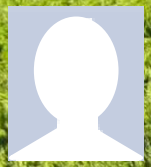
Lesson Module Checklist
- Slides –
- Properties -
- Flash cards –
- First minute quiz –
- Web calendar summary –
- Web book pages –
- Commands –
- Lab

- CCC Confer wall paper –
- labx1 and Project posted -

- Materials uploaded –
- Backup slides, CCC info, handouts on flash drive -
- Check that room headset is charged Aptos (backup) –

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


Sean C.

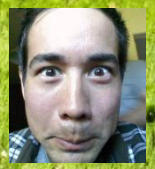

Don

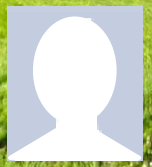

Carlile


Andrew


Sean Fa.


Carter


Sean Fy.


Dajan

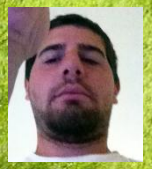

Bryn


Rita


Kelly


Ben


Ray


Michael

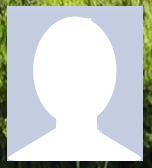

Evan


Josh


Carlos


Gustavo


Jessica


Evie

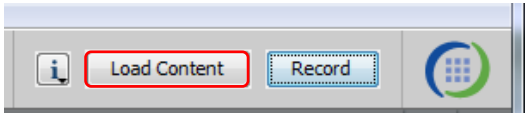

Jacob


Humberto

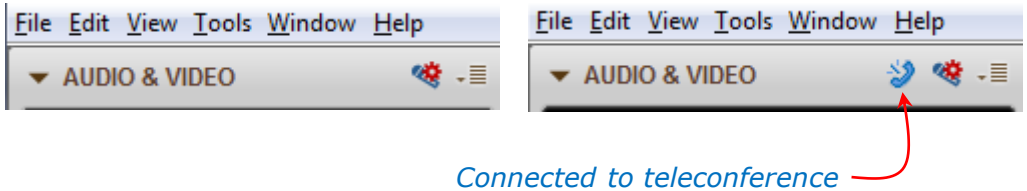

Chad

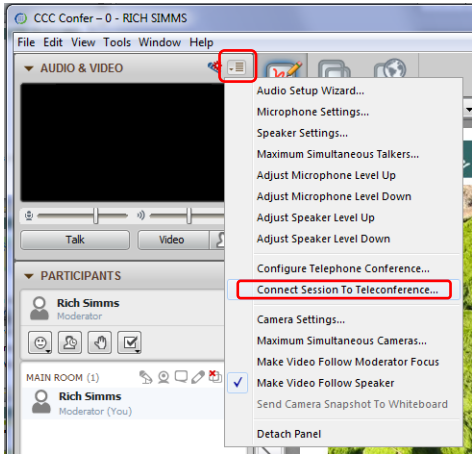
Email me (risimms@cabrillo.edu) a relatively current photo of your face for 3 points extra credit

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

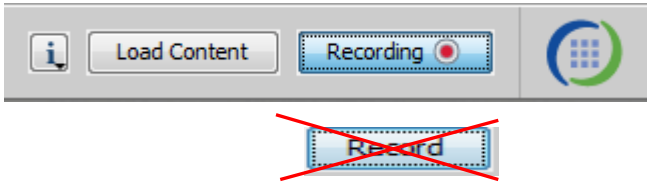


[ ] Connect session to Teleconference





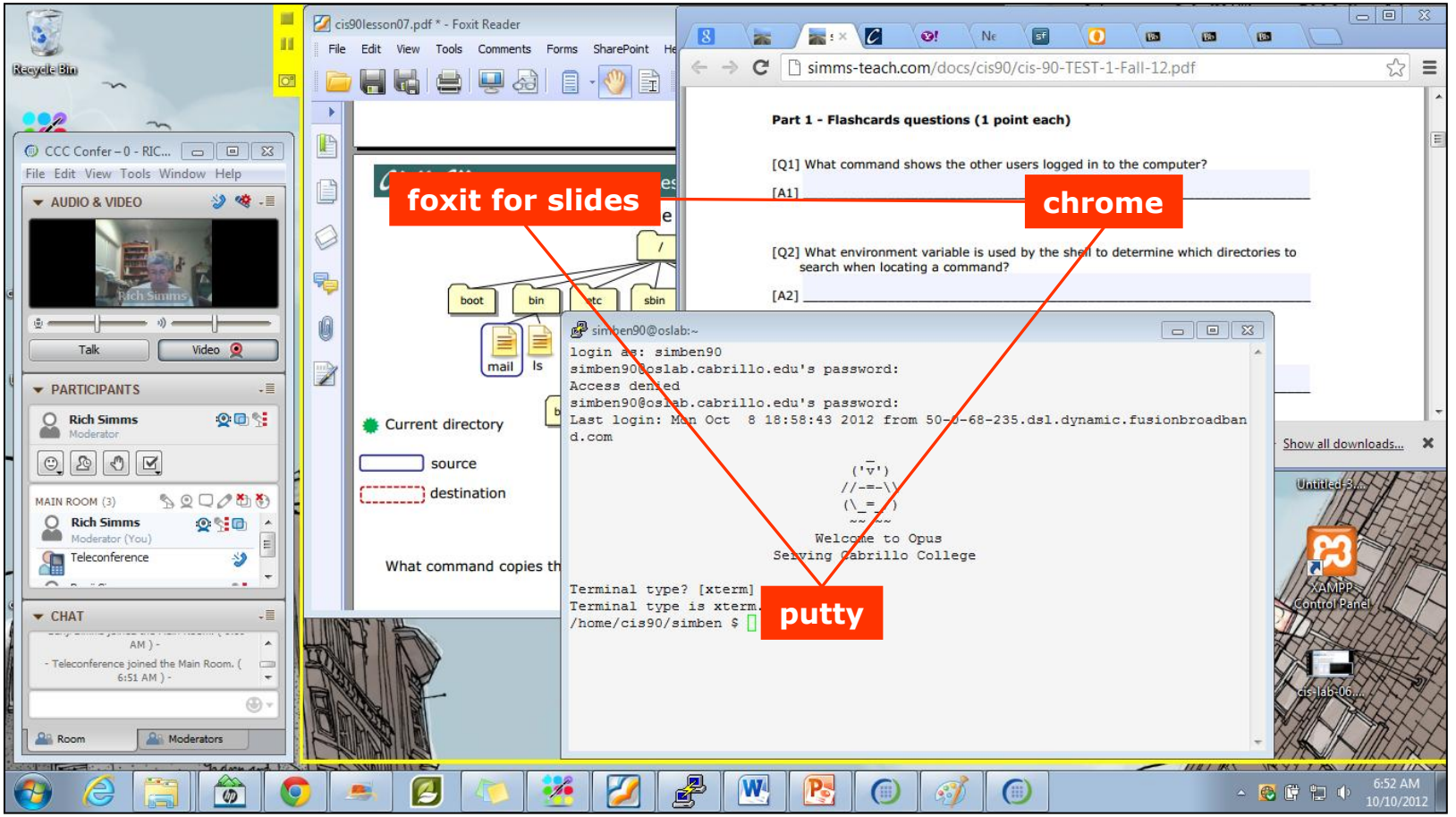
[ ] Is recording on?



[ ] Toggle Talk button to not use Mic

[ ] Video (webcam) optional

[ ] layout and share apps



Quiz

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

See electronic white board

email answers to: risimms@cabrillo.edu

(answers must be emailed within the first few minutes of class for credit)

Shell Scripting and Printing

Objectives	Agenda
• Be able to print, view the print queue and cancel print jobs	• Quiz • Housekeeping • Refresh • Shell scripting • Printing

Questions

Previous material and assignment

1. Previous material
2. Lab 10

Life without a path (surviving Lab 10)

The Path

- The path is a list of directories each containing commands, programs and scripts.
- The path is used by the shell to locate commands to run.
- The PATH variable defines the directories (separated by ":"s) and the search order.
- If the path is not defined then each command to run must be specified using it's absolute pathname

```
/home/cis90/simben $ echo $PATH
/usr/lib/qt-
3.3/bin:/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbi
n:/sbin:/home/cis90/simben/../bin:/home/cis90/simben/bin:.
/home/cis90/simben $
```

What is the fourth directory on this path?

Can you name a command found in the fourth directory of the path?

The Path

```
/home/cis90/simben $ type tty
tty is hashed (/usr/bin/tty)
```

The tty command is in the /usr/bin directory

```
/home/cis90/simben $ oldpath=$PATH
/home/cis90/simben $ unset PATH
```

Backup your current path

```
/home/cis90/simben $ tty
-bash: tty: No such file or directory
```

The tty command can no longer be run by typing just it's name

```
/home/cis90/simben $ /usr/bin/tty
/dev/pts/0
```

Instead the full absolute pathname must be used

```
/home/cis90/simben $ PATH=$oldpath
/home/cis90/simben $ tty
/dev/pts/0
```

Restore your path to what it was

Class Activity: Life without a path

Backup and remove your path variable:

```
/home/cis90/simben $ oldpath=$PATH

/home/cis90/simben $ unset PATH
/home/cis90/simben $ echo $PATH

/home/cis90/simben $
```

The Path

If the path is not defined then each command to run must be specified using an absolute pathname

```
/home/cis90/simben $ ls letter
-bash: ls: No such file or directory
```

```
/home/cis90/simben $ /bin/ls letter
letter
/home/cis90/simben $
```

The Path

Some commands still work without a path … why?

```
/home/cis90/simben $ echo "I want my path back"
I want my path back

/home/cis90/simben $ type echo
echo is a shell builtin

/home/cis90/simben $ type type
type is a shell builtin
```

The Path

Fixing the path, one directory at a time …

```
/home/cis90/simben $ ls letter
-bash: ls: No such file or directory
```

*The **ls** command is in /bin so lets put that on the path*

```
/home/cis90/simben $ PATH=/bin
/home/cis90/simben $ ls letter
letter
```

```
/home/cis90/simben $ stat letter
-bash: stat: command not found
```

*The **stat** command is in /usr/bin so lets append that directory too*

```
/home/cis90/simben $ PATH=$PATH:/usr/bin
/home/cis90/simben $ stat letter
  File: `letter'
  Size: 1059            Blocks: 16         IO Block: 4096
regular file
Device: fd00h/64768d    Inode: 102594      Links: 1
Access: (0644/-rw-r--r--)  Uid: ( 1000/simben90)   Gid: (
90/   cis90)
Access: 2012-04-30 15:43:28.000000000 -0700
Modify: 2012-03-20 10:31:30.000000000 -0700
Change: 2012-04-30 07:34:30.000000000 -0700
```

The Path

```
/home/cis90/simben $ allscripts
-bash: allscripts: command not found
```

```
/home/cis90/simben $ PATH=$PATH:/home/cis90/bin
/home/cis90/simben $ allscripts
```

*The **allscripts** shell script is in /home/cis90/bin so lets add that directory to the path as well*

```
**************************************************************
*                Fall 2012 CIS 90 Online Projects            *
**************************************************************
1) Andrew
2) Ben
3) Benji
4) Bryn
5) Carlile
6) Carlos
  <snipped>
21) Ray
22) Rita
23) Sean C.
24) Sean F.
25) Shahram

99) Exit

Enter Your Choice:
```

The Path

```
/home/cis90/simben $ datecal
bash: datecal: command not found
```

*The **datecal** shell script is in your own bin directory so lets add that to the path as well*

```
/home/cis90/simben $ PATH=$PATH:/home/cis90/simben/bin
/home/cis90/simben $ datecal

Tue May  8 14:30:59 PDT 2012

     April 2012              May 2012               June 2012
Su Mo Tu We Th Fr Sa    Su Mo Tu We Th Fr Sa    Su Mo Tu We Th Fr Sa
 1  2  3  4  5  6  7                1  2  3  4  5                1  2
 8  9 10 11 12 13 14     6  7  8  9 10 11 12     3  4  5  6  7  8  9
15 16 17 18 19 20 21    13 14 15 16 17 18 19    10 11 12 13 14 15 16
22 23 24 25 26 27 28    20 21 22 23 24 25 26    17 18 19 20 21 22 23
29 30                   27 28 29 30 31          24 25 26 27 28 29 30


/home/cis90/simben $
```

The Path

```
/home/cis90/simben $ dogbone
-bash: dogbone: command not found
```

*The **dogbone** shell script is in the current directory but not on the path*

```
/home/cis90/simben $ ./dogbone
What is your name? Benji
What is your favorite bone? Chicken
Hi Benji, your favorite bone is Chicken
```

How can I run a script in the current directory without having to put a ./ in front of it?

The Path

Easy … add "here" or "." to the path

```
/home/cis90/simben $ dogbone
-bash: dogbone: command not found
```

```
/home/cis90/simben $ PATH=$PATH:.
/home/cis90/simben $ dogbone
What is your name? Benji
What is your favorite bone? Chicken
Hi Benji, your favorite bone is Chicken
```

The Path

Rebuilding the path by appending directories one at a time

```
/home/cis90/simben $ unset PATH
/home/cis90/simben $ echo $PATH

/home/cis90/simben $ PATH=/bin
/home/cis90/simben $ echo $PATH
/bin
```

Start with /bin which has all the essential UNIX/Linux commands

```
/home/cis90/simben $ PATH=$PATH:/usr/bin
/home/cis90/simben $ echo $PATH
/bin:/usr/bin
```

Append /usr/bin which has hundreds of additional UNIX/Linux commands

```
/home/cis90/simben $ PATH=$PATH:/home/cis90/bin
/home/cis90/simben $ echo $PATH
/bin:/usr/bin:/home/cis90/bin
```

Append the CIS 90 class bin directory

```
/home/cis90/simben $ PATH=$PATH:/home/cis90/simben/bin
/home/cis90/simben $ echo $PATH
/bin:/usr/bin:/home/cis90/bin:/home/cis90/simben/bin
```

Append your own student bin directory

```
/home/cis90/simben $ PATH=$PATH:.
/home/cis90/simben $ echo $PATH
/bin:/usr/bin:/home/cis90/bin:/home/cis90/simben/bin:.
```

Append the current directory

CIS 90 class bin directory

Student bin directory

Current directory

The Path

Making the path permanent using .bash_profile

```
/home/cis90/simben $ cat .bash_profile
# .bash_profile

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

# User specific environment and startup programs

PATH=$PATH:/home/cis90/bin:$HOME/bin:.
BASH_ENV=$HOME/.bashrc
USERNAME=""
PS1='$PWD $ '
export USERNAME BASH_ENV PATH
umask 002
set -o ignoreeof
stty susp
eval `tset -s -m vt100:vt100 -m :\?${TERM:-ansi} -r -Q `

/home/cis90/simben $
```

This customizes the normal path by appending the class bin directory, the student's bin directory and the "current" directory

Housekeeping

Previous material and assignment

1. Lab 10 due 11:59PM tonight

2. The Extra Credit Labs X1 and X2 (30 points each) area available.

3. The Final Project is available.

Sage advice:

Get one "practice" task script working in your project today before you leave class today.

Review the final project grading rubric to see how many points you have completed so far with your practice script.

Cabrillo College Student Senate Holiday Food Drive

Second Harvest Food Bank A Plate For Kate

by **Carter Frost** » Wed Nov 14, 2012 4:28 pm

Carter Frost

Posts: 20
Joined: Wed Aug 29, 2012 2:29 pm

Second Harvest Food Bank A Plate For Kate
http://thefoodbank.org/plateforkate
The Cabrillo College Student Senate is doing a fundraiser for Second Harvest Food Bank

There are many families in need of food this coming holiday season. Whether you donate food or money, it will go towards making sure families don't go hungry! Every $1 donated will feed a family of four.

Cabrillo's Goal: 20,000 Pounds of Food
Food barrels and green piggies will be located throughout campus to collect donations for Second Harvest Holiday Food Drive.

I have placed a green piggy in your classroom.

Methods of donation:
1. Place non-perishable food in the barrels around campus.
Canned products and any non-perishable food items are accepted.
Can't find a barrel? Come to SAC East and there is a barrel located in the lobby.
All food collected will be distributed to both the Cabrillo Food Pantry and the Second Harvest Food Bank.

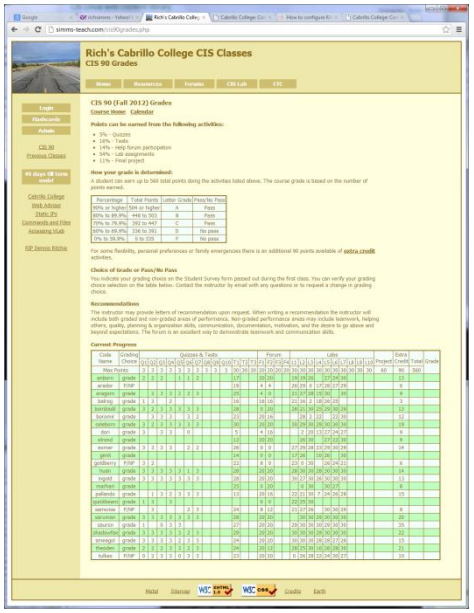
2. Feed the Green Piggies located around campus and in your classroom with your spare change. (They also don't mind a helping of a few dollars as well!)

3. Make a Monetary Donation
Make checks payable to the Second Harvest Food Bank you can drop it in a green pig or off at SAC East.

Note to instructor: The Green Pig Is Lurking In "The 2501 Closet"

Managing your grade

Use the web page



http://simms-teach.com/cis90grades.php

Use Jesse's checkgrades script

```
anborn: 72% (302 of 417 points)
arador: 58% (244 of 417 points)
aragorn: 71% (298 of 417 points)
balrog: 52% (220 of 417 points)
bombadil: 92% (384 of 417 points)
boromir: 69% (288 of 417 points)
celeborn: 104% (436 of 417 points)
dori: 56% (234 of 417 points)
elrond: 68% (287 of 417 points)
eomer: 82% (344 of 417 points)
gimli: 29% (125 of 417 points)
goldberry: 64% (267 of 417 points)
huan: 105% (439 of 417 points)
ingold: 97% (405 of 417 points)
marhari: 61% (255 of 417 points)
pallando: 75% (315 of 417 points)
samwise: 73% (306 of 417 points)
saruman: 97% (406 of 417 points)
sauron: 103% (431 of 417 points)
shadowfax: 104% (434 of 417 points)
smeagol: 95% (399 of 417 points)
theoden: 93% (388 of 417 points)
tulkas: 82% (346 of 417 points)
```

As of November 18, 2012

Managing your grade

Percentage	Total Points	Letter Grade	Pass/No Pass
90% or higher	504 or higher	A	Pass
80% to 89.9%	448 to 503	B	Pass
70% to 79.9%	392 to 447	C	Pass
60% to 69.9%	336 to 391	D	No pass
0% to 59.9%	0 to 335	F	No pass

Points gone by
- 9 quizzes - 27 points
- 2 tests - 60 points
- 3 forum periods - 60 points
- 9 labs - 270 points

417 points

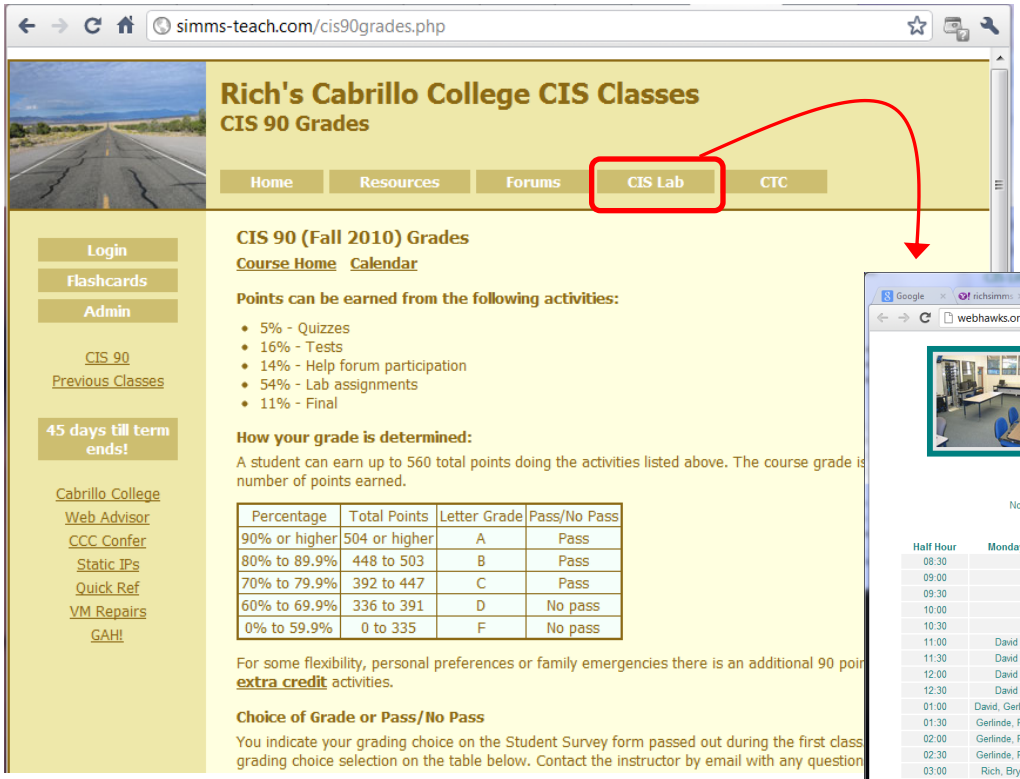
Points yet to earn
- 1 quizzes - 3 points
- 1 test - 30 points
- 1 forum periods - 20 points
- 1 labs - 30 points
- 1 final project - 60 points

143 points

- Plus extra credit - up to 90 points

Managing your grade

Getting extra help for CIS 90

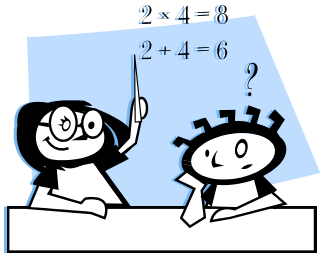


Come by the lab and get help from instructors and student assistants

Managing your grade

Getting extra help for CIS 90

- Rich's Office Hours Wed 4:20-5:10pm in Room 2501 (right after class) or TBA (contact me)
- Ask questions on the Forum at:
  http://opus.cabrillo.edu/forum/



Final Exam

Can **not** be taken online using CCC Confer

It will be held in room 2501 on Wednesday, Dec 12th from 1:00 to 3:50PM (hard stop, no extension time period)

If you know you can't make this date you will need to contact the instructor, in advance, to arrange an exam **EARLIER** in the week.

No makeups after the Wednesday exam

Practice test will be available

	12/12	**Test #3 (the final exam)** **Time** • 1:00PM - 3:50PM in Room 2501 **Materials** • Presentation slides (download) • Test (download)		5 posts Lab X1 Lab X2

Spring 2013 Linux Classes

CIS 90 Introduction to UNIX/Linux

Provides a technical overview of the UNIX/Linux operating system, including hands-on experience with commands, files, and tools. Recommended Preparation: CS 1L or CIS 172.

Transfer Credit: CSU.

Section	Days	Times	Units	Instructor	Room
78467	TH	01:15PM-04:20PM	3.00	R.Simms	OL

Section 78467 is an ONLINE course. Meets weekly throughout the semester online at the scheduled times by remote technology using CCC Confer. For details, see instructor's web page at go.cabrillo.edu/online.

CIS 192AB UNIX/Linux Network Administration

Build and monitor network infrastructures, and install, configure, and protect services on Linux TCP/IP networks. Prerequisite: CIS 81 and CIS 90. Recommended Preparation: CIS 191AB. Repeatability: May be taken 2 times.

Section	Days	Times	Units	Instructor	Room
79995	T	05:30PM-09:35PM	4.00	R.Simms	OL
&	Arr.	Arr.		R.Simms	OL

Section 79995 is an ONLINE course. Meets weekly throughout the semester online during the scheduled times by remote technology using CCC Confer with an additional 4 hr 5 min online lab per week. Students will be required to show that they meet the course prerequisites. For details, see instructor's web page at go.cabrillo.edu/online.

- Dear Sir or Madam:
  >
  > TWENTIETH CENTURY FOX FILM CORPORATION and its affiliated companies
  > (collectively, "FOX") are the exclusive owners of copyrights in
  > motion pictures.
  >
  > It has come to our attention that California State University Network
  > is the service provider for the IP address listed below, from which
  > unauthorized copying and distribution (downloading, uploading, file
  > serving, file "swapping" or other similar activities) of FOX'S
  > property is taking place. The documentation included at the end of
  > this notice specifies the location of the infringement. We believe
  > that the Internet access of the user engaging in this infringement
  > is provided by California State University Network or a downstream
  > service provider who purchases this connectivity from California
  > State University Network.

DO NOT USE THE SCHOOL NETWORK FOR PIRATING INTELLECTUAL PROPERTY THAT BELONGS TO OTHERS!

- > Evidentiary Information:
  > Notice ID: 22264275815
  > Asset: DARKEST HOUR (2011), THE
  > Protocol: BitTorrent
  > IP Address: 207.62.184.250
  > DNS: pat-cis-students.cabrillo.edu
  > Port ID: 3116
  > File Name: The.Darkest.Hour.2011.720p.BRRip.x264.AC3-26K
  > File Size: 1791679401
  > Timestamp: 2012-11-07 22:33:59.057 GMT Last Seen Date: 2012-11-07
  > 22:33:59.057 GMT

Final Project Grading Rubric

Grading rubric (60 points maximum)

Possible Points	Requirements
30	Implementing all five tasks (6 points each): • Requirements for each task: – Minimum of 10 "original" script command lines – Has comments to explain what it does – Has user interaction
25	You don't have to do all of these but do at least five: • Redirecting stdin (5 points) • Redirecting stdout (5 points) • Redirecting stderr (5 points) • Use of permissions (5 points) • Use of filename expansion characters (5 points) • Use of absolute path (5 points) • Use of relative path (5 points) • Use of a PID (5 points) • Use of inodes (5 points) • Use of links (5 points) • Use of a GID or group (5 points) • Use of a UID or user (5 points) • Use of a signal (5 points) • Use of piping (5 points) • Use of an environment variable (5 points) • Use of /bin/mail (5 points) • Use of a conditional (5 points) The maximum for this section are 25 points.
5	Present your script in front of the class
Points lost	
-15	Fails to run from **allscripts**
-15	Other students in the class are unable to read and execute your script.
-15	Error messages are displayed when running one or more tasks
-up to 90	No credit for any task which contains unoriginal script code that: • Doesn't give full credit to the original author • Doesn't indicate where the code was obtained from • Doesn't include licensing terms • Violates copyright or licensing terms
Extra credit	
30	Up to three additional tasks (10 points each)

Final Project forum

Use the forum effectively to get scripting help

Not so good …

Preview:
Help!

My script is getting weird error

- Homer

Not enough information has been provided on this post for others to help

Use the forum effectively to get scripting help

Better … but requires viewer to log into Opus and you may have modified the script since posting

Preview:

Help!

My script is getting weird error

My script is here:
/home/cis90/milhom/bin/myscript

And this is the error:

CODE: SELECT ALL

```
/home/cis90/simben/bin $ ./script99
simben90
-rwxr-x--- 1 simben90 cis90 10489 Apr 30 07:33 /home/cis90/simben/bin/myscript
./script99: line 8: unexpected EOF while looking for matching `"'
./script99: line 16: syntax error: unexpected end of file
/home/cis90/simben/bin $
```

- Homer

This post provides the location of the script and the error message which enables others to help you find and fix the problem

Use the forum effectively to get scripting help

B | i | u | Quote | Code | List | List= | [*] | Img | URL | Flash | Normal | Font colour

Preview:

Help!

My script is getting weird error

This is the script:

CODE: SELECT ALL

```
#!/bin/bash
# Test script
#
echo $LOGNAME
dir=/home/cis90/simben
ls -l $dir/bin/myscript
if [ -f "$dir/bin/myscript ]; then
  echo you have a myscript file in the bin directory
else
  echo there is no myscript file in your bin directory!]
fi
exit
```

And this is the error:

CODE: SELECT ALL

```
/home/cis90/simben/bin $ ./script99
simben90
-rwxr-x--- 1 simben90 cis90 10489 Apr 30 07:33 /home/cis90/simben/bin/myscript
./script99: line 8: unexpected EOF while looking for matching `"'
./script99: line 16: syntax error: unexpected end of file
/home/cis90/simben/bin $
```

- Homer

Best …

This post shows both the script and the error using code tags which enables others to help you find and fix the problem.

The thread will also benefit future CIS 90 students

Scripting Tips

echo

Silence is golden

Many UNIX commands that run successfully produce no output

```
[simben90@opus bin]$ alias details=file
[simben90@opus bin]$ cp quiet quiet.bak
[simben90@opus bin]$ value=002
[simben90@opus bin]$ umask $value
[simben90@opus bin]$ cat quiet > /dev/null
[simben90@opus bin]$ > important_file
```

Silence is golden

Running or sourcing a script full of UNIX commands that produce no output …. still produces no output!

```
[simben90@opus bin]$ cat quiet
alias details=file
cp quiet quiet.bak
value=002
umask $value
cat quiet > /dev/null
> important_file
```

```
[simben90@opus bin]$ quiet
[simben90@opus bin]$
```

```
[simben90@opus bin]$ source quiet
[simben90@opus bin]$
```

Silence is golden

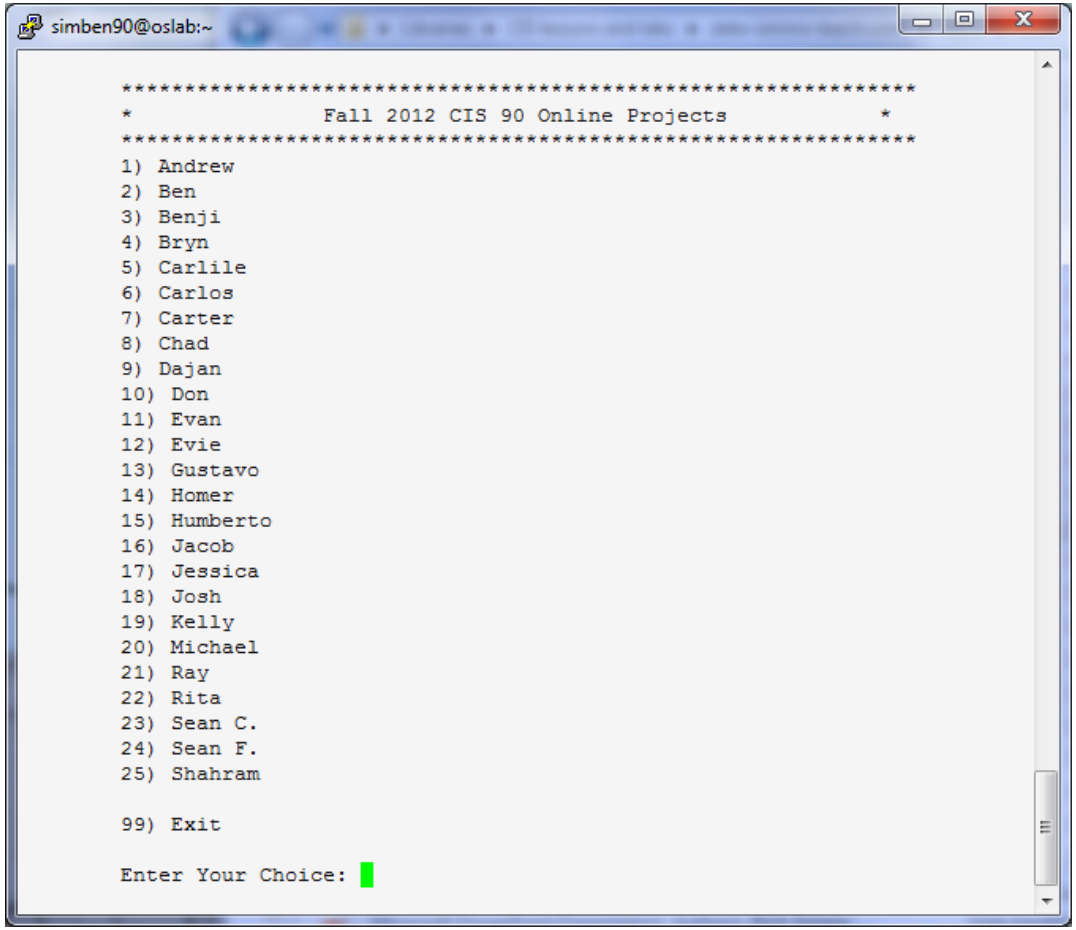
*Shell **script developers use the echo command** to provide interaction, feedback and tracing (for debugging) with the scripts they write.*

```
[simben90@opus bin]$ cat quiet
alias details=file
cp quiet quiet.bak
value=002
umask $value
echo TRACE: value=$value
cat quiet > /dev/null
echo "Quiet script successfully completed"

[simben90@opus bin]$ quiet
TRACE: value=002
Quiet script successfully completed
```

Final Project permissions

Final Project



Before leaving class today you want to make sure you can run your script from ***allscripts***

Permissions

A past forum post …

Ha Ha Class

by [redacted] on Tue May 12, 2009 12:22 pm

I'm sure this is some kind of payback for last week "Hacking" attempt

I will find out who did this

~~~~~~

ps. Im going to pass

Uh, oh … someone got hacked!
~~~~~~

Permissions

rsimms@oslab:/home/cis90/bin

```
[rsimms@oslab bin]$ ls -l /home/cis90/*/bin/myscript
-rwxr-xr-x. 1 calsea90 cis90   560 Nov 14 14:25 /home/cis90/calsea/bin/myscript
-rwxrwxr-x. 1 davdon90 cis90   510 Nov 14 14:28 /home/cis90/davdon/bin/myscript
-rwxrwxr-x. 1 evaand90 cis90   518 Nov 14 14:25 /home/cis90/evaand/bin/myscript
-rwxrwxr-x. 1 farsha90 cis90   551 Nov 14 14:25 /home/cis90/farsha/bin/myscript
-rwxrwxr-x. 1 frocar90 cis90   557 Nov 14 14:23 /home/cis90/frocar/bin/myscript
-rwxrwxr-x. 1 hendaj90 cis90   752 Nov 17 12:07 /home/cis90/hendaj/bin/myscript
-rwxrwxr-x. 1 kanbry90 cis90   510 Nov 14 14:24 /home/cis90/kanbry/bin/myscript
-rw-rw-r--. 1 kenrit90 cis90   481 Nov 14 22:35 /home/cis90/kenrit/bin/myscript
-rwxrwxr-x. 1 libkel90 cis90   587 Nov 14 14:21 /home/cis90/libkel/bin/myscript
-rwxrwxrwx. 1 lyoben90 cis90   508 Nov 17 11:33 /home/cis90/lyoben/bin/myscript
-rwxr-xr-x. 1 marray90 cis90  1711 Nov 15 00:33 /home/cis90/marray/bin/myscript
-rwxr-x--x. 1 milhom90 cis90   615 Nov 14 14:23 /home/cis90/milhom/bin/myscript
-rwxrwxr-x. 1 noreva90 cis90   551 Nov 14 14:21 /home/cis90/noreva/bin/myscript
-rwxr-x---. 1 simben90 cis90 10489 Nov 10 15:23 /home/cis90/simben/bin/myscript
-rwxrwxr-x. 1 verevi90 cis90   594 Nov 14 14:36 /home/cis90/verevi/bin/myscript
[rsimms@oslab bin]$
```

Which myscript files can only be edited by their owner? Which ones could be edited by anyone in the CIS 90 class? Which ones could be edited by anyone on Opus?

Overall

rsimms@oslab:/home/cis90/bin

```
[rsimms@oslab bin]$ ./checkmyscripts
Sean ==> -rwxr-xr-x. 1 calsea90 cis90 560 Nov 14 14:25 calsea/bin/myscript
Don ==> -rwxrwxr-x. 1 davdon90 cis90 510 Nov 14 14:28 davdon/bin/myscript
Carlile ==> ls: cannot access ellcar/bin/myscript: No such file or directory
Andrew ==> -rwxrwxr-x. 1 evaand90 cis90 518 Nov 14 14:25 evaand/bin/myscript
Shahram ==> -rwxrwxr-x. 1 farsha90 cis90 551 Nov 14 14:25 farsha/bin/myscript
Carter ==> -rwxrwxr-x. 1 frocar90 cis90 557 Nov 14 14:23 frocar/bin/myscript
Sean ==> ls: cannot access fyosea/bin/myscript: No such file or directory
Dajan ==> -rwxrwxr-x. 1 hendaj90 cis90 752 Nov 17 12:07 hendaj/bin/myscript
Bryn ==> -rwxrwxr-x. 1 kanbry90 cis90 510 Nov 14 14:24 kanbry/bin/myscript
Rita ==> -rw-rw-r--. 1 kenrit90 cis90 481 Nov 14 22:35 kenrit/bin/myscript
Kelly ==> -rwxrwxr-x. 1 libkel90 cis90 587 Nov 14 14:21 libkel/bin/myscript
Ben ==> -rwxrwxrwx. 1 lyoben90 cis90 508 Nov 17 11:33 lyoben/bin/myscript
Ray ==> -rwxr-xr-x. 1 marray90 cis90 1711 Nov 15 00:33 marray/bin/myscript
Chad ==> ls: cannot access mescha/bin/myscript: No such file or directory
Michael ==> ls: cannot access mesmic/bin/myscript: No such file or directory
Homer ==> -rwxr-x--x. 1 milhom90 cis90 615 Nov 14 14:23 milhom/bin/myscript
Evan ==> -rwxrwxr-x. 1 noreva90 cis90 551 Nov 14 14:21 noreva/bin/myscript
Josh ==> ls: cannot access potjos/bin/myscript: No such file or directory
Carlos ==> ls: cannot access ramcar/bin/myscript: Permission denied
Gustavo ==> ls: cannot access ramgus/bin/myscript: No such file or directory
Jessica ==> ls: cannot access rawjes/bin/myscript: No such file or directory
Duke ==> ls: cannot access rodduk/bin/myscript: No such file or directory
Benji ==> -rwxr-x---. 1 simben90 cis90 10489 Nov 10 15:23 simben/bin/myscript
Evie ==> -rwxrwxr-x. 1 verevi90 cis90 594 Nov 14 14:36 verevi/bin/myscript
Jacob ==> ls: cannot access wiljac/bin/myscript: No such file or directory
Humberto ==> ls: cannot access zamhum/bin/myscript: Permission denied
[rsimms@oslab bin]$
```

Which myscript files cannot by run by the class?

Class Activity

Note: One of the requirements for the final project is setting permissions on your script so that all cis90 members can run it.

To meet this requirement use:

cd bin
chmod 750 myscript
ls –l myscript

umask

again!

Permissions

Why can other classmates write to my scripts?

Before Lab 10

```
/home/cis90/rodduk/bin $ umask
0002
/home/cis90/rodduk/bin $ rm newscript; touch newscript
/home/cis90/rodduk/bin $ ls -l newscript
-rw-rw-r-- 1 rodduk cis90 0 Nov 23 16:17 newscript
/home/cis90/rodduk/bin $ chmod +x newscript
/home/cis90/rodduk/bin $ ls -l newscript
-rwxrwxr-x 1 rodduk cis90 0 Nov 23 16:17 newscript
```

After Lab 10

```
/home/cis90ol/simmsben $ umask
0006
/home/cis90ol/simmsben $ rm newscript; touch newscript
/home/cis90ol/simmsben $ ls -l newscript
-rw-rw---- 1 simmsben cis90ol 0 May 12 08:44 newscript
/home/cis90ol/simmsben $ chmod +x newscript
/home/cis90ol/simmsben $ ls -l newscript
-rwxrwx--x 1 simmsben cis90ol 0 May 12 08:44 newscript
```

Because your umask setting gives group members write permission on any new files you create!

Permissions

```
[rodduk90@opus bin]$ cat /home/cis90/rodduk/.bash_profile
# .bash_profile

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

# User specific environment and startup programs

PATH=$PATH:$HOME/../bin:$HOME/bin:.
BASH_ENV=$HOME/.bashrc
USERNAME=""
PS1='$PWD $ '
export USERNAME BASH_ENV PATH
umask 002
set -o ignoreeof
stty susp
eval `tset -s -m vt100:vt100 -m :\?${TERM:-ansi} -r -Q `
```

Note your umask is defined in .bash_profile which runs every time you login. In lab 10 your change this setting to 006.

Class Activity

- Change your umask to 026
- Can group or other users modify your new files now?
- Try it, **touch** a new file and check the permissions with **ls -l**
- How would you make this a permanent umask setting?

Scripting Tips

date

Utilizing the date command

The date command prints the current date and time

```
/home/cis90/simben $ date
Tue Nov 20 15:54:13 PST 2012
```

The **$** metacharacter provides the "value" of both variables, e.g. $PS1 or commands, e.g. $(***command***):

```
/home/cis90/simben $ echo $PS1
$PWD $

/home/cis90/simben $ echo $(grep love poems/Shakespeare/* | wc -l)
11

/home/cis90/simben $ myname=$(grep $LOGNAME /etc/passwd | cut -f5 -d":")
/home/cis90/simben $ echo $myname
Benji Simms
```

Utilizing the date command

```
/home/cis90/simben $ date
Wed Nov 26 15:35:53 PST 2008


/home/cis90/simben $ date +'%r'
04:14:26 PM
/home/cis90/simben $ time=$(date +'%r')
/home/cis90/simben $ echo "At the tone the time will be $time"
At the tone the time will be 04:15:02 PM


/home/cis90/simben $ date +'%A'
Tuesday
/home/cis90/simben $ day=$(date +'%A')
/home/cis90/simben $ echo "Today is $day"
Today is Tuesday
```

See the man page on date for lots of other % codes

Class Activity

Your turn, make a script by adding the following two lines to a file named *mydate* using the vi editor:

```
echo "Hola $LOGNAME"
echo Today is $(date +'%m/%d/%Y')
```

Give the script execute permissions and run it:

```
/home/cis90/simben $ chmod +x mydate
/home/cis90/simben $ mydate
Hola simben90
Today is 11/20/2012
```

tips on script names

Don't name your scripts "script"

```
[simben90@opus bin]$ ls -l script
-rwxr-x--- 1 simben90 cis90 47 Nov 23 16:44 script
```

```
[simben90@opus bin]$ cat script
echo "Hello from the script file named script"
```

What would happen if your ran the script above?

Don't name your scripts "script"

```
[simben90@opus bin]$ cat script
echo "Hello from the script file named script"

[simben90@opus bin]$ script
Script started, file is typescript
```

Why the heck doesn't my script do what it's supposed to do?

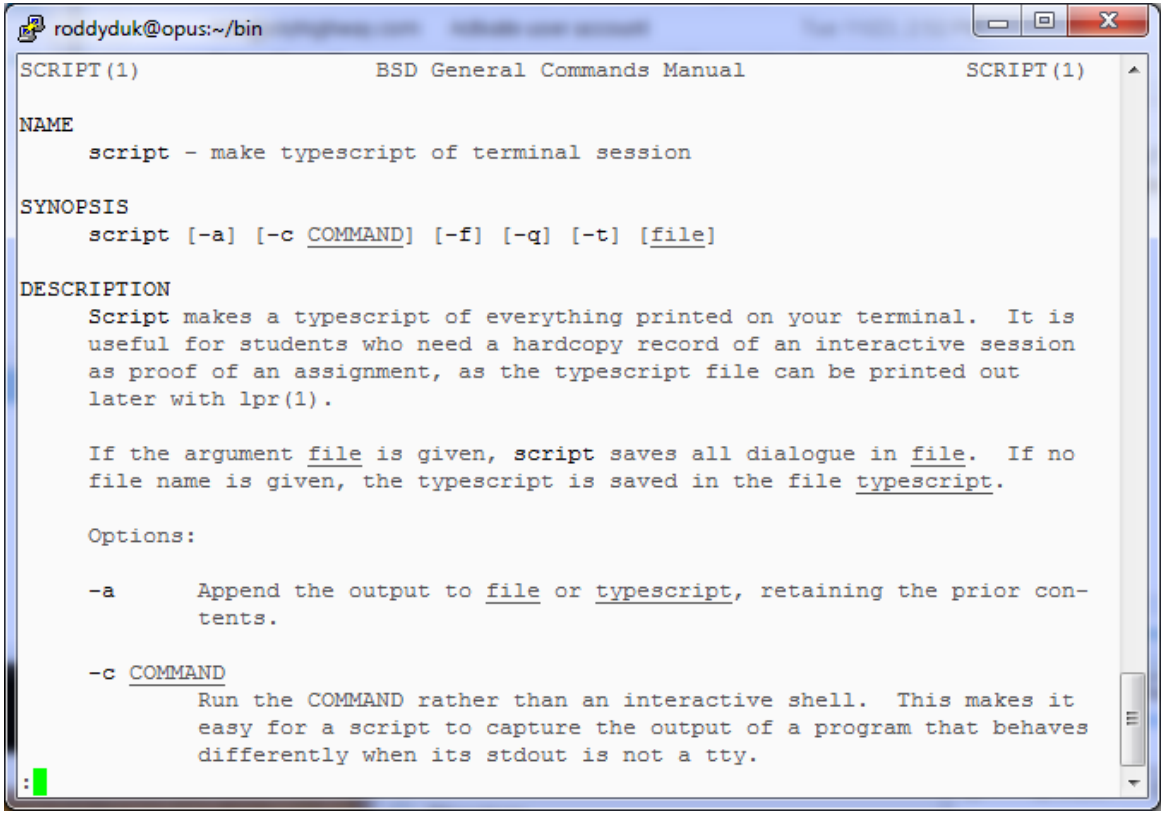
```
[simben90@opus bin]$ Where is my script?
bash: Where: command not found
[simben90@opus bin]$ exit
Script done, file is typescript
[simben90@opus bin]$ cat typescript
Script started on Wed 13 May 2009 08:00:02 AM PDT
[simben90@opus bin]$ Where is my script?
bash: Where: command not found
[simben90@opus bin]$ exit

Script done on Wed 13 May 2009 08:00:47 AM PDT
[simben90@opus bin]$
```

Don't name your scripts "script"

Why doesn't script do what it is supposed to do? … because script is the name of an existing UNIX command!

```
[simben90@opus bin]$ man script
[simben90@opus bin]$
```



Don't name your scripts "script"

There are (at least) two files named script on Opus

```
[simben90@opus bin]$ type script
script is hashed (/usr/bin/script)
[simben90@opus bin]$ file /usr/bin/script
/usr/bin/script: ELF 32-bit LSB executable, Intel 80386, version 1
(SYSV), for GNU/Linux 2.6.9, dynamically linked (uses shared libs),
for GNU/Linux 2.6.9, stripped
```

```
[simben90@opus bin]$ type /home/cis90/simben/bin/script
/home/cis90/simben/bin/script is /home/cis90/simben/bin/script
[simben90@opus bin]$ file /home/cis90/simben/bin/script
/home/cis90/simben/bin/script: ASCII text
[simben90@opus bin]$
```

***Question**: Why did bash run the script in /usr/bin instead of the script in /home/cis90/simben/bin?*

Don't name your scripts "script"

***Question**: Why did bash run the script in /usr/bin instead of the script in /home/cis90/simben/bin?*

*The Linux **script** command is in this directory*

```
[simben90@opus bin]$ echo $PATH
/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:/home/cis90/bin:
/home/cis90/simben/bin:.
```

*Our script, named **script**, is in this directory*

***Answer**: bash searches the path in the order the directories are listed. It finds the script command in /user/bin first.*

Don't name your scripts "script"

To override the PATH you can always specify an absolute pathname to the file you want to run:

```
[simben90@opus bin]$ /home/cis90/simben/bin/script
Hello from the script file named script

[simben90@opus bin]$ ./script
Hello from the script file named script
```

Note the shell treats the . above as "here" which in this case is /home/cis90/simben/bin

Try the script command

- Use the **script** command to start recording
- Type various commands of your choice
- Type **exit** or hit **Ctrl-D** to end recording
- Use **cat typescript** to see what you recorded

This would be a good way to record a session such as working one of the lab assignments for future reference.

Review

```
function runningScript ()
{
```

The rules of the road for variables

- Rule 1: A child process can only see variables the parent has exported.
- Rule 2: A child process cannot change the parent's variables.

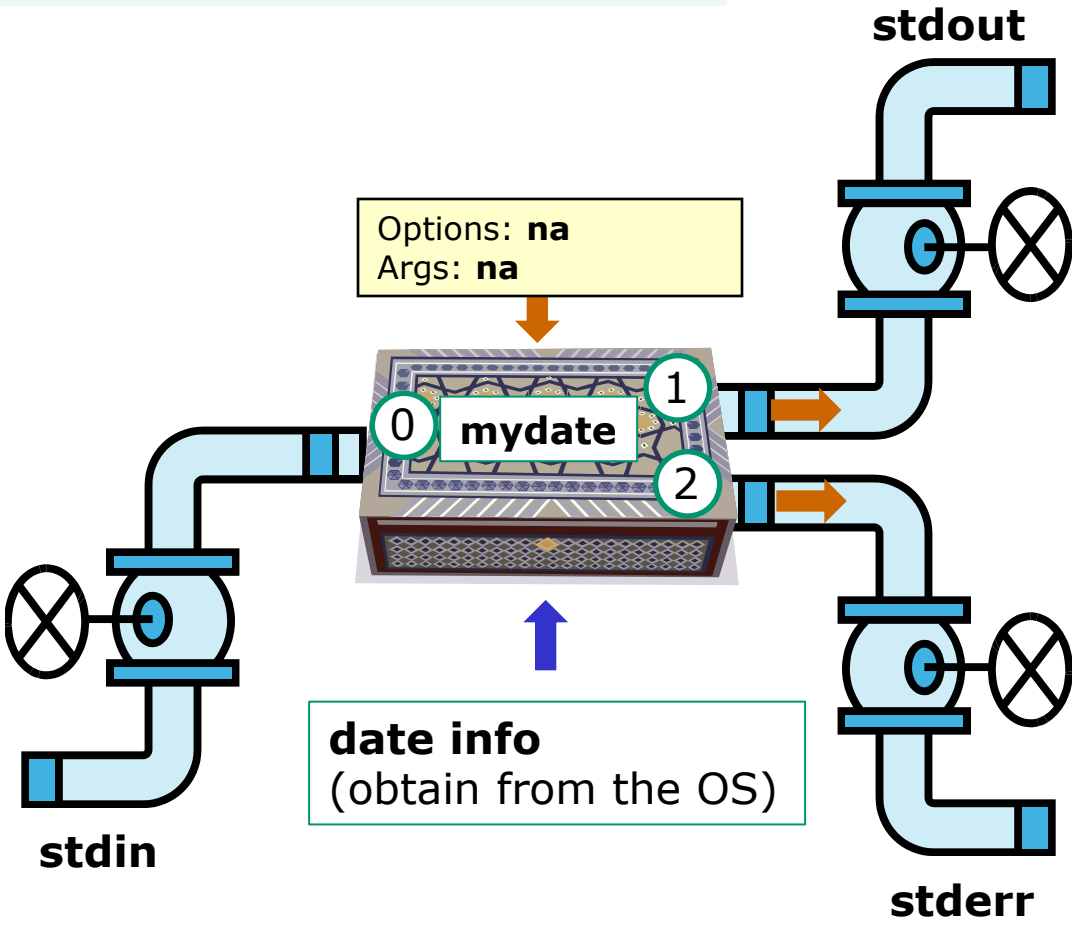
Running a Script

```
/home/cis90/simben $ cat mydate
#!/bin/bash
echo "Hola $LOGNAME"
date +'%m/%d/%Y'
echo $myvar1 $myvar2 $myvar3
```

Add this line to the last script we made

Running a Script

$ mydate



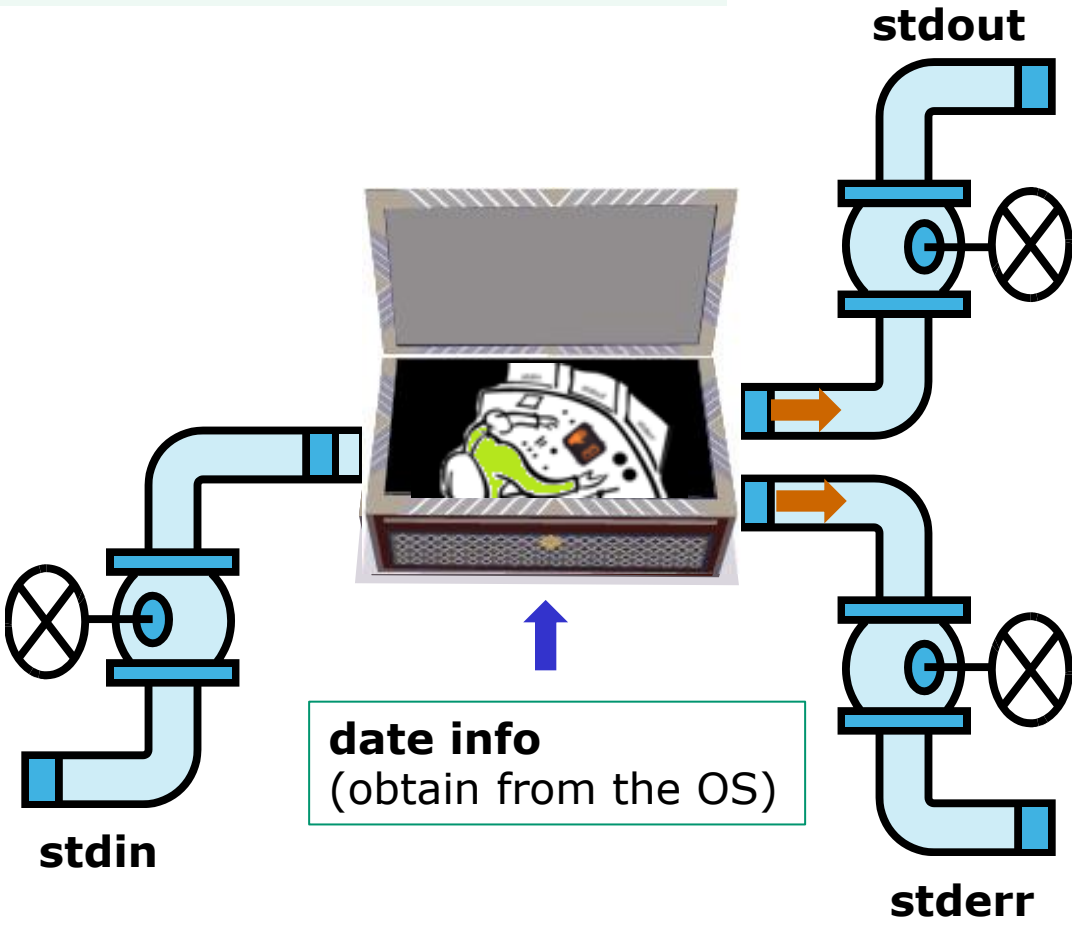

```
Hola simben90
05/09/2012
```

*In this example, ouptput from **myscript** goes to stdout.*

stdout has not been redirected so it goes to the default terminal device (your screen).

Running a Script

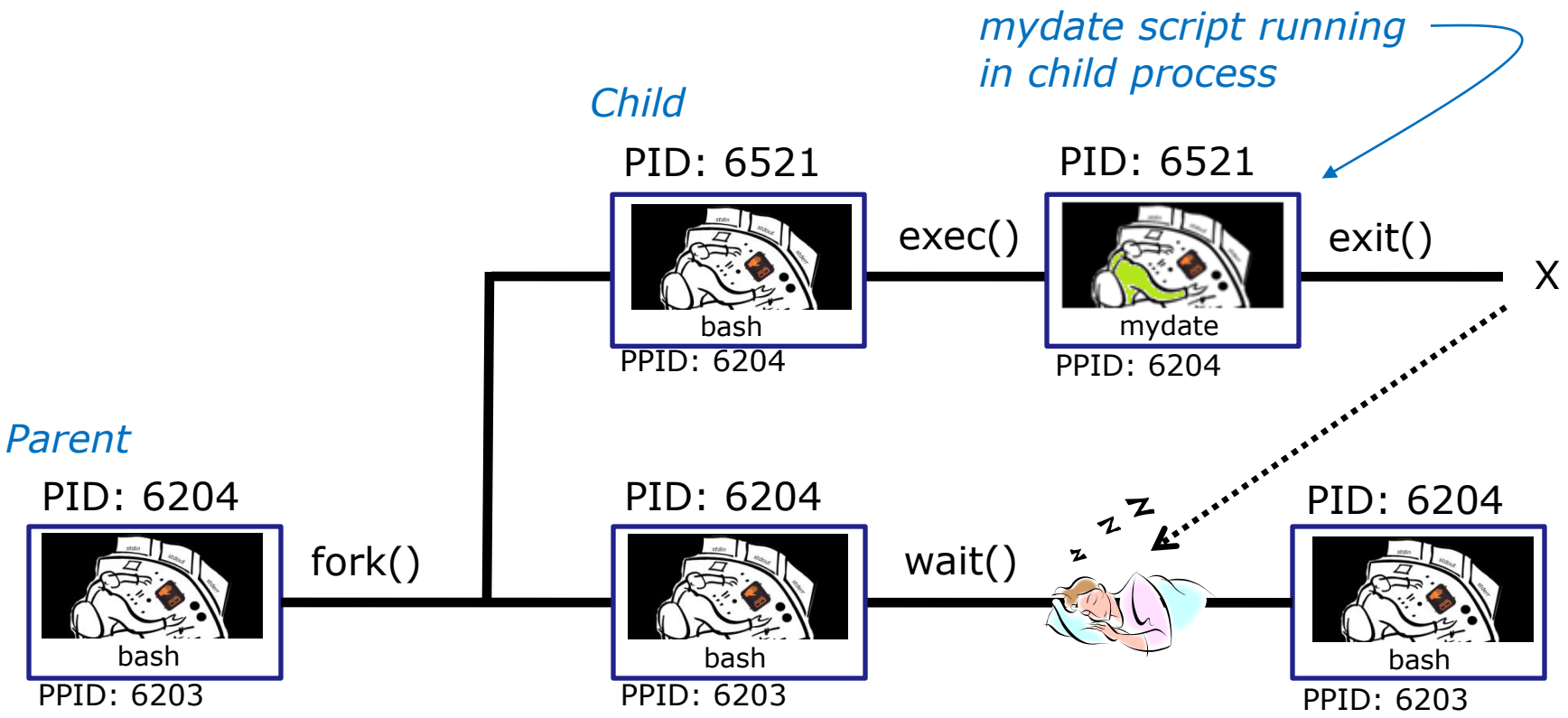
```
$ mydate
```




```
Hola simben90
05/09/2012
```

A sneak peek into memory to see what our process looks like!

Running a Script



*Whenever you run any command, program, or script it runs as a **child process***

Running a Script

```
/home/cis90/simben $ cat mydate
#!/bin/bash
echo "Hola $LOGNAME"
date +'%m/%d/%Y'
echo $myvar1 $myvar2 $myvar3
```

In the parent process, initialize the three variables

```
/home/cis90/simben $ myvar1=Tic; myvar2=Tac; myvar3=Toe
/home/cis90/simben $ echo $myvar1 $myvar2 $myvar3
Tic Tac Toe
```

*What happens if we run **mydate** now?*

Running a Script

```
/home/cis90/simben $ cat mydate
#!/bin/bash
echo "Hola $LOGNAME"
date +'%m/%d/%Y'
echo $myvar1 $myvar2 $myvar3

/home/cis90/simben $ myvar1=Tic; myvar2=Tac; myvar3=Toe
/home/cis90/simben $ echo $myvar1 $myvar2 $myvar3
Tic Tac Toe

/home/cis90/simben $ mydate
Hola simben90
05/09/2012

/home/cis90/simben $
```

*Running **mydate***
(as a child process)

Why no Tic Tac Toe output?

Running a Script

```
/home/cis90/simben $ export myvar1
/home/cis90/simben $ mydate
Hola simben90
05/09/2012
Tic

/home/cis90/simben $ export myvar2
/home/cis90/simben $ mydate
Hola simben90
05/09/2012
Tic Tac

/home/cis90/simben $ export myvar3
/home/cis90/simben $ mydate
Hola simben90
05/09/2012
Tic Tac Toe
```

Rule 1: A child process can only see variables the parent has exported

Running a Script

```
/home/cis90/simben $ echo $myvar1 $myvar2 $myvar3
Tic Tac Toe

/home/cis90/simben $ cat mydate
#!/bin/bash
echo "Hola $LOGNAME"
date +'%m/%d/%Y'
echo $myvar1 $myvar2 $myvar3
myvar1=red myvar2=white myvar3=blue
echo $myvar1 $myvar2 $myvar3

/home/cis90/simben $ mydate
Hola simben90
05/09/2012
Tic Tac Toe
red white blue

/home/cis90/simben $ echo $myvar1 $myvar2 $myvar3
Tic Tac Toe
```

Add these new lines

Rule 2: A child process cannot change the parent's variables.

Running a Script

Unless we want them to

```
/home/cis90/simben $ echo $myvar1 $myvar2 $myvar3
Tic Tac Toe

/home/cis90/simben $ source mydate
Hola simben90
05/09/2012
Tic Tac Toe
red white blue

/home/cis90/simben $ echo $myvar1 $myvar2 $myvar3
red white blue
```

Sourcing a script causes the instructions to be run in the parent process. A child process is not created

```
}
while no-comprende
do
  runningScript
done
```

Printers

Sneak Peak for CIS 90 Students

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)

Back then:
- Serial cable
- Parallel printer cable

Printer Configuration

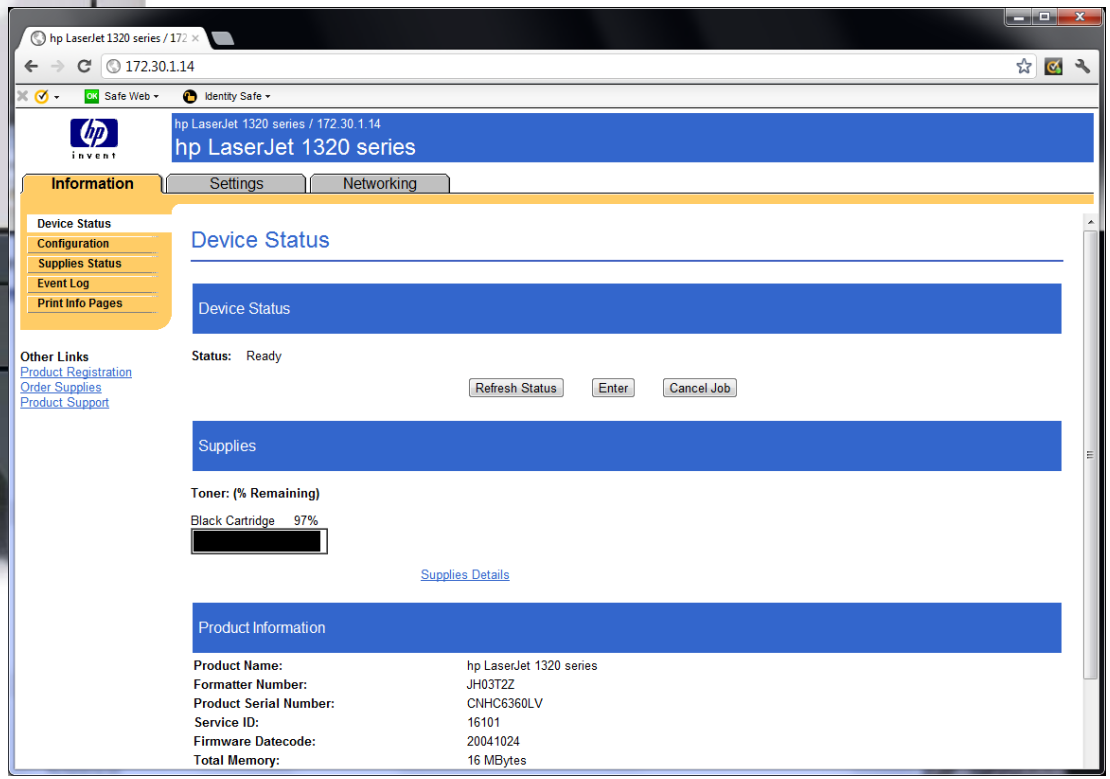
CUPS

Example printer configuration

Printer: HP LaserJet 1320n
Connection: LAN

CUPS

The LaserJets have a web-based management utility

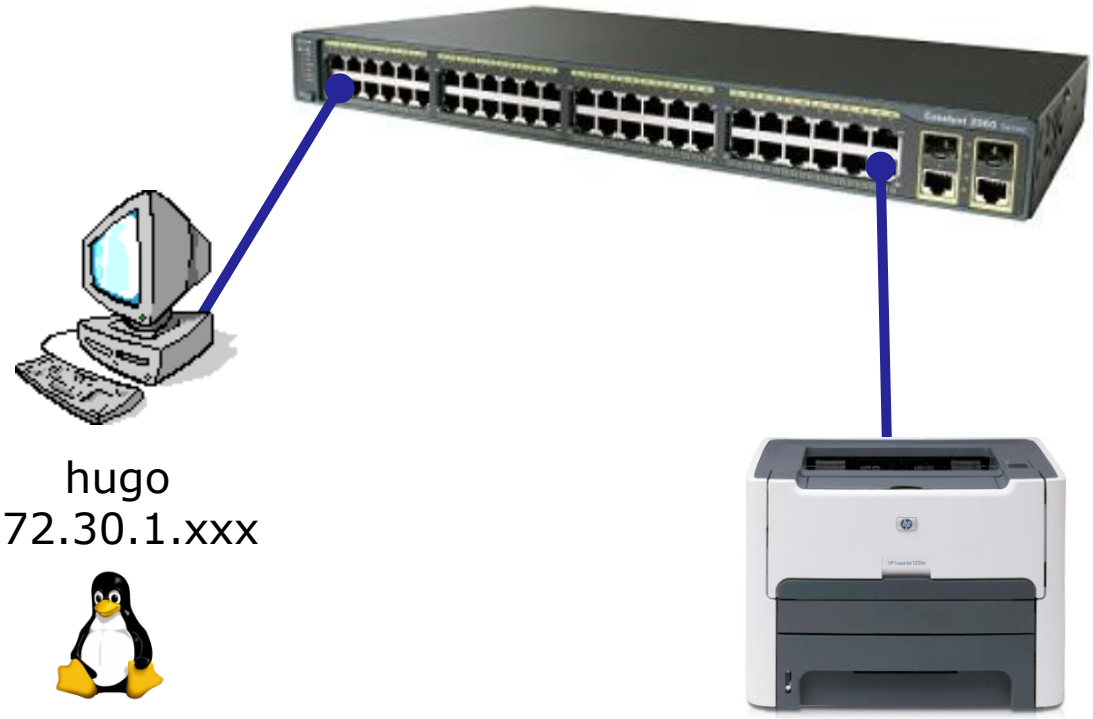


IP Address for this 1320n is 172.30.1.14

CUPS

CUPS

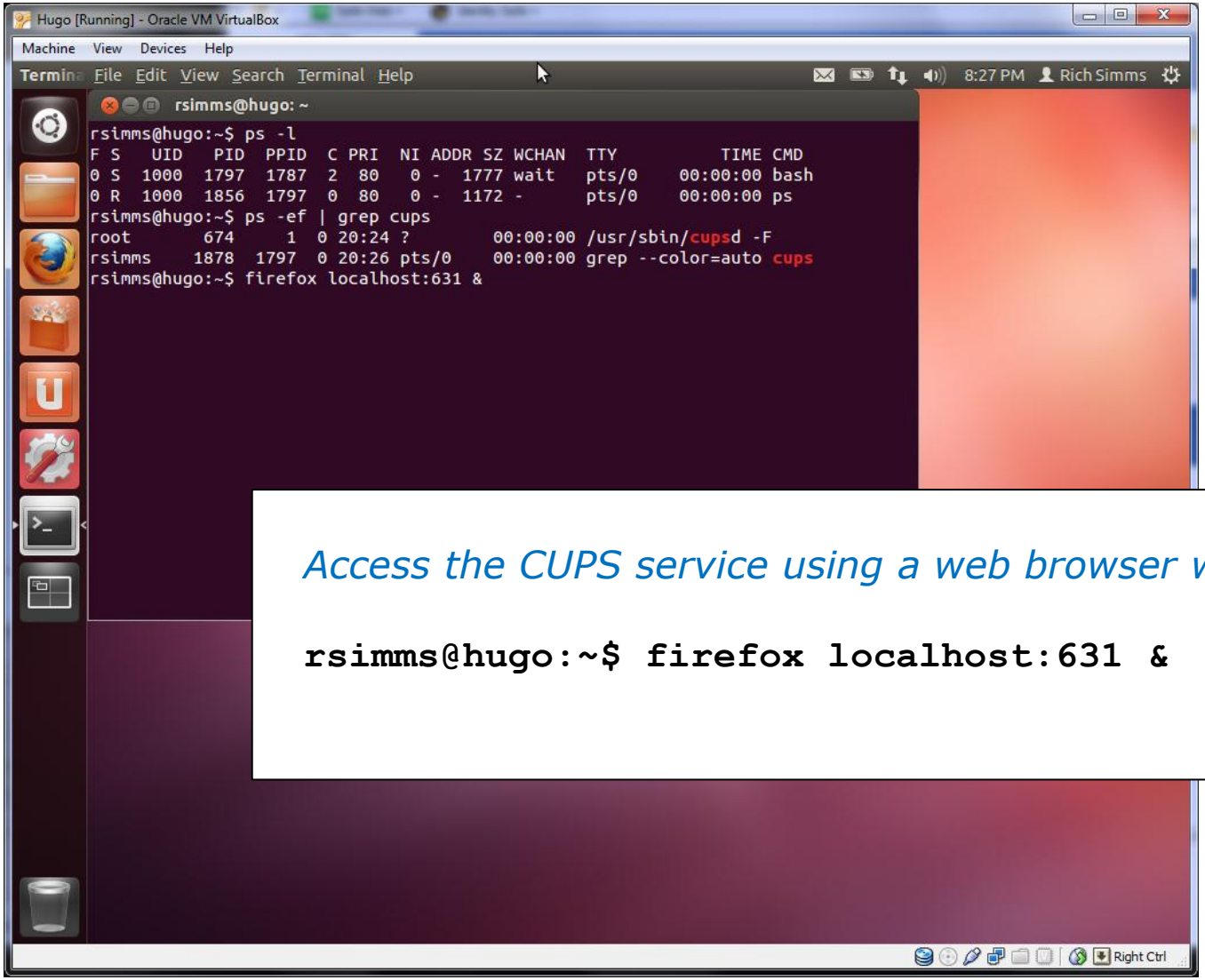
This example will show how to add the HP 1320n as a networked printer.



hugo
172.30.1.xxx

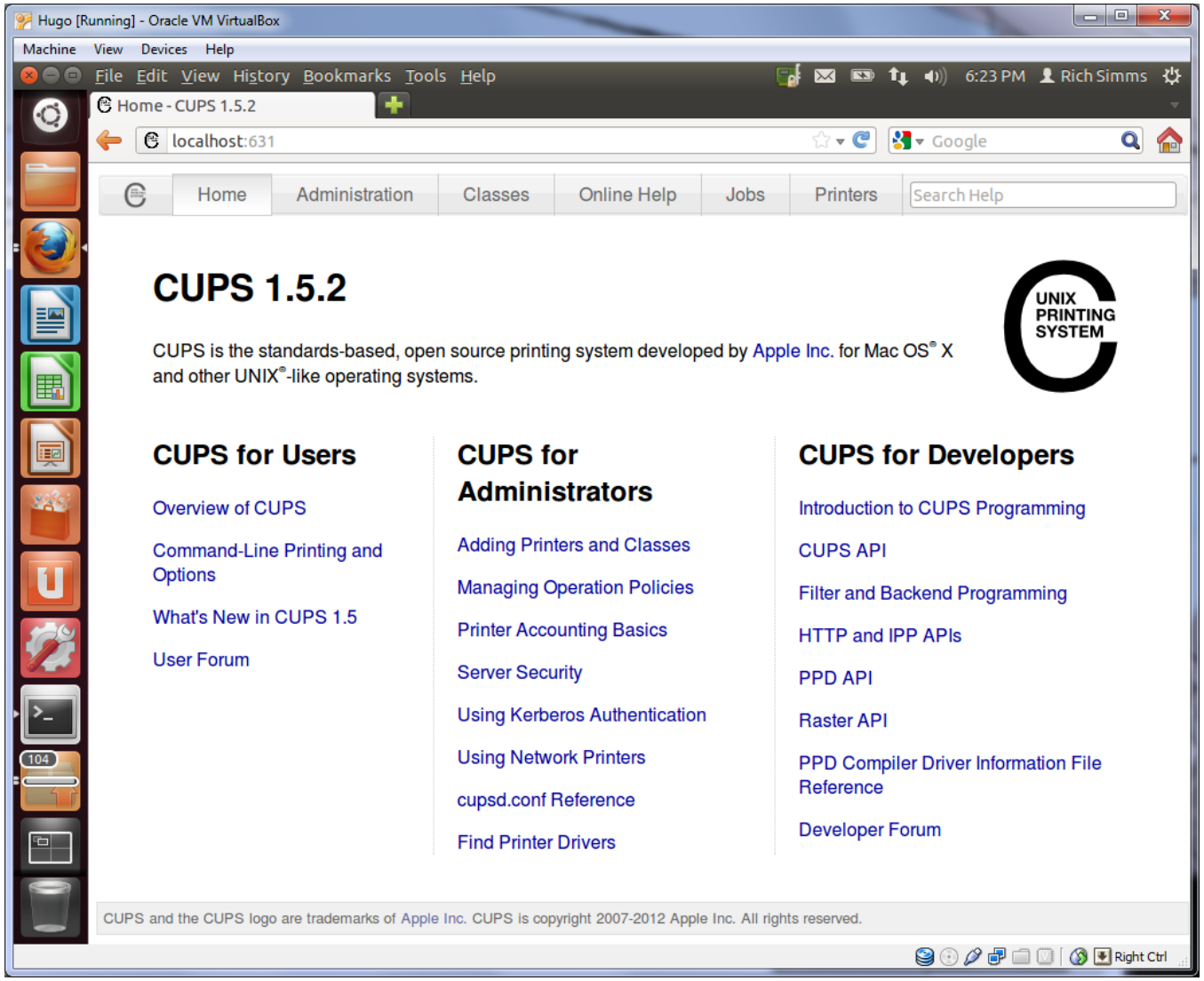
hp1320n
172.30.1.14

CUPS

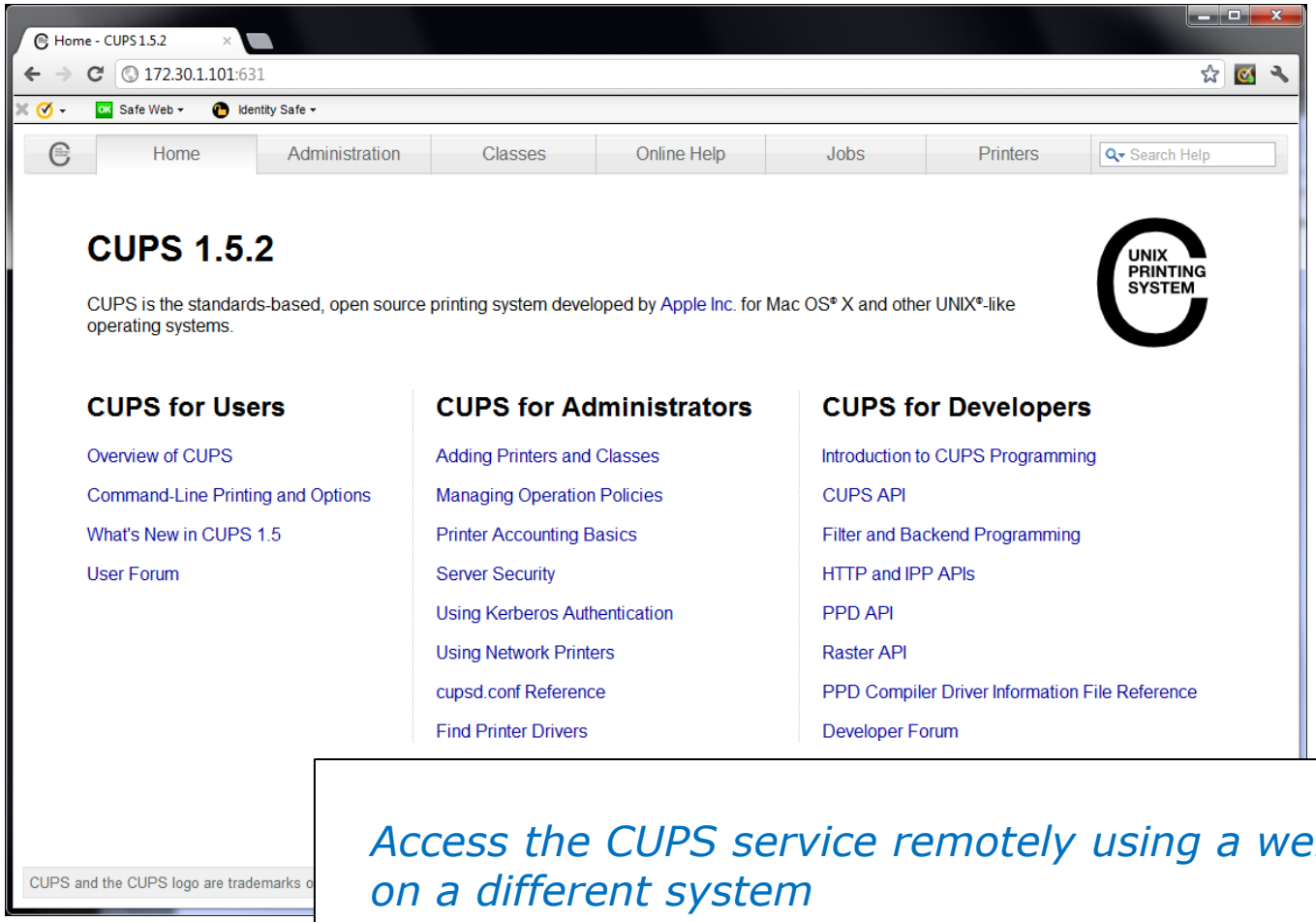


Access the CUPS service using a web browser with

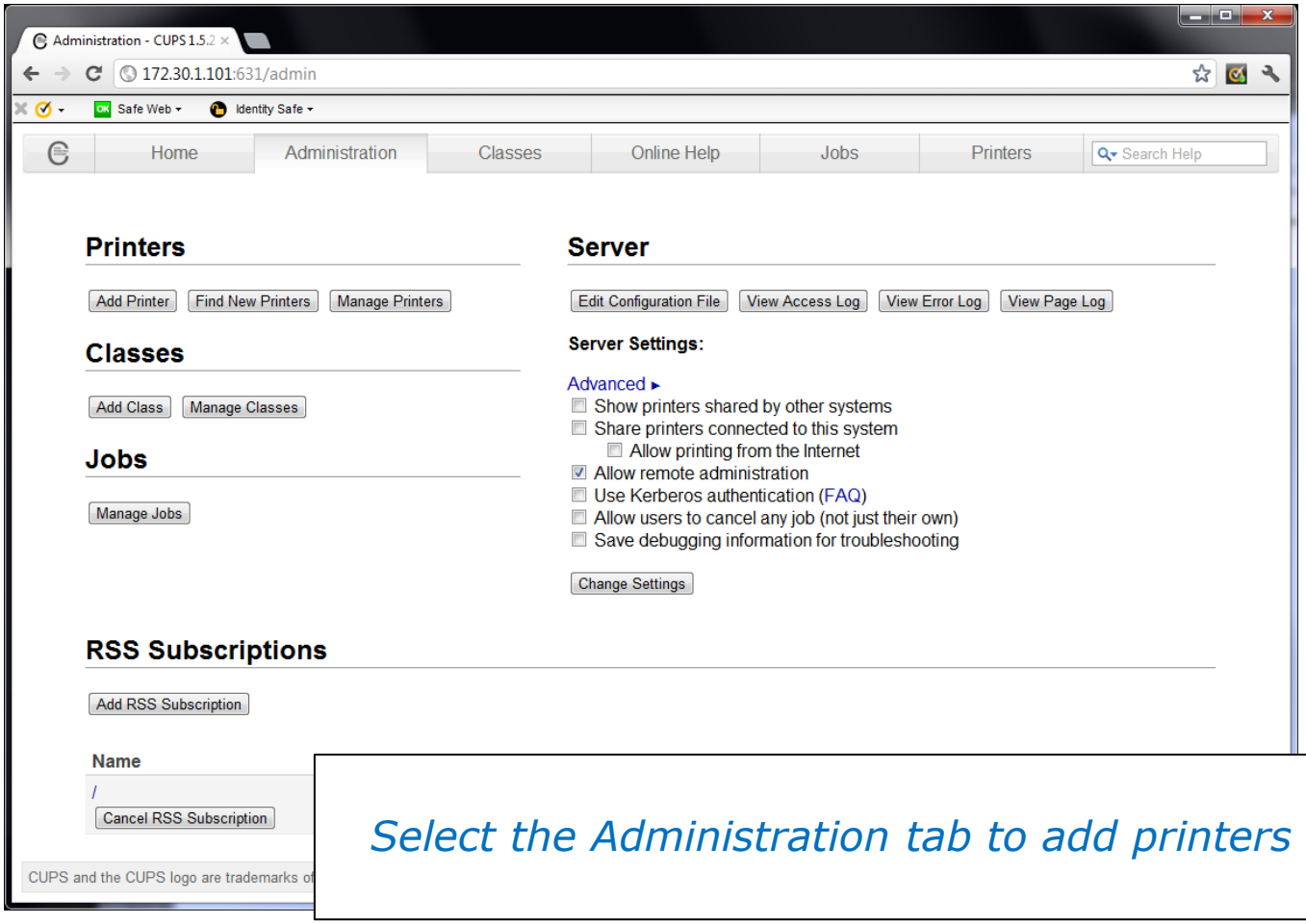
```
rsimms@hugo:~$ firefox localhost:631 &
```


Hugo [Running] - Oracle VM VirtualBox
Machine View Devices Help
File Edit View History Bookmarks Tools Help
6:23 PM Rich Simms
Home - CUPS 1.5.2
localhost:631
Google
Home
Administration
Classes
Online Help
Jobs
Printers
Search Help
CUPS 1.5.2
CUPS is the standards-based, open source printing system developed by Apple Inc. for Mac OS® X and other UNIX®-like operating systems.
UNIX PRINTING SYSTEM
CUPS for Users
Overview of CUPS
Command-Line Printing and Options
What's New in CUPS 1.5
User Forum
CUPS for Administrators
Adding Printers and Classes
Managing Operation Policies
Printer Accounting Basics
Server Security
Using Kerberos Authentication
Using Network Printers
cupsd.conf Reference
Find Printer Drivers
CUPS for Developers
Introduction to CUPS Programming
CUPS API
Filter and Backend Programming
HTTP and IPP APIs
PPD API
Raster API
PPD Compiler Driver Information File Reference
Developer Forum
CUPS and the CUPS logo are trademarks of Apple Inc. CUPS is copyright 2007-2012 Apple Inc. All rights reserved.
Right Ctrl

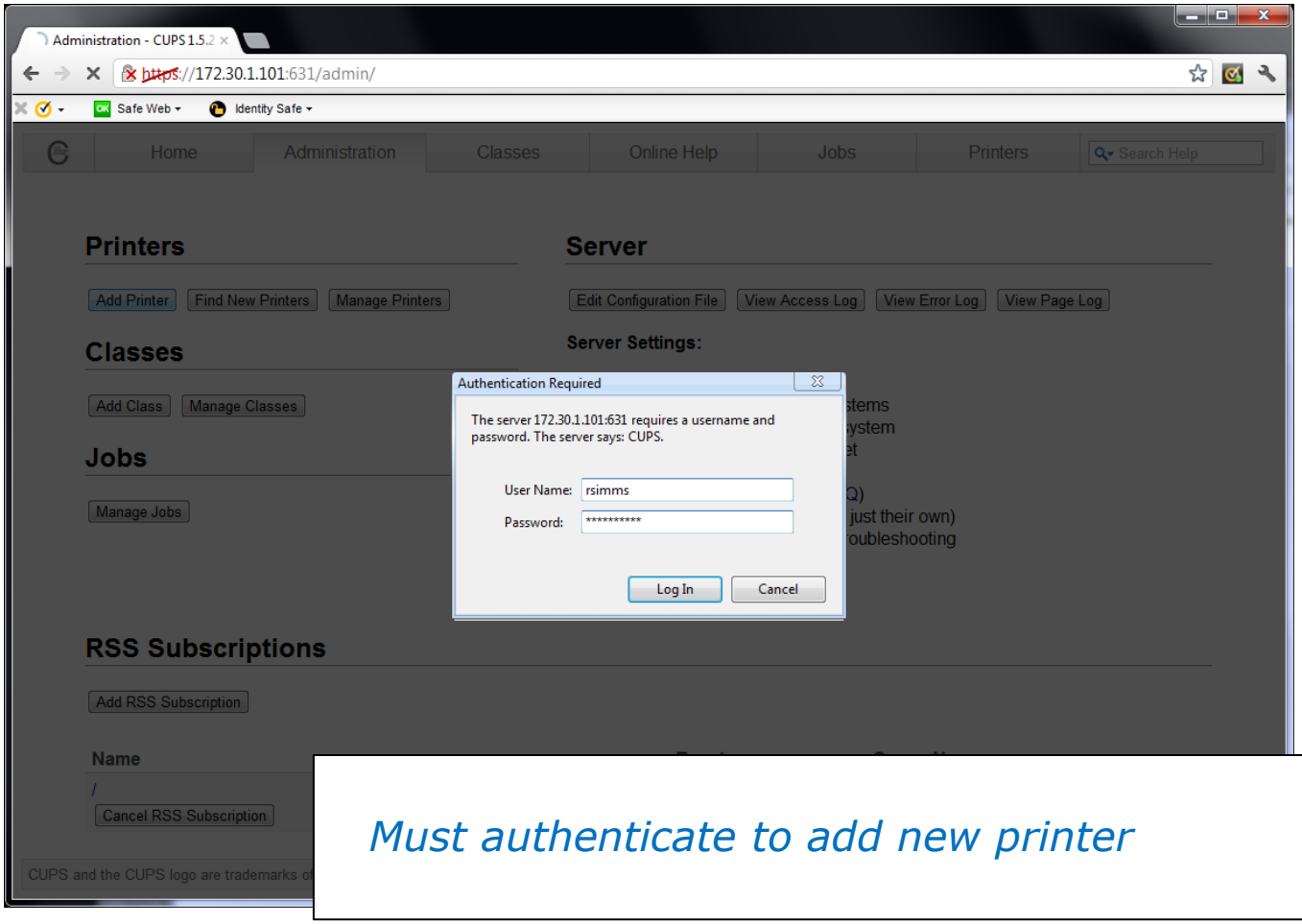
CIS 192



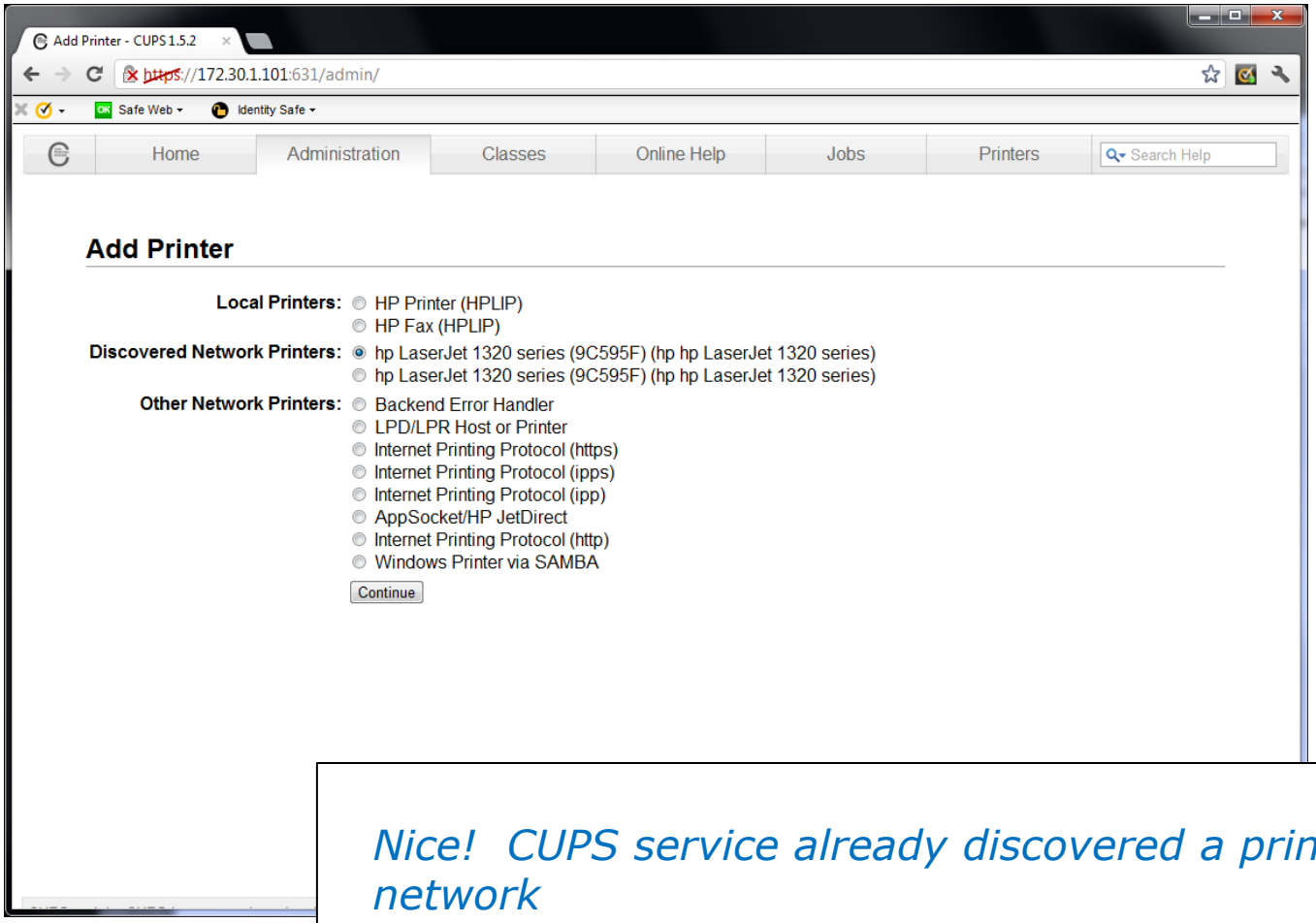
Access the CUPS service remotely using a web browser on a different system



Select the Administration tab to add printers

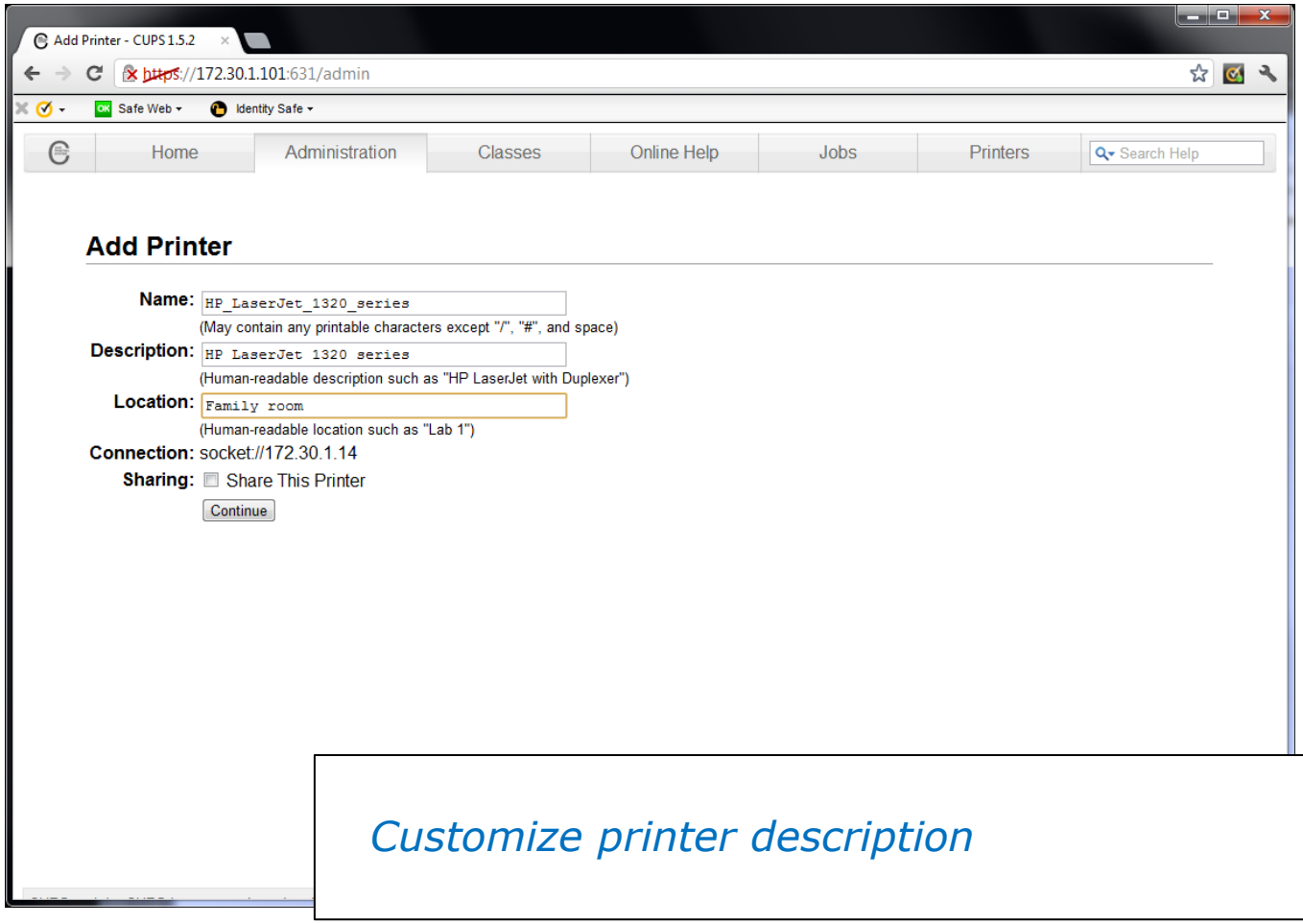


Must authenticate to add new printer

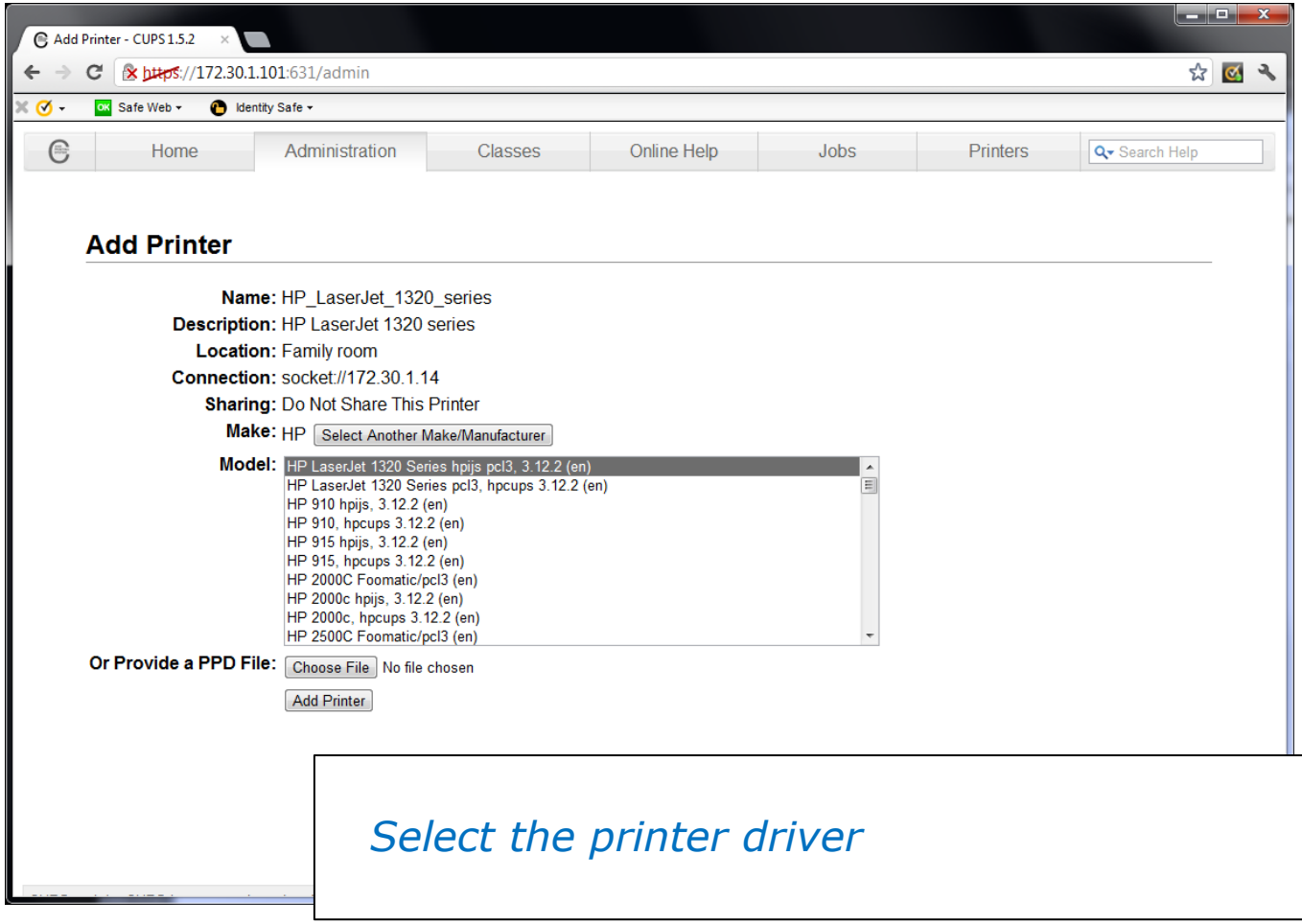


Nice! CUPS service already discovered a printer on the network

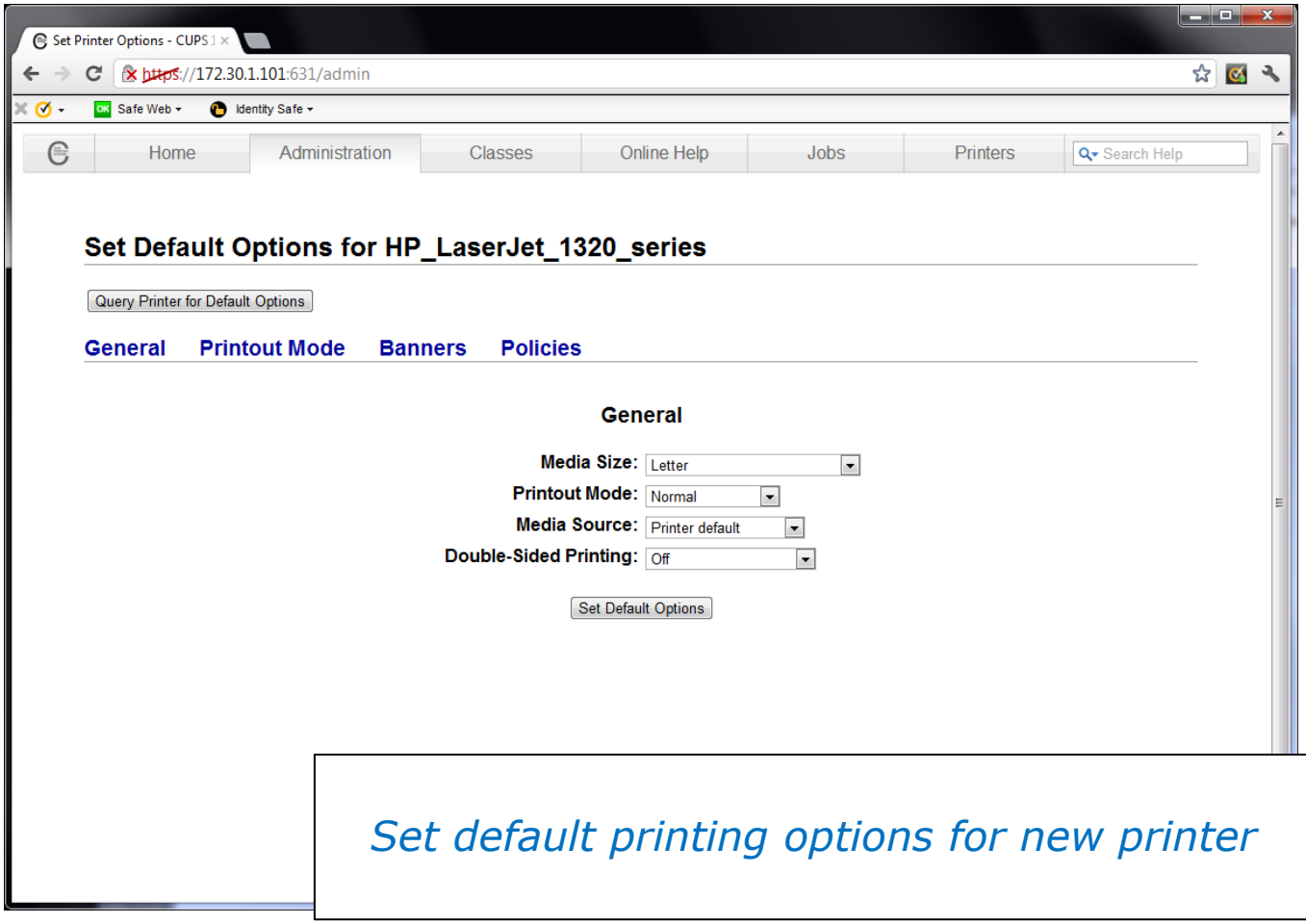
CIS 192



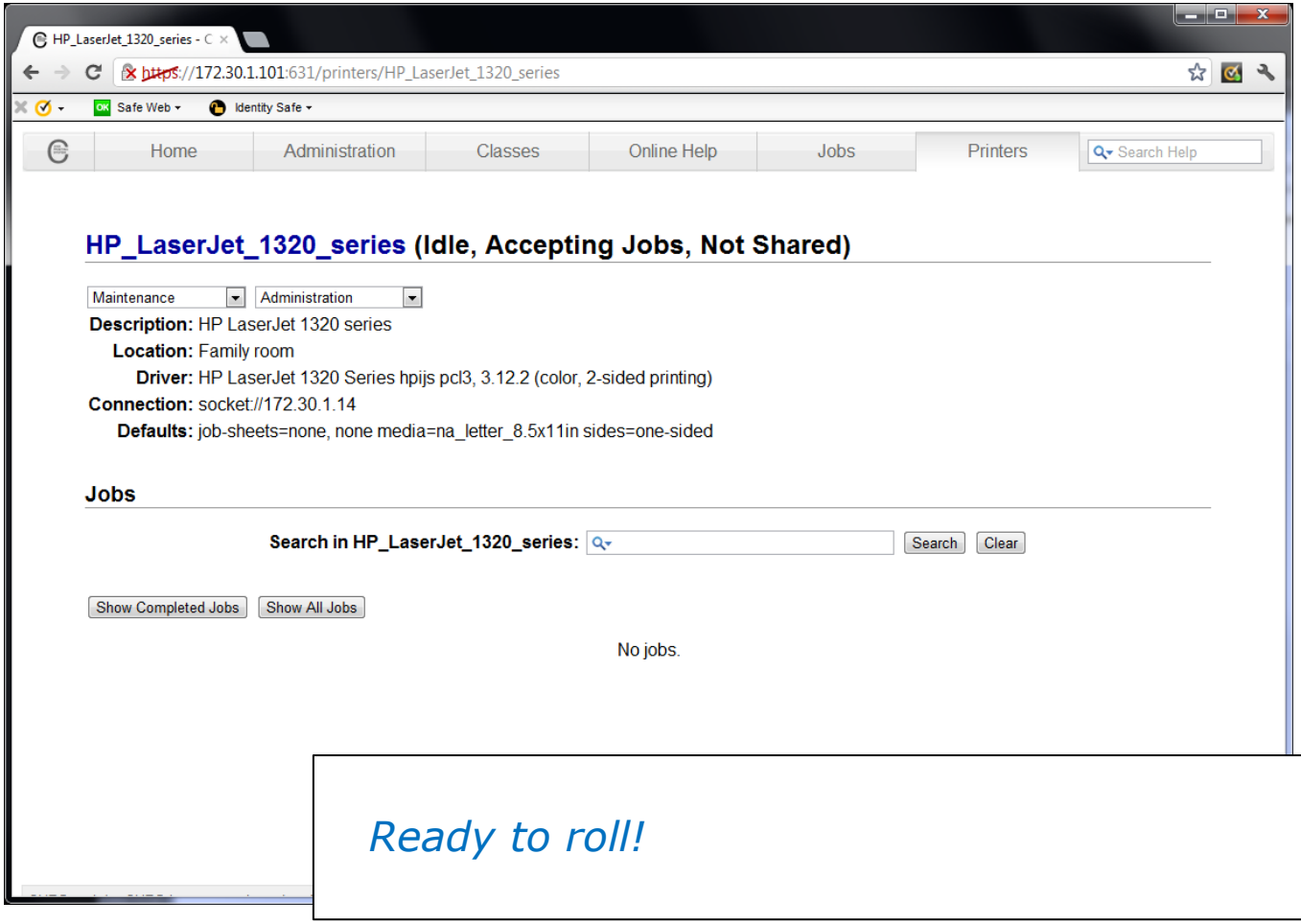
Customize printer description



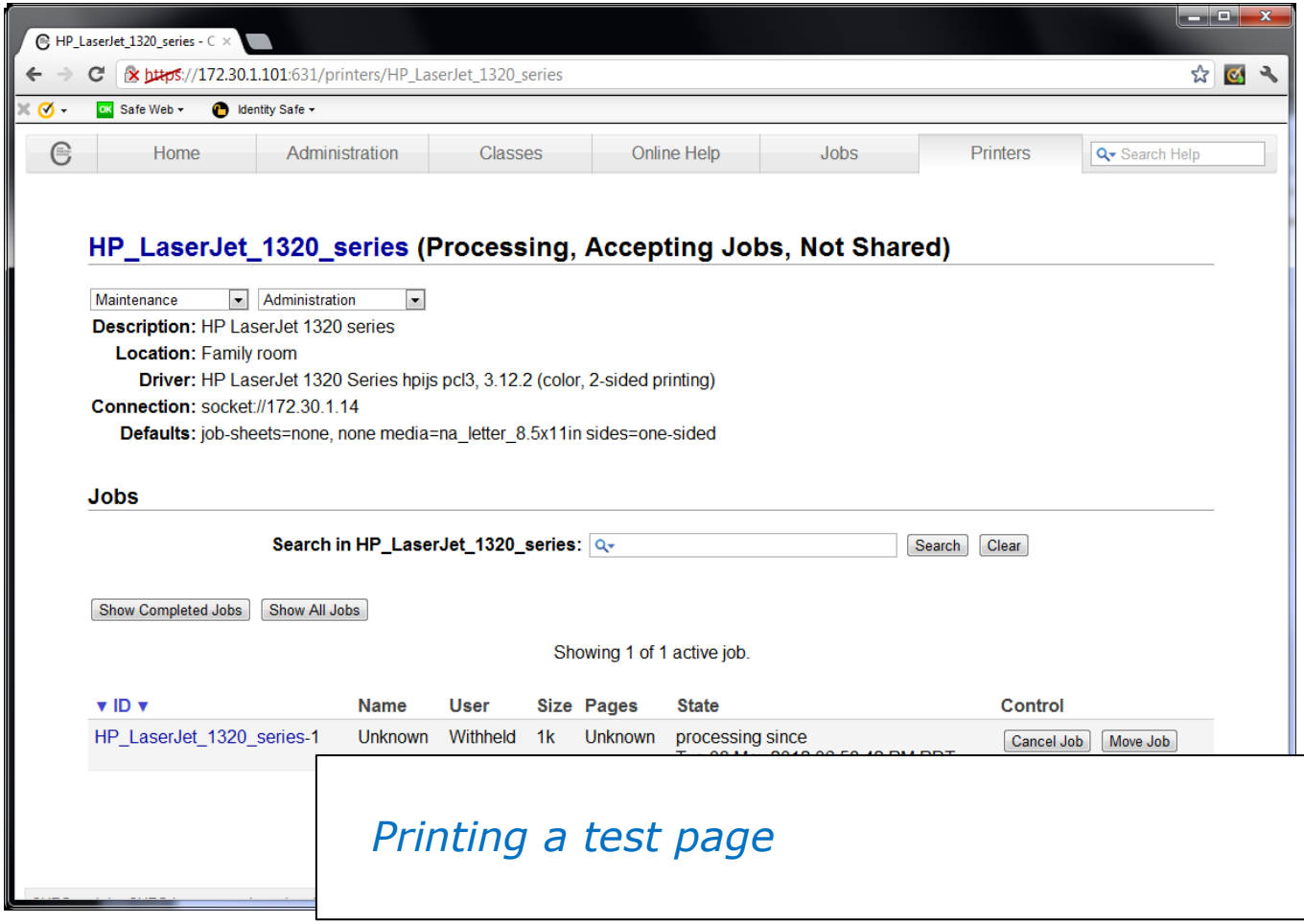
Select the printer driver



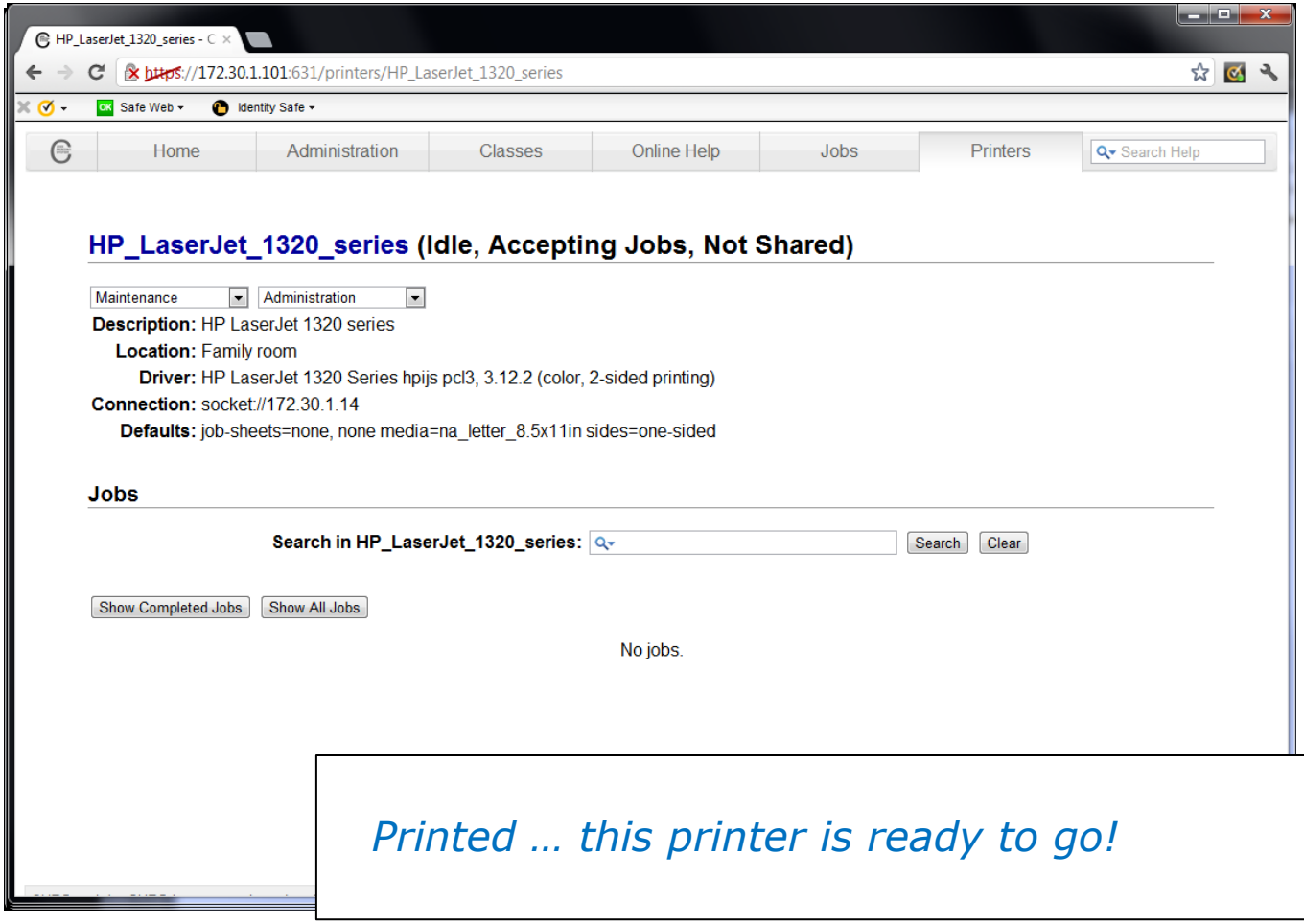
Set default printing options for new printer



Ready to roll!



Printing a test page



Printed … this printer is ready to go!

Printing in Linux

Printing Commands

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

BSD (Berkeley Software Distribution) based print subsystem
- lpr (to print)
- lpq (queue management)
- lprm (to remove jobs)

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

BSD is a branch of UNIX developed at the University of California, Berkeley

CUPS

lpstat command

*Use **lpstat** to show spooled print jobs, available and default printers*

```
rsimms@hugo:~$ lpstat

rsimms@hugo:~$ lpstat -p
printer HP_LaserJet_1320_series is idle.  enabled since Tue 08 May
2012 08:46:45 PM PDT

rsimms@hugo:~$ lpstat -p -d
printer HP_LaserJet_1320_series is idle.  enabled since Tue 08 May
2012 08:46:45 PM PDT
system default destination: HP_LaserJet_1320_series
```

The –p option will show the available printers

The –d option will identify the default printer

CUPS
lpstat command

On Opus

```
/home/cis90/simben $ lpstat -p -d
printer charlie disabled since Tue 26 Jan 2010 05:03:19 PM PST -
        I don't really exist
printer hplaser disabled since Tue 26 Jan 2010 04:58:14 PM PST -
        Out of paper
system default destination: charlie
```

There are two "pretend" printers named charlie and hplaser on Opus

CUPS

lp and lpr commands

*Use **lp** (or **lpr**) to print files*

```
/home/cis90/simben $ lp lab10
request id is hplaser-5 (1 file(s))

/home/cis90/simben $ lp -d hplaser lab10
request id is hplaser-6 (1 file(s))
```

*With **lp**, use the –d option to manually select a printer*

```
/home/cis90/simben $ lpr lab10

/home/cis90/simben $ lpr -P hplaser lab10
```

*With **lpr**, use the –P option to manually select a printer*

CUPS

lp and lpr commands

```
/home/cis90/simben $ echo "Print Me Quietly" | lpr -P hplaser
/home/cis90/simben $
```

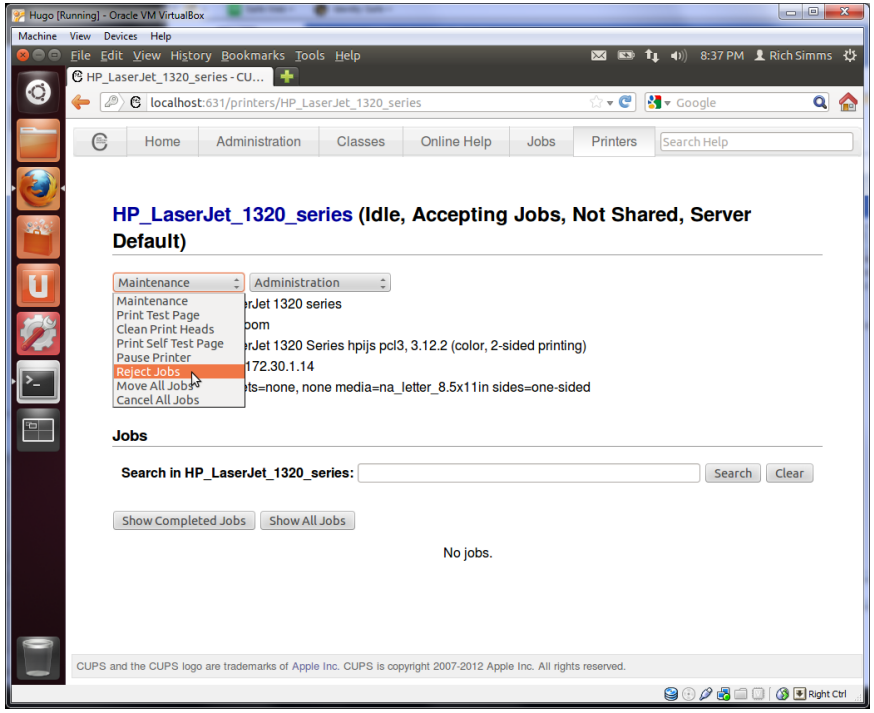
Note that both lp and lpr will read from stdin.

This allows output from another command to be piped in

Managing Print Jobs

CUPS

Rejecting Jobs



CIS 192

Clicking the Reject Jobs *selection 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 ~]#
```

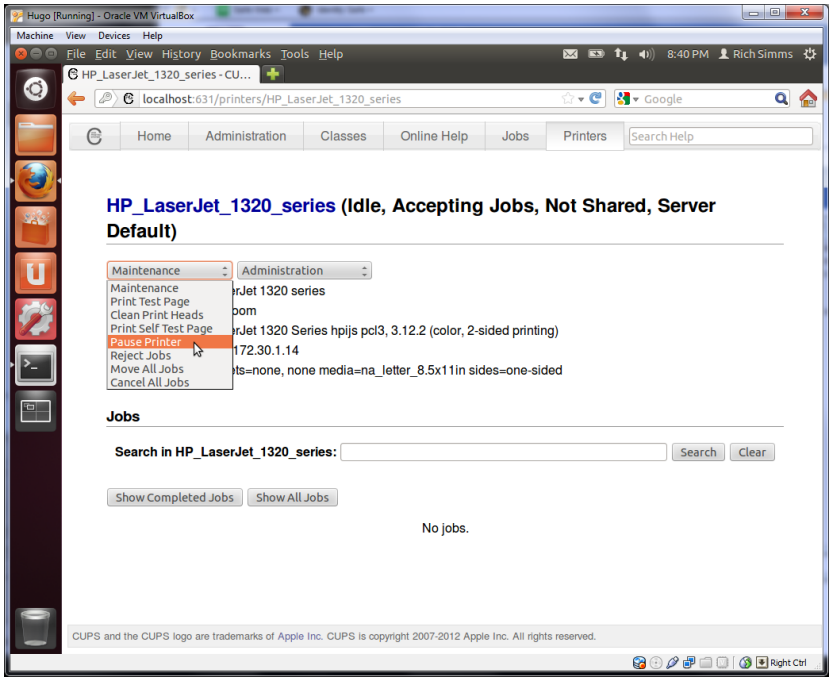
CUPS

Pausing the Printer

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


[root@benji ~]# lpq
hp7550 is not ready
Rank    Owner   Job     File(s)
Total Size
1st     root    22      myfile
1024 bytes


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



Clicking the **Pause Printer** *selection on the web based utility will still allow jobs to be spooled*

CIS 192

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

Wrap up

Commands:

lp, lpr	- Linux print command
cancel, lprm	- cancel print job
lpq, lpstat	- Show print queue

Web:

http://hostname:631	- CUPS web based management utility
http://hostname:9100	- HP JetDirect printer

Next Class

Assignment: Check Calendar Page on web site to see what is due next week.

No Quiz

No Lab due

Work on final projects

Optional extra credit labs

Final Project Workshop

- See if you can get one "starter" task scripted and working before leaving class today.
- Grade your starter script using the Final Project rubric

Implementing all five tasks (6 points each): • Requirements for each task: –Minimum of 10 "original" script command lines –Has one or more non-generic comments to explain what it is doing –Has user interaction
You don't have to do all of these but do at least five: • Redirecting stdin (5 points) • Redirecting stdout (5 points) • Redirecting stderr (5 points) • Use of permissions (5 points) • Use of filename expansion characters (5 points) • Use of absolute path (5 points) • Use of relative path (5 points) • Use of a PID (5 points) • Use of inodes (5 points) • Use of links (5 points) • Use of scheduling (5 points) • Use of a GID or group (5 points) • Use of a UID or user (5 points) • Use of a /dev/tty device (5 points) • Use of a signal (5 points) • Use of piping (5 points) • Use of an environment variable (5 points) • Use of /bin/mail (5 points) • Use of a conditional (5 points) The maximum for this section is 25 points.

Backup