

CS-344 - Unix Operating System Fundamentals

Lecture 5
Specifying Instructions to the Shell,
Controlling User Processes and
Managing
&
Printing and Archiving Large Files

Based on slides created by
Dr. Bangalore for the
Spring 2005 offering of
the course

Declaring and Using Variables in the Shell

- Declaring Variables: `variable_name = value`
- Display Variables: `echo $variable_name`
- Examples:
 - a = Hello
 - b = date
 - \$b
 - c = `date`
 - d = "You are currently logged on to `hostname`"
 - e = "Today's date is: \$b"
 - f = "Today's date is: \$c"
 - echo \$e \$f

Overwriting Existing Files

- ❑ When output is redirected to a file, if the file already exists it's overwritten.
- ❑ To avoid overwriting existing files set the noclobber variable: `set -o noclobber`
- ❑ When noclobber variable is set, the shell complains that the file exists
- ❑ To overwrite a file when noclobber variable is set, use `>!` instead of `>` for output redirection
- ❑ Example: `date >! filename`

7/8/2005

4

Avoiding Accidental Removal of Files

- ❑ Unlike Windows, in UNIX when files are deleted they cannot be undeleted
- ❑ `noclobber` feature is an instruction to the shell not to the UNIX commands (e.g., `cp`, `mv`, `rm`)
- ❑ To avoid accidental removal of files use `-i` option with `cp`, `mv`, `rm`
- ❑ set alias for `cp`, `mv`, `rm`, etc. to avoid using `-i` every time that you use those commands
 - `alias rm='rm -i'`
 - `alias cp='cp -i'`
 - `alias mv='mv -i'`
- ❑ To override an alias use `"\command"` instead of command (e.g., `\rm filename`)

7/8/2005

5

Redirecting Error Messages

- ❑ Utilities write to the error stream when an error occurs during their execution
 - `ls -l filename` (if the filename does not exist or the file permissions are not sufficient)
- ❑ We have seen output redirection using `>` and `>>`. To redirect error messages use `>2>` or `>2>>` (in bash)
- ❑ The shell assigns `1` for the standard output stream and `2` for the standard error stream (`>` is same as `>1`)
- ❑ To redirect both error and output together use `>2>&1` after the filename
 - `ls -l myfile xyz > outerr 2>&1`

7/8/2005

6

Local & Environment Variables

- Use "export variable_name"
 - a=Hello
 - export a
 - export b=World
 - export PATH=\${PATH}::~/bin
- Remove variable - unset
 - unset a
- List all variables - set
- List environment variables - env

7/8/2005

10

Filename Completion with Shell

- When listing or editing files typing complete filenames accurately could be difficult
- The shell can help in this problem:
 - Using wildcard characters
 - Using file completion with TAB (BASH)
- Enter part of the filename/directory and press TAB
- If a unique file/directory exists the shell will complete it, otherwise it will display all possible options

7/8/2005

11

Job Control Commands (I)

- When the shell executes a command it waits until the utility completes
- If we do not like to wait, we can instruct the shell to execute the job in background by adding & at the end of the command (e.g., sleep 10 &)
- To suspend current job, press CTRL-Z
- To kill a running job with interrupt signal, press CTRL-C
- To kill a running job with quit signal, CTRL-\

7/8/2005

12

Job Control Commands (II)

- ❑ To list all jobs background and stopped jobs use the command `"jobs"`
- ❑ To put suspended job into background use `"bg %job_num"`
- ❑ To bring a background job into foreground use `"fg %job_num"`
- ❑ To kill a job use `"kill %job_num"`

7/8/2005

13

Details about User Processes

- ❑ To list user processes use `"ps"` command (on some systems you can also use `"top"`)
- ❑ To obtain detailed information about all processes use `"ps -ef"`
- ❑ To list processes owned by you use `"ps -u $USER"`
- ❑ To obtain a long listing of your processes use `"ps -l"`

```
$ ps -l
FS UID  PID  PPID C PRNI  ADDR  SZ  WCHAN TTY  TIME CMD
8 S 511 22817 22815 0 51 20  ?  309  ? pts/1  0:00 bash
```

- ❑ To kill a particular process use `"kill -9 PID"`

7/8/2005

14

Working with Large Files, Printing (I)

- ❑ `view`: similar to `vi` with the read-only flag set
- ❑ `split`: split long files into smaller files
 - `split -l linecount filename suffix`
- ❑ `lp`: printing files (also `lpr -P`)
 - `lp -d printername filename`
- ❑ `lpstat`: check printer status (also `lpq -P`)
 - `lpstat -d printername`

7/8/2005

15

Working with Large Files, Printing (II)

- ❑ **cancel:** to remove jobs from print queue
(also *lprm*)
 - *cancel jobid*
- ❑ to remove control characters from a file send to the printer use: *col -bx*
 - *cat filename | col -bx | lp -dprinter*

7/8/2005

16

Pagination of files with pr

- ❑ pagination based on number of lines:
pr -l lines filename
- ❑ customize page headers:
pr -h 'My Header' filename
- ❑ numbered listing: *pr -n filename*
- ❑ skipping pages: *pr +npages filename*
- ❑ printing multiple files side-by-side:
pr -m file1 file2 (lines that does not fit are truncated)
- ❑ print as multiple columns: *pr -ncols filename*

7/8/2005

17

Installing software

- ❑ Steps involved:
 - Download
 - Configure
 - Make
- ❑ Download and install Apache ant software
 - Download tar gzip file from:
<http://government-grants.org/mirrors/apache.org/ant/source/apache-ant-1.6.2-src.tar.gz>
 - Copy this file to one of the blazer machines
 - Install the software in your home directory
 - Add "ant" to your path
- ❑ To recover the files of a gzip/tar file:
 - *gzip -dv apache-ant-1.6.2-src.tar.gz*
 - *tar -xvf apache-ant-1.6.2-src.tar*

7/8/2005

18
