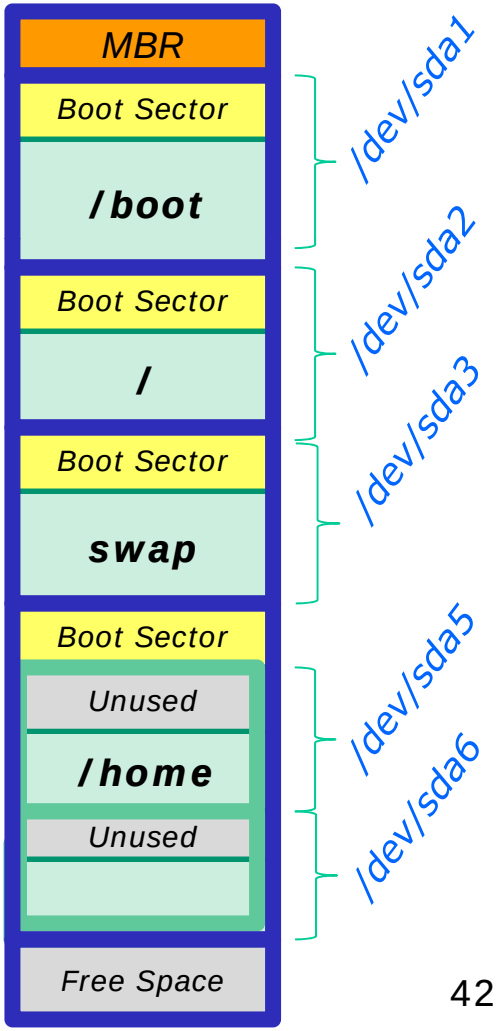


/dev/sda1 is mounted on /

This file tree uses only one partition on a single drive

Mounting File Systems

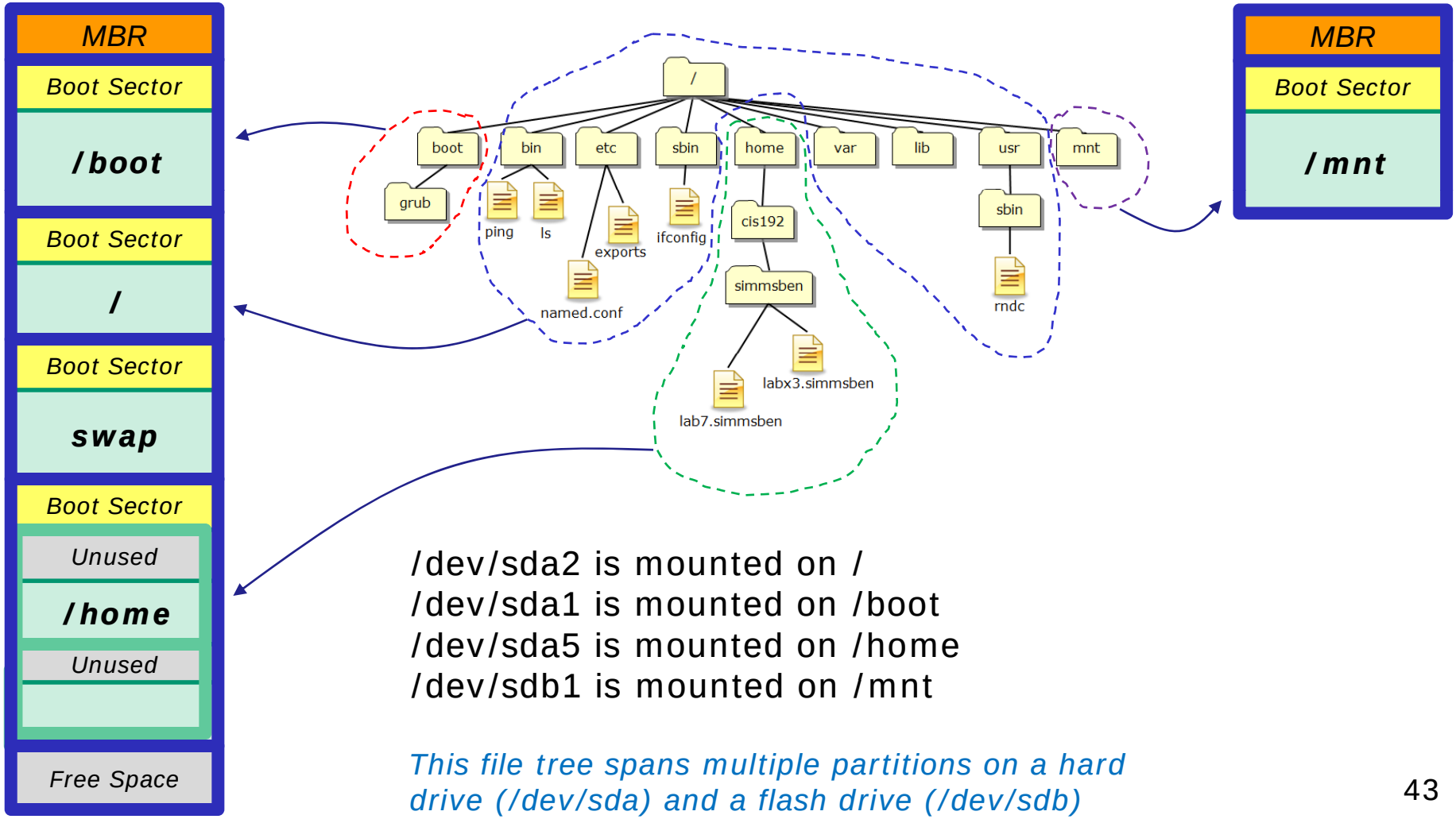




/dev/sda2 is mounted on /
/dev/sda1 is mounted on /boot
/dev/sda5 is mounted on /home

This file tree spans multiple partitions on a drive

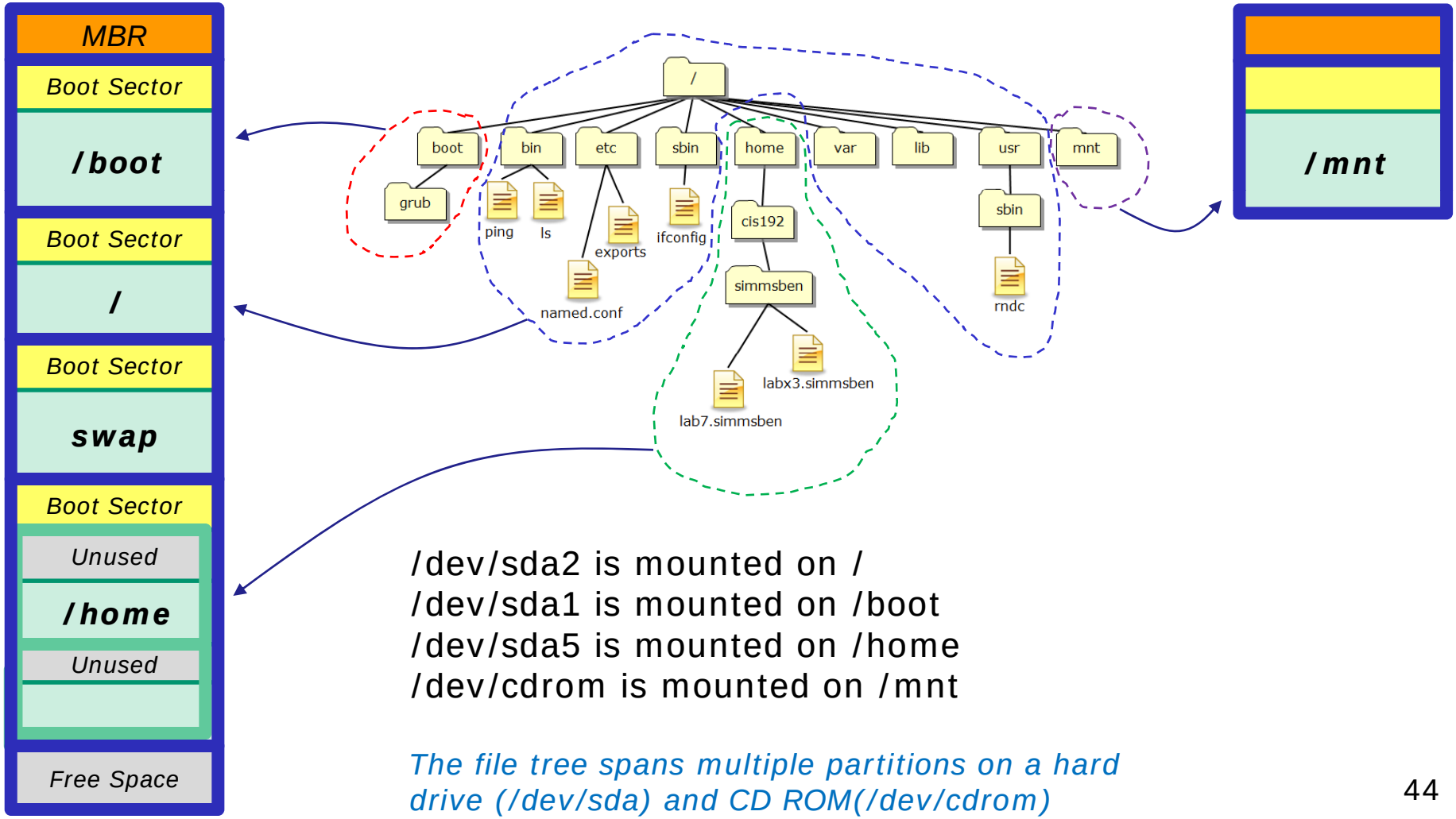
Mounting File Systems



/dev/sda2 is mounted on /
/dev/sda1 is mounted on /boot
/dev/sda5 is mounted on /home
/dev/sdb1 is mounted on /mnt

This file tree spans multiple partitions on a hard drive (/dev/sda) and a flash drive (/dev/sdb)

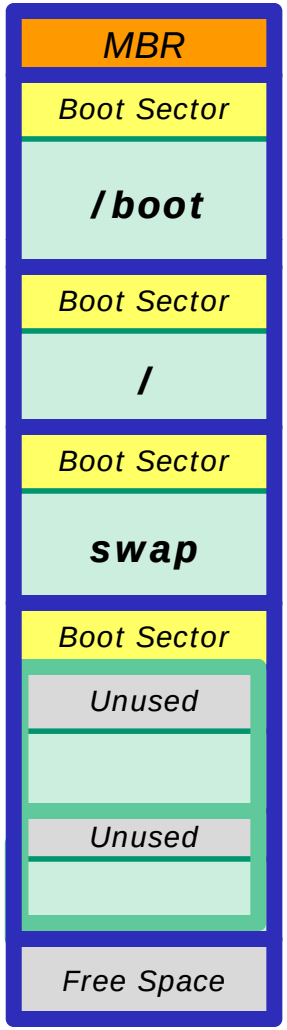
Mounting File Systems

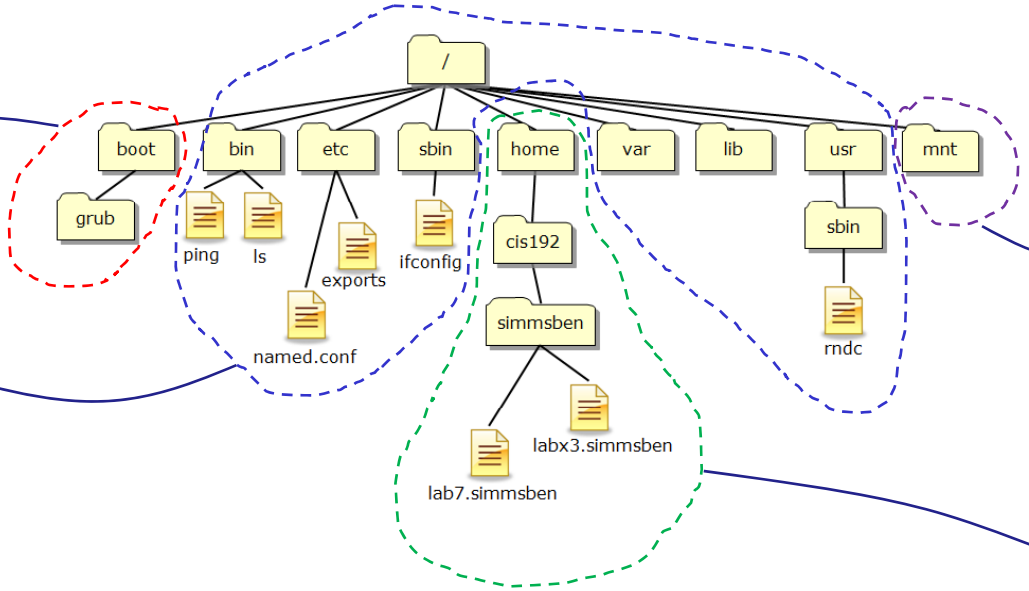


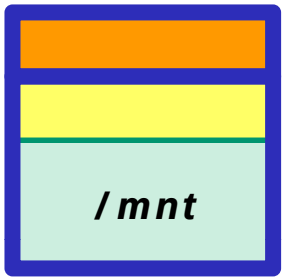
/dev/sda2 is mounted on /
/dev/sda1 is mounted on /boot
/dev/sda5 is mounted on /home
/dev/cdrom is mounted on /mnt

The file tree spans multiple partitions on a hard drive (/dev/sda) and CD ROM(/dev/cdrom)

Mounting File Systems







NFS exported directory on remote system

Hiro:/home

/dev/sda2 is mounted on /
/dev/sda1 is mounted on /boot
Hiro:/home is mounted on /home
/dev/cdrom is mounted on /mnt

This file tree spans multiple partitions on a hard drive (/dev/sda) and CD ROM(/dev/cdrom)

Mounting File Systems

Mounting commands

- syntax: **mount** *device-file directory*
  - example: **mount /dev/cdrom /mnt**

    Mount the CD on the /mnt directory
  - example: **mount /dev/sdb1 /home**

    Mount the 1st partition of the 2nd drive on the /home directory
- syntax: **umount *device-file* | *directory***
  - example: **umount /mnt**

Mount information

1. /etc/fstab *what to mount at boot time*
2. /etc/mtab *current mount status*

Understanding mount command output

Showing current mount status

On a rh9 VM

```
[root@rh9 root]# mount
/dev/sda2 on / type ext3 (rw)
none on /proc type proc (rw)
/dev/sda1 on /boot type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda5 on /home type ext3 (rw)
none on /dev/shm type tmpfs (rw)

[root@rh9 root]# cat /etc/mtab
/dev/sda2 / ext3 rw 0 0
none /proc proc rw 0 0
/dev/sda1 /boot ext2 rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
/dev/sda5 /home ext3 rw 0 0
none /dev/shm tmpfs rw 0 0
[root@rh9 root]#
```

*The **mount** command without any arguments shows current mount status. Same info is in /etc/mtab*

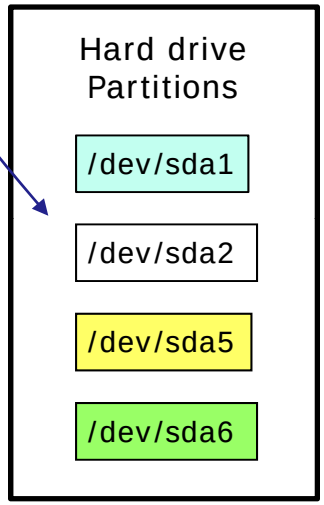
Note the relationship between the mount command output and /etc/mtab

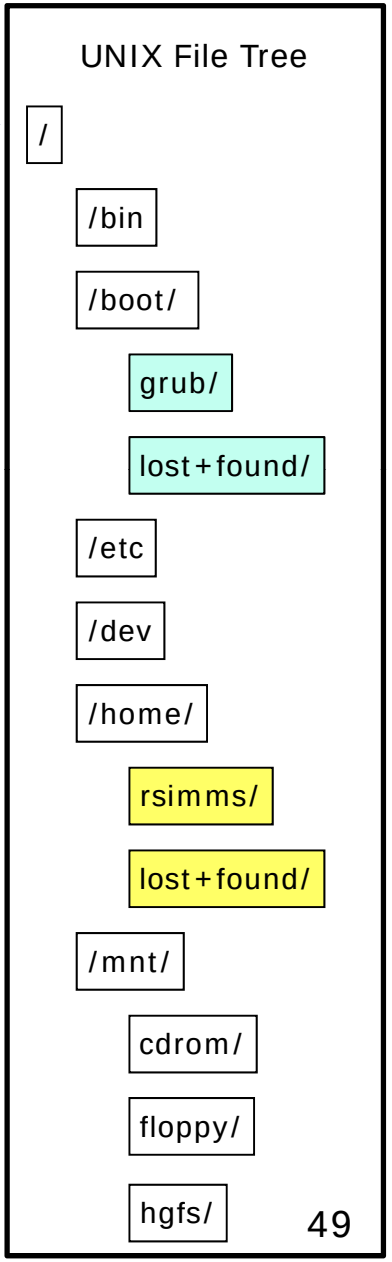
Showing current mount status

On a rh9 VM

```
[root@rh9 root]# mount
/dev/sda2 on / type ext3 (rw)
none on /proc type proc (rw)
/dev/sda1 on /boot type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda5 on /home type ext3 (rw)
none on /dev/shm type tmpfs (rw)

[root@rh9 root]# cat /etc/mtab
/dev/sda2 / ext3 rw 0 0
none /proc proc rw 0 0
/dev/sda1 /boot ext2 rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
/dev/sda5 /home ext3 rw 0 0
none /dev/shm tmpfs rw 0 0
[root@rh9 root]#
```

The second partition on the hard drive is mounted on the / directory in the UNIX file tree





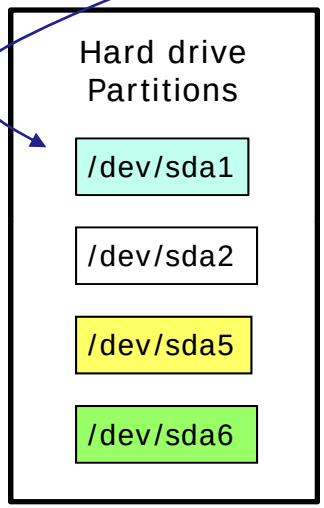
Using mount command output to understand how the file tree maps to storage devices

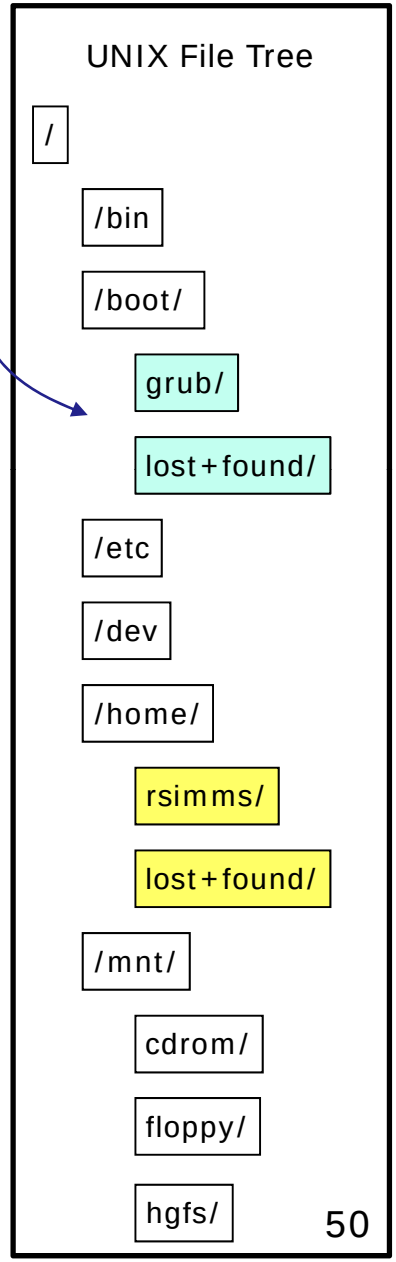
Showing current mount status

On a rh9 VM

```
[root@rh9 root]# mount
/dev/sda2 on / type ext3 (rw)
none on /proc type proc (rw)
/dev/sda1 on /boot type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda5 on /home type ext3 (rw)
none on /dev/shm type tmpfs (rw)

[root@rh9 root]# cat /etc/mtab
/dev/sda2 / ext3 rw 0 0
none /proc proc rw 0 0
/dev/sda1 /boot ext2 rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
/dev/sda5 /home ext3 rw 0 0
none /dev/shm tmpfs rw 0 0
[root@rh9 root]#
```

The first partition on the hard drive is mounted on the /boot directory in the UNIX file tree






```
[root@rh9 root]# ls -F /boot
/boot:
boot.b                kernel.h       module-info-2.4.20-6  vmlinuz@
chain.b               lost+found/    os2_d.b               vmlinuz-2.4.20-6
config-2.4.20-6       message        System.map@
grub/                 message.ja     System.map-2.4.20-6
initrd-2.4.20-6.img   module-info@   vmlinux-2.4.20-6
```

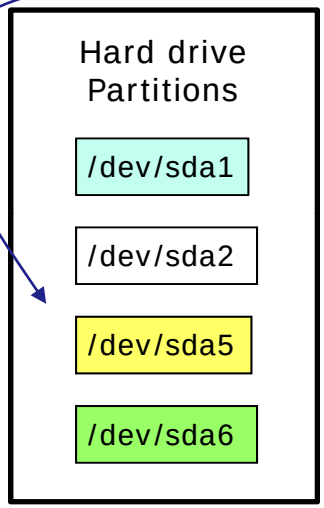
Using mount command output to understand how the file tree maps to storage devices

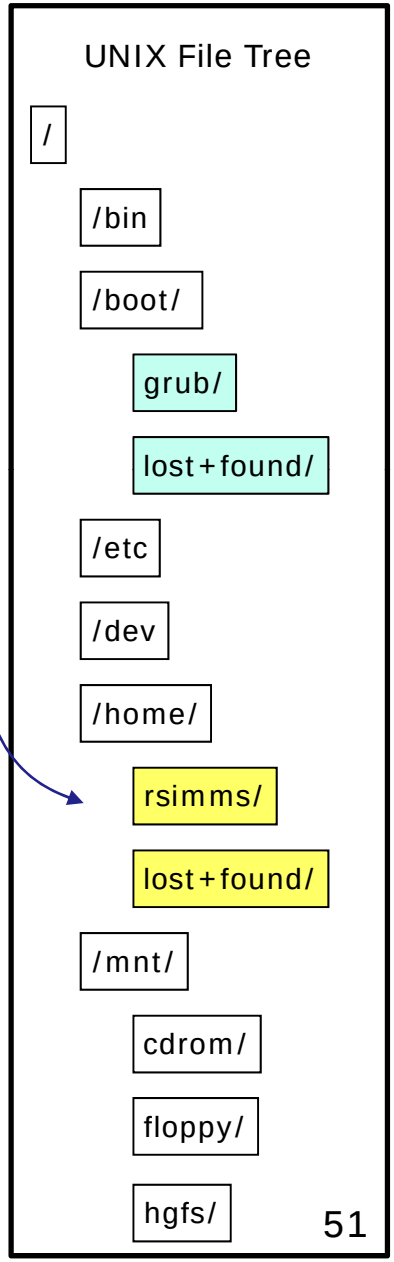
Showing current mount status

On a rh9 VM

```
[root@rh9 root]# mount
/dev/sda2 on / type ext3 (rw)
none on /proc type proc (rw)
/dev/sda1 on /boot type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda5 on /home type ext3 (rw)
none on /dev/shm type tmpfs (rw)

[root@rh9 root]# cat /etc/mtab
/dev/sda2 / ext3 rw 0 0
none /proc proc rw 0 0
/dev/sda1 /boot ext2 rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
/dev/sda5 /home ext3 rw 0 0
none /dev/shm tmpfs rw 0 0
[root@rh9 root]#



[root@rh9 root]# ls -F /home

/home:
lost+found/  rsimms/
```

The fifth partition on the hard drive is mounted on the /home directory in the UNIX file tree





Using mount command output to understand how the file tree maps to storage devices

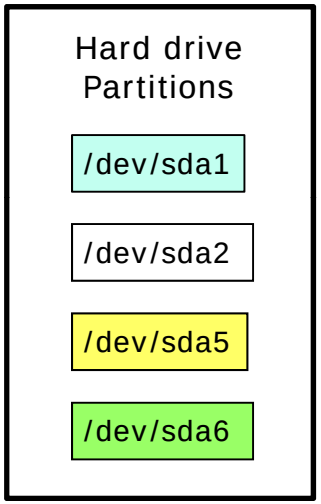
The great cover-up

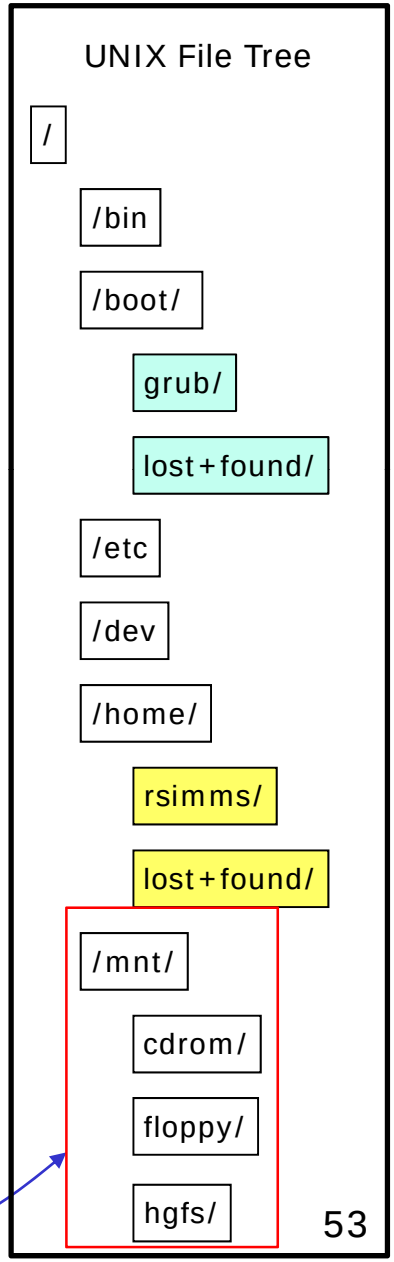
Where did those files go?

Showing current mount status

On a rh9 VM

```
[root@rh9 root]# mount
/dev/sda2 on / type ext3 (rw)
none on /proc type proc (rw)
/dev/sda1 on /boot type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda5 on /home type ext3 (rw)
none on /dev/shm type tmpfs (rw)

[root@rh9 root]# cat /etc/mtab
/dev/sda2 / ext3 rw 0 0
none /proc proc rw 0 0
/dev/sda1 /boot ext2 rw 0 0
none /dev/pts devpts rw,gid=5,mode=620 0 0
/dev/sda5 /home ext3 rw 0 0
none /dev/shm tmpfs rw 0 0
[root@rh9 root]#

[root@rh9 root]# ls -F /home /boot /mnt
/boot:
boot.b                kernel.h      module-info-2.4.20-6  vmlinuz@
chain.b               lost+found/   os2_d.b               vmlinuz-2.4.20-6
config-2.4.20-6       message       System.map@
grub/                 message.ja    System.map-2.4.20-6
initrd-2.4.20-6.img   module-info@  vmlinux-2.4.20-6

/home:
lost+found/  rsimms/

/mnt:
cdrom/  floppy/  hgfs/
```



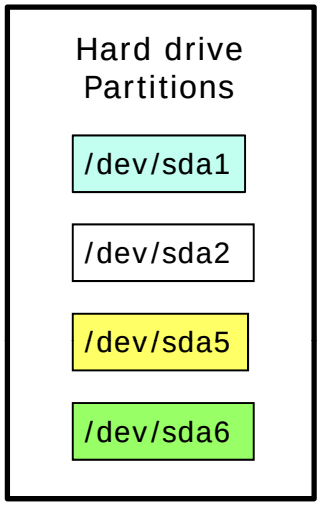


*/dev/sda6 is **not** mounted yet, note contents of the /mnt directory*

Making and mounting a filesystem

On the rh9 VM

```
[root@rh9 root]# mkfs -t ext3 /dev/sda6
mke2fs 1.32 (09-Nov-2002)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
26104 inodes, 104391 blocks
5219 blocks (5.00%) reserved for the super user
First data block=1
13 block groups
8192 blocks per group, 8192 fragments per group
2008 inodes per group
Superblock backups stored on blocks:
        8193, 24577, 40961, 57345, 73729

Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 39 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```



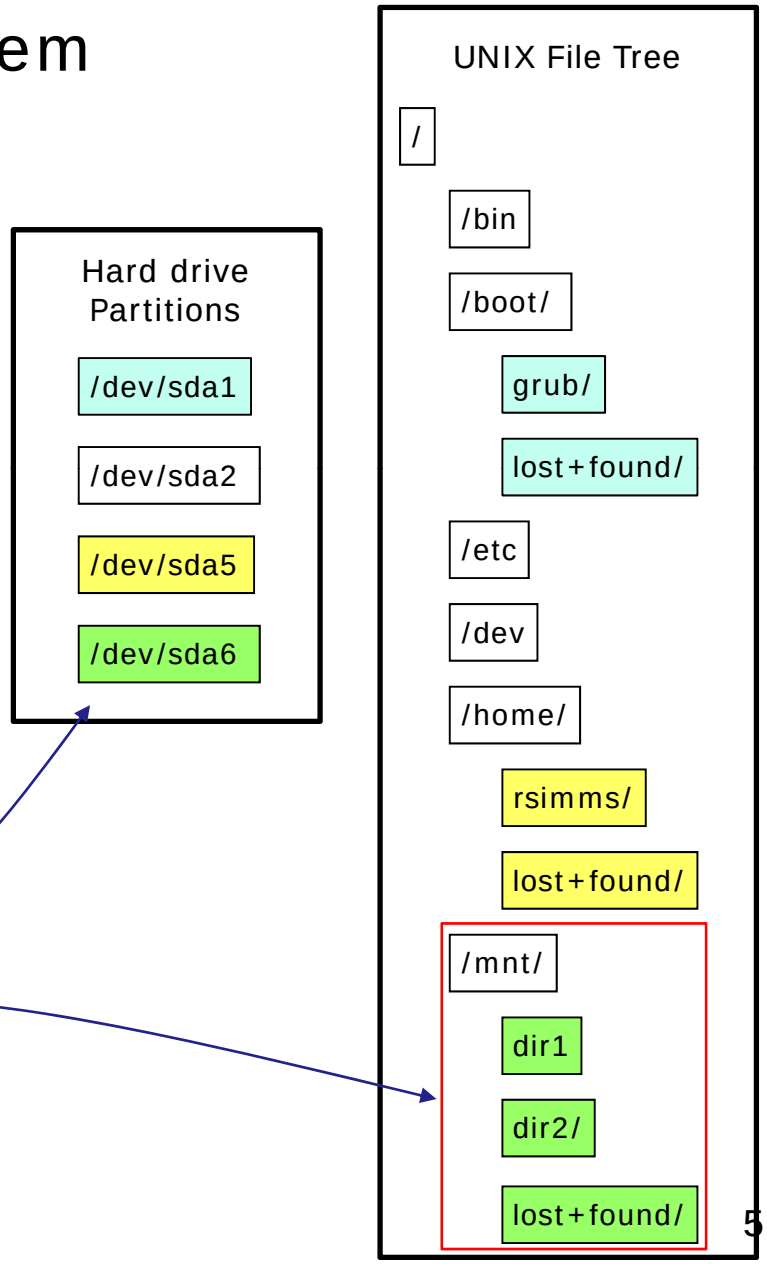
Lets make a new ext3 filesystem on /dev/sda6

Making and mounting a filesystem

On the rh9 VM

```
[root@rh9 root]# mount /dev/sda6 /mnt
[root@rh9 root]# cd /mnt
[root@rh9 mnt]# mkdir dir1 dir2
[root@rh9 mnt]# ls
dir1  dir2  lost+found
[root@rh9 mnt]# cd
[root@rh9 root]# ls /mnt
dir1  dir2  lost+found
```

*Lets mount the new filesystem on **/mnt** and populate it with some example directories*

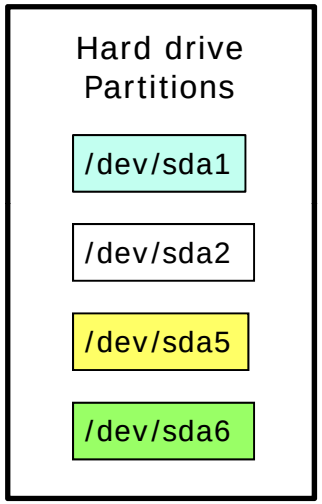


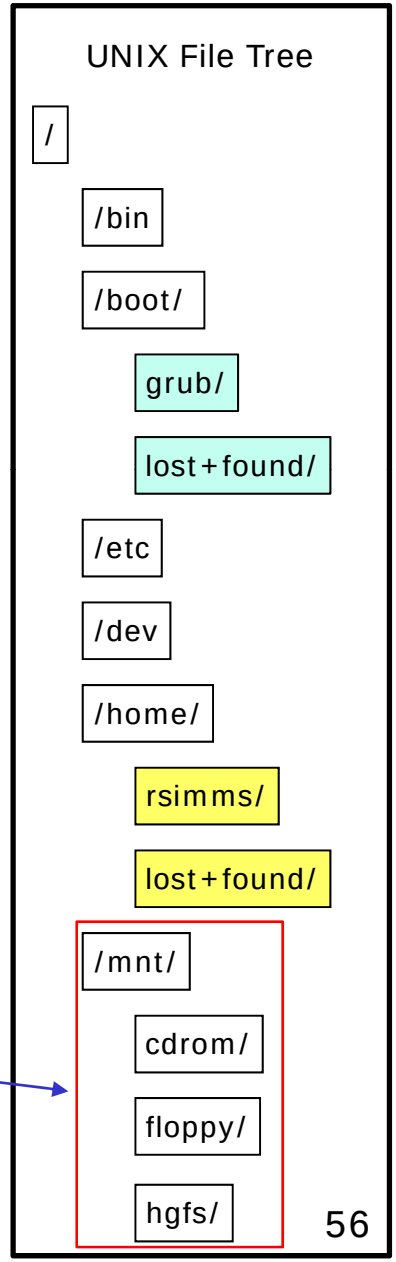
Making and mounting a filesystem

On the rh9 VM

```
[root@rh9 root]# umount /mnt
[root@rh9 root]# ls /mnt
cdrom  floppy  hgfs
[root@rh9 root]#
```

The umount command reverts the UNIX file tree back to the original directories

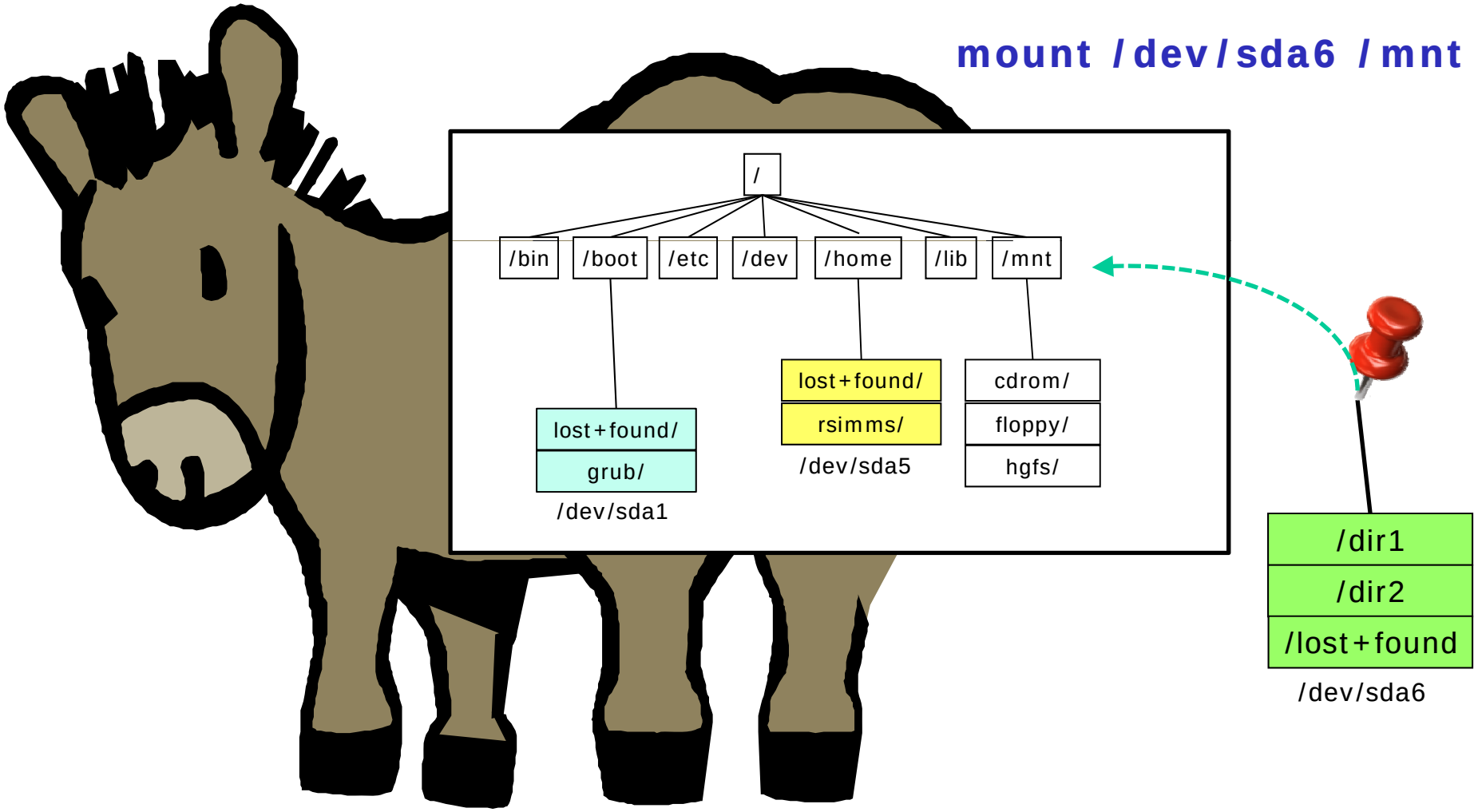




Pin the tail on the donkey

Mounting File Systems

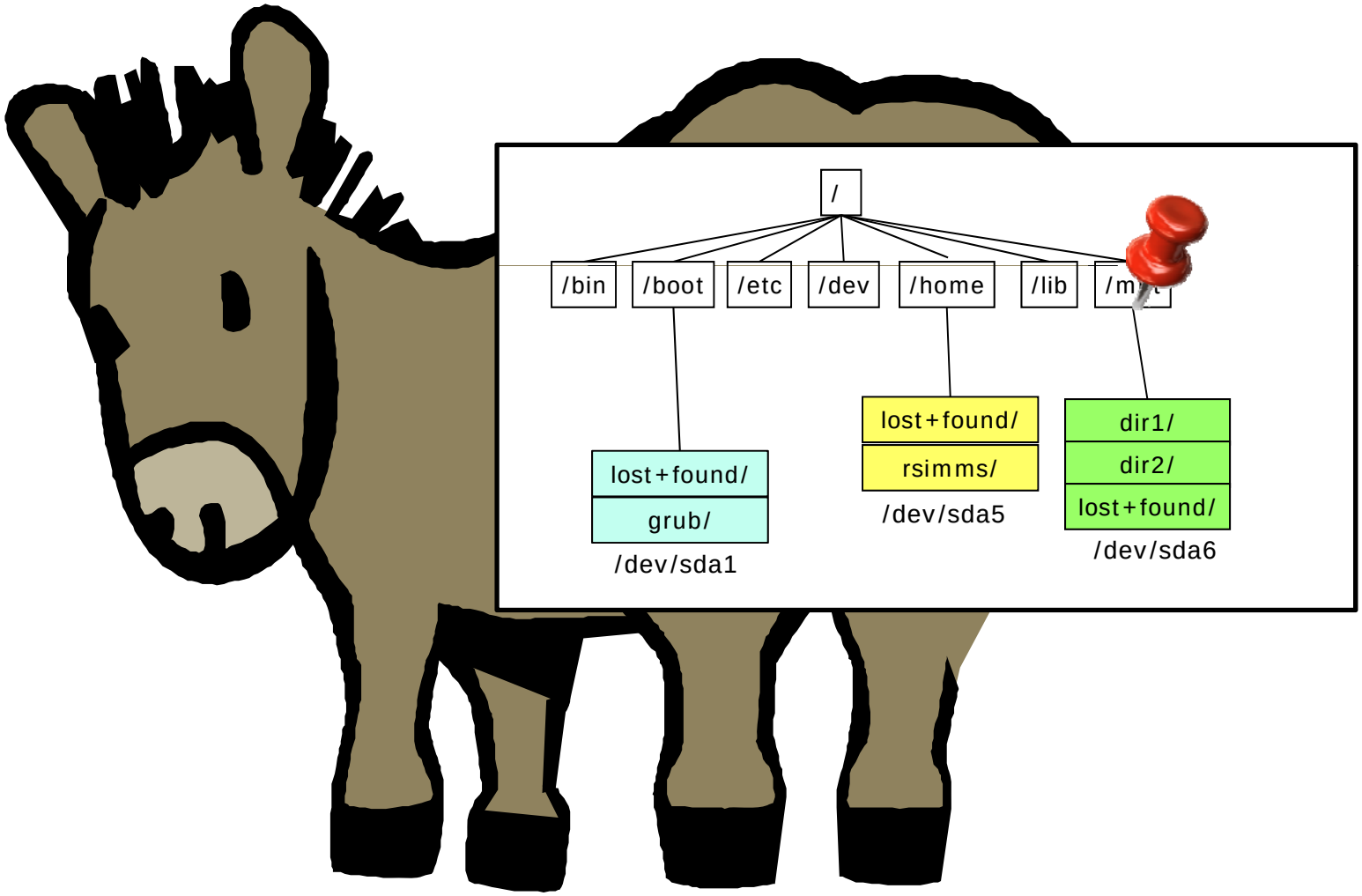
Like pinning the tail on the donkey



Initially the /mnt directory is mapped to files on /dev/sda2

Mounting File Systems

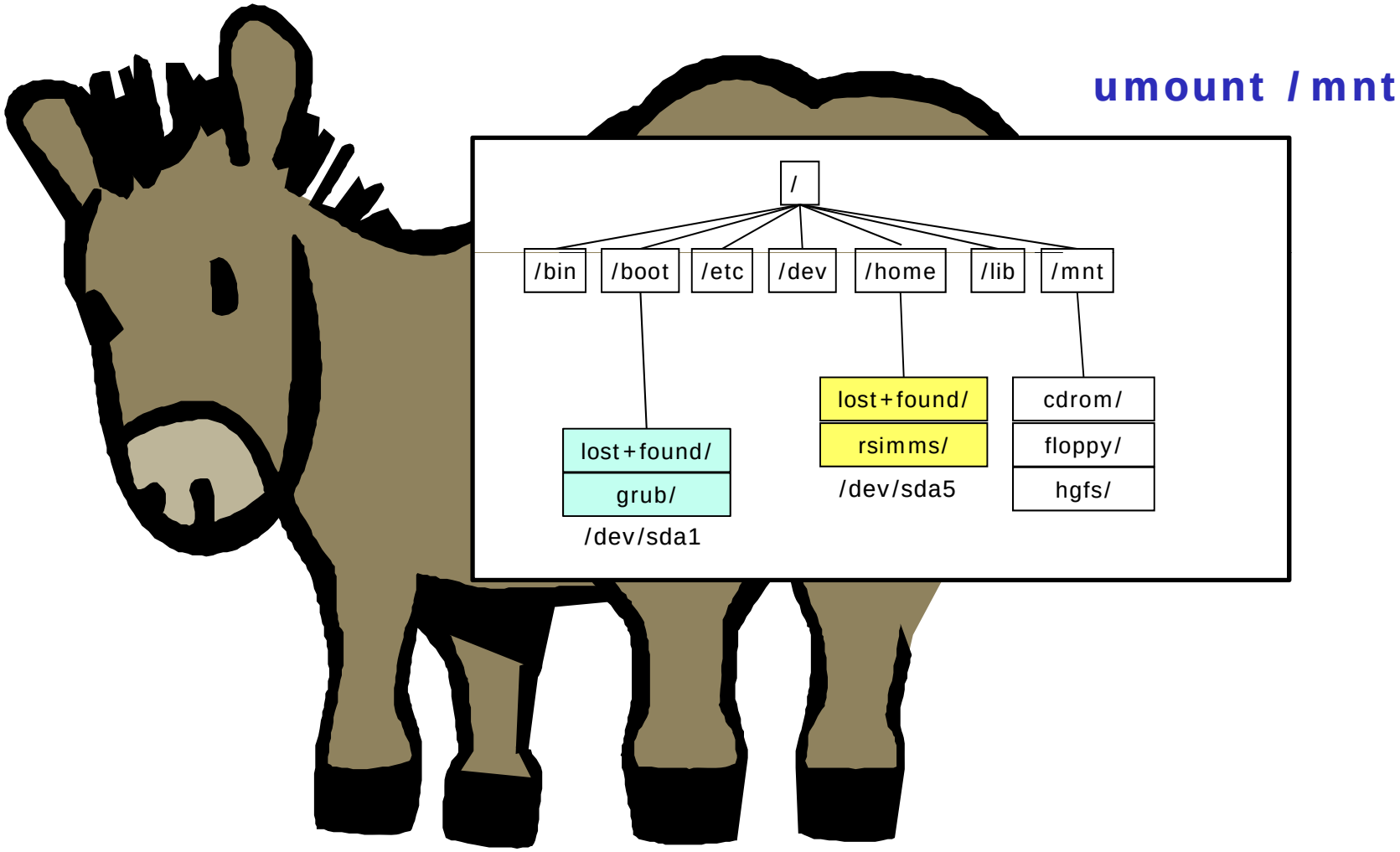
Like pinning the tail on the donkey



After the mount command, /mnt is mapped to /dev/sda6 files

Mounting File Systems

Like pinning the tail on the donkey



After the umount command, /mnt is mapped back to files on /dev/sda2

Mounting File Systems
Caveats

- Don’t mount a file system to a directory you are in.
- You can’t un-mount a file system you have cd’ed into:

```
[root@rh9 mnt]# umount /mnt
umount: /mnt: device is busy

[root@rh9 mnt]# cd
[root@rh9 root]# umount /mnt
[root@rh9 root]#
```

Must cd out of the mounted directory before it can be unmounted

Mounting File Systems

mount command and /etc/mtab

device	mount point	file system type	mount options	dump frequency	fsck pass
[root@rh9	root]# **mount**				
/dev/sda2	on /	type ext3	(rw)		
none	on /proc	type proc	(rw)		
/dev/sda1	on /boot	type ext2	(rw)		
none	on /dev/pts	type devpts	(rw,gid=5,mode=620)		
/dev/sda5	on /home	type ext3	(rw)		
none	on /dev/shm	type tmpfs	(rw)		
[root@rh9	root]# **cat**	**/etc/mtab**			
/dev/sda2	/	ext3	rw	0	0
none	/proc	proc	rw	0	0
/dev/sda1	/boot	ext2	rw	0	0
none	/dev/pts	devpts	rw,gid=5,mode=620	0	0
/dev/sda5	/home	ext3	rw	0	0
none	/dev/shm	tmpfs	rw	0	0

Note: spaces added to output above for readability

Mounting File Systems

/etc/fstab

/etc/fstab is used to automatically mount file systems at boot time

*Use **man fstab** for details*

only 1's will be backed up by dump

fsck order, 0's not checked

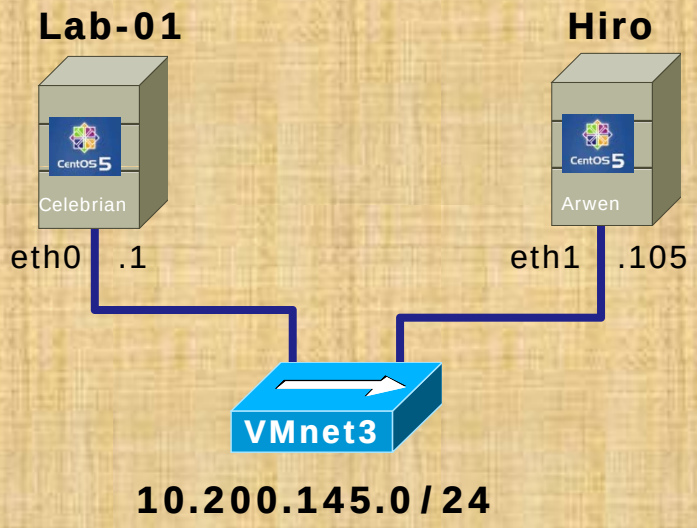
```
[root@rh9 root]# cat /etc/fstab
```

device	mount point	file system type	mount options	dump frequency	fsck pass
LABEL=/	/	ext3	defaults	1	1
LABEL=/boot	/boot	ext2	defaults	1	2
none	/dev/pts	devpts	gid=5,mode=620	0	0
LABEL=/home	/home	ext3	defaults	1	2
none	/proc	proc	defaults	0	0
none	/dev/shm	tmpfs	defaults	0	0
/dev/sda3	swap	swap	defaults	0	0
/dev/cdrom	/mnt/cdrom	udf,iso9660	noauto,owner,kudzu,ro	0	0
/dev/fd0	/mnt/floppy	auto	noauto,owner,kudzu	0	0

Huh? So what the heck partition is LABEL=/? Use tune2fs to show volume names (labels)

```
[root@rh9 root]# tune2fs -l /dev/sda1 | grep name
Filesystem volume name:   /boot
[root@rh9 root]# tune2fs -l /dev/sda2 | grep name
Filesystem volume name:   /
[root@rh9 root]# tune2fs -l /dev/sda5 | grep name
Filesystem volume name:   /home
[root@rh9 root]#
```

Mounts



On Arwen (Hiro)

- Issue **fdisk –l** (lower case L) to see partitions
- Issue **mount** command to see how partitions are mounted on UNIX file tree
- Review /etc/mtab
- Review /etc/fstab

Hmmm, where is /dev/sda2?

LVM sidetrack

Logical Volume Manager

```
[root@hiro ~]# fdisk -l

Disk /dev/sda: 5368 MB, 5368709120 bytes
255 heads, 63 sectors/track, 652 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          13      104391   83  Linux
/dev/sda2              14         652     5132767+  8e  Linux LVM
[root@hiro ~]#
```

```
[root@hiro ~]# mount
/dev/mapper/VolGroup00-LogVol00 on / type ext3 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda1 on /boot type ext3 (rw)
tmpfs on /dev/shm type tmpfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
nfsd on /proc/fs/nfsd type nfsd (rw)
[root@hiro ~]#
```

So what partition is this?

Where is /dev/sda2 mounted?

Logical Volume Manager

```
[root@hiro ~]# cat /etc/fstab
/dev/VolGroup00/LogVol00 /                       ext3    defaults        1 1
LABEL=/boot              /boot                   ext3    defaults        1 2
tmpfs                    /dev/shm                tmpfs   defaults        0 0
devpts                   /dev/pts                devpts  gid=5,mode=620  0 0
sysfs                    /sys                    sysfs   defaults        0 0
proc                     /proc                   proc    defaults        0 0
/dev/VolGroup00/LogVol01 swap                    swap    defaults        0 0
[root@hiro ~]#

[root@hiro ~]# tune2fs -l /dev/sda1 | grep volume
Filesystem volume name:   /boot
[root@hiro ~]#

[root@hiro ~]# tune2fs -l /dev/VolGroup00/LogVol00 | grep volume
Filesystem volume name:   <none>
[root@hiro ~]#
```

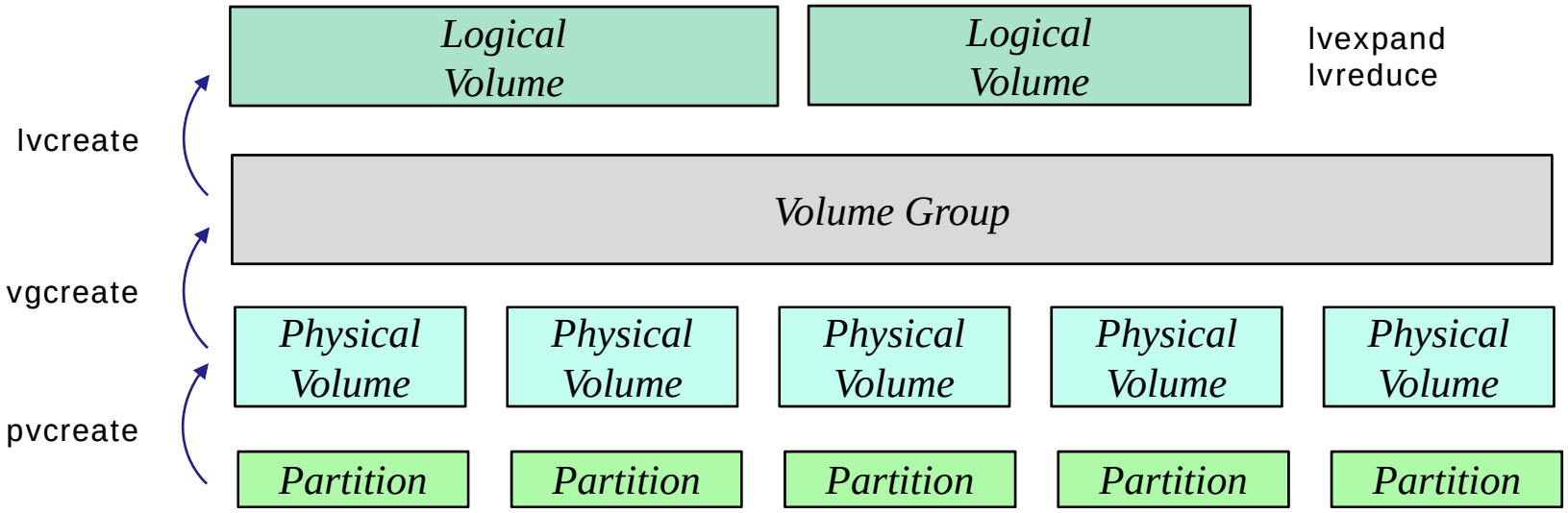
The /boot labeled partition is /dev/sda1

dev/VolGroup00/LogVol10x are logical volumes created by LVM

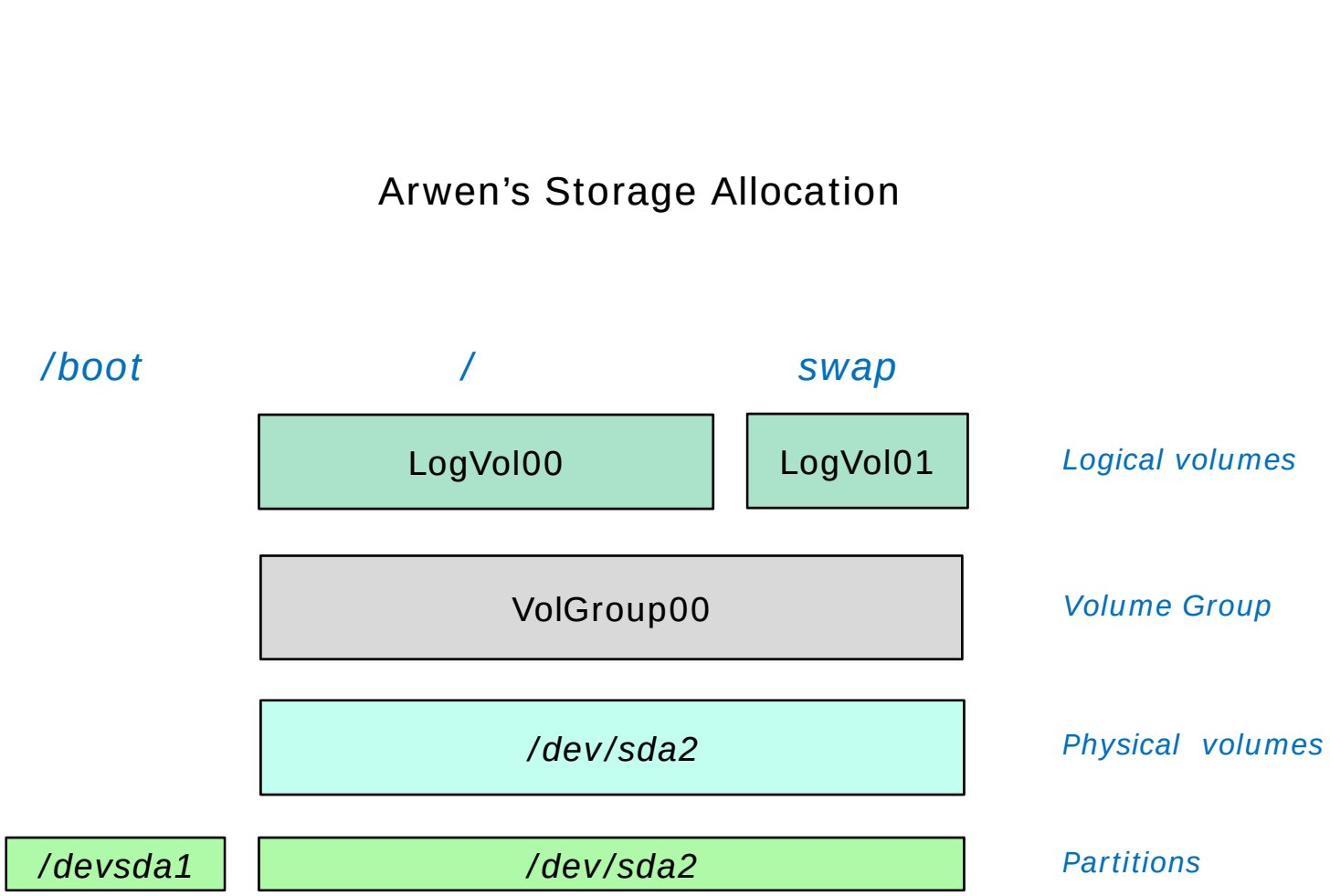
Note, LVM is used with default installations of Centos and RHEL 5.

Logical Volume Manager

- LVM lets you spread one or more logical volumes across one or more physical partitions.
- LVM provides more flexibility (with some additional complexity) for future changes.
- Logical volumes are very easy to resize compared to partitions.



Logical Volume Manager



Logical Volume Manager

LVM Commands:

pvscan - shows physical volumes

pvdisplay – shows physical volume info

vgscan – scans for volume groups

vgdisplay – shows volume group info

lvscan – scans for logical volumes

lvdisplay – shows logical volume info

Logical Volume Manager

```
[root@hiro ~]# pvscan
  PV /dev/sda2   VG VolGroup00   lvm2 [4.88 GB / 0    free]
  Total: 1 [4.88 GB] / in use: 1 [4.88 GB] / in no VG: 0 [0   ]
[root@hiro ~]#
```

← *Physical volume information*

```
[root@hiro ~]# pvdisplay
  --- Physical volume ---
  PV Name               /dev/sda2
  VG Name               VolGroup00
  PV Size               4.89 GB / not usable 20.47 MB
  Allocatable           yes (but full)
  PE Size (KByte)       32768
  Total PE              156
  Free PE               0
  Allocated PE          156
  PV UUID               Hj6Fx7-Ars0-7MMp-voCc-0L92-uPgN-Q8s4xr

[root@hiro ~]#
```

Note: /dev/sda2 is being used in the volume group VolGroup00

LVM Commands:

pvscan - shows physical volumes
pvdisplay – shows physical volume info

Logical Volume Manager

```
[root@hiro ~]# vgscan
  Reading all physical volumes.  This may take a while...
  Found volume group "VolGroup00" using metadata type lvm2
```

Volume group scan

```
[root@hiro ~]# vgdisplay
  --- Volume group ---
  VG Name               VolGroup00
  System ID
  Format                lvm2
  Metadata Areas        1
  Metadata Sequence No  3
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                2
  Open LV               2
  Max PV                0
  Cur PV                1
  Act PV                1
  VG Size               4.88 GB
  PE Size               32.00 MB
  Total PE              156
  Alloc PE / Size       156 / 4.88 GB
  Free  PE / Size       0 / 0
  VG UUID               fOKPku-aXsN-L1Ro-4yZo-Fi38-nA8R-zz41jT
```

Volume group display

LVM Commands:

vgscan – scans for volume groups
vgdisplay – shows volume group info

lvscan – scans for logical volumes
lvdisplay – shows logical volume info

```
[root@hiro ~]# lvscan
  ACTIVE            '/dev/VolGroup00/LogVol00' [3.88 GB] inherit
  ACTIVE            '/dev/VolGroup00/LogVol01' [1.00 GB] inherit
```

Logical volume scan

Logical Volume Manager

```
[root@hiro ~]# lvdisplay
  --- Logical volume ---
  LV Name                /dev/VolGroup00/LogVol00
  VG Name                VolGroup00
  LV UUID                y275PJ-CqY7-vZ9f-gCod-X72l-HgWo-QwC4T7
  LV Write Access        read/write
  LV Status              available
  # open                 1
  LV Size                3.88 GB
  Current LE             124
  Segments               1
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           253:0

  --- Logical volume ---
  LV Name                /dev/VolGroup00/LogVol01
  VG Name                VolGroup00
  LV UUID                xaOVc6-xP9C-wpT5-Ve2U-NA9p-Io9k-cPLQ4F
  LV Write Access        read/write
  LV Status              available
  # open                 1
  LV Size                1.00 GB
  Current LE             32
  Segments               1
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           253:1
```

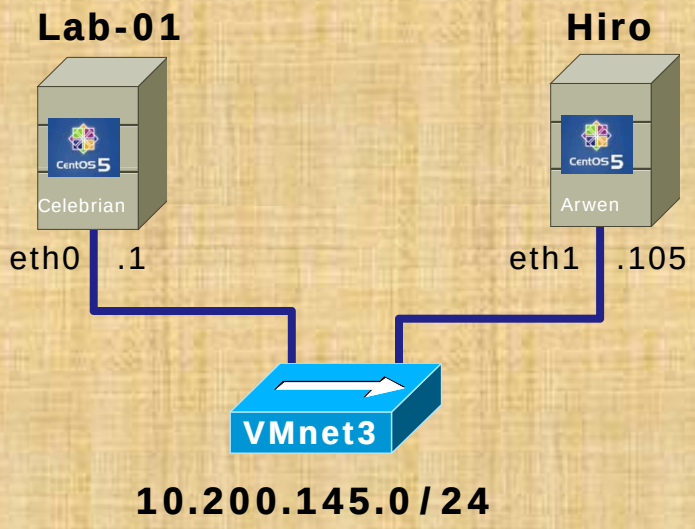
Logical volume display

Mounted on /

Note, these logical volumes relate back to what we saw in /etc/fstab

Used as swap partition

LVM



Try some LVM commands on Arwen (Hiro)

- **pvscan**
- **pvdisplay /dev/sda2**
- **vgscan**
- **vgdisplay VolGroup00**
- **lvscan**
- **lvdisplay /dev/VolGroup00/LogVol00**
- **lvdisplay**

RPC

Remote Procedure Call

- In programming you use procedures (AKA functions) to encapsulate common functionality.
- Functions can be defined in the program or be available in a function library with can be linked statically or dynamically.
- The RPC (Remote Procedure Call) protocol is a way for a program to use a procedure on a remote system over the network.
- Services like NFS and NIS use this remote procedure call technology.
- A port mapper is used by RPC based programs to map a UDP or TCP port to a specific RPC program.
- Sun created the first popular implementation of RPC which is now called ONC (Open Network Computing) RPC. See RFC 1831.

Port Mapper

(portmap)

Port Mapper

- The portmap service maps RPC (Remote Procedure Call) program numbers and versions to transport specific ports.
- An RPC service like NFS or NIS will tell portmap the port it is listening on and what RPC programs it will serve.

```
[root@hiro ~]# rpcinfo -p
   program vers proto   port
    100000    2   tcp    111  portmapper
    100000    2   udp    111  portmapper
    100024    1   udp    602  status
    100024    1   tcp    605  status
    100011    1   udp    786  rquotad
    100011    2   udp    786  rquotad
    100011    1   tcp    789  rquotad
    100011    2   tcp    789  rquotad
    100003    2   udp   2049  nfs
    100003    3   udp   2049  nfs
    100003    4   udp   2049  nfs
    100021    1   udp  42112  nlockmgr
    100021    3   udp  42112  nlockmgr
    100021    4   udp  42112  nlockmgr
    100003    2   tcp   2049  nfs
    100003    3   tcp   2049  nfs
    100003    4   tcp   2049  nfs
    100021    1   tcp  51723  nlockmgr
    100021    3   tcp  51723  nlockmgr
    100021    4   tcp  51723  nlockmgr
    100005    1   udp    814  mountd
    100005    1   tcp    817  mountd
    100005    2   udp    814  mountd
    100005    2   tcp    817  mountd
    100005    3   udp    814  mountd
    100005    3   tcp    817  mountd
[root@hiro ~]#
```

The –p option for probing the port mapper to get open port and RPC program numbers

nfs (RPC program number 100003) version 2, 3 and 4 is running and can be reached at TCP port 2049 or UDP port 2049

tcp *listening*

```
[root@hiro ~]# netstat -tl
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State
tcp        0      0 hiro.rivendell:2208        *:*                         LISTEN
tcp        0      0 *:nfs                      *:*                         LISTEN
tcp        0      0 *:dhcp-failover            *:*                         LISTEN
tcp        0      0 *:51723                    *:*                         LISTEN
tcp        0      0 *:sunrpc                   *:*                         LISTEN
tcp        0      0 *:x11                      *:*                         LISTEN
tcp        0      0 *:817                      *:*                         LISTEN
tcp        0      0 *:789                      *:*                         LISTEN
tcp        0      0 hiro.rivendell:ipp         *:*                         LISTEN
tcp        0      0 hiro.rivendell:smtp        *:*                         LISTEN
tcp        0      0 *:soap-beep                *:*                         LISTEN
tcp        0      0 *:830                      *:*                         LISTEN
tcp        0      0 hiro.rivendell:2207        *:*                         LISTEN
tcp        0      0 *:x11                      *:*                         LISTEN
tcp        0      0 *:ssh                      *:*                         LISTEN
```

nfs

port mapper

*Use **netstat** command to show open ports*

tcp *listening* *numerical, no name resolution*

```
[root@hiro ~]# netstat -tln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State
tcp        0      0 127.0.0.1:2208              0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:2049      nfs       0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:647               0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:51723             0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:111     port mapper 0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:6000              0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:817               0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:789               0.0.0.0:*                   LISTEN
tcp        0      0 127.0.0.1:631             0.0.0.0:*                   LISTEN
tcp        0      0 127.0.0.1:25              0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:605               0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:830               0.0.0.0:*                   LISTEN
tcp        0      0 127.0.0.1:2207            0.0.0.0:*                   LISTEN
tcp        0      0 :::6000                   :::*                        LISTEN
tcp        0      0 :::22                     :::*                        LISTEN
[root@hiro ~]#
```

*Use **netstat** command to show open ports*

udp *listening*

```
[root@hiro ~]# netstat -ul
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State
udp        0      0 *:42112                     *:*
udp        0      0 *:nfs                       *:*
udp        0      0 *:dwr                       *:*
udp        0      0 *:786                       *:*
udp        0      0 *:814                       *:*
udp        0      0 *:827                       *:*
udp        0      0 *:38485                     *:*
udp        0      0 *:xmlrpc-beep               *:*
udp        0      0 *:mdns                      *:*
udp        0      0 *:sunrpc                    *:*
udp        0      0 *:ipp                       *:*
udp        0      0 *:1023                      *:*
udp        0      0 *:53387                     *:*
udp        0      0 *:mdns                      *:*
[root@hiro ~]#
```

nfs

port mapper

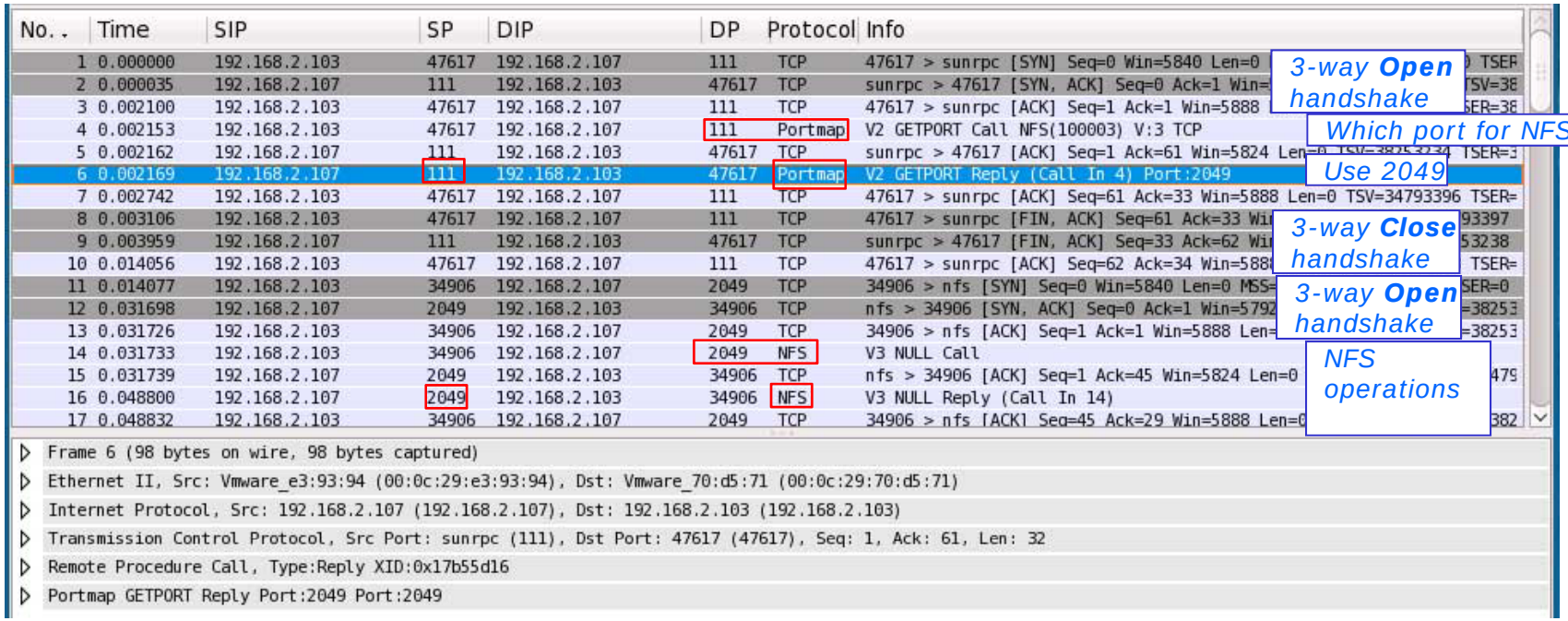
*Use **netstat** command to show open ports*

udp *listening* *numerical, no name resolution*

```
[root@hiro ~]# netstat -uln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State
udp        0      0 0.0.0.0:42112               0.0.0.0:*
udp        0      0 0.0.0.0:2049    nfs         0.0.0.0:*
udp        0      0 0.0.0.0:644                 0.0.0.0:*
udp        0      0 0.0.0.0:786                 0.0.0.0:*
udp        0      0 0.0.0.0:814                 0.0.0.0:*
udp        0      0 0.0.0.0:827                 0.0.0.0:*
udp        0      0 0.0.0.0:38485               0.0.0.0:*
udp        0      0 0.0.0.0:602                 0.0.0.0:*
udp        0      0 0.0.0.0:5353                0.0.0.0:*
udp        0      0 0.0.0.0:111     port mapper 0.0.0.0:*
udp        0      0 0.0.0.0:631                 0.0.0.0:*
udp        0      0 0.0.0.0:1023                0.0.0.0:*
udp        0      0 :::53387                    :::*
udp        0      0 :::5353                     :::*
```

*Use **netstat** command to show open ports*

Port Mapper Example – client wants to use NFS service



1-3 Client (.103) initiates connection with Port Mapper (on port 111) on the NFS server (.107)

4: Client (.103) requests the port to use for the NFS service

6: Server (.107) responds with port 2049

8-10: Client (.103) closes connection (an abbreviated 3-way handshake)

11: Client (.103) initiates connection with NFS service on port 2049

NFS

NFS

What is NFS?

NFS is the Network File System. It allows a system to mount a remote directory.

NFS Example

Hiro

NFS Server

This will be the "remote" system which is configured as a NFS server

Here is an example directory we want to share

```
[root@hiro ~]# ls -l /depot
total 20
-rw-r--r-- 1 root root 0 Apr 19 16:36 file1
-rw-r--r-- 1 root root 0 Apr 19 16:36 file2
-rw-r--r-- 1 root root 0 Apr 19 16:36 file3
-rw-r--r-- 1 root root 0 Apr 19 16:36 file4
-rw-r--r-- 1 root root 0 Apr 19 16:36 file5
```

NFS Example

Hiro

NFS Server

Here is an example directory we want to share

```
[root@hiro ~]# ls -l /depot
total 20
-rw-r--r-- 1 root root 0 Apr 19 16:36 file1
-rw-r--r-- 1 root root 0 Apr 19 16:36 file2
-rw-r--r-- 1 root root 0 Apr 19 16:36 file3
-rw-r--r-- 1 root root 0 Apr 19 16:36 file4
-rw-r--r-- 1 root root 0 Apr 19 16:36 file5
```

This is the directory we want to share

Here is how you share this directory with others

```
[root@hiro ~]# cat /etc/exports
/depot          *(ro,sync)
```

Add a line to /etc/exports

```
[root@hiro ~]# service nfs start
Starting NFS services:                     [  OK  ]
Starting NFS quotas:                       [  OK  ]
Starting NFS daemon:                       [  OK  ]
Starting NFS mountd:                       [  OK  ]
[root@hiro ~]#
```

Start the NFS services

NFS Example

Hiro

NFS Server

Add line for shared directory to /etc/exports

```
[root@hiro ~]# cat /etc/exports
/depot          *(ro,sync)
```

Options, ro=read only, sync=writes data to disk instead of buffering

*The hostnames that can mount it, *=all*

The directory to export

Breaking it down, this is the line added to /etc/exports. It defines the directory to be share, who can access it, and how it is shared.

NFS Example

Hiro

NFS Server

Start the NFS services

```
root@hiro ~]# service nfs start

Starting NFS services:                                     [  OK  ]
```

Synchronizes /var/lib/nfs/xtab with /etc/exports (exportfs –r) command and sets ports for lockd (file locking daemon) to listen on

```
Starting NFS quotas:                                       [  OK  ]
```

rquotad (remote quota daemon) answers clients' queries about user quotas for exported NFS filesystems.

```
Starting NFS daemon:                                       [  OK  ]
```

nsfd (NFS daemon) provides the actual file transfer service

```
Starting NFS mountd:                                       [  OK  ]
```

mountd (Mount daemon) handles mount requests from clients

Breaking it down, this shows all the daemons comprising NFS starting up

NFS Example

Hiro

NFS Server

```
[root@hiro ~]# ls -li /depot
total 20
950279 -rw-r--r-- 1 root root 0 Apr 19 16:36 file1
950280 -rw-r--r-- 1 root root 0 Apr 19 16:36 file2
950281 -rw-r--r-- 1 root root 0 Apr 19 16:36 file3
950282 -rw-r--r-- 1 root root 0 Apr 19 16:36 file4
950283 -rw-r--r-- 1 root root 0 Apr 19 16:36 file5
[root@hiro ~]#
```

This is the directory on Hiro being shared

Mount the remote directory on the client Arwen

Arwen

NFS Client

```
[root@arwen ~]# showmount -e hiro
Export list for hiro:
/depot       *

[root@lab-01 mnt]# mkdir /depot

[root@arwen ~]# mount hiro:/depot /depot

[root@arwen ~]# ls -li /depot
total 20
950279 -rw-r--r-- 1 root root 0 Apr 19 16:36 file1
950280 -rw-r--r-- 1 root root 0 Apr 19 16:36 file2
950281 -rw-r--r-- 1 root root 0 Apr 19 16:36 file3
950282 -rw-r--r-- 1 root root 0 Apr 19 16:36 file4
950283 -rw-r--r-- 1 root root 0 Apr 19 16:36 file5
[root@arwen ~]#
```

shows all exported directories on the server

Mount the remote share (exported directory) on Arwen

Voilá

NFS Example

Hiro

NFS Server

```
[root@hiro ~]# ls -l /depot
total 20
-rw-r--r-- 1 root root 0 Apr 19 16:36 file1
-rw-r--r-- 1 root root 0 Apr 19 16:36 file2
-rw-r--r-- 1 root root 0 Apr 19 16:36 file3
-rw-r--r-- 1 root root 0 Apr 19 16:36 file4
-rw-r--r-- 1 root root 0 Apr 19 16:36 file5
```

Mount the remote directory on the another client, Lab-01

Lab-01

NFS Client

```
[root@lab-01 mnt]# showmount -e hiro
Export list for hiro:
/depot       *

[root@lab-01 mnt]# mkdir /depot

[root@lab-01 mnt]# mount hiro:/depot /depot

[root@lab-01 mnt]# ls -l /depot
total 20
-rw-r--r-- 1 root root 0 Apr 19 16:36 file1
-rw-r--r-- 1 root root 0 Apr 19 16:36 file2
-rw-r--r-- 1 root root 0 Apr 19 16:36 file3
-rw-r--r-- 1 root root 0 Apr 19 16:36 file4
-rw-r--r-- 1 root root 0 Apr 19 16:36 file5
[root@lab-01 mnt]#
```

shows the NFS servers export list

Can also use –t nfs option to specify remote share is NFS

NFS Example

Hiro


```
[root@hiro ~]# cat /etc/exports
/depot          *(ro,sync)
[root@hiro ~]#
```

NFS Server

Arwen

NFS Client

```
[root@arwen ~]# showmount -a hiro
All mount points on hiro:
192.168.2.103:/depot
192.168.2.105:/depot

[root@arwen ~]# mount
< snipped >
hiro:/depot on /depot type nfs (rw,addr=192.168.2.107)
[root@arwen ~]#
```

The -a option on showmount shows current mounts (not reliable)

The mount command by itself shows all current mount points

Lab-01

NFS Client

```
[root@lab-01 mnt]# showmount -a hiro
All mount points on hiro:
192.168.2.103:/depot
192.168.2.105:/depot

[root@lab-01 ~]# mount
< snipped >
hiro:/depot on /depot type nfs (rw,addr=192.168.2.107)
[root@lab-01 ~]#
```

NFS Example

Hiro

NFS Server

```
[root@hiro ~]# cat /etc/exports
/depot          *(ro,sync)
[root@hiro ~]#
```

Note: The mounted directory is only rw if configured that way on the NFS server

Arwen

NFS Client

```
[root@arwen ~]# showmount -a hiro
All mount points on hiro:
192.168.2.103:/depot
192.168.2.105:/depot

[root@arwen ~]# mount
< snipped >
hiro:/depot on /depot type nfs (rw,addr=192.168.2.107)
[root@arwen ~]#
```

Lab-01

NFS Client

```
[root@lab-01 mnt]# showmount -a hiro
All mount points on hiro:
192.168.2.103:/depot
192.168.2.105:/depot

[root@lab-01 ~]# mount
< snipped >
hiro:/depot on /depot type nfs (rw,addr=192.168.2.107)
[root@lab-01 ~]#
```

Service Applications

Steps to installing services

1. Install software package using **yum**, **rpm** or build from source code
2. Customize service's configuration file
3. Modify the firewall to allow access to the service
4. Customize SELinux context settings to allow use
5. Start the service
6. Configure service to automatically start when system boots
7. Monitor and verify service is running
8. Troubleshoot as necessary
9. Monitor log files as appropriate
10. Configure additional security

NFS Overview

The Network File System Components

1. The *portmap* package

NFS is designed around the Remote Procedure Call API and utilizes the portmap daemon for its operation.

2. The *nfs-utils* package has five components:

- **rpc.nfsd** Primary NFS component. Handles all NFS requests, and provides the main engine for NFS to work.
- **rpc.mountd** Handles permission evaluation before allowing the client to mount an export.
- **rpc.quotad** Interfaces with the quota manager to ensure that client file system quotas are preserved.
- **rpc.statd** Monitors UDP and TCP traffic during NFS operation. It reports crashes and reboots to the lock manager.
- **rpc.lockd** Manages file-locking requests, and on crashed systems, this component provides lock recovery.

3. Automount is also included in the *nfs-utils* package, and is used for mounting NFS directories dynamically as they are accessed by users.

NFS Overview

Client-Server Operations

An NFS server, serving files and directories to remote clients, can be a client at the same time as it is serving.

- NFSv3 is capable of running over both UDP and TCP
- rpc.lockd and rpc.statd run on both the server and the client

Implements NSM (Network Status Monitor) RPC protocol and provides reboot notification so locking can be updated when the NFS server crashes

Starts the NFS lock manager on kernels. Locking prevents more than one user accessing and modifying a file at the same time.

NFS

Steps to installing services

Step 1 *Install software package using **yum**, **rpm** or build from source code*

```
[root@lab-01 ~]# rpm -qa | grep nfs
nfs-utils-lib-1.0.8-7.6.el5
nfs-utils-1.0.9-42.el5

[root@lab-01 ~]# rpm -qa | grep port
portmap-4.0-65.2.2.1
```

Installed by default on Red Hat and CentOS. Note: NFS uses portmapper for making RPC (remote procedure calls)

NFS

The Network File System Packages

```
[root@hiro ~]# rpm -qi nfs-utils
[root@lab-01 ~]# rpm -qi nfs-utils
Name        : nfs-utils                    Relocations: (not relocatable)
Version     : 1.0.9                             Vendor: CentOS
Release     : 42.el5                         Build Date: Thu 03 Sep 2009 02:14:31 PM PDT
Install Date: Wed 24 Feb 2010 06:50:19 AM PST      Build Host: builder16.centos.org
Group       : System Environment/Daemons    Source RPM: nfs-utils-1.0.9-42.el5.src.rpm
Size        : 768535                           License: GPL
Signature   : DSA/SHA1, Wed 09 Sep 2009 08:56:17 AM PDT, Key ID a8a447dce8562897
Summary     : NFS utlilities and supporting clients and daemons for the kernel NFS server.
Description :
The nfs-utils package provides a daemon for the kernel NFS server and
related tools, which provides a much higher level of performance than the
traditional Linux NFS server used by most users.

This package also contains the showmount program.  Showmount queries the
mount daemon on a remote host for information about the NFS (Network File
System) server on the remote host.  For example, showmount can display the
clients which are mounted on that host.

This package also contains the mount.nfs and umount.nfs program.
[root@lab-01 ~]#
```

nfs-utils: The NFS server programs

NFS

The Network File System Packages

```
[root@hiro ~]# rpm -qi nfs-utils-lib
Name        : nfs-utils-lib                 Relocations: (not relocatable)
Version     : 1.0.8                              Vendor: CentOS
Release     : 7.6.el5                       Build Date: Thu 03 Sep 2009 01:35:40 PM PDT
Install Date: Wed 24 Feb 2010 06:47:37 AM PST      Build Host: builder10.centos.org
Group       : System Environment/Libraries   Source RPM: nfs-utils-lib-1.0.8-7.6.el5.src.rpm
Size        : 105265                            License: GPL
Signature   : DSA/SHA1, Sat 19 Sep 2009 08:53:58 PM PDT, Key ID a8a447dce8562897
URL         : http://www.citi.umich.edu/projects/nfsv4/linux/
Summary     : Network File System Support Library
Description :
Support libaries that are needed by the commands and
daemons the nfs-utils rpm.
[root@hiro ~]#
```

nfs-utils-lib: support library for the NFS programs

NFS

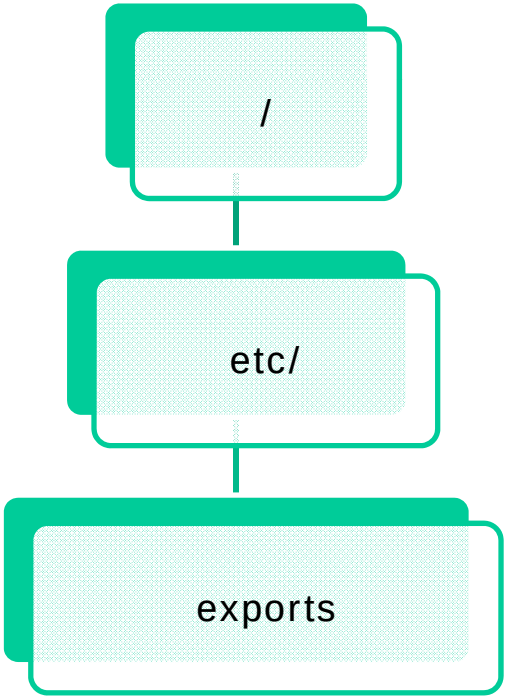
The Network File System Packages

```
[root@hiro ~]# rpm -qi portmap
Name        : portmap                      Relocations: (not relocatable)
Version     : 4.0                               Vendor: CentOS
Release     : 65.2.2.1                      Build Date: Sat 06 Jan 2007 02:10:02 AM PST
Install Date: Wed 24 Feb 2010 06:47:57 AM PST      Build Host: builder5.centos.org
Group       : System Environment/Daemons    Source RPM: portmap-4.0-65.2.2.1.src.rpm
Size        : 58245                            License: BSD
Signature   : DSA/SHA1, Tue 03 Apr 2007 05:27:00 PM PDT, Key ID a8a447dce8562897
Summary     : A program which manages RPC connections.
Description :
The portmapper program is a security tool which prevents theft of NIS
(YP), NFS and other sensitive information via the portmapper.  A
portmapper manages RPC connections, which are used by protocols like
NFS and NIS.

The portmap package should be installed on any machine which acts as a
server for protocols using RPC.
[root@hiro ~]#
```

portmap: The port mapper is used by NFS for RPCs (remote procedure calls)

Installing and Configuring DNS service

Step 2 Customize the configuration file



The exports file specifies directories and access controls for remote access by clients

NFS

Step 2 *Customize service's configuration file*

/etc/exports

Syntax:

absolute-directory-path [machine-specifier(option,options...)]...

directory to share

who to share it with

no spaces (single argument)

- Machine Specifiers
  - hostname
  - IP address
  - network addresses

- Options
  - ro, rw *read only, read-write (default)*
  - root_squash *squashes special root powers for UID 0 and GID 0, runs as nobody (default)*
  - no_root_squash *allows normal root access*
  - secure *allows remote access only from a privileged port (< 1024)*
  - insecure *allows access from any port*
  - sync *writes to disk are not buffered*

*See **man exports** for more details*

Server-side NFS

/etc/exports file syntax:

absolute-directory-path [machine-specifier(option,options...)]...

/etc/exports file examples:

```
[root@hiro ~]# cat /etc/exports
/depot lab-01(rw) arwen(ro)
```

exports depot directory to lab-01 (read-write) and to arwen (read only)

```
[root@hiro ~]# cat /etc/exports
/depot 192.168.2.0/24(rw)
```

exports depot directory all hosts on 192.168.2.0/24 network (read-write)

```
[root@hiro ~]# cat /etc/exports
/depot 192.168.2.0/255.255.255.0(rw)
```

same as above using netmask rather than prefix for network selection

These examples showing different ways to share /depot on Hiro to other systems

Server-side NFS

Another /etc/exports example

exports /depot directory to lab-01 (read-write) and to arwen (read only)

exports /home/cis192 to all hosts on 192.168.2.0/24 network as read-write, normal root access (dangerous) and replies are made after only disk writes have completed

```
[root@hiro ~]# cat /etc/exports
/depot lab-01(rw) arwen(ro)
/home/cis192 192.168.2.0/24(rw,no_root_squash,sync)
/home/guest *(rw,sync)
```

exports /home/guest to all as read-write and replies are made only after disk writes have completed

*Use **exportfs –rv** or **service nfs restart** whenever you make changes to the /etc/exports file*

```
[root@hiro ~]# exportfs -rv
exporting lab-01:/depot
exporting arwen:/depot
exporting 192.168.2.0/24:/home/cis192
exporting *:/home/guest
```

Server side NFS

exportfs – command to dynamically update exported files

Options:

-a *Exports or unexport all entries in the exports file.*
-r *Re-export all entries.*
-u client:[export] *Removes (unexport) from the specified host.*
-o options *Overrides NFS options in /etc/exports file.*
-v *Display output in verbose mode.*

Examples:

exportfs –rv (use after making changes to /etc/exports)

exportfs –ua (shuts down all exported directories)

Server side NFS

Step 3 *Firewall modifications*

Because NFS security is based on host or IP addresses (which can be spoofed) it's a good idea to have keep NFS access local and block any connections from the Internet.

- Block non-local access to UPD and TCP ports 2049 (NFS)
- Block non-local access to UDP and TCP port 111 (port mapper)

Port Mapper (the portmap daemon) selects random ports so you will need to allow connections with any hosts needing the service.

Don't allow Internet access to your NFS server!

Server side NFS

Step 4 *SELinux*

- By default, the appropriate SELinux booleans are set to allow NFS to operate in enforcing mode.
- Note: the following SELinux booleans must be ON for NFS to operate:

```
[root@hiro ~]# setsebool nfs_export_all_rw=1
[root@hiro ~]# setsebool nfs_export_all_ro=1

[root@hiro ~]# getsebool nfs_export_all_ro
nfs_export_all_ro --> on

[root@hiro ~]# getsebool nfs_export_all_rw
nfs_export_all_rw --> on
```

Server side NFS

Step 5 *Start service*

```
[root@hiro ~]# service nfs start
Starting NFS services:                                     [  OK  ]
Starting NFS quotas:                                       [  OK  ]
Starting NFS daemon:                                       [  OK  ]
Starting NFS mountd:                                       [  OK  ]
```

Server side NFS

If service is already running use the following to reread configuration files:

service named restart

or

exportfs -rv

Server side NFS

Step 6 *Configure automatic service startup*

To automatically start NFS service at system boot use:

```
[root@hiro ~]# chkconfig nfs on
[root@hiro ~]# chkconfig --list nfs
nfs             0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

To automatically start Port Mapper service at system boot use:

```
[root@hiro ~]# chkconfig portmap on
[root@hiro ~]# chkconfig --list portmap
portmap         0:off   1:off   2:on    3:on    4:on    5:on    6:off
[root@hiro ~]#
```

Server side NFS

Step 7 *Monitor and verify service is running*

NFS service

```
[root@hiro ~]# service nfs status
rpc.mountd (pid 5150) is running...
nfsd (pid 5147 5146 5145 5144 5143 5142 5141 5140) is running...
rpc.rquotad (pid 5135) is running...
[root@hiro ~]#
```

Server side NFS

Step 7 *Monitor and verify service is running*

NFS processes

```
[root@hiro ~]# ps -ef | grep nfs
root      5139     7  0 17:58 ?        00:00:00 [nfsd4]
root      5140     1  0 17:58 ?        00:00:00 [nfsd]
root      5141     1  0 17:58 ?        00:00:00 [nfsd]
root      5142     1  0 17:58 ?        00:00:00 [nfsd]
root      5143     1  0 17:58 ?        00:00:00 [nfsd]
root      5144     1  0 17:58 ?        00:00:00 [nfsd]
root      5145     1  0 17:58 ?        00:00:00 [nfsd]
root      5146     1  0 17:58 ?        00:00:00 [nfsd]
root      5147     1  0 17:58 ?        00:00:00 [nfsd]
root      5283  3848  0 18:13 pts/0    00:00:00 grep nfs
[root@hiro ~]# ps -ef | grep rpc
rpc       1836     1  0 14:10 ?        00:00:00 portmap
root      1867     7  0 14:10 ?        00:00:00 [rpciod/0]
root      1873     1  0 14:10 ?        00:00:00 rpc.statd
root      1907     1  0 14:10 ?        00:00:01 rpc.idmapd
root      5135     1  0 17:58 ?        00:00:00 rpc.rquotad
root      5150     1  0 17:58 ?        00:00:00 rpc.mountd
root      5285  3848  0 18:13 pts/0    00:00:00 grep rpc
[root@hiro ~]# ps -ef | grep lockd
root        10     7  0 14:09 ?        00:00:00 [kblockd/0]
root      5138     1  0 17:58 ?        00:00:00 [lockd]
root      5294  3848  0 18:16 pts/0    00:00:00 grep lockd
```

port mapper

[nfsd] *Primary NFS component. Handles all NFS requests, and provides the main engine for NFS to work.*

rpc.statd *Monitors UDP and TCP traffic during NFS operation. It reports crashes and reboots to the lock manager.*

rpc.idmapd *For NFSv4 to map UIDs/GIDs to names*

rpc.rquotad *Interfaces with the quota manager to ensure that client file system quotas are preserved.*

rpc.mountd *Handles permission evaluation before allowing the client to mount an export.*

[lockd] *Manages file-locking requests, and on crashed systems, this component provides lock recovery.*

Server side NFS

Step 7 *Verify service is running*

netstat

```
[root@hiro ~]# netstat -tln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State
tcp        0      0 127.0.0.1:2208              0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:2049      NFS       0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:651                 0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:783                 0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:111     Port Mapper 0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:47536               0.0.0.0:*                   LISTEN
tcp        0      0 127.0.0.1:631               0.0.0.0:*                   LISTEN
tcp        0      0 0.0.0.0:665                 0.0.0.0:*                   LISTEN
tcp        0      0 127.0.0.1:25                0.0.0.0:*                   LISTEN
tcp        0      0 127.0.0.1:2207              0.0.0.0:*                   LISTEN
tcp        0      0 :::22                       :::*                        LISTEN
```

*Use **netstat –tl** command to see what port names your system is listening for requests on*

Server side NFS

Step 7 *Verify service is running*

netstat

```
[root@hiro ~]# netstat -uln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State
udp        0      0 0.0.0.0:2049      NFS      0.0.0.0:*
udp        0      0 0.0.0.0:648               0.0.0.0:*
udp        0      0 0.0.0.0:777               0.0.0.0:*
udp        0      0 0.0.0.0:780               0.0.0.0:*
udp        0      0 0.0.0.0:662               0.0.0.0:*
udp        0      0 0.0.0.0:43325             0.0.0.0:*
udp        0      0 0.0.0.0:5353              0.0.0.0:*
udp        0      0 0.0.0.0:111     Port Mapper   0.0.0.0:*
udp        0      0 0.0.0.0:631               0.0.0.0:*
udp        0      0 0.0.0.0:57595             0.0.0.0:*
udp        0      0 :::37930                  :::*
udp        0      0 :::5353                   :::*
```

*Use **netstat –ul** command to see what port names your system is listening for requests on*

Server side NFS

Step 8 *Troubleshooting*

Problem: share stops working

Client error message:

```
[root@lab-01 ~]# ls /depot
ls: /depot: Stale NFS file handle
```

Solution:

You mounted a remote directory on the client, then removed that directory from the /etc/export list on the NFS server

To fix: export the share again

Server side NFS

Step 8 *Troubleshooting*

Problem: Cannot show exported directories on remote server

Client error message:

```
[root@lab-01 ~]# showmount -a hiro
mount clntudp_create: RPC: Port mapper failure - RPC: Unable to receive
```

Solution:

This will appear on the client when the firewall on the server is blocking port mapper connections

To fix: Modify firewall on server to allow connections from local hosts

Server side NFS

Step 9 *Monitor log files*

```
[root@hiro ~]# cat /var/log/messages | grep nfs
Apr 20 14:04:34 hiro kernel: nfsd: last server has exited
Apr 20 14:04:34 hiro kernel: nfsd: unexporting all filesystems
Apr 20 14:10:17 hiro rpc.statd[1873]: statd running as root. chown
/var/lib/nfs/statd/sm to choose different user
Apr 20 14:29:10 hiro kernel: Installing knfsd (copyright (C) 1996 okir@monad.swb.de).
Apr 20 14:29:10 hiro kernel: NFSD: Using /var/lib/nfs/v4recovery as the NFSv4 state
recovery directory
Apr 20 14:40:08 hiro kernel: nfsd: last server has exited
Apr 20 14:40:08 hiro kernel: nfsd: unexporting all filesystems
Apr 20 14:44:54 hiro kernel: NFSD: Using /var/lib/nfs/v4recovery as the NFSv4 state
recovery directory
Apr 20 17:46:02 hiro setsebool: The nfs_export_all_ro policy boolean was changed to 0
by root
Apr 20 17:46:23 hiro setsebool: The nfs_export_all_rw policy boolean was changed to 0
by root
Apr 20 17:50:00 hiro setsebool: The nfs_export_all_rw policy boolean was changed to 1
by root
Apr 20 17:50:05 hiro setsebool: The nfs_export_all_ro policy boolean was changed to 1
by root
Apr 20 17:58:07 hiro kernel: nfsd: last server has exited
Apr 20 17:58:07 hiro kernel: nfsd: unexporting all filesystems
Apr 20 17:58:14 hiro kernel: NFSD: Using /var/lib/nfs/v4recovery as the NFSv4 state
recovery directory
```

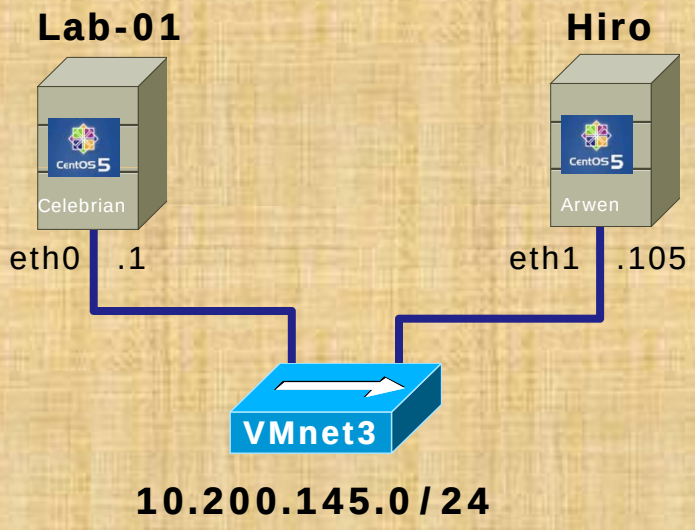
*Use **tail –f /var/log/messages** to monitor in real time*

Server side NFS

Step 10 *Configure additional security*

- Use consistent UIDs and GIDs across systems as account and group names are not used
- Set appropriate permissions
- Export only what you need to and only to those who need it.
- Use wildcards sparingly
- Protect your DNS server (avoid spoofing attacks)
- Protect portmap with TCP wrappers
- Use NSFv4 instead of earlier versions to authenticate users rather than client systems
- Take CIS 193!

Try it, you will like it



On Arwen (Hiro)

- Create a /depot directory and add some example files
- Add to /etc/exports:
  /depot *(ro,sync)
- Disable firewall with **service iptables stop**
- **service nfs start**

Set up a "share" on Hiro

Client-side NFS

Clients merely need to mount the exported directories to a local directory as if it were a file system.

- Syntax: **mount [-t nfs] [-*o options...*] *servername:export mountdir***

- Options:
  - rw *read-write (must be exported this way)*
  - hard *if a NFS server goes down service will hang (blocked) till available again*
  - udp *use UDP as the transport protocol (default)*
  - soft *if a NFS server goes down service will return an error*
  - intr *allows user to interrupt a blocked operation and return an error*
  - ro *read-only*
  - tcp *use TCP as the transport protocol*

- showmount command
  - **showmount -e *servername*** *shows the available exports*
  - **showmount –a *servername*** *shows current exports being shared*

*See **man mount** and **man nfs** for more details*

Client-side NFS

• NFS mount examples:

mount hiro:/depot /depot

mount-t nfs hiro:/depot /depot

mount –t nfs –o rw,hard,intr hiro:/depot /depot

• Using showmount command examples:

```
[root@lab-01 depot]# showmount -a hiro
All mount points on hiro:
192.168.2.103:/depot
192.168.2.105:/depot
192.168.2.105:/home/cis192
192.168.2.105:/home/guest
```

Shows current exports in use

```
[root@lab-01 depot]# showmount -e hiro
Export list for hiro:
/home/guest  *
/home/cis192 192.168.2.0/24
/depot       arwen,lab-01
```

Shows available exports

Client-side NFS

To automate mounting with /etc/fstab

- **Manual** mount:

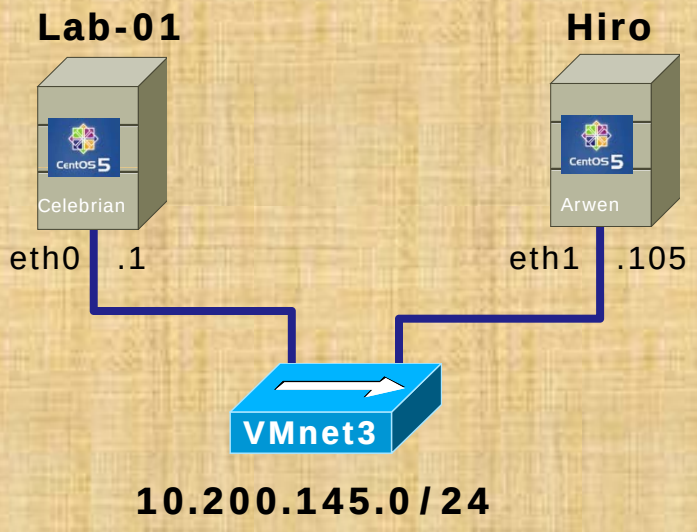
```
[root@lab-01 ~]# mount -t nfs  hiro:/depot /depot
```

- **Automated** mount:

```
[root@lab-01 ~]# cat /etc/fstab
/dev/VolGroup00/LogVol00 /                       ext3    defaults        1 1
LABEL=/boot              /boot                   ext3    defaults        1 2
tmpfs                    /dev/shm                tmpfs   defaults        0 0
devpts                   /dev/pts                devpts  gid=5,mode=620  0 0
sysfs                    /sys                    sysfs   defaults        0 0
proc                     /proc                   proc    defaults        0 0
/dev/VolGroup00/LogVol01 swap                    swap    defaults        0 0
hiro:/depot              /depot                  nfs rw,addr=192.168.2.107 0 0
```

One way to do this … do a manual mount and test the share. If satisfied, copy the line in /etc/mtab (current mounts) to /etc/fstab (mounts to perform at startup)

Try it, you will like it



On Arwen (Hiro)

- Create a /depot directory and add some example files
- Add to /etc/exports:
  /depot *(ro,sync)
- **service nfs start**
- **showmount –e localhost**
- Disable firewall with **service iptables stop**

On Lab-01 (Celebrian):

- **mkdir /depot**
- **showmount –e hiro**
- **mount hiro:/depot /depot**
- **ls –l /depot**
- **showmount –a hiro**

Printing

Printer Configuration

- Printer Modes
  - Text Mode - Accepts ASCII characters
  - Graphic mode - Accepts separate commands and data
- Graphic-mode Commands and Languages
  - Hewlett Packard's PCL
  - Adobe's PostScript
- Types of Printer Drivers
- Print Filters

Printing in Linux

Linux printing commands

- The LP and LPRNG systems
  - **lpr** or **lp** copies the print job to the spool dir
  - **lpd** daemon checks the spool dir for jobs
  - Control and Data files are copied to the printer queue
  - **lpc** controls and configures lpd daemon
  - **lpq** queries the **lpd** daemon about print jobs
  - **lpstat** gives the status of the lp system
  - **cancel** or **lprm**
- The **C**ommon **U**NIX **P**rint **S**ystem
  - Uses modified versions of the same commands as LP by making use of symbolic links
  - Provides a web-based interface to the print system: *http://localhost:631/printes*
  - Configuration files and drivers for CUPS printers are in */etc/cups*.
  - Print jobs are spooled as a pair of data and control files in */var/spool/cups*.

lp Examples:

- **lpr -P printer filetoprint**
- **lp -d printer filetoprint**
- **lp -d printer -h host filetoprint**

CUPS Printing

Main goals of CUPS software

- Cross-platform
- Web-based Management
- Separate hardware dependencies from logical printers
- Compatible with older lp model

Relevant Commands:

- redhat-config-printer
- lpadmin
- http://localhost:631/admin

Printers

Two predominate types of printers

- *Thermal inkjet technology*
- *Laser, drum, toner technology*

Two predominate types of printers

- *Thermal inkjet technology*
- *Laser, drum, toner technology*

So many ways to hook them up …

Now:

- *Network*
- *USB*
- *Wireless (Bluetooth, IR)*
- *PictBridge (USB based)*

Back then:

- *Serial cable*
- *Parallel printer cable*

Printer Configuration

Printing

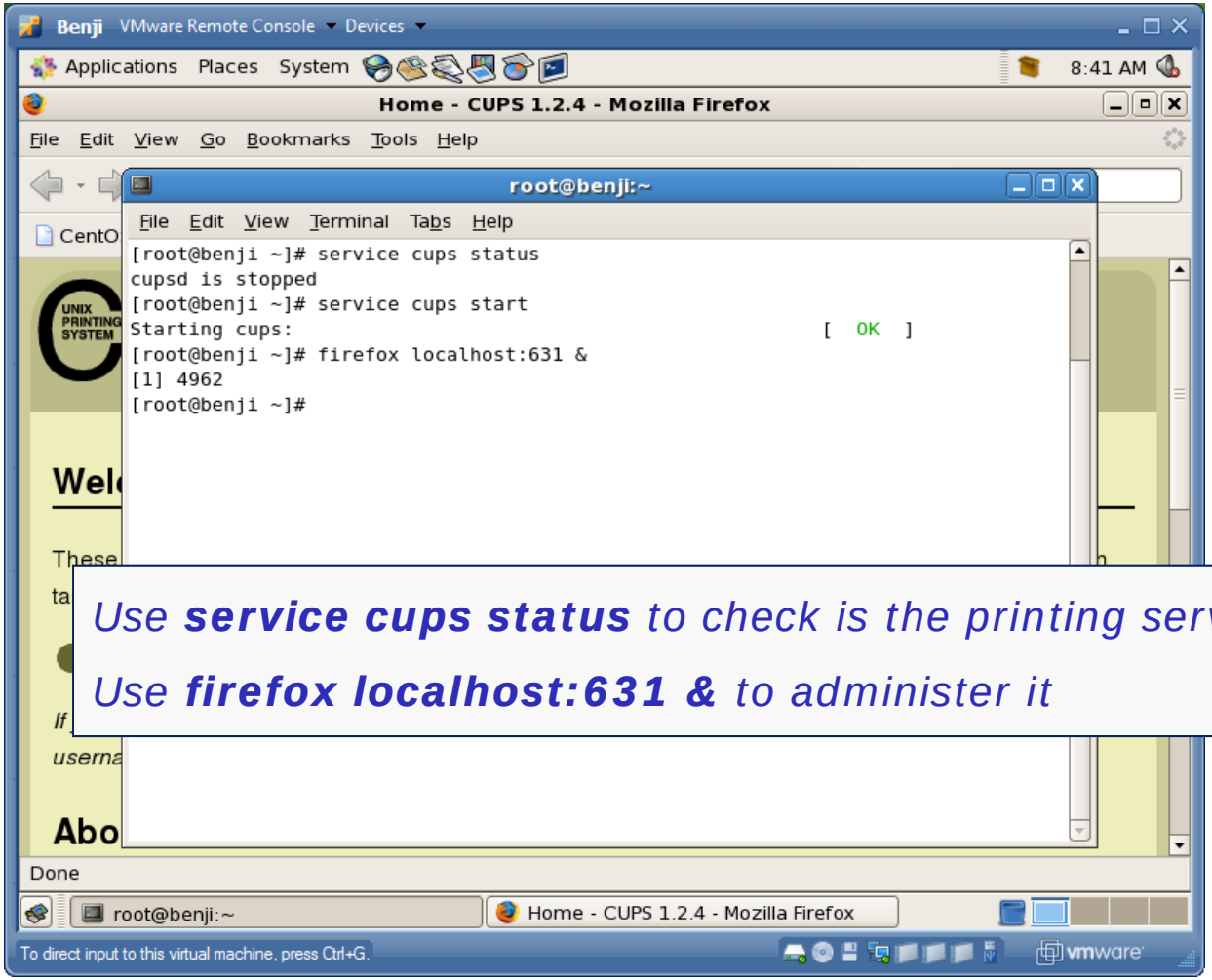
System V based print subsystem
- **lp** (to print)
- **lpstat** (queue management)
- **cancel** (to remove jobs)

BSD based print subsystem
- **lpr** (to print)
- **lpq** (queue management)
- **lprm** (to remove jobs)

CUPS - Common UNIX Printing System
- Provides both System V and Berkeley based command-line interfaces
- Supports new Internet Printing Protocol
- Works with Samba

We will be just looking at CUPS

CUPS



*Use **service cups status** to check is the printing service in running*

*Use **firefox localhost:631 &** to administer it*

Service Configuration

CUPS Summary

Step 1 **yum install cups** (if not already installed)
- Dependencies: cups-libs, poppler, poppler-util (poppler is used to make PDF documents)

Step 2 Configuration files/tools:
Web GUI at http://localhost:631
or edit /etc/cups/printers.conf

Step 3 Firewall: Open UDP 631 & TCP 631

Step 4 SELinux: enforcing or permissive

Step 5 **service cups start** (also **stop** and **restart**)

Step 6 **chkconfig cups on** (or **off**)

Step 7 Monitor or verify service is running:
service cups status
ps –ef | grep cupsd
netstat –tln | grep 631
netstat –uln | grep 631
Spool files in /var/spool/cups

Step 8 Troubleshoot (check logs, firewall & network settings)

Step 9 Log files: /var/log/cups/*

Step 10 Additional security:
Remote access to web GUI must be enabled
http://www.cups.org/documentation.php/doc-1.4/security.html

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 UDP and TCP protocol traffic to port 631 is allowed.

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

Sidetrack – Service command

```
[root@hiro ~]# service cups status
cupsd (pid 4665) is running...
[root@hiro ~]# firefox localhost:631 &
[2] 12812

[root@hiro ~]# service cups status
cupsd (pid 4665) is running...

[root@hiro ~]# service cups restart
Stopping cups:                                             [  OK  ]
Starting cups:                                             [  OK  ]

[root@hiro ~]# service cups stop
Stopping cups:                                             [  OK  ]

[root@hiro ~]# service cups start
Starting cups:                                             [  OK  ]
```

Note: the effects of the service command changes will last until the next shutdown.

They are not permanent.

Sidetrack – Service command

```
[root@hiro ~]# type service
service is hashed (/sbin/service)
[root@hiro ~]# file /sbin/service
/sbin/service: Bourne shell script text executable
[root@hiro ~]# cat /sbin/service
#!/bin/sh

. /etc/init.d/functions

VERSION="`basename $0` ver. 0.91"
USAGE="Usage: `basename $0` < option > | --status-all | \
[ service_name [ command | --full-restart ] ]"
SERVICE=
SERVICEDIR="/etc/init.d"
OPTIONS=

if [ $# -eq 0 ]; then
   echo "${USAGE}" >&2
   exit 1
fi
" ]; then
< snipped>
[root@hiro ~]#
```

*Use the **type** and **file** commands to find the location and type of commands*

The service command is a viewable shell script and can be viewed

Sidetrack – chkconfig command

```
[root@hiro ~]# chkconfig --list cups
cups            0:off   1:off   2:on    3:on    4:on    5:on    6:off

[root@hiro ~]# chkconfig cups off

[root@hiro ~]# chkconfig --list cups
cups            0:off   1:off   2:off   3:off   4:off   5:off   6:off

[root@hiro ~]# chkconfig cups on

[root@hiro ~]# chkconfig --list cups
cups            0:off   1:off   2:on    3:on    4:on    5:on    6:off

[root@hiro ~]# chkconfig --level 2 cups off

[root@hiro ~]# chkconfig --list cups
cups            0:off   1:off   2:off   3:on    4:on    5:on    6:off
```

*The **chkconfig** command is used to configure which services are started when the system boots up.*

Sidetrack – chkconfig command

```
[root@benji rc.d]# chkconfig --list cups
cups            0:off   1:off   2:off   3:on    4:on    5:on    6:off
[root@benji rc.d]# cd /etc/rc.d
[root@benji rc.d]# ls –R

< snipped >

./rc0.d:
K01smartd                       K15gpm          K74nscd             K89pand
K02avahi-daemon                 K20nfs          K74ntpd             K89rdisc
K02avahi-dnsconfd               K24irda         K75netfs            K90bluetooth
K02dhcdbd                       K25sshd         K85mdmonitor        K90network
K02haldaemon                    K30sendmail     K85mdmpd            K90restorecond
K02NetworkManager               K30spamassassin K85messagebus       K91capi
K02NetworkManagerDispatcher     K35vncserver    K86nfslock          K91isdn
K02oddjobd                      K35winbind      K87irqbalance       K92ip6tables
K03yum-updatesd                 K50ibmasm       K87mcstrans         K92iptables
K05anacron                      K56acpid        K87portmap          K95firstboot
K05atd                          K60crond        K88auditd           K95kudzu
K05conman                       K68rpcidmapd    K88pcscd            K99cpuspeed
K05saslauthd                    K69rpcgssd      K88syslog           K99microcode_ctl
K10cups                         K69rpcsvcgssd   K88wpa_supplicant   K99readahead_early
K10hplip                        K72autofs       K89dund             K99readahead_later
K10psacct                       K73ypbind       K89hidd             S00killall
K10xfs                          K74apmd         K89netplugd         S01halt

< snipped >
```

Sidetrack – chkconfig command

```
[root@benji rc.d]# chkconfig --list cups
cups            0:off   1:off   2:off   3:on    4:on    5:on    6:off
[root@benji rc.d]# cd /etc/rc.d
[root@benji rc.d]# ls –R

< snipped >

./rc1.d:
K01smartd                   K15gpm          K74nscd             K89pand
K02avahi-daemon             K20nfs          K74ntpd             K89rdisc
K02avahi-dnsconfd           K24irda         K75netfs            K90bluetooth
K02dhcdbd                   K25sshd         K85mdmonitor        K90network
K02haldaemon                K30sendmail     K85mdmpd            K90restorecond
K02NetworkManager           K30spamassassin K85messagebus       K91capi
K02NetworkManagerDispatcher K35vncserver    K86nfslock          K91isdn
K02oddjobd                  K35winbind      K87irqbalance       K92ip6tables
K03yum-updatesd             K50ibmasm       K87mcstrans         K92iptables
K05anacron                  K56acpid        K87portmap          K95firstboot
K05atd                      K60crond        K88auditd           K95kudzu
K05conman                   K68rpcidmapd    K88pcscd            K99microcode_ctl
K05saslauthd                K69rpcgssd      K88syslog           K99readahead_early
K10cups                     K69rpcsvcgssd   K88wpa_supplicant   K99readahead_later
K10hplip                    K72autofs       K89dund             S06cpuspeed
K10psacct                   K73ypbind       K89hidd             S99single
K10xfs                      K74apmd         K89netplugd

< snipped >
```

Sidetrack – chkconfig command

```
[root@benji rc.d]# chkconfig --list cups
cups            0:off   1:off   2:off   3:on    4:on    5:on    6:off
[root@benji rc.d]# cd /etc/rc.d
[root@benji rc.d]# ls –R

< snipped >

./rc2.d:
K02avahi-daemon                 K35winbind          K89netplugd          S12syslog
K02avahi-dnsconfd               K50ibmasm           K89pand              S13irqbalance
K02dhcdbd                       K56acpid            K89rdisc             S13mcstrans
K02haldaemon                    K68rpcidmapd        K90bluetooth         S15mdmonitor
K02NetworkManager               K69rpcgssd          K91capi              S25pcscd
K02NetworkManagerDispatcher     K69rpcsvcgssd       K95firstboot         S26apmd
K02oddjobd                      K72autofs           K95kudzu             S26hidd
K03yum-updatesd                 K73ypbind           K99readahead_later   S50hplip
K05atd                          K74nscd             S00microcode_ctl     S55sshd
K05conman                       K74ntpd             S04readahead_early   S80sendmail
K05saslauthd                    K75netfs            S06cpuspeed          S85gpm
K10cups                         K85mdmpd            S08ip6tables         S90crond
K10psacct                       K85messagebus       S08iptables          S90xfs
K20nfs                          K86nfslock          S09isdn              S95anacron
K24irda                         K87portmap          S10network           S99local
K30spamassassin                 K88wpa_supplicant   S10restorecond       S99smartd
K35vncserver                    K89dund             S11auditd

< snipped >
```

Sidetrack – chkconfig command

```
[root@benji rc.d]# chkconfig --list cups
cups            0:off   1:off   2:off   3:on    4:on    5:on    6:off
[root@benji rc.d]# cd /etc/rc.d
[root@benji rc.d]# ls –R

< snipped >

./rc3.d:
K02avahi-dnsconfd               K74ntpd                 S10network      S44acpid
K02dhcdbd                       K85mdmpd                S10restorecond  S50hplip
K02NetworkManager               K88wpa_supplicant       S11auditd       S55cups
K02NetworkManagerDispatcher     K89dund                 S12syslog       S55sshd
K02oddjobd                      K89netplugd             S13irqbalance   S80sendmail
K05conman                       K89pand                 S13mcstrans     S85gpm
K05saslauthd                    K89rdisc                S13portmap      S90crond
K10psacct                       K90bluetooth            S14nfslock      S90xfs
K20nfs                          K91capi                 S15mdmonitor    S95anacron
K24irda                         K99readahead_later      S18rpcidmapd    S95atd
K30spamassassin                 S00microcode_ctl        S19rpcgssd      S97yum-updatesd
K35vncserver                    S04readahead_early      S22messagebus   S98avahi-daemon
K35winbind                      S05kudzu                S25netfs        S98haldaemon
K50ibmasm                       S06cpuspeed             S25pcscd        S99firstboot
K69rpcsvcgssd                   S08ip6tables            S26apmd         S99local
K73ypbind                       S08iptables             S26hidd         S99smartd
K74nscd                         S09isdn                 S28autofs

< snipped >
```

Sidetrack – chkconfig command

```
[root@benji rc.d]# chkconfig --list cups
cups            0:off   1:off   2:off   3:on    4:on    5:on    6:off
[root@benji rc.d]# cd /etc/rc.d
[root@benji rc.d]# ls –R

< snipped >

./rc4.d:
K02avahi-dnsconfd               K74ntpd                 S09isdn         S28autofs
K02dhcdbd                       K85mdmpd                S10network      S44acpid
K02NetworkManager               K88wpa_supplicant       S10restorecond  S50hplip
K02NetworkManagerDispatcher     K89dund                 S11auditd       S55cups
K02oddjobd                      K89netplugd             S12syslog       S55sshd
K05conman                       K89pand                 S13irqbalance   S80sendmail
K05saslauthd                    K89rdisc                S13mcstrans     S85gpm
K10psacct                       K90bluetooth            S13portmap      S90crond
K20nfs                          K91capi                 S14nfslock      S90xfs
K24irda                         K95firstboot            S15mdmonitor    S95anacron
K30spamassassin                 K99readahead_later      S18rpcidmapd    S95atd
K35vncserver                    S00microcode_ctl        S19rpcgssd      S97yum-updatesd
K35winbind                      S04readahead_early      S22messagebus   S98avahi-daemon
K50ibmasm                       S05kudzu                S25netfs        S98haldaemon
K69rpcsvcgssd                   S06cpuspeed             S25pcscd        S99local
K73ypbind                       S08ip6tables            S26apmd         S99smartd
K74nscd                         S08iptables             S26hidd

< snipped >
```

Sidetrack – chkconfig command

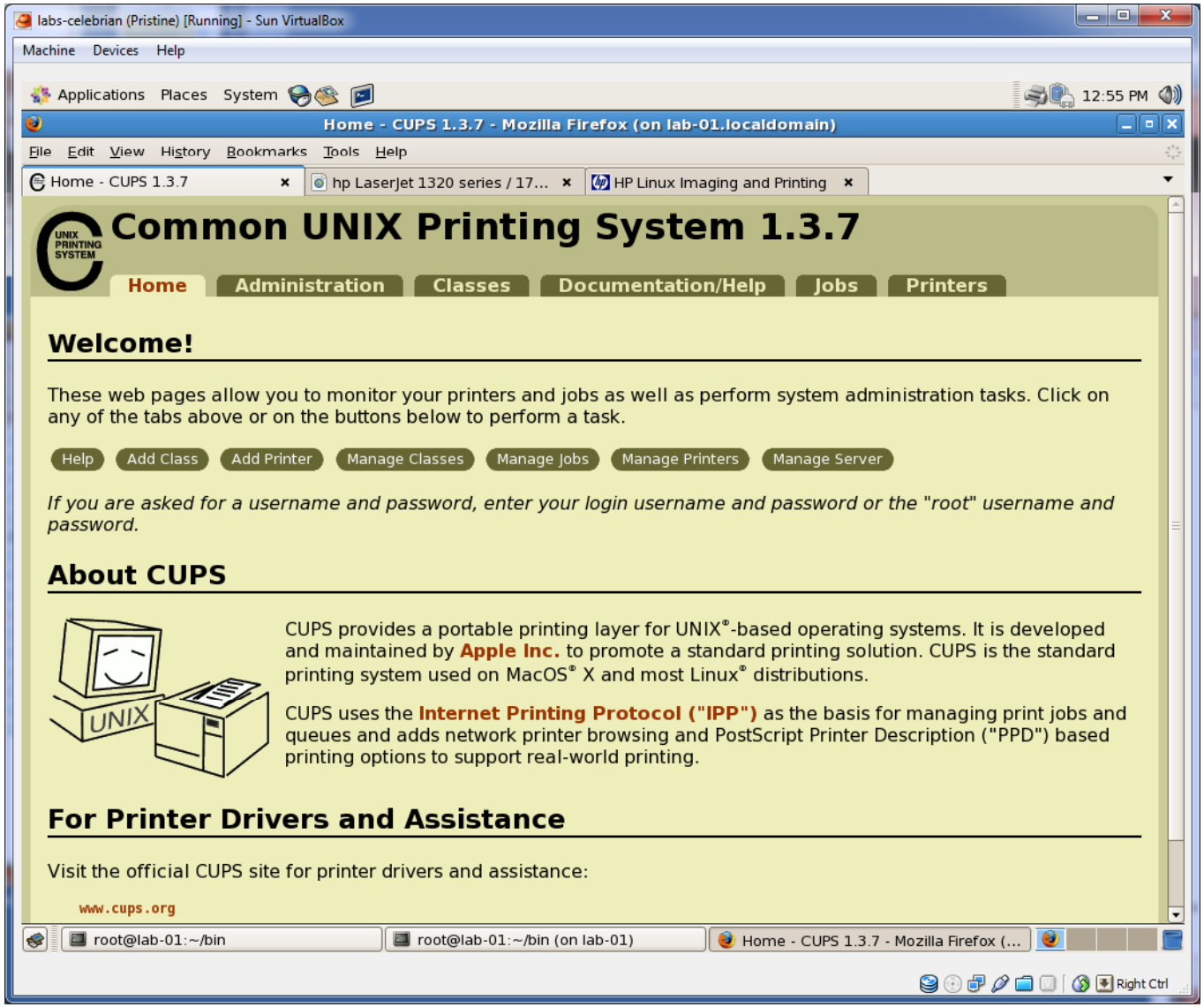
```
[root@benji rc.d]# chkconfig --list cups
cups            0:off   1:off   2:off   3:on    4:on    5:on    6:off
[root@benji rc.d]# cd /etc/rc.d
[root@benji rc.d]# ls –R

< snipped >

./rc5.d:
K02avahi-dnsconfd               K74ntpd             S10restorecond  S50hplip
K02dhcdbd                       K85mdmpd            S11auditd       S55cups
K02NetworkManager               K88wpa_supplicant   S12syslog       S55sshd
K02NetworkManagerDispatcher     K89dund             S13irqbalance   S80sendmail
K02oddjobd                      K89netplugd         S13mcstrans     S85gpm
K05conman                       K89pand             S13portmap      S90crond
K05saslauthd                    K89rdisc            S14nfslock      S90xfs
K10psacct                       K90bluetooth        S15mdmonitor    S95anacron
K20nfs                          K91capi             S18rpcidmapd    S95atd
K24irda                         S00microcode_ctl    S19rpcgssd      S96readahead_later
K30spamassassin                 S04readahead_early  S22messagebus   S97yum-updatesd
K35vncserver                    S05kudzu            S25netfs        S98avahi-daemon
K35winbind                      S06cpuspeed         S25pcscd        S98haldaemon
K50ibmasm                       S08ip6tables        S26apmd         S99firstboot
K69rpcsvcgssd                   S08iptables         S26hidd         S99local
K73ypbind                       S09isdn             S28autofs       S99smartd
K74nscd                         S10network          S44acpid

< snipped >
```

Sidetrack – chkconfig command

```
[root@benji rc.d]# chkconfig --list cups
cups            0:off   1:off   2:off   3:on    4:on    5:on    6:off
[root@benji rc.d]# cd /etc/rc.d
[root@benji rc.d]# ls –R

< snipped >

./rc6.d:
K01smartd                       K15gpm          K74nscd             K89pand
K02avahi-daemon                 K20nfs          K74ntpd             K89rdisc
K02avahi-dnsconfd               K24irda         K75netfs            K90bluetooth
K02dhcdbd                       K25sshd         K85mdmonitor        K90network
K02haldaemon                    K30sendmail     K85mdmpd            K90restorecond
K02NetworkManager               K30spamassassin K85messagebus       K91capi
K02NetworkManagerDispatcher     K35vncserver    K86nfslock          K91isdn
K02oddjobd                      K35winbind      K87irqbalance       K92ip6tables
K03yum-updatesd                 K50ibmasm       K87mcstrans         K92iptables
K05anacron                      K56acpid        K87portmap          K95firstboot
K05atd                          K60crond        K88auditd           K95kudzu
K05conman                       K68rpcidmapd    K88pcscd            K99cpuspeed
K05saslauthd                    K69rpcgssd      K88syslog           K99microcode_ctl
K10cups                         K69rpcsvcgssd   K88wpa_supplicant   K99readahead_early
K10hplip                        K72autofs       K89dund             K99readahead_later
K10psacct                       K73ypbind       K89hidd             S00killall
K10xfs                          K74apmd         K89netplugd         S01reboot

< snipped >
```

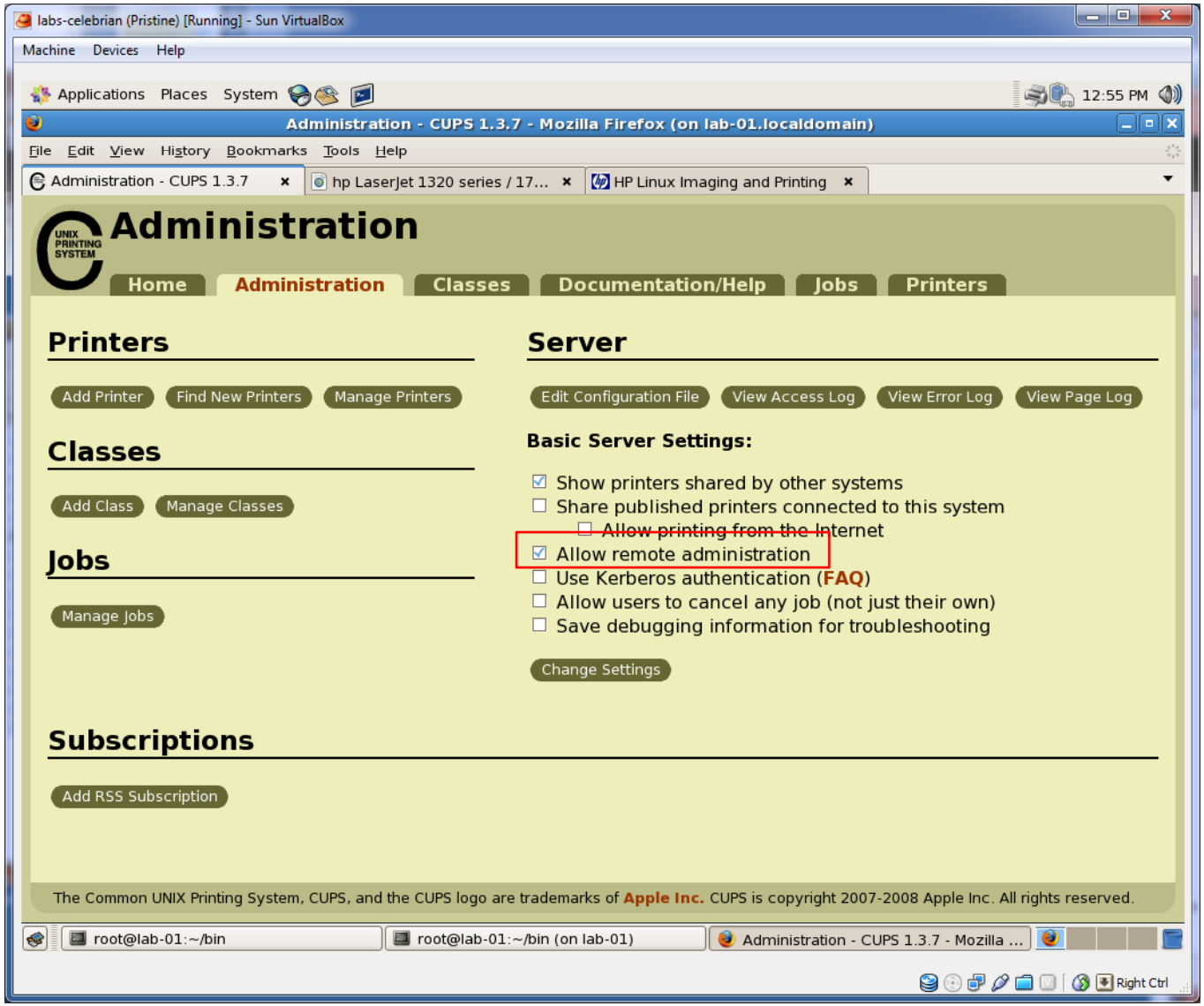
Back to Printer Configuration

CUPS



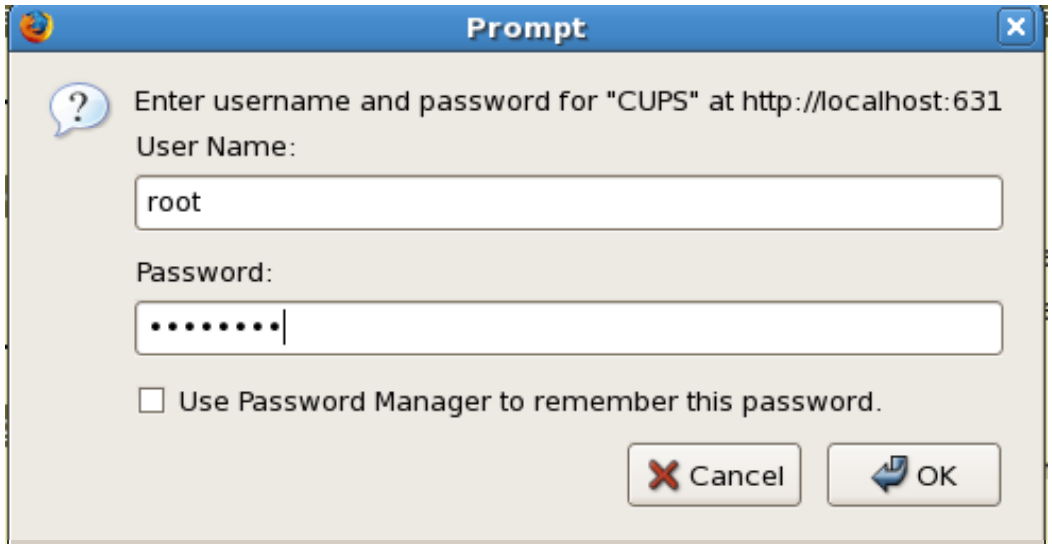
To enable remote access click on Administration tab ...

CUPS



... and scroll down to allow remote administration

CUPS

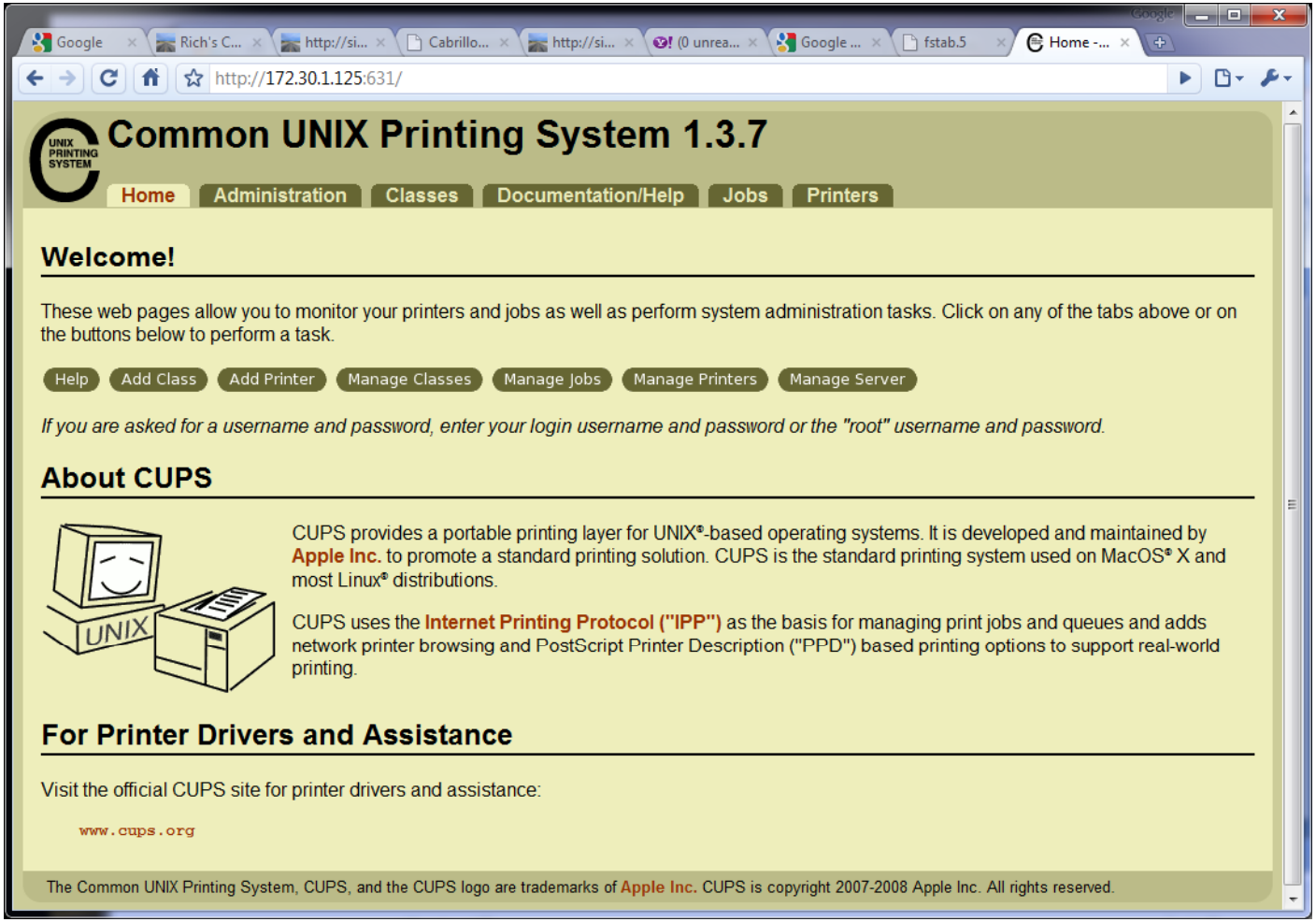


If prompted, authenticate yourself as root to confirm change to allow remote administration

CUPS service will restart

CUPS

Browse to http://xxx.xxx.xxx.xxx:631/ from another system



You may now manage the CUPS service remotely from another system.

CUPS

Next step is to add printers

Printer: HP LaserJet 1320n
Connection: LAN

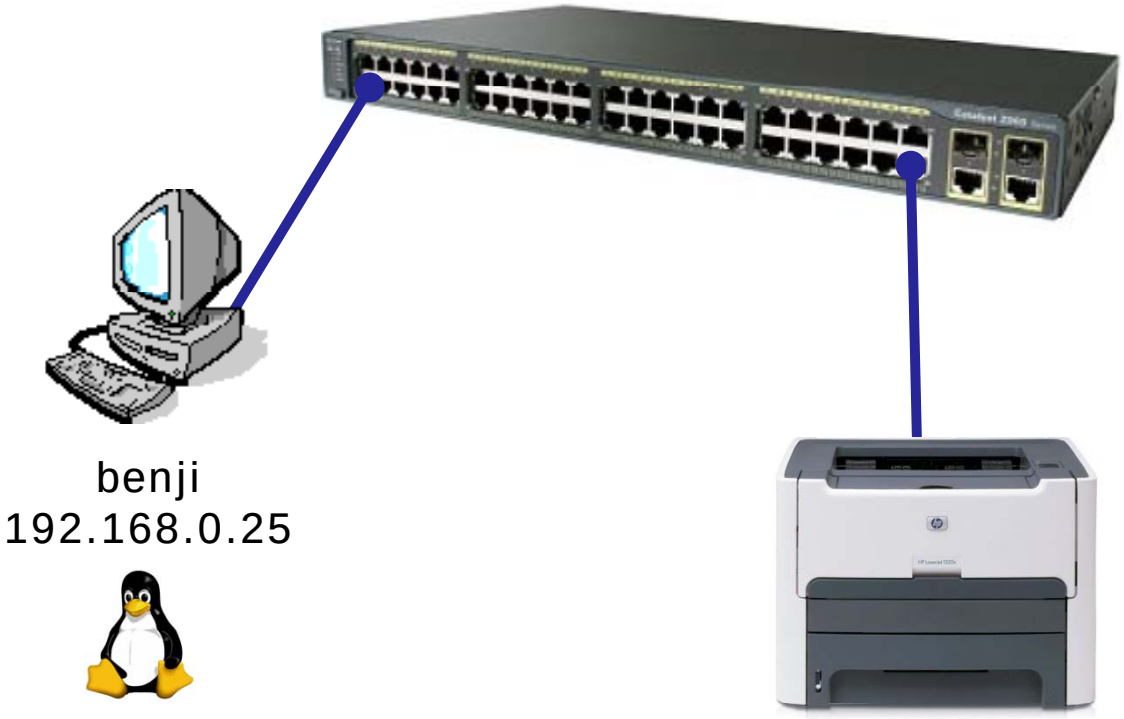
CUPS

The LaserJets also have a web-based management utility

IP Address for this 1320n is 192.168.0.12

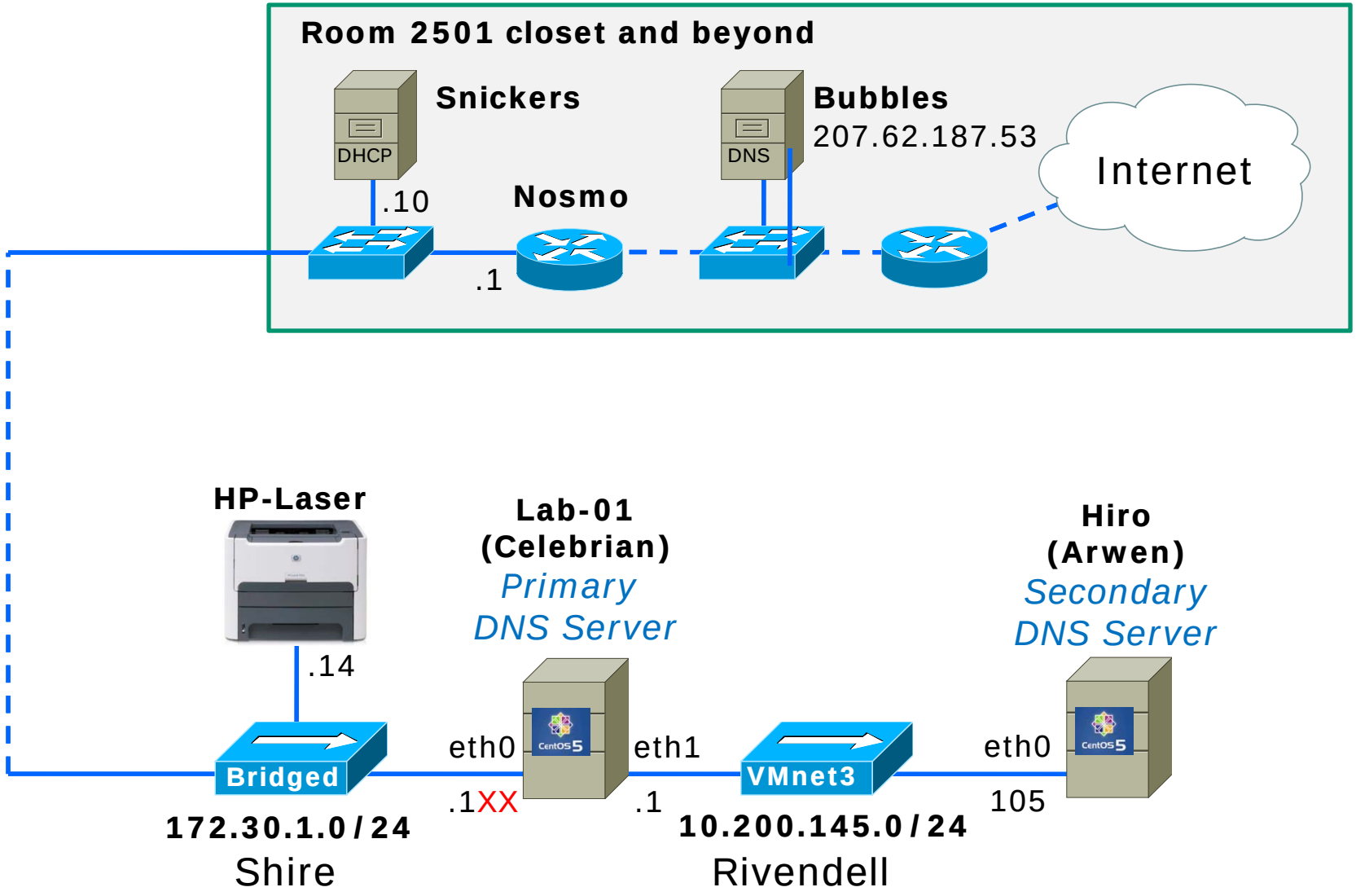
CUPS

This example will show how to add the HP 1320n as a networked printer.



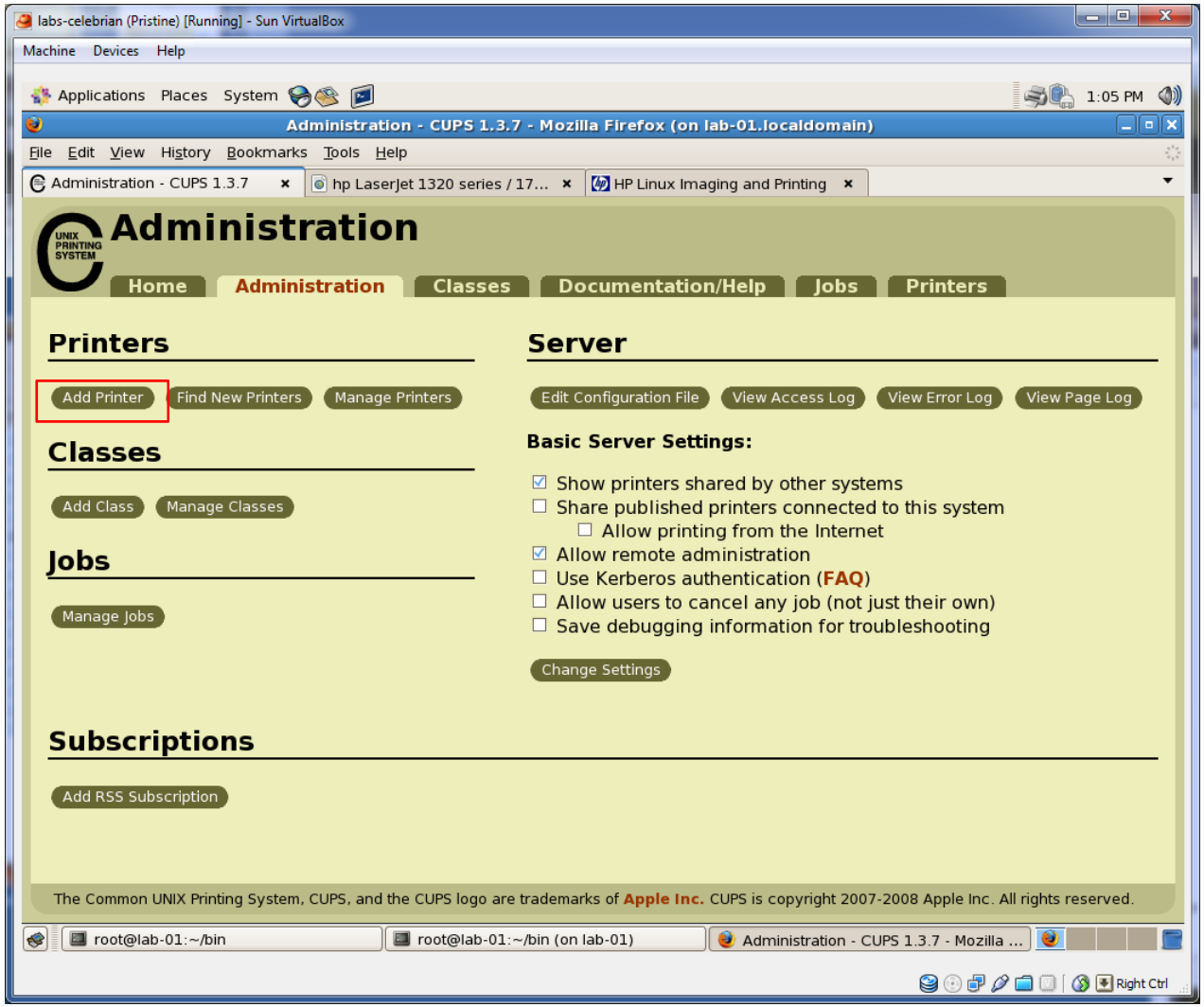
benji
192.168.0.25

hp1320n
192.168.0.12



Recable VMs as necessary

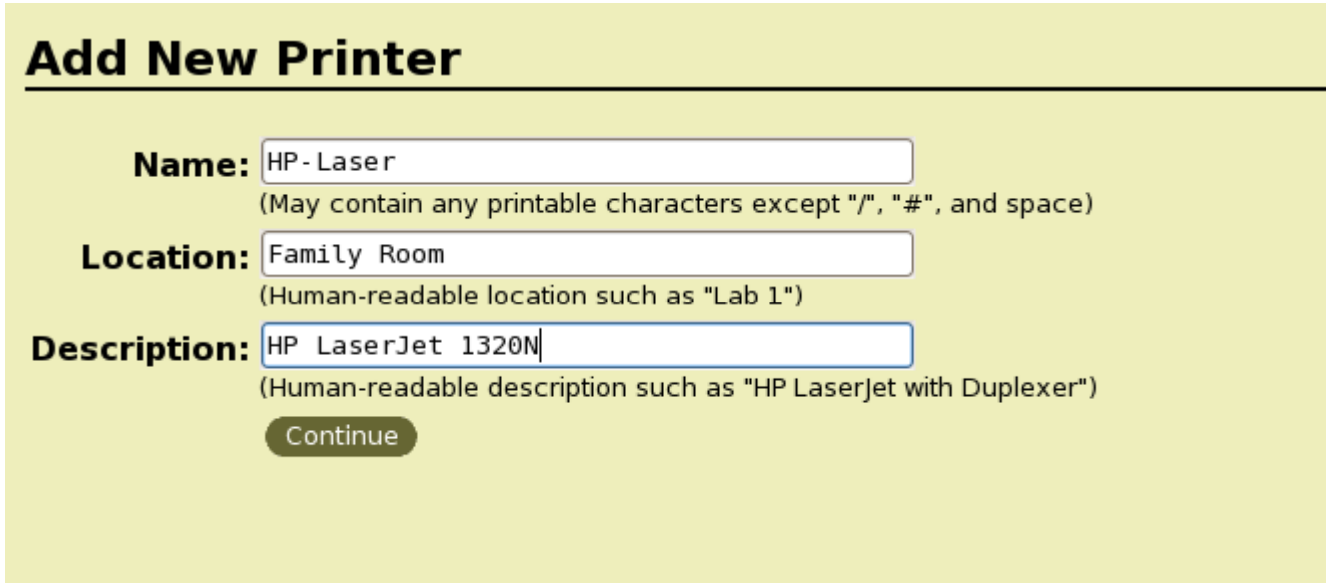
CUPS



To add in HP 1320N printer ...

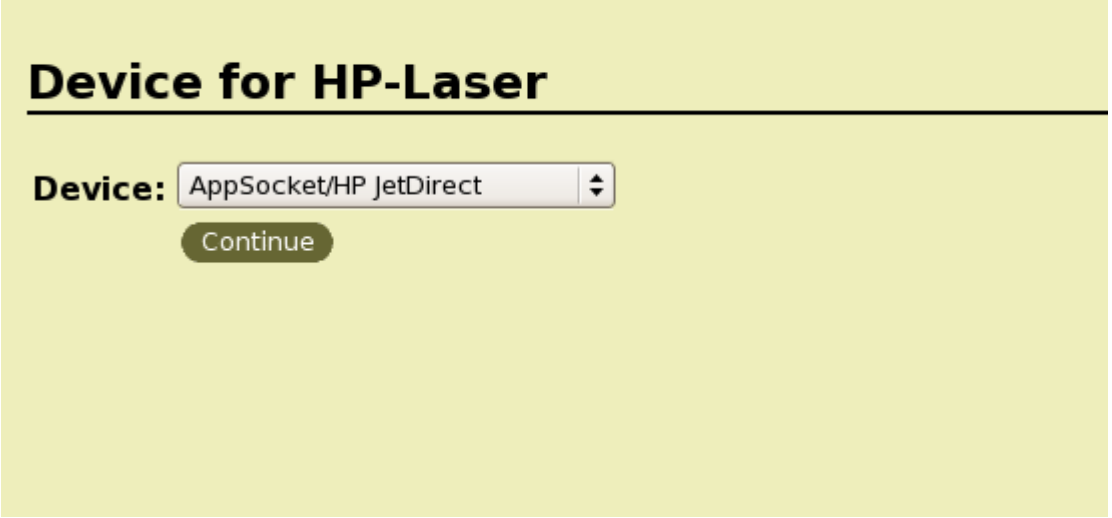
... the first step is to click the Add Printer button

CUPS



Now we can add the LaserJet

CUPS

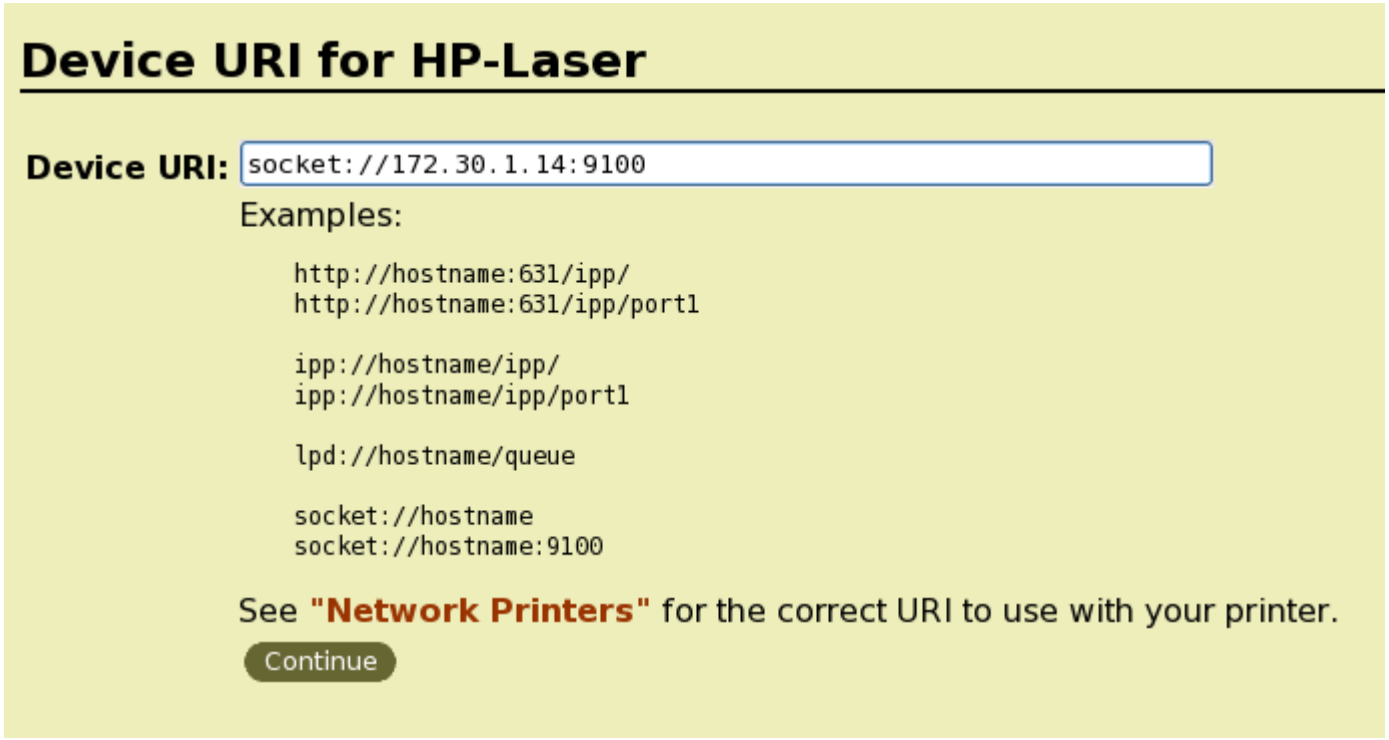


We will use JetDirect.

JetDirect is a small printer server built into many of HP's printers.

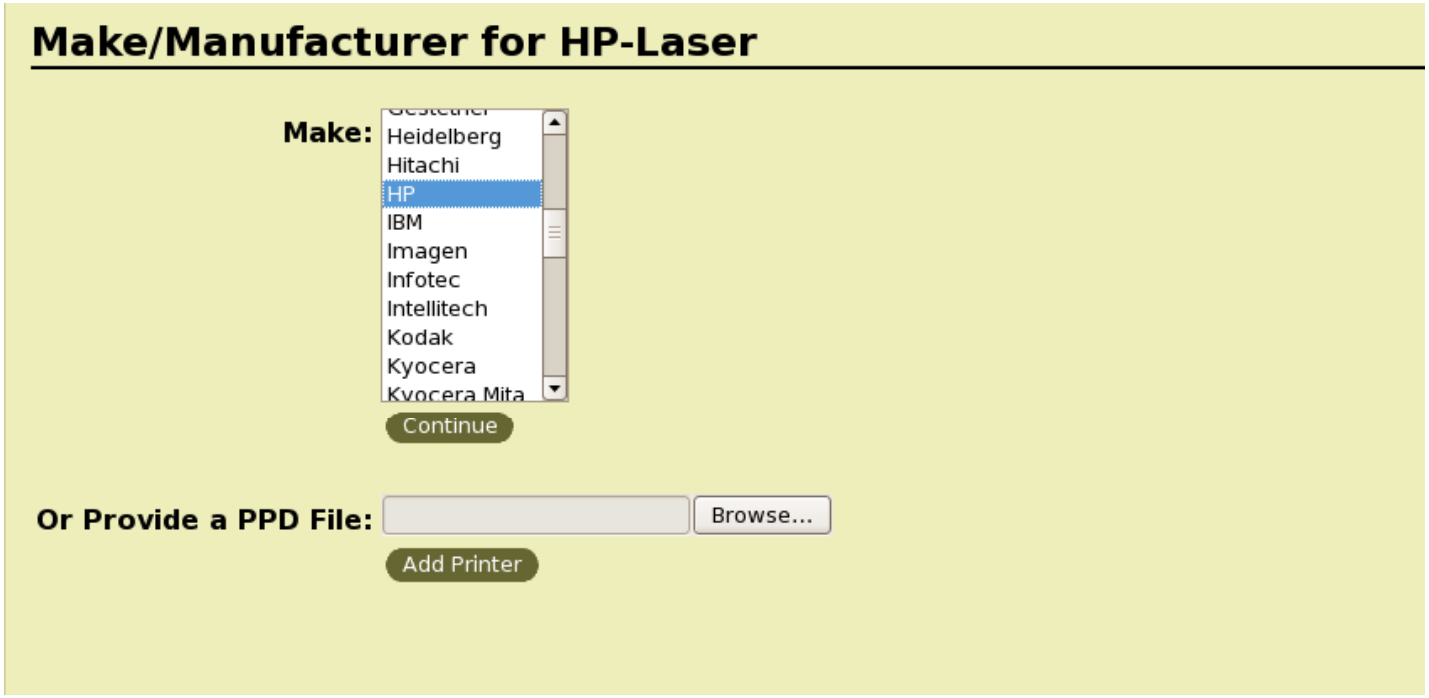
CUPS

socket://172.30.1.14:9100 – Note JetDirect uses port 9100



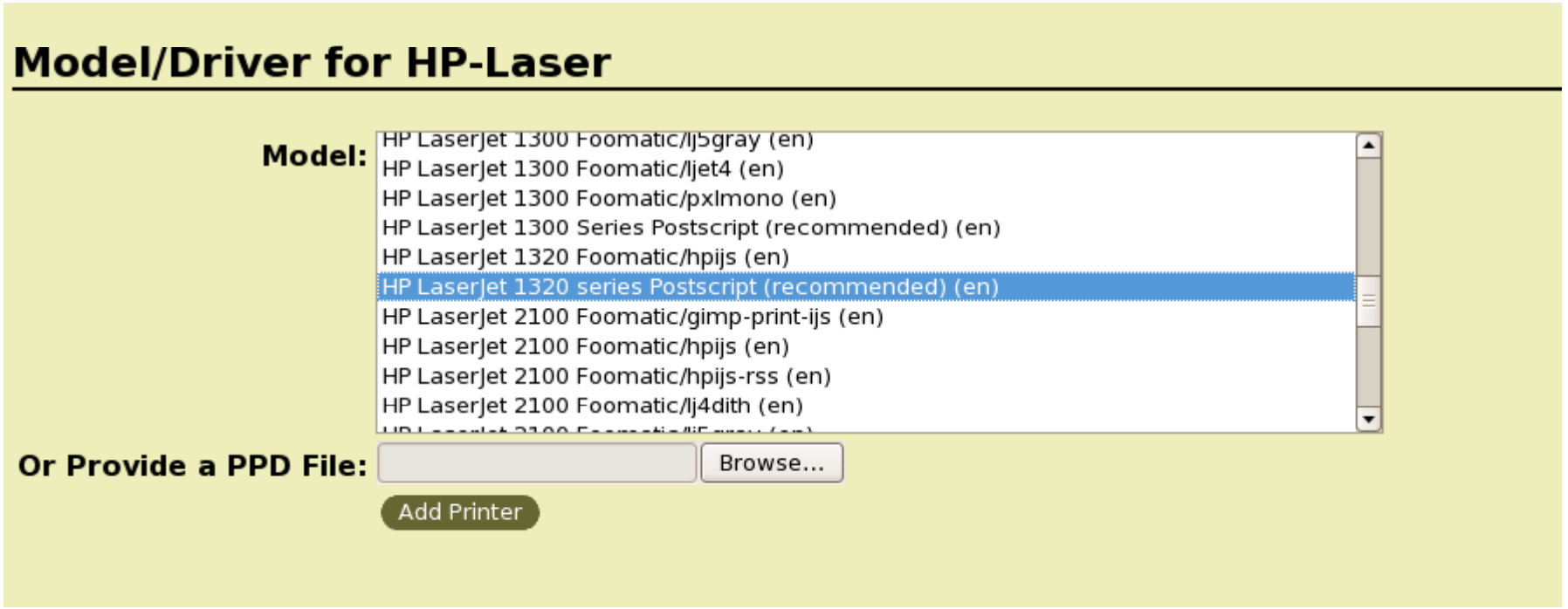
This defines how to communicate with the printer

CUPS



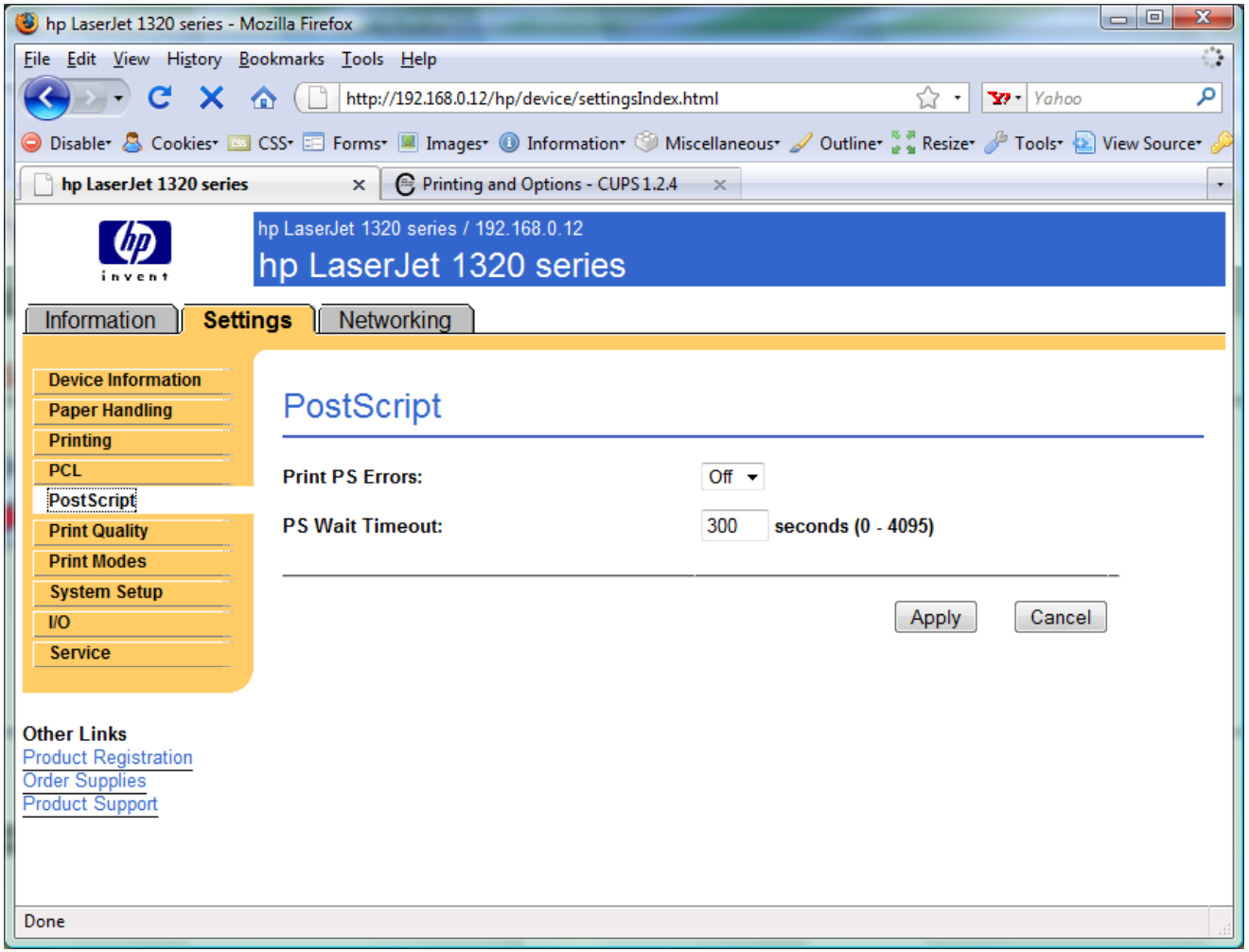
Must select printer vendor from long list

CUPS



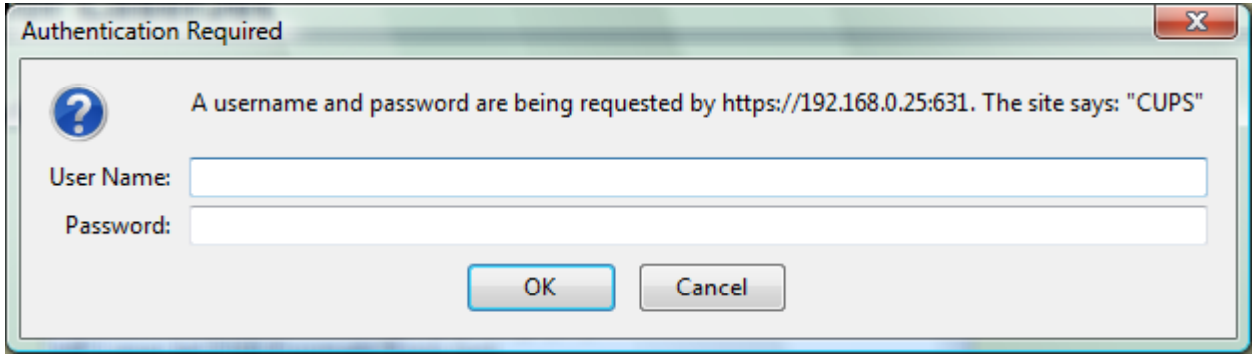
We will choose hp LaserJet 1320 series Postscript (recommended) (en)

CUPS



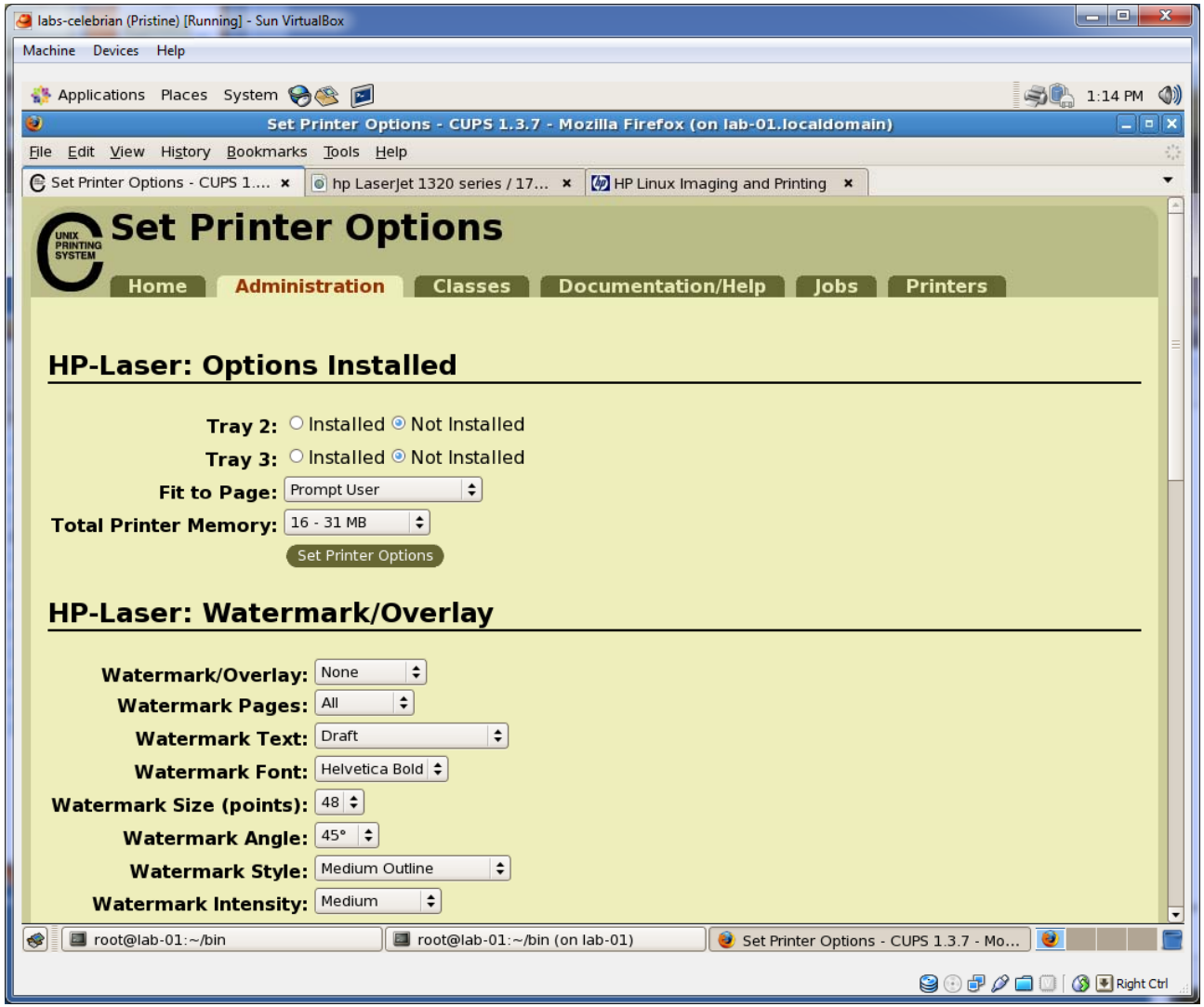
Lets double check the printer supports PostScript … it does

CUPS



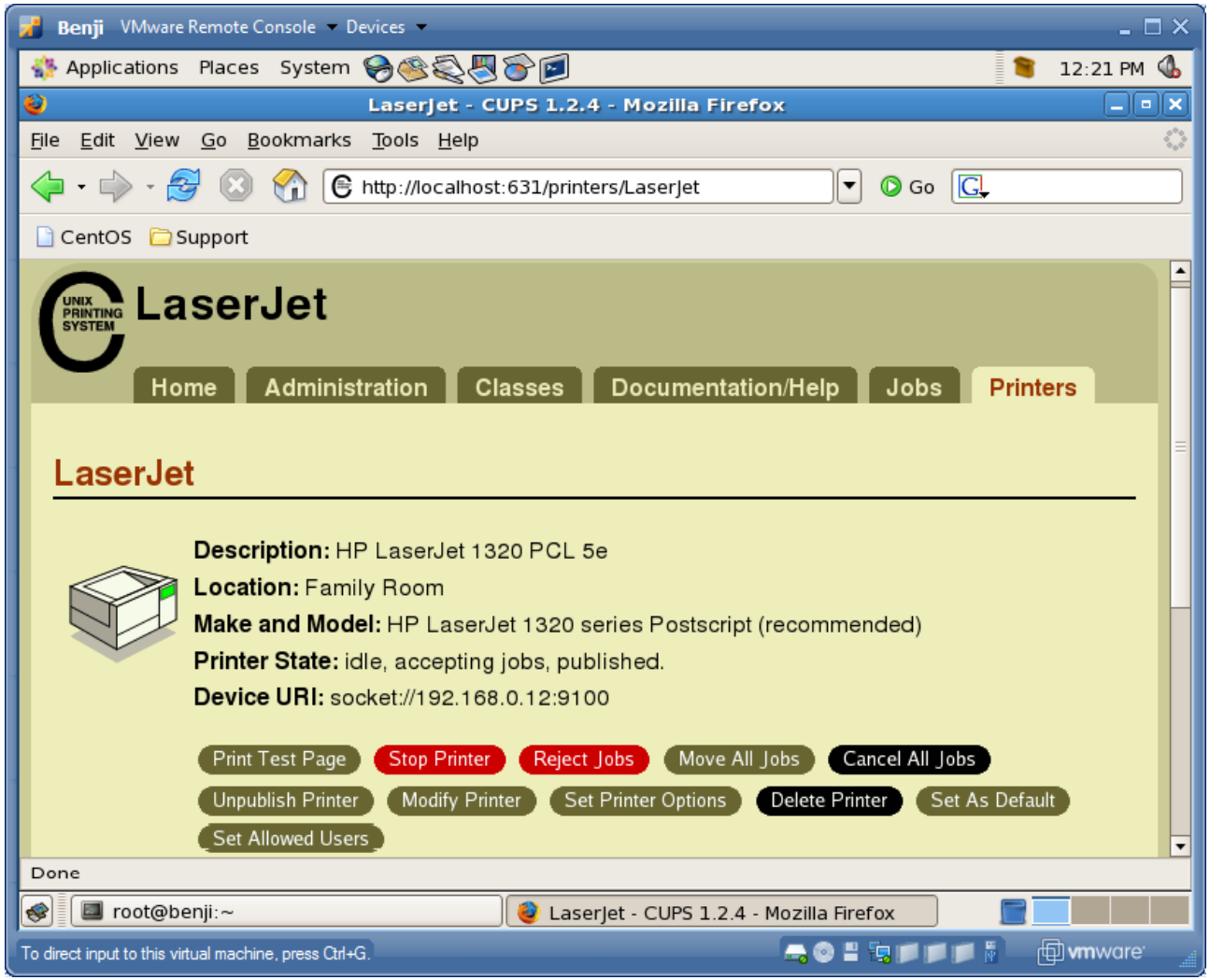
To finally add the printer it may be necessary to authenticate as root

CUPS



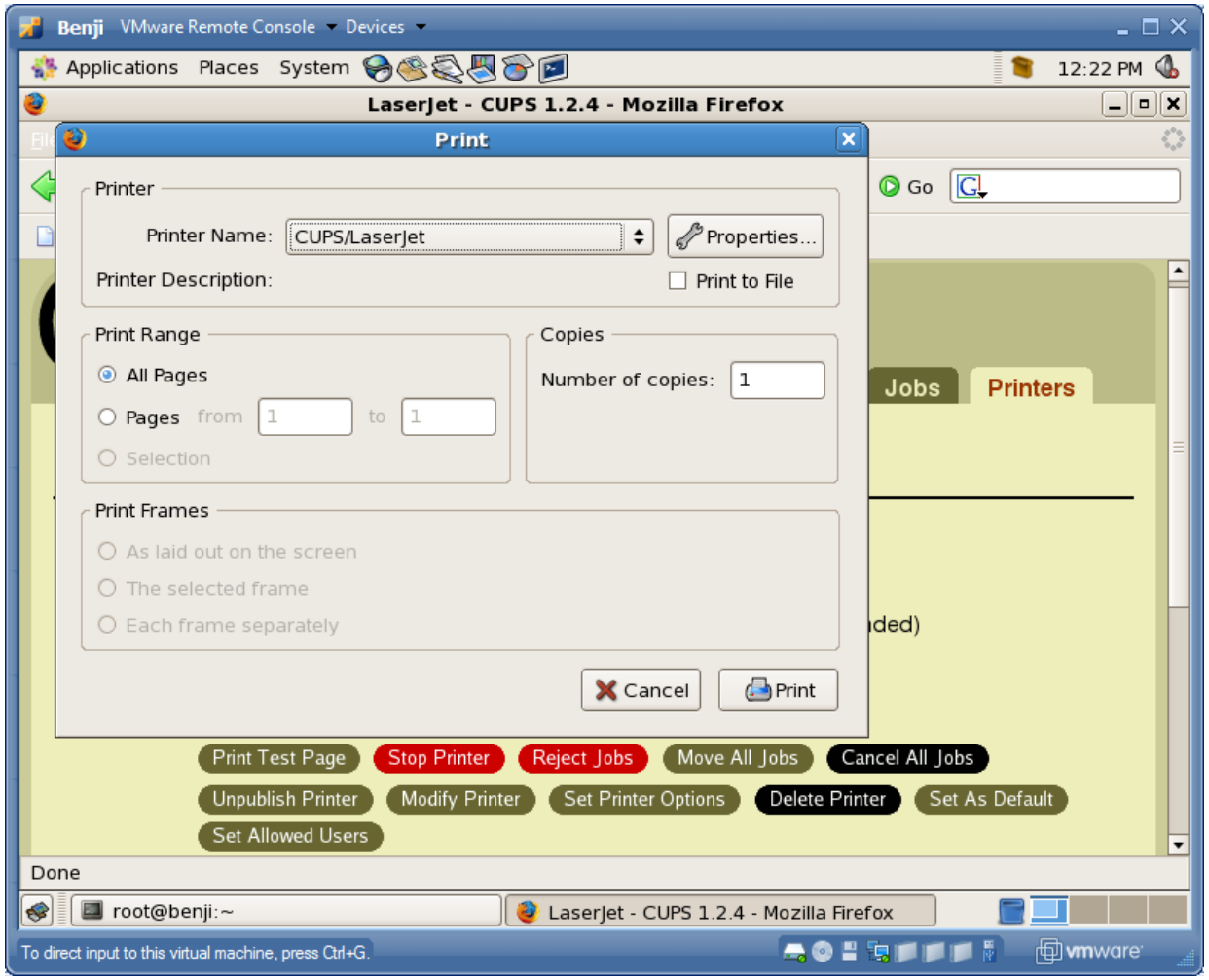
Printer has been added

CUPS



View of newly added printer from Printer tab

CUPS



Lets test the printer by printing this CUPS web page to it ... and it works.

CUPS

Lets add second printer

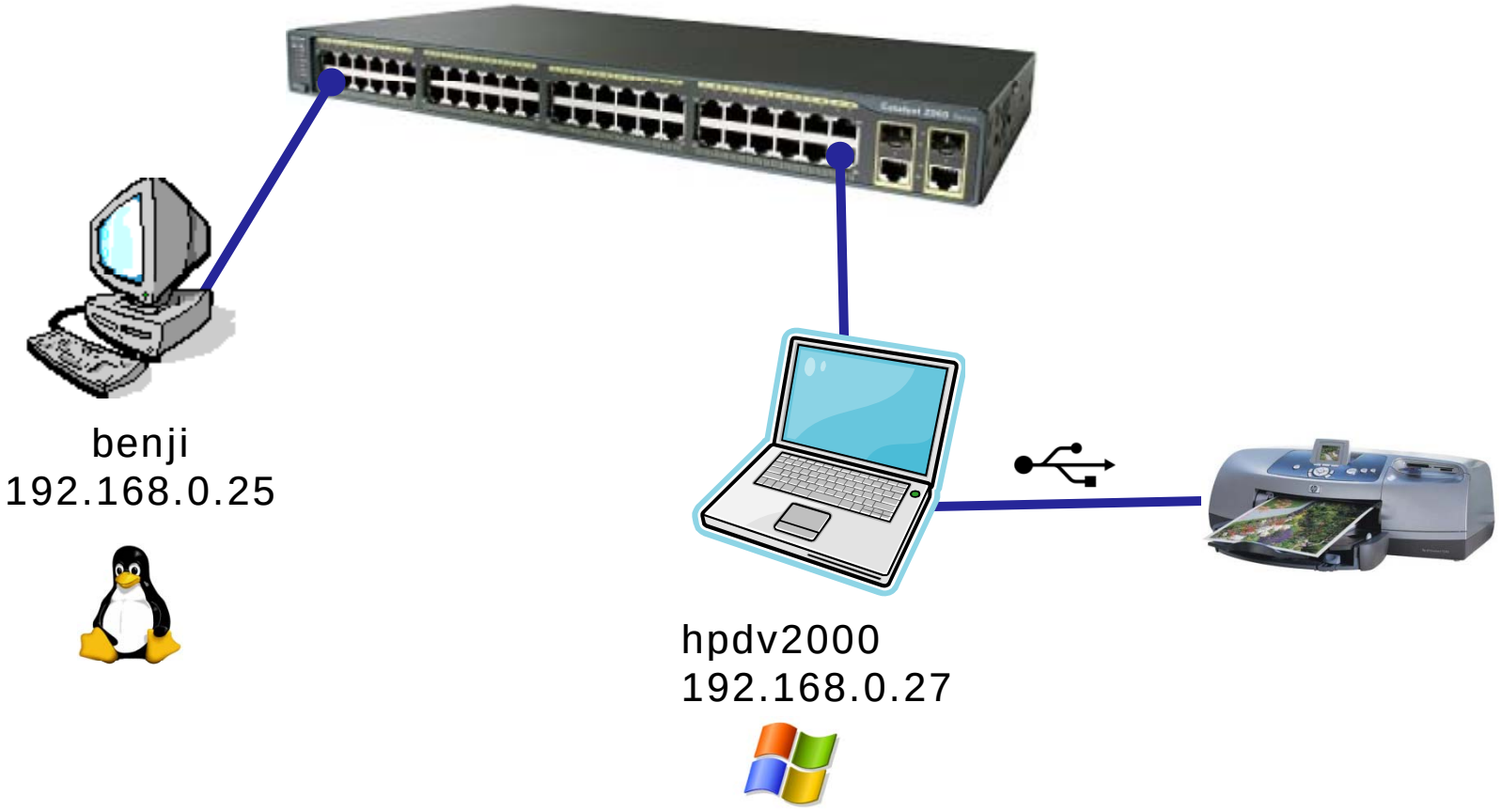
Printer: hp photosmart 7550 (color inkjet technology)
Connection: USB

Sidetrack – The previous 7550 "Hot Lips"

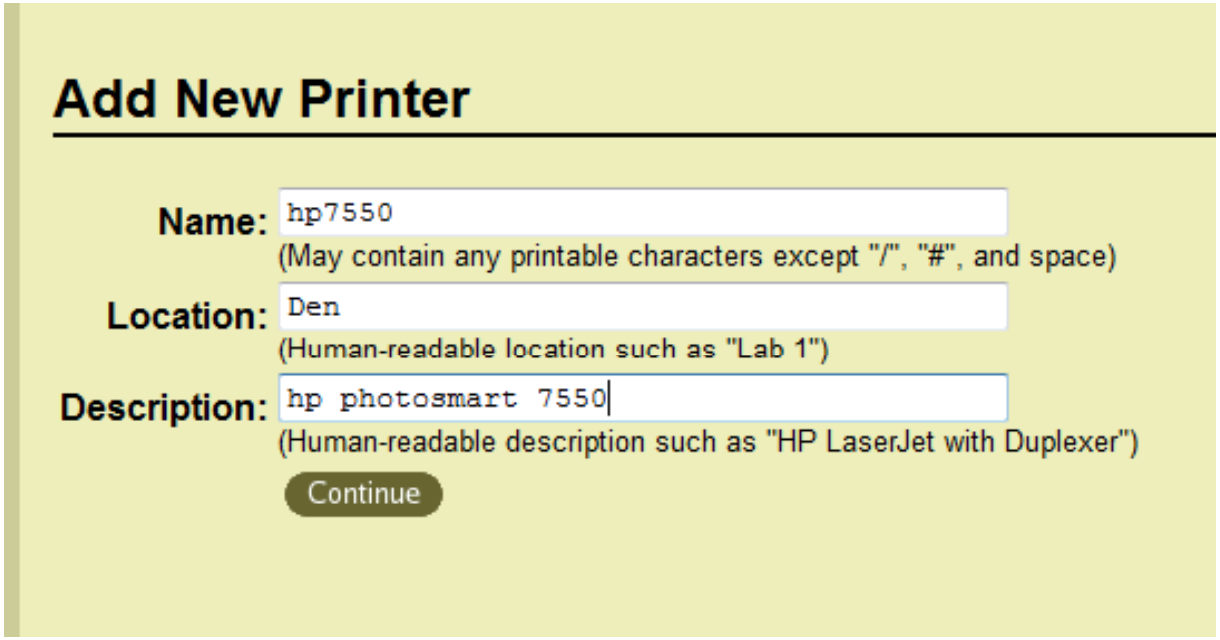
6 G's of acceleration
8-pen turret
Grit wheel technology from HP Labs

CUPS

The second printer is connected by USB to a Windows notebook computer

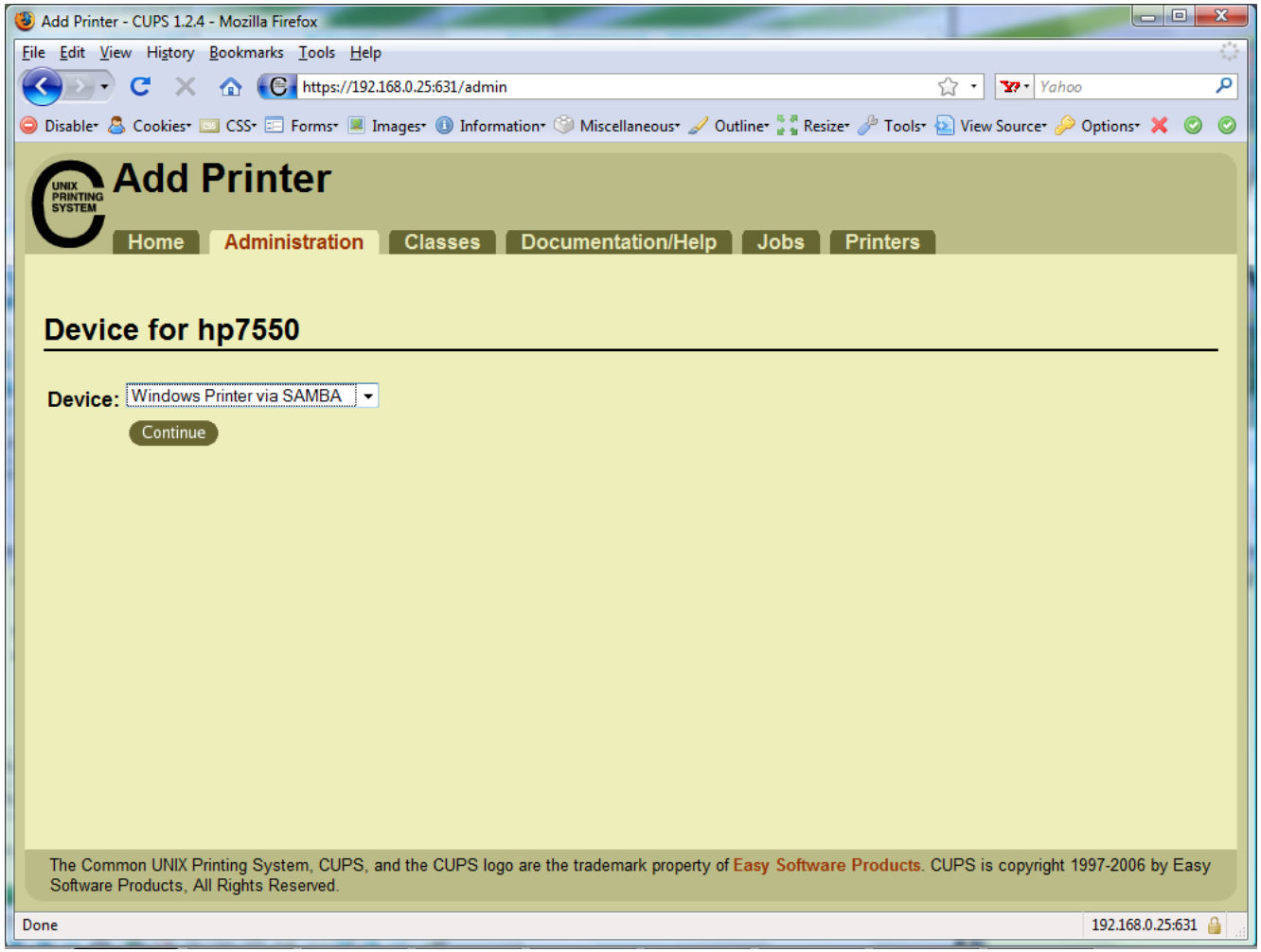


CUPS



First step is the same which is to fill out basic information on printer

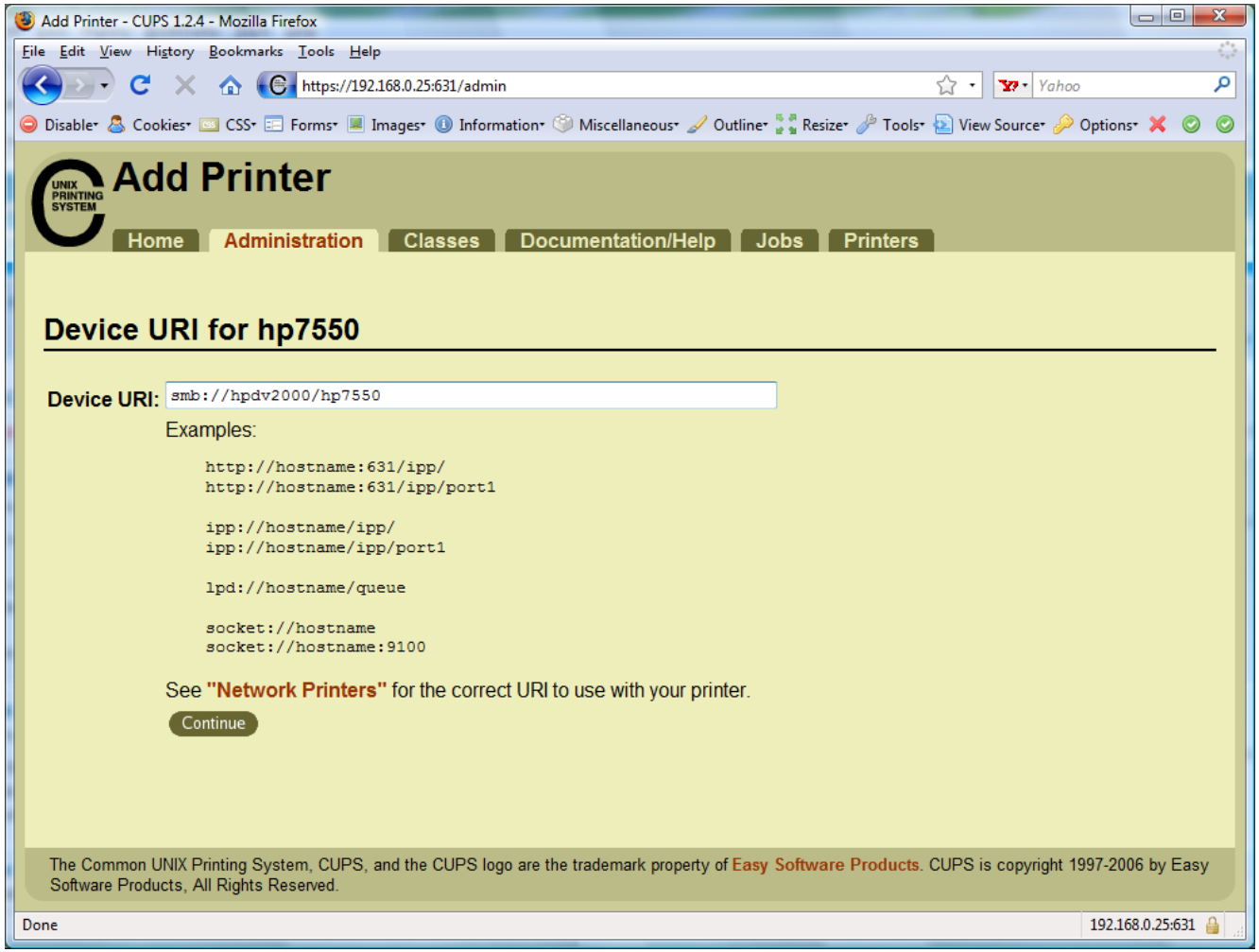
CUPS



For this connection we will use Samba. Samba implements Windows file and print services sharing on Linux.

Note Windows uses SMB (Server Message Block) protocol to implement these services

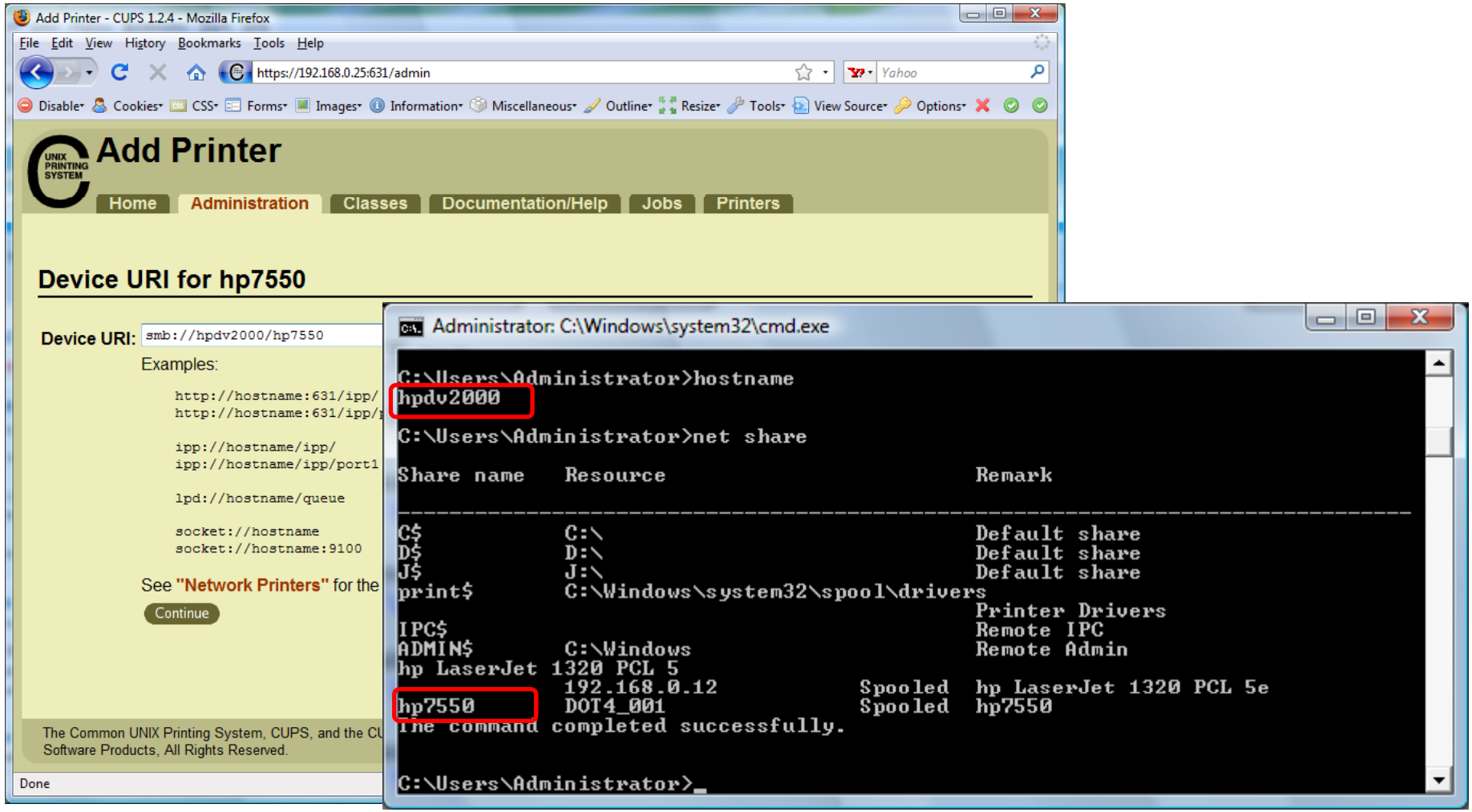
CUPS



Will need to specify the Windows print share

CUPS

Will need to specify the Windows print share as //hostname/printsharename



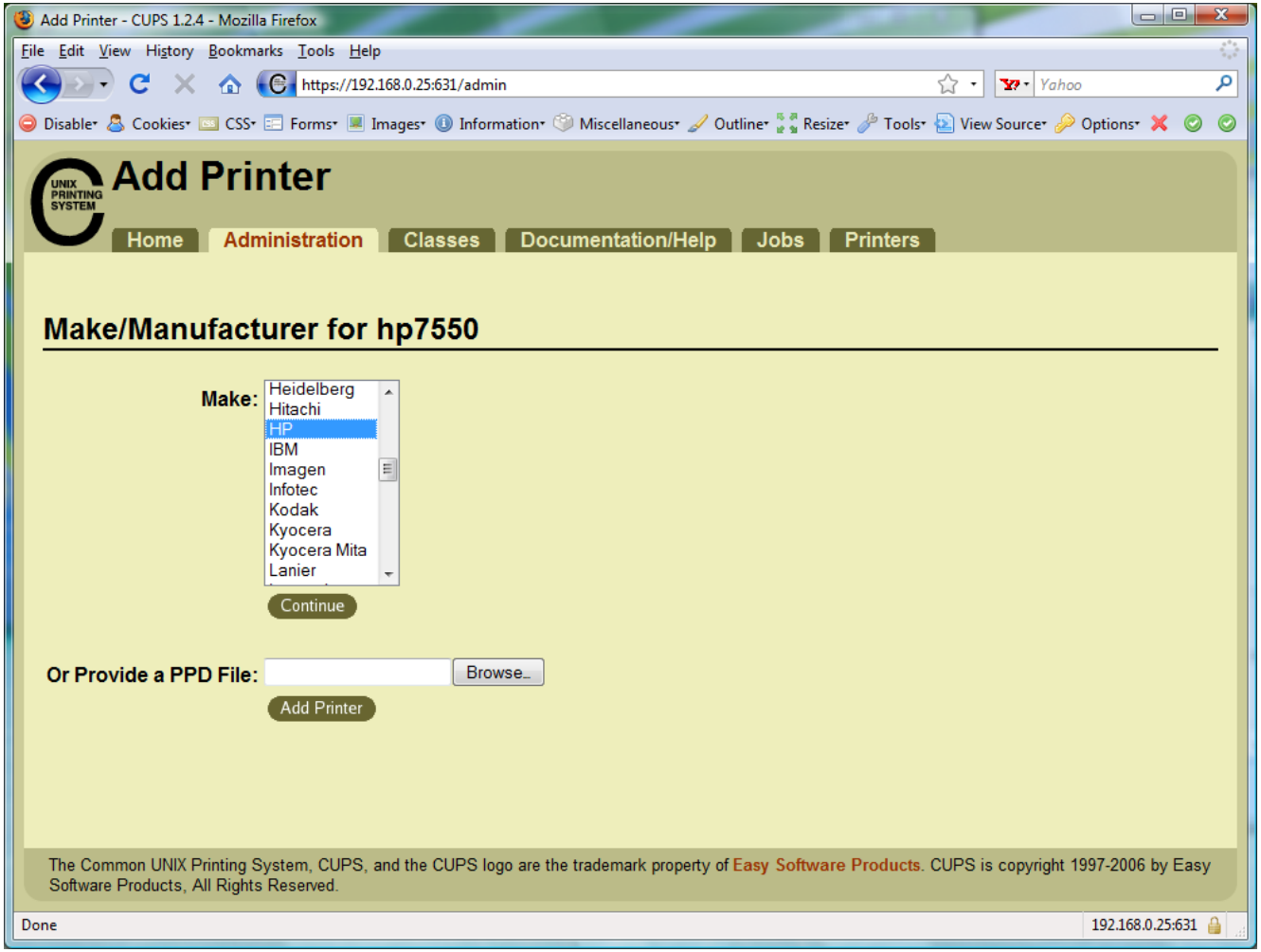
CUPS

Ways to specify a Windows share

	Username and password Not required
This machine is in the same workgroup	`smb://server/sharename`
This machine is in a different workgroup	`smb://workgroup/server/sharename`

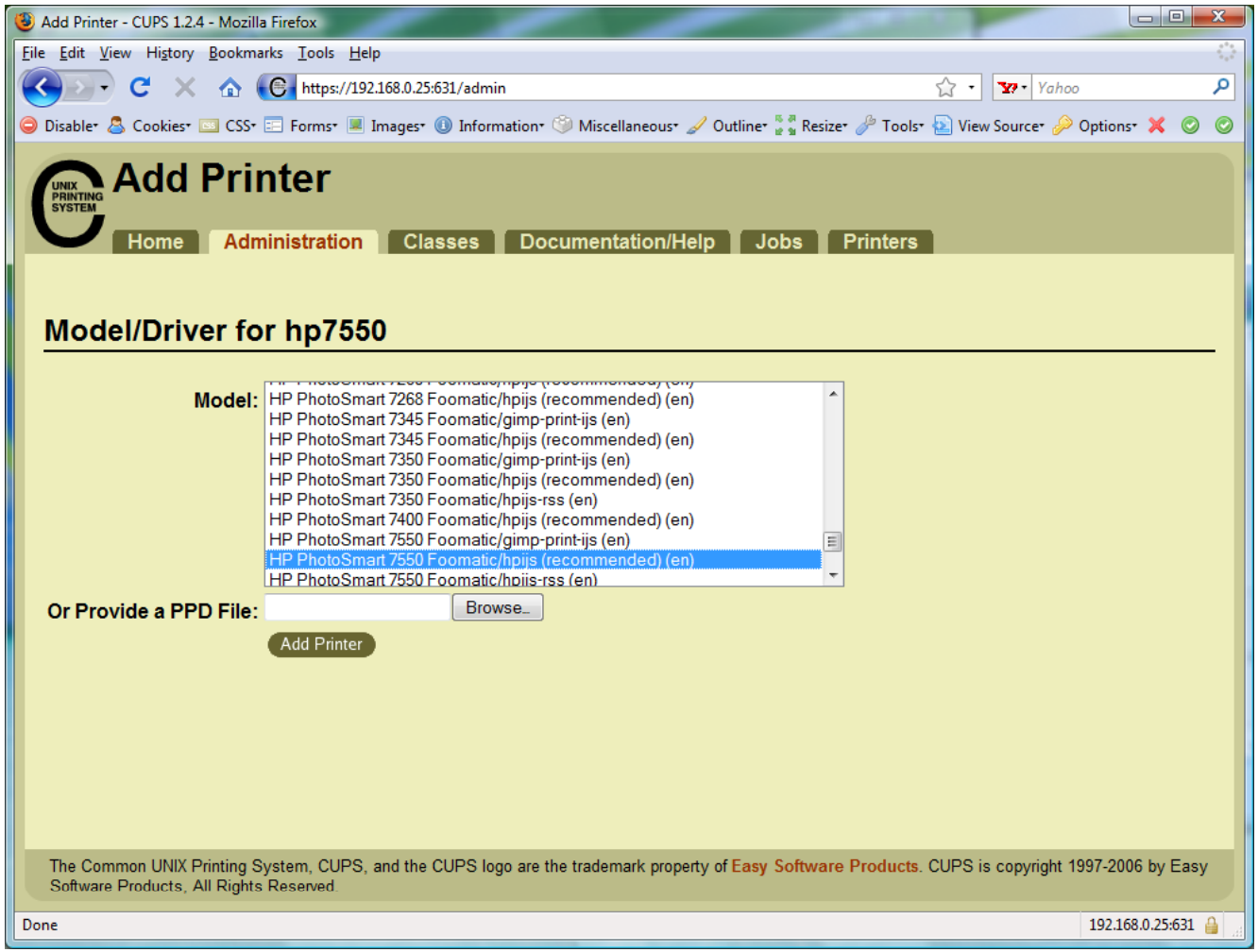
	Username and password required
This machine is in the same workgroup	`smb://username:password@server/sharename`
This machine is in a different workgroup	`smb://username:password@workgroup/server/sharename`

CUPS



Select make of printer

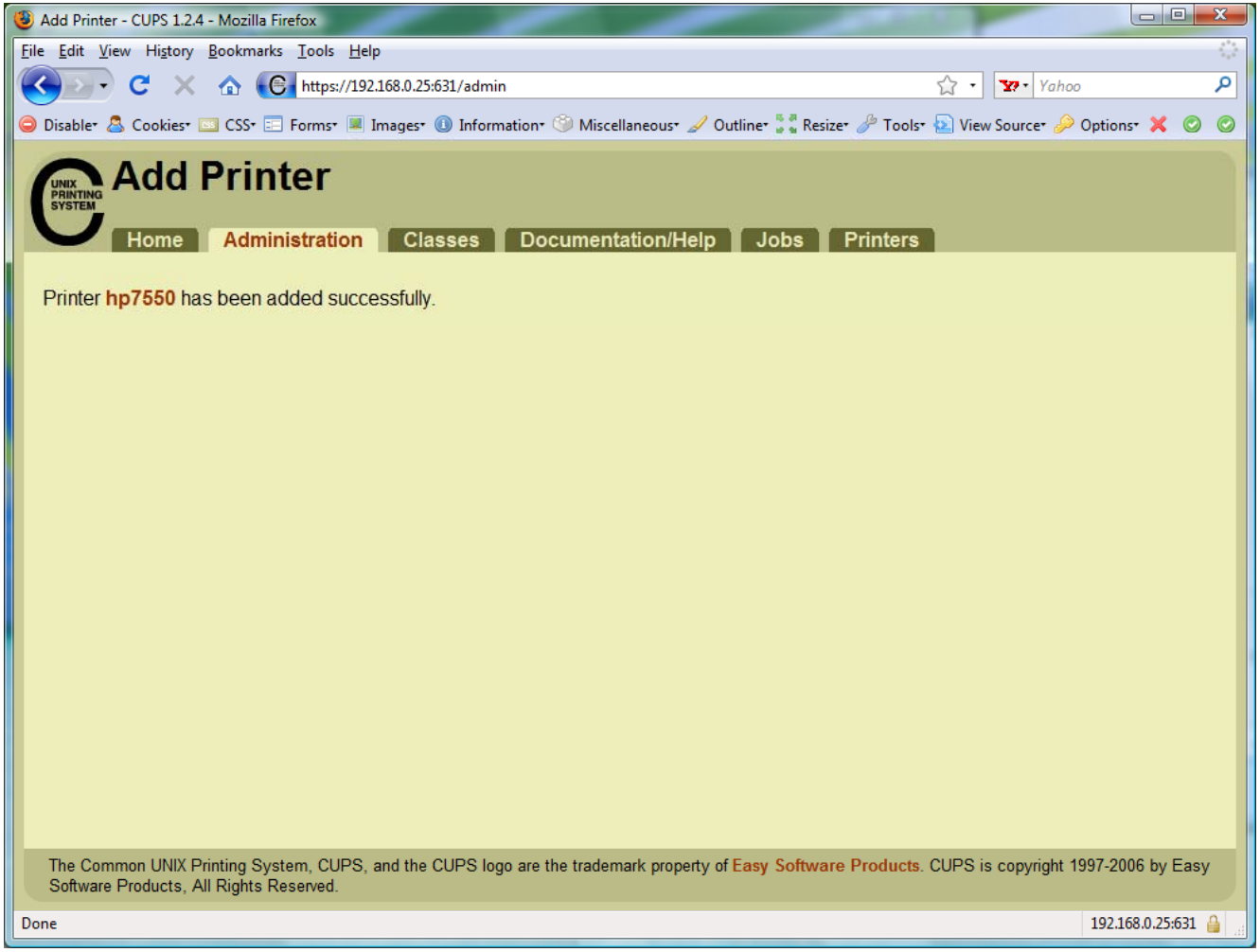
CUPS



Select model of printer

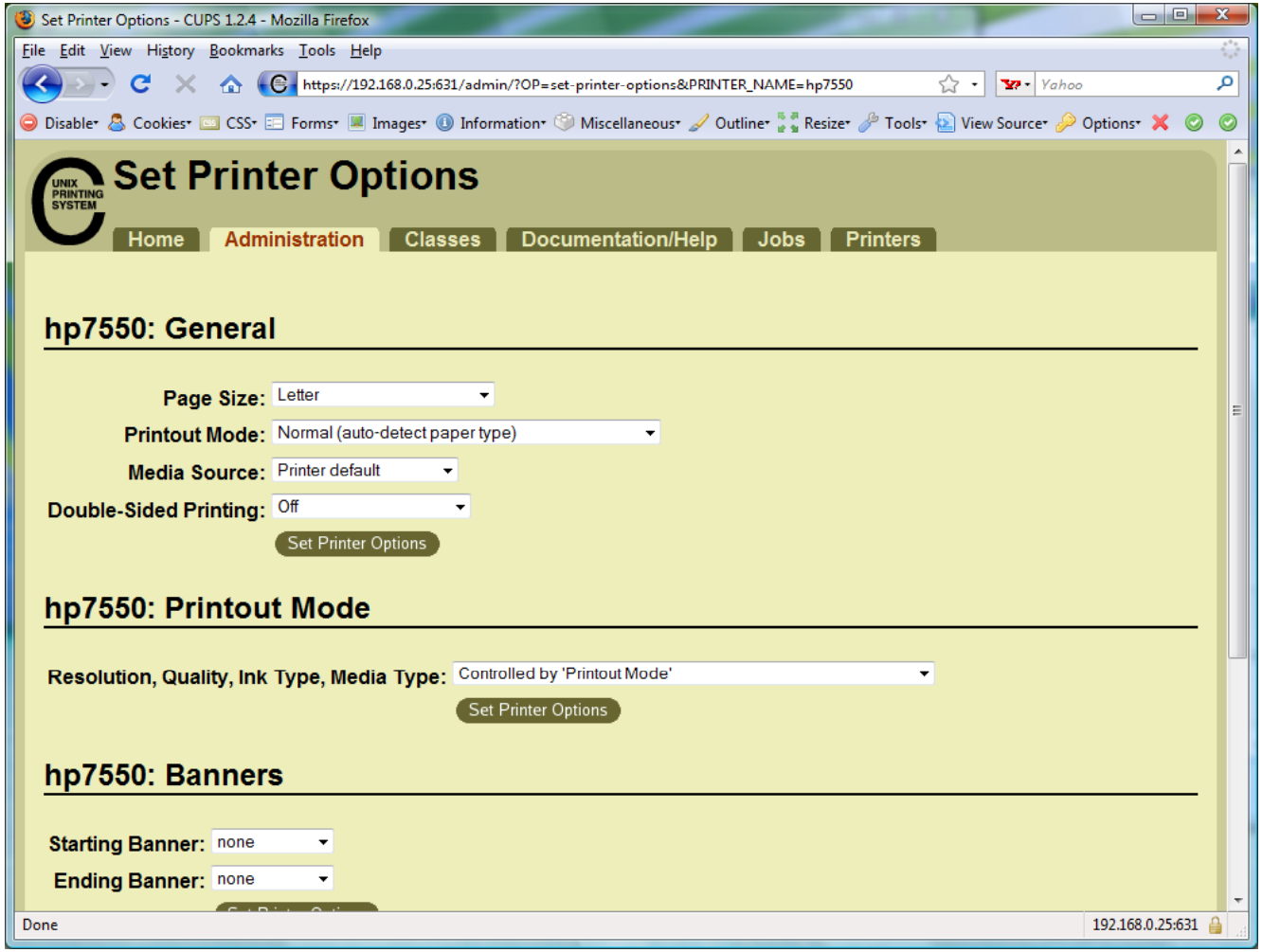
HP PhotoSmart 7550 Foomatic/hpijs (recommended) (en)

CUPS



Printer has been added

CUPS



View and set options as needed

Before using the printer we need to check that SAMBA is installed

Printing in Linux

CUPS

lpstat command

Show available printers

```
[root@benji ~]# lpstat -p -d
printer hp7550 is idle.  enabled since Fri 14 Nov 2008 05:01:28 PM PST
printer LaserJet is idle.  enabled since Fri 14 Nov 2008 12:23:27 PM PST
system default destination: hp7550
[root@benji ~]#
```

The –p option will show the available printers

The –d option will identify the default printer

CUPS

lp and lpr commands

Print some files

```
[root@benji ~]# lp -d hp7550 myfile
request id is hp7550-14 (1 file(s))

[root@benji ~]# lpr -P hp7550 myfile
[root@benji ~]#
```

*Either **lp** or **lpr** commands will print myfile to the selected printer*

Print output from a command or program

```
program | lp
program | lp -d printer

program | lpr
program | lpr -P printer
```

Note that both lp and lpr will read from stdin.

This allows output from another command to be piped in

CUPS

convert command

JPEG files need to be converted to postscript before printing with lp or lpr commands

```
[root@benji Desktop]# convert benji-500x420.jpg benji-500x420.ps
[root@benji Desktop]# lp benji-500x420.ps
request id is hp7550-29 (1 file(s))
[root@benji Desktop]# lpq
hp7550 is not ready
Rank    Owner   Job     File(s)                         Total Size
1st     root    28      benji-500x420.ps                1284096 bytes
2nd     root    29      benji-500x420.ps                1284096 bytes

[root@benji Desktop]# cancel 29

[root@benji Desktop]# cd /var/spool/cups/
[root@benji cups]# ls
0000001b  c00009  c00012  c00015  c00018  c00021  c00024  c00027  d00028-001
c00001    c00010  c00013  c00016  c00019  c00022  c00025  c00028  tmp
c00008    c00011  c00014  c00017  c00020  c00023  c00026  c00029
[root@benji cups]# ls tmp
```

*To get the **convert** command use:*
yum install ImageMagick

Configuring CUPS

CUPS

Set the default printer

```
[root@benji ~]# cat /etc/cups/printers.conf
# Printer configuration file for CUPS v1.2.4
# Written by cupsd on 2008-11-16 03:06
<DefaultPrinter hp7550>
Info hp photosmart 7550
Location Den
DeviceURI smb://hpdv2000/hp7550
State Idle
StateTime 1226791825
Accepting Yes
Shared Yes
JobSheets none none
QuotaPeriod 0
PageLimit 0
KLimit 0
OpPolicy default
ErrorPolicy stop-printer
</Printer>
```

Must restart CUPS after changing configuration file with:

service cups restart

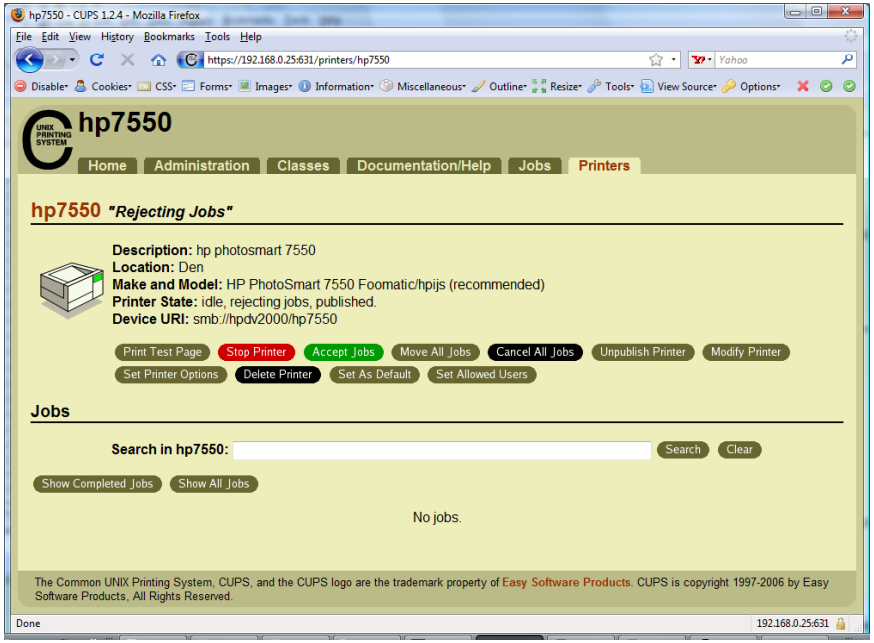

```
[root@benji ~]# lpoptions -d hp7550
job-sheets=none,none printer-info='hp photosmart 7550' printer-is-accepting-jobs=1 printer-is-shared=1
printer-make-and-model='HP PhotoSmart 7550 Foomatic/hpijs (recommended)' printer-state=3 printer-state-
change-time=1226778411 printer-state-reasons=none printer-type=36892
[root@benji ~]#
```

Three ways to set the default printer – edit configuration file, command or GUI

Managing Print Jobs

CUPS

Rejecting new print jobs



Clicking the Reject Jobs *button on the web based utility will reject further jobs*

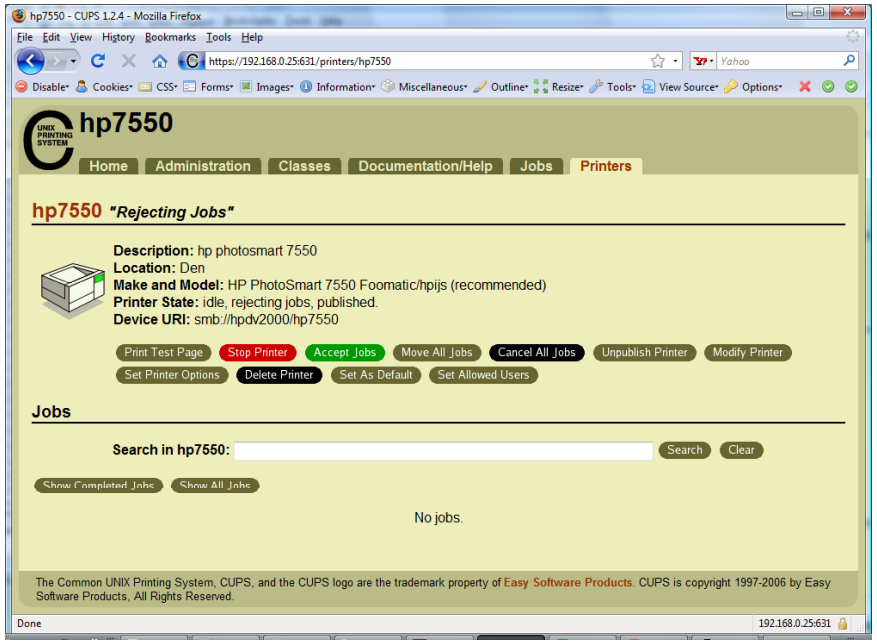
```
[root@benji ~]# lp myfile
lp: Destination "hp7550" is not accepting jobs.
[root@benji ~]#

[root@benji ~]# lpr myfile
lpr: Destination "hp7550" is not accepting jobs.
[root@benji ~]#
```

No more printing can be done now and jobs will not be spooled

CUPS

Stopping the printer

```
[root@benji ~]# lp myfile
request id is hp7550-22 (1 file(s))
[root@benji ~]# lpr myfile
[root@benji ~]# lp myfile
request id is hp7550-24 (1 file(s))
[root@benji ~]# lpr myfile

[root@benji ~]# lpq
hp7550 is not ready
Rank    Owner   Job     File(s)
Total Size
1st     root    22      myfile
1024 bytes
2nd     root    23      myfile
1024 bytes
3rd     root    24      myfile
1024 bytes
4th     root    25      myfile
1024 bytes
[root@benji ~]# lpstat
hp7550-22               root              1024    Sat 15
Nov 2008 12:20:23 PM PST
hp7550-23               root              1024    Sat 15
Nov 2008 12:20:28 PM PST
hp7550-24               root              1024    Sat 15
Nov 2008 12:20:31 PM PST
hp7550-25               root              1024    Sat 15
Nov 2008 12:20:34 PM PST
```



Clicking the Stop Printer *button on the web based utility will still allow jobs to be spooled*

CUPS

Showing jobs waiting to print

```
[root@benji ~]# lpq
hp7550 is not ready
Rank    Owner   Job     File(s)
Total Size
1st     root    22      myfile
1024 bytes
2nd     root    23      myfile
1024 bytes
3rd     root    24      myfile
1024 bytes
4th     root    25      myfile
1024 bytes
```

*Use **lpq** or **lpstat** to show spooled print jobs*

```
[root@benji ~]# lpstat
hp7550-22               root              1024   Sat 15
Nov 2008 12:20:23 PM PST
hp7550-23               root              1024   Sat 15
Nov 2008 12:20:28 PM PST
hp7550-24               root              1024   Sat 15
Nov 2008 12:20:31 PM PST
hp7550-25               root              1024   Sat 15
Nov 2008 12:20:34 PM PST
```

CUPS

Removing/canceling pending print jobs

```
[root@benji ~]# lpq
hp7550 is not ready
Rank    Owner   Job     File(s)
Total Size
1st     root    22      myfile
1024 bytes
2nd     root    23      myfile
1024 bytes
3rd     root    24      myfile
1024 bytes
4th     root    25      myfile
1024 bytes

[root@benji ~]# cancel 22
[root@benji ~]# cancel 23
[root@benji ~]# lprm 24
[root@benji ~]# lprm 25

[root@benji ~]# lpq
hp7550 is not ready
no entries

[root@benji ~]# lpstat
[root@benji ~]#
```

*Use **lpq** or **lpstat** to show the spooled print jobs*

*Use **cancel** or **lprm** to remove print jobs*

Spool Files

CUPS

Spool files in /var/spool/cups

When you print a new job, the response includes a number that can identify the spoolfile

```
[root@benji ~]# lp myfile
request id is hp7550-27 (1 file(s))

[root@benji ~]# ls /var/spool/cups/
0000001b  c00009  c00012  c00015  c00018  c00021  c00024  c00027
c00001    c00010  c00013  c00016  c00019  c00022  c00025  d00027-001
c00008    c00011  c00014  c00017  c00020  c00023  c00026  tmp
[root@benji ~]#

[root@benji ~]# file /var/spool/cups/*27*
/var/spool/cups/c00027:     PDP-11 UNIX/RT ldp
/var/spool/cups/d00027-001: ASCII English text
[root@benji ~]#
```

Spooled print files are kept in pairs, one control file and one data file, in /var/spool/cups

CUPS

Spool file contents in /var/spool/cups

Print job #27

```
[root@benji ~]# ls /var/spool/cups/
0000001b  c00009  c00012  c00015  c00018  c00021  c00024  c00027
c00001    c00010  c00013  c00016  c00019  c00022  c00025  d00027-001
c00008    c00011  c00014  c00017  c00020  c00023  c00026  tmp
[root@benji ~]#


[root@benji ~]# cat /var/spool/cups/d00027-001
Hello There,
  How is it going.  Ready for some salsa?  Benji is getting ready for
the big trip.
-Rich
-[root@benji ~]# xxd -l 128 /var/spool/cups/c00027
-0000000: 0101 0002 0000 0001 0147 0012 6174 7472  .........G..attr
-0000010: 6962 7574 6573 2d63 6861 7273 6574 0005  ibutes-charset..
-0000020: 7574 662d 3848 001b 6174 7472 6962 7574  utf-8H..attribut
-0000030: 6573 2d6e 6174 7572 616c 2d6c 616e 6775  es-natural-langu
-0000040: 6167 6500 0565 6e2d 7573 0245 000b 7072  age..en-us.E..pr
-0000050: 696e 7465 722d 7572 6900 1f69 7070 3a2f  inter-uri..ipp:/
-0000060: 2f6c 6f63 616c 686f 7374 2f70 7269 6e74  /localhost/print
-0000070: 6572 732f 6870 3735 3530 4200 196a 6f62  ers/hp7550B..job
-[root@benji ~]#
```

The data file is ascii and the control file is binary when printing a text file

printcap file

printcap file

The printcap file is used by applications that are hardcoded to look at the printcap file for available printers.

```
[root@benji ~]# cat /etc/printcap
# This file was automatically generated by cupsd(8) from the
# /etc/cups/printers.conf file.  All changes to this file
# will be lost.
LaserJet|HP LaserJet 1320 PCL 5e:rm=benji.localdomain:rp=LaserJet:
hp7550|hp photosmart 7550:rm=benji.localdomain:rp=hp7550:
nada|Not a real printer:rm=benji.localdomain:rp=nada:
[root@benji ~]#
```

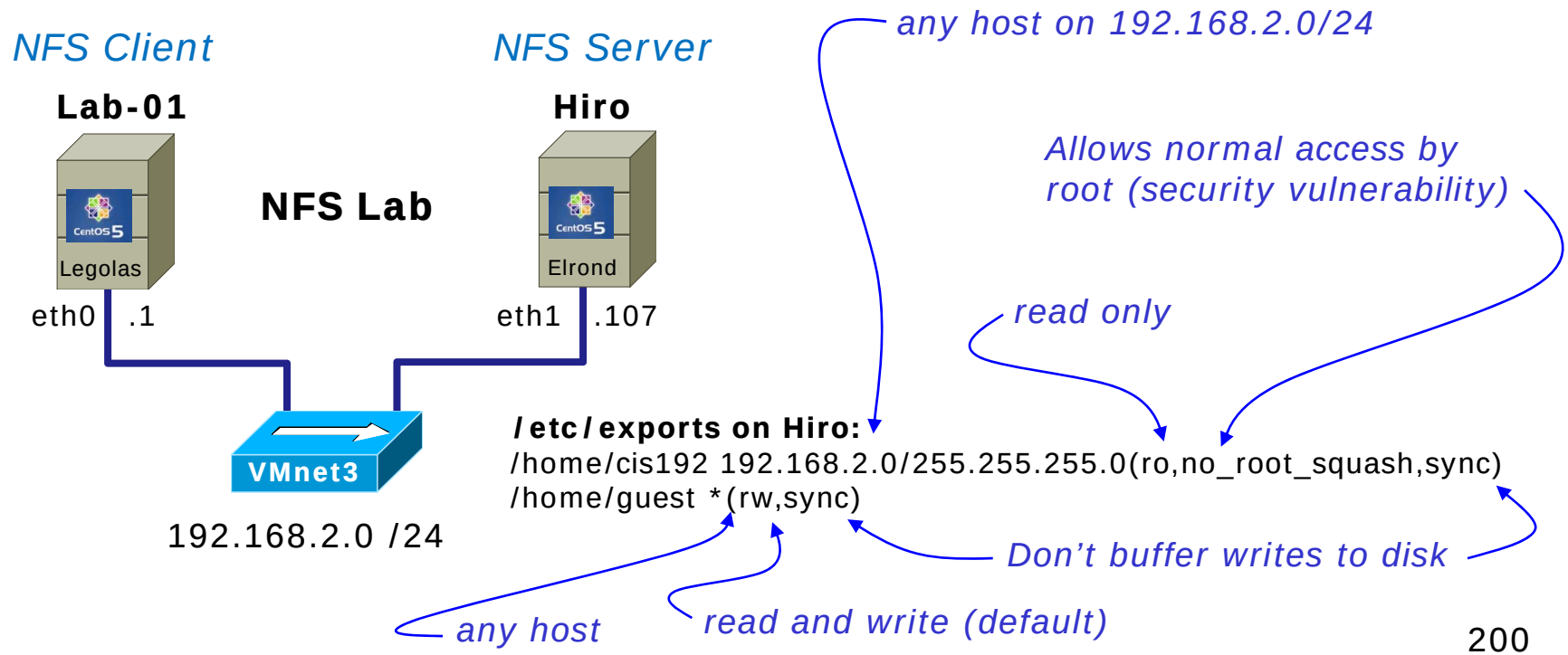
Exercise: CUPS

- Enable the CUPS service to run at system startup
- Turn on the CUPS service
- Add a "fake" HP LaserJet 1320N and disable it.
- Practice printing to your fake printer.

Lab X3

Lab X3 (NFS)

- Legolas and Elrond get new hostnames
- Export two /home directories on Hiro
- Mount Hiro's directories Lab-01



Wrap

References

RPC

- http://publib.boulder.ibm.com/infocenter/systems/index.jsp?topic=/com.ibm.aix.progcomm/doc/progcomc/rpc_portmap.htm

Port Mapper

- http://en.wikipedia.org/wiki/Portmap
- http://publib.boulder.ibm.com/infocenter/systems/index.jsp?topic=/com.ibm.aix.progcomm/doc/progcomc/rpc_portmap.htm

NFS

- http://www.cabrillo.edu/~jgriffin/CIS192/files/lesson11.html

- http://www.redhat.com/docs/manuals/enterprise/RHEL-3-Manual/ref-guide/s1-nfs-server-export.html
- http://blog.taragana.com/index.php/archive/full-disclosure-nis-security-hole-full-access-by-nis-client-root/
- http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/custom-guide/s1-nfs-mount.html
- http://linux.about.com/library/cmd/blcmdl8_rpc.statd.htm

LVM

- http://advait.wordpress.com/2008/09/23/logical-volume-manager-and-logical-volumes-linux/
- http://www.linuxconfig.org/Linux_lvm_-_Logical_Volume_Manager
- http://blog.timc3.com/2006/03/19/lvm2-and-adding-disks/

New commands, daemons:

- mount
- pvcreate, pvscan, pvdispaly
- lvcreate, lvscan, lvdisplay
- vgcreate, vgscan, vgdisplay

- rpcinfo
- netstat
- showmount
- exportfs

- lp or lpr
- convert
- lpstat
- lpq
- cancel or lprm
- lpoptions

Configuration files

- /etc/exports
- /etc/mtab
- /etc/fstab

- /var/spool/cups

Next Class

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

Lab X3 NFS is available now

Quiz questions for next class:

- To configure an NFS server, what file must be edited to specify the directories to be shared ?
- What is one way you might fix a “Stale NFS file handle” error?
- What URL would be used to browse to the local CUPS web-based configuration utility?

Backup

Classroom Static IP addresses for VM's

Station	IP	Static 1
Instructor	172.30.1.100	172.30.1.125
Station-01	172.30.1.101	172.30.1.126
Station-02	172.30.1.102	172.30.1.127
Station-03	172.30.1.103	172.30.1.128
Station-04	172.30.1.104	172.30.1.129
Station-05	172.30.1.105	172.30.1.130
Station-06	172.30.1.106	172.30.1.131
Station-07	172.30.1.107	172.30.1.132
Station-08	172.30.1.108	172.30.1.133
Station-09	172.30.1.109	172.30.1.134
Station-10	172.30.1.110	172.30.1.135
Station-11	172.30.1.111	172.30.1.136
Station-12	172.30.1.112	172.30.1.137

Station	IP	Static 1
Station-13	172.30.1.113	172.30.1.138
Station-14	172.30.1.114	172.30.1.139
Station-15	172.30.1.115	172.30.1.140
Station-16	172.30.1.116	172.30.1.141
Station-17	172.30.1.117	172.30.1.142
Station-18	172.30.1.118	172.30.1.143
Station-19	172.30.1.119	172.30.1.144
Station-20	172.30.1.120	172.30.1.145
Station-21	172.30.1.121	172.30.1.146
Station-22	172.30.1.122	172.30.1.147
Station-23	172.30.1.123	172.30.1.148
Station-24	172.30.1.124	172.30.1.149

Note the static IP address for your station to use in the next class exercise

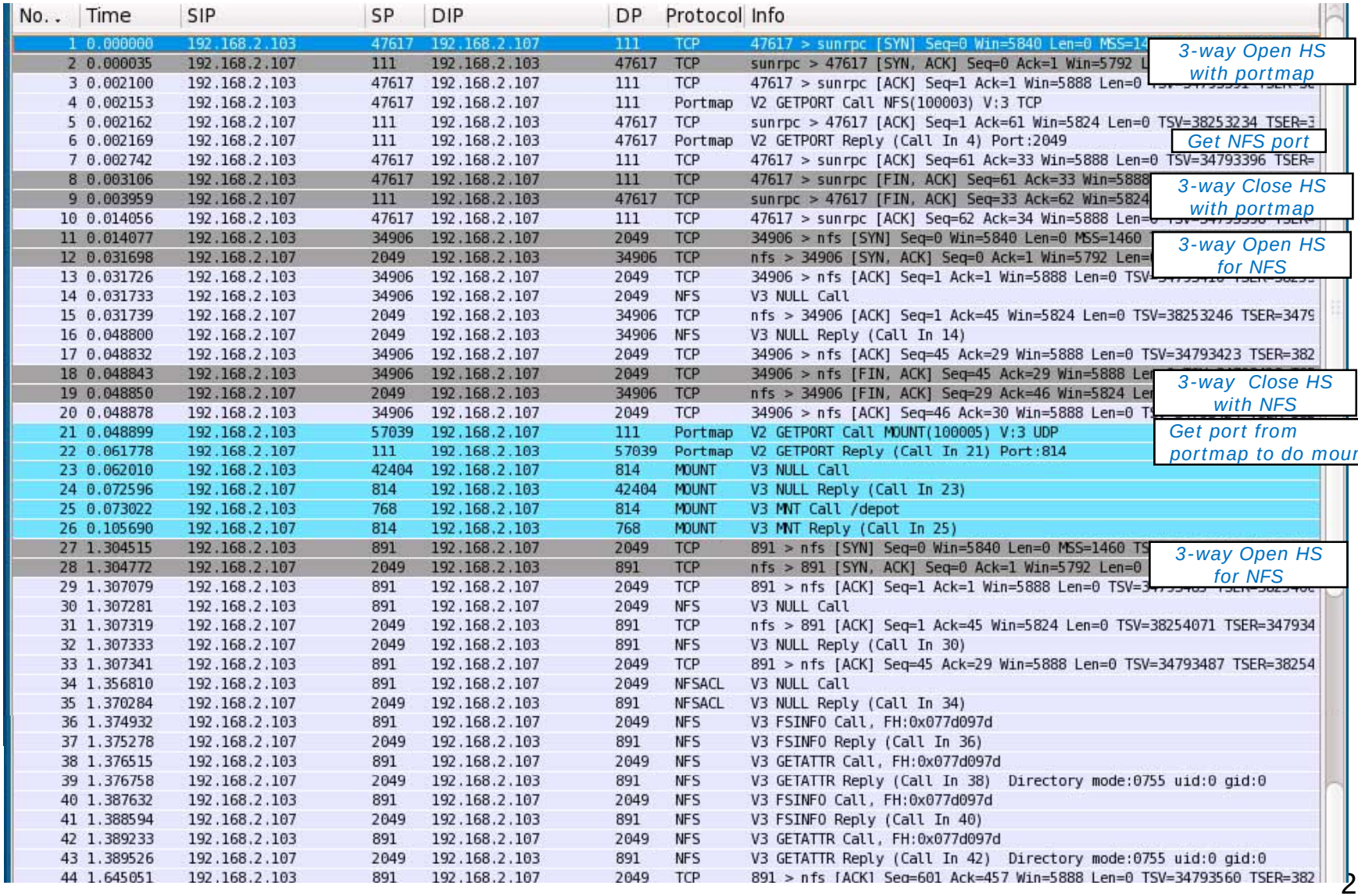
Classroom DHCP IP allocation pools table by station number

Station	IP	Start	End
01	172.30.1.101	172.30.1.50	172.30.1.54
02	172.30.1.102	172.30.1.55	172.30.1.59
03	172.30.1.103	172.30.1.60	172.30.1.64
04	172.30.1.104	172.30.1.65	172.30.1.69
05	172.30.1.105	172.30.1.70	172.30.1.74
06	172.30.1.106	172.30.1.75	172.30.1.79
07	172.30.1.107	172.30.1.80	172.30.1.84
08	172.30.1.108	172.30.1.85	172.30.1.89
09	172.30.1.109	172.30.1.90	172.30.1.94
10	172.30.1.110	172.30.1.95	172.30.1.99
11	172.30.1.111	172.30.1.200	172.30.1.204
12	172.30.1.112	172.30.1.205	172.30.1.209

Station	IP	Start	End
13	172.30.1.101	172.30.1.210	172.30.1.214
14	172.30.1.102	172.30.1.215	172.30.1.219
15	172.30.1.103	172.30.1.220	172.30.1.224
16	172.30.1.104	172.30.1.225	172.30.1.229
17	172.30.1.105	172.30.1.230	172.30.1.234
18	172.30.1.106	172.30.1.235	172.30.1.239
19	172.30.1.107	172.30.1.240	172.30.1.244
20	172.30.1.108	172.30.1.245	172.30.1.249
21	172.30.1.109	172.30.1.250	172.30.1.254
22	172.30.1.110	172.30.1.30	172.30.1.34
23	172.30.1.111	172.30.1.35	172.30.1.39
24	172.30.1.112	172.30.1.20	172.30.1.44
Instruct	172.30.1.100	172.30.1.45	172.30.1.49

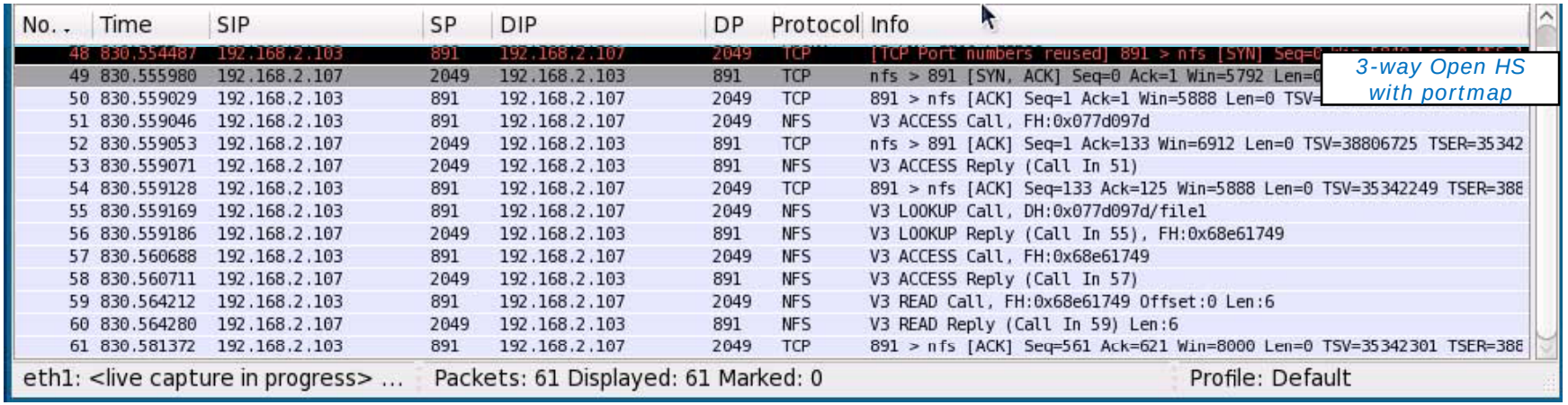
Use these pools of addresses based on your station number to avoid conflicts on the classroom network

```
[root@arwen ~]# mount hiro:/depot /depot
```



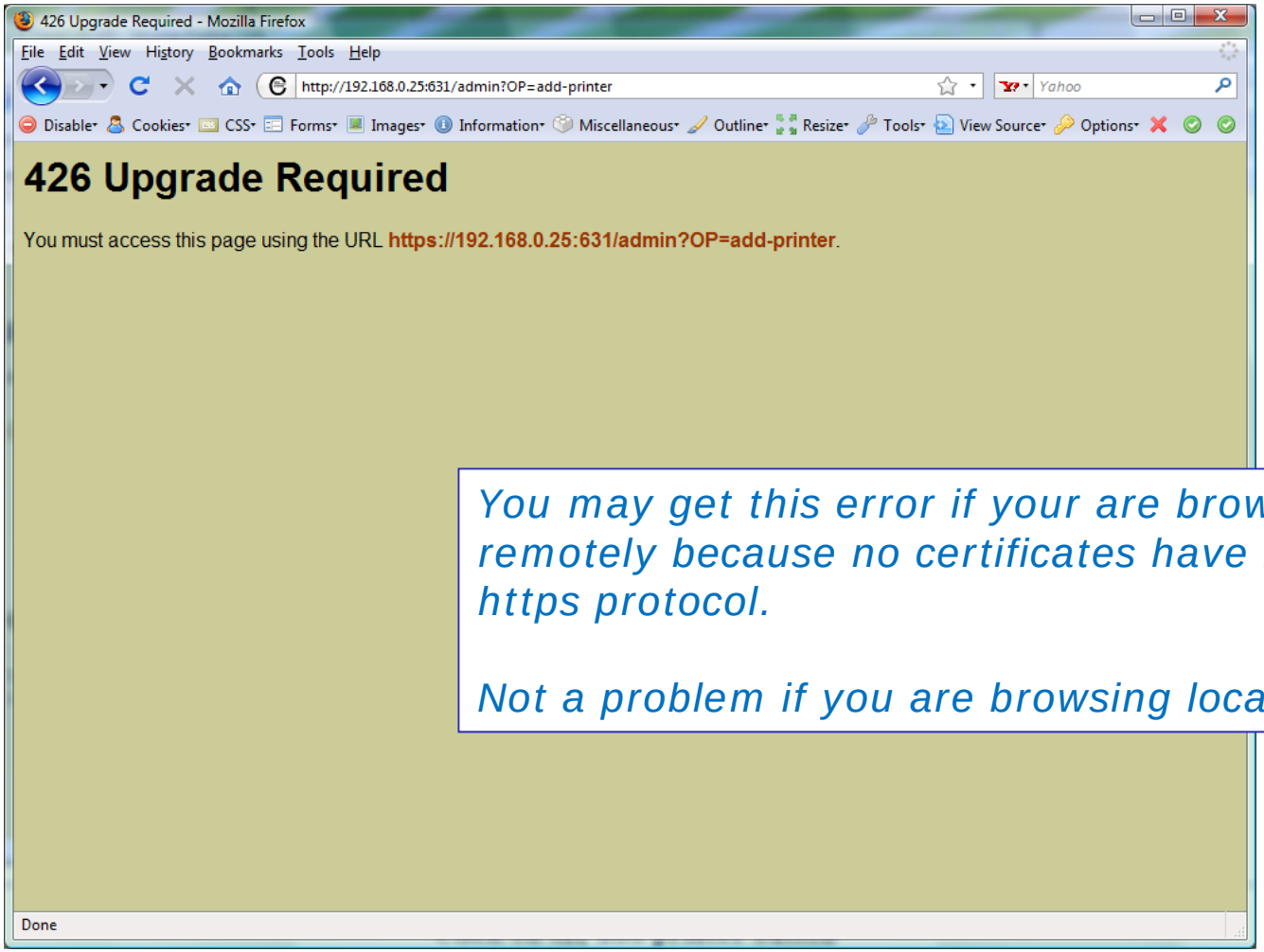
No.	Time	SIP	SP	DIP	DP	Protocol	Info
1	0.000000	192.168.2.103	47617	192.168.2.107	111	TCP	47617 > sunrpc [SYN] Seq=0 Win=5840 Len=0 MSS=14
2	0.000035	192.168.2.107	111	192.168.2.103	47617	TCP	sunrpc > 47617 [SYN, ACK] Seq=0 Ack=1 Win=5792 L
3	0.002100	192.168.2.103	47617	192.168.2.107	111	TCP	47617 > sunrpc [ACK] Seq=1 Ack=1 Win=5888 Len=0
4	0.002153	192.168.2.103	47617	192.168.2.107	111	Portmap	V2 GETPORT Call NFS(100003) V:3 TCP
5	0.002162	192.168.2.107	111	192.168.2.103	47617	TCP	sunrpc > 47617 [ACK] Seq=1 Ack=61 Win=5824 Len=0 TSV=38253234 TSER=3
6	0.002169	192.168.2.107	111	192.168.2.103	47617	Portmap	V2 GETPORT Reply (Call In 4) Port:2049
7	0.002742	192.168.2.103	47617	192.168.2.107	111	TCP	47617 > sunrpc [ACK] Seq=61 Ack=33 Win=5888 Len=0 TSV=34793396 TSER=
8	0.003106	192.168.2.103	47617	192.168.2.107	111	TCP	47617 > sunrpc [FIN, ACK] Seq=61 Ack=33 Win=5888
9	0.003959	192.168.2.107	111	192.168.2.103	47617	TCP	sunrpc > 47617 [FIN, ACK] Seq=33 Ack=62 Win=5824
10	0.014056	192.168.2.103	47617	192.168.2.107	111	TCP	47617 > sunrpc [ACK] Seq=62 Ack=34 Win=5888 Len=0
11	0.014077	192.168.2.103	34906	192.168.2.107	2049	TCP	34906 > nfs [SYN] Seq=0 Win=5840 Len=0 MSS=1460
12	0.031698	192.168.2.107	2049	192.168.2.103	34906	TCP	nfs > 34906 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=
13	0.031726	192.168.2.103	34906	192.168.2.107	2049	TCP	34906 > nfs [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSV
14	0.031733	192.168.2.103	34906	192.168.2.107	2049	NFS	V3 NULL Call
15	0.031739	192.168.2.107	2049	192.168.2.103	34906	TCP	nfs > 34906 [ACK] Seq=1 Ack=45 Win=5824 Len=0 TSV=38253246 TSER=3479
16	0.048800	192.168.2.107	2049	192.168.2.103	34906	NFS	V3 NULL Reply (Call In 14)
17	0.048832	192.168.2.103	34906	192.168.2.107	2049	TCP	34906 > nfs [ACK] Seq=45 Ack=29 Win=5888 Len=0 TSV=34793423 TSER=382
18	0.048843	192.168.2.103	34906	192.168.2.107	2049	TCP	34906 > nfs [FIN, ACK] Seq=45 Ack=29 Win=5888 Len
19	0.048850	192.168.2.107	2049	192.168.2.103	34906	TCP	nfs > 34906 [FIN, ACK] Seq=29 Ack=46 Win=5824 Len
20	0.048878	192.168.2.103	34906	192.168.2.107	2049	TCP	34906 > nfs [ACK] Seq=46 Ack=30 Win=5888 Len=0 TS
21	0.048899	192.168.2.103	57039	192.168.2.107	111	Portmap	V2 GETPORT Call MOUNT(100005) V:3 UDP
22	0.061778	192.168.2.107	111	192.168.2.103	57039	Portmap	V2 GETPORT Reply (Call In 21) Port:814
23	0.062010	192.168.2.103	42404	192.168.2.107	814	MOUNT	V3 NULL Call
24	0.072596	192.168.2.107	814	192.168.2.103	42404	MOUNT	V3 NULL Reply (Call In 23)
25	0.073022	192.168.2.103	768	192.168.2.107	814	MOUNT	V3 MNT Call /depot
26	0.105690	192.168.2.107	814	192.168.2.103	768	MOUNT	V3 MNT Reply (Call In 25)
27	1.304515	192.168.2.103	891	192.168.2.107	2049	TCP	891 > nfs [SYN] Seq=0 Win=5840 Len=0 MSS=1460 TS
28	1.304772	192.168.2.107	2049	192.168.2.103	891	TCP	nfs > 891 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0
29	1.307079	192.168.2.103	891	192.168.2.107	2049	TCP	891 > nfs [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSV=
30	1.307281	192.168.2.103	891	192.168.2.107	2049	NFS	V3 NULL Call
31	1.307319	192.168.2.107	2049	192.168.2.103	891	TCP	nfs > 891 [ACK] Seq=1 Ack=45 Win=5824 Len=0 TSV=38254071 TSER=347934
32	1.307333	192.168.2.107	2049	192.168.2.103	891	NFS	V3 NULL Reply (Call In 30)
33	1.307341	192.168.2.103	891	192.168.2.107	2049	TCP	891 > nfs [ACK] Seq=45 Ack=29 Win=5888 Len=0 TSV=34793487 TSER=38254
34	1.356810	192.168.2.103	891	192.168.2.107	2049	NFSACL	V3 NULL Call
35	1.370284	192.168.2.107	2049	192.168.2.103	891	NFSACL	V3 NULL Reply (Call In 34)
36	1.374932	192.168.2.103	891	192.168.2.107	2049	NFS	V3 FSINFO Call, FH:0x077d097d
37	1.375278	192.168.2.107	2049	192.168.2.103	891	NFS	V3 FSINFO Reply (Call In 36)
38	1.376515	192.168.2.103	891	192.168.2.107	2049	NFS	V3 GETATTR Call, FH:0x077d097d
39	1.376758	192.168.2.107	2049	192.168.2.103	891	NFS	V3 GETATTR Reply (Call In 38) Directory mode:0755 uid:0 gid:0
40	1.387632	192.168.2.103	891	192.168.2.107	2049	NFS	V3 FSINFO Call, FH:0x077d097d
41	1.388594	192.168.2.107	2049	192.168.2.103	891	NFS	V3 FSINFO Reply (Call In 40)
42	1.389233	192.168.2.103	891	192.168.2.107	2049	NFS	V3 GETATTR Call, FH:0x077d097d
43	1.389526	192.168.2.107	2049	192.168.2.103	891	NFS	V3 GETATTR Reply (Call In 42) Directory mode:0755 uid:0 gid:0
44	1.645051	192.168.2.103	891	192.168.2.107	2049	TCP	891 > nfs [ACK] Seq=601 Ack=457 Win=5888 Len=0 TSV=34793560 TSER=382

```
[root@arwen ~]# cat /depot/file1
file1
```



Client		Server	
IP:	*192.168.2.103*	IP:	*192.168.2.107*
Port:	*891*	Port:	*2049*

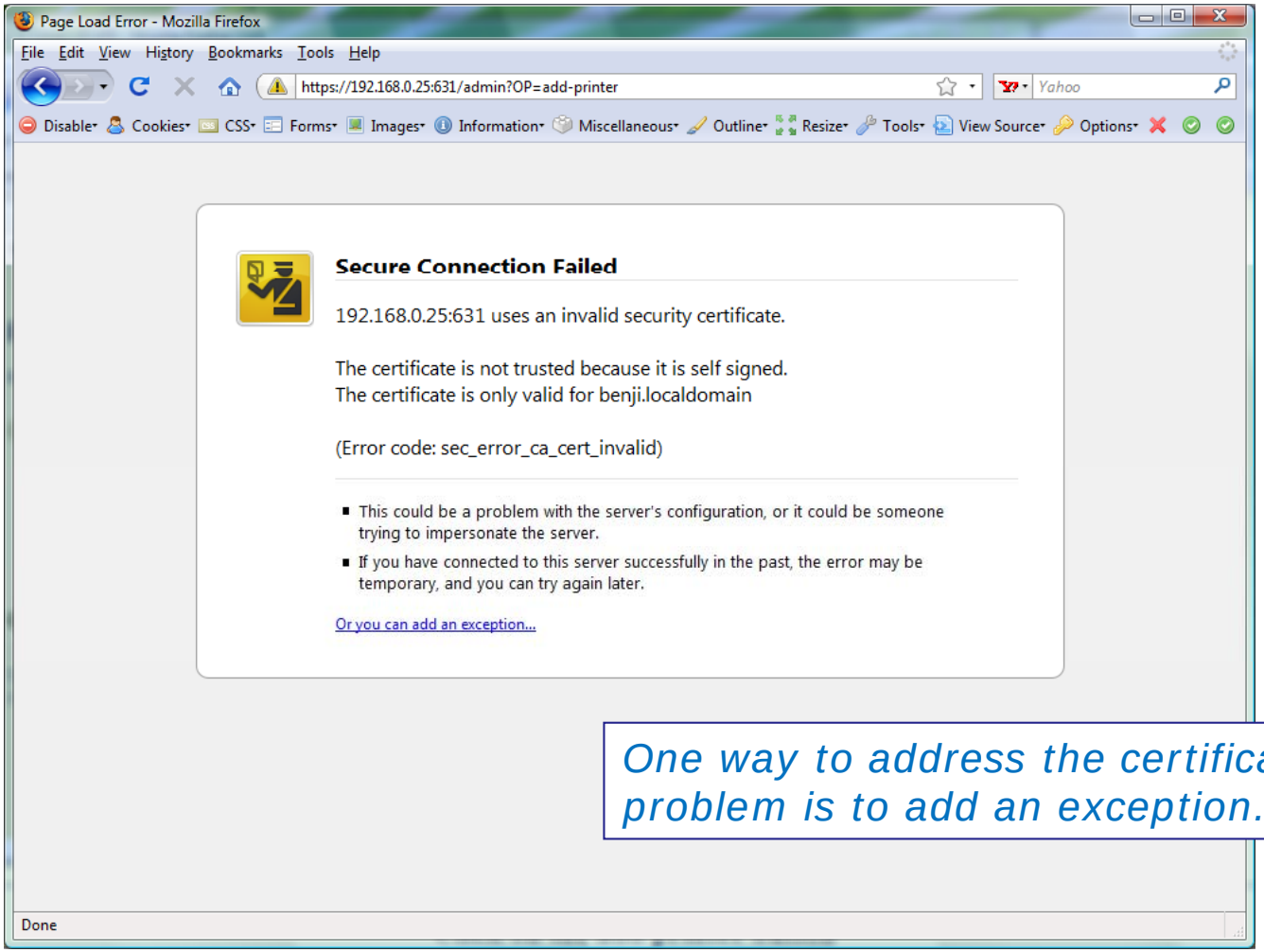
CUPS



You may get this error if your are browsing in remotely because no certificates have been set up for https protocol.

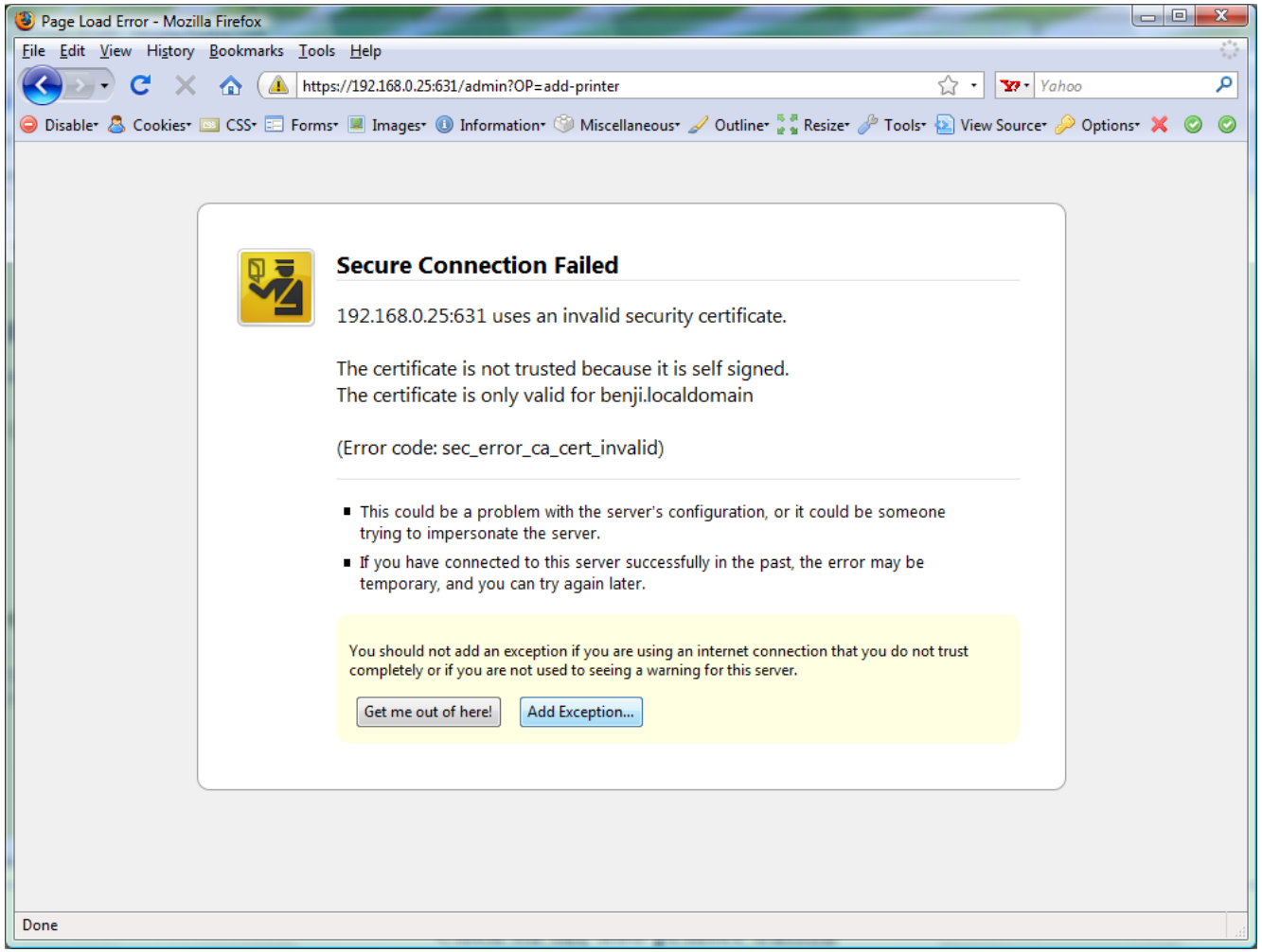
Not a problem if you are browsing locally

CUPS



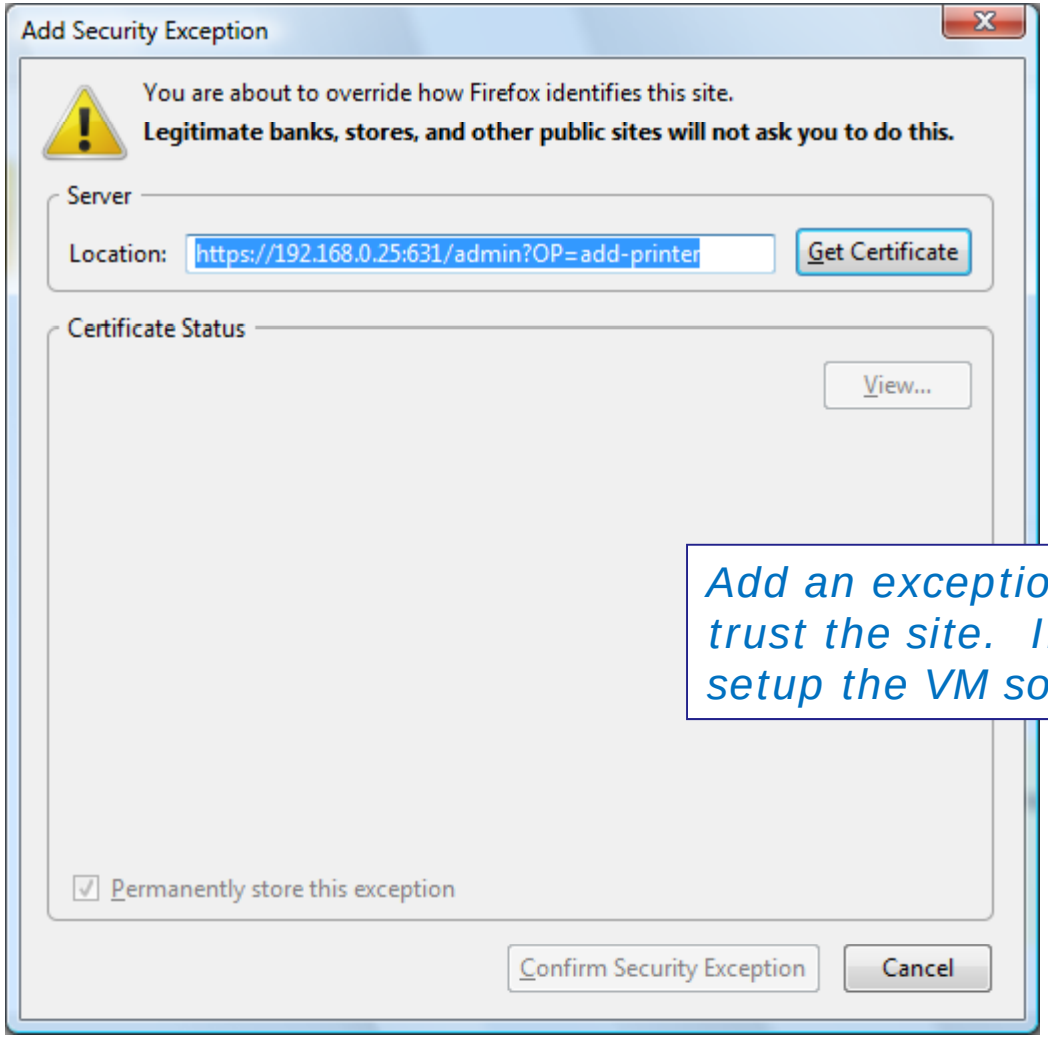
One way to address the certificate problem is to add an exception.

CUPS



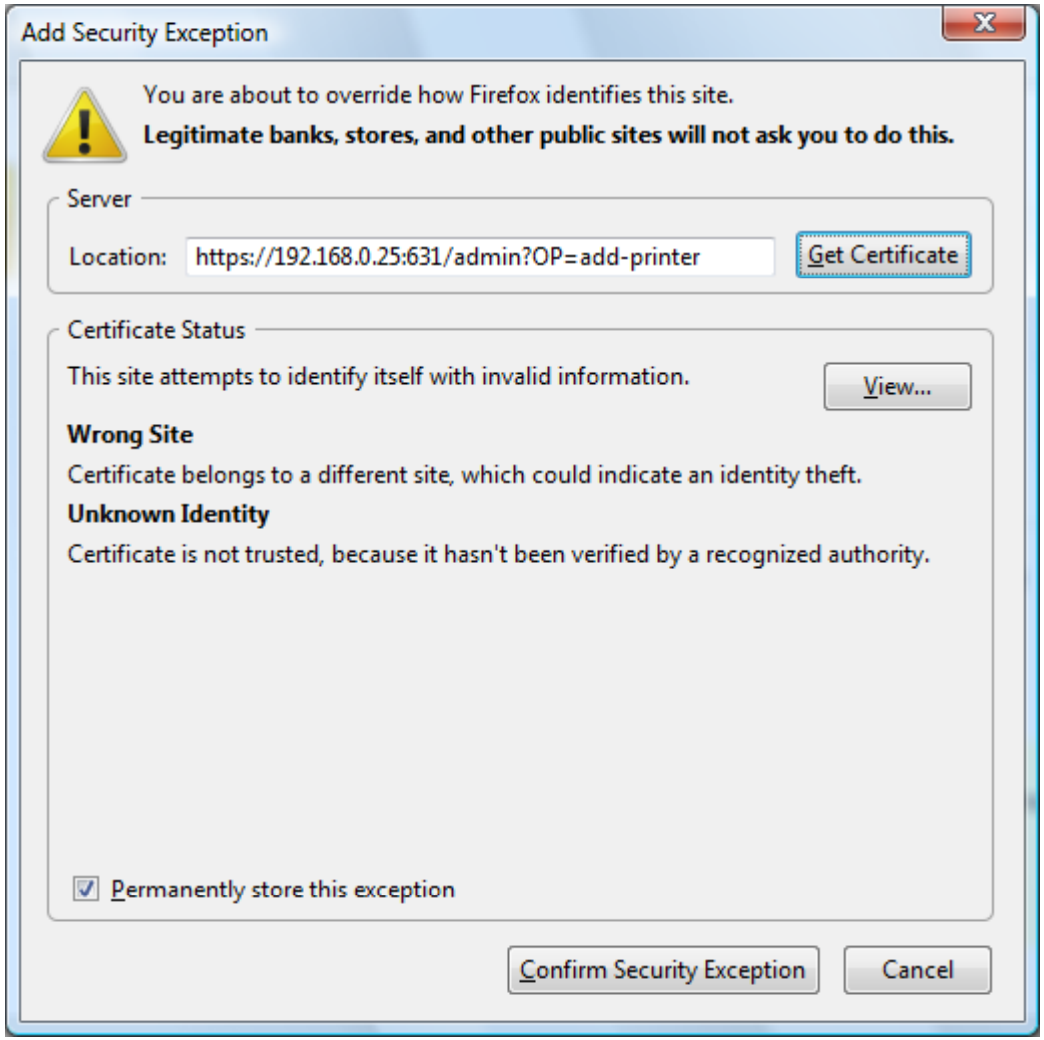
This adds the exception

CUPS



Add an exception only when you do trust the site. In this case we built and setup the VM so we trust it.

CUPS



Click Confirm Security Exception button