

Shell Scripting II

IT2758 - OS

Agenda

- ▶ date, wc, tail
- ▶ Substitution
- ▶ Redirectors
- ▶ Filters
- ▶ Archiving / Backingup

date



- ▶ **date [OPTION] . . . [+FORMAT]**
- ▶ Running date with no options will output the system date and time, as in the following output:

```
[prompt]$ date
```

Thu Feb 8 16:47:32 MST 2001

- ▶ Set the system date and time to November 20, 2003, 12:48 PM.

```
[prompt]$ date -s "11/20/2003 12:48:00"
```

- ▶ Outputs the date and time in the following format:

```
[prompt]$ date "DATE: +%m/%d/%y%nTIME: %H:%M:%S"
```

DATE: 02/08/01

TIME: 16:44:55

wc & tail

- ▶ **wc** [*OPTION*] . . . [*FILE*] . . .
 - ▶ Command to print newline, word, and byte counts for each file
 - ▶ Example
 - ▶ [prompt]\$ **wc -l /etc/passwd**

- ▶ **tail** [*OPTION*] . . . [*FILE*] . . .
 - ▶ tail – output the last part of files
 - ▶ Example
 - ▶ [prompt]\$ **tail -3 /etc/passwd**

Substitution

- ▶ `$ (date +"%m%d%Y %T")`
- ▶ The characters "`$()`" tell the shell, "substitute the results of the enclosed command." In our script, we want the shell to insert the results of the command `date +"%m%d%Y %T"` which expresses the current date and time.

Substitution & Redirection

```
[prompt]$ ls -al > opFile_$(date +"%m_%d_%Y")
```

- ▶ Generate a directory listing with command **ls**, and **redirect** the output to a file **opFile_XXXX** where XXXX is the current day, month, and year in the file name.
- ▶ Includes using bash **command substitution**, running the date command in a subshell and inserting that output into the original command.

Redirectors

Re-directs standard output (terminal) to file or command

- ▶ command > file
- ▶ command >> file
- ▶ command < file
- ▶ command | command

Filters

► Wildcards * and ?

► The asterix *

- For representing 0 or more characters

► The question ?

- For representing a single character

```
[prompt]$ ls *.sh  
23.sh 225.sh 522.sh
```

```
[prompt]$ ls 5*  
55.txt 522.sh
```

```
[prompt]$ ls ?5.*  
25.txt 55.txt 25.bin
```

```
[prompt]$ ls 2*.???  
2.txt 25.txt 225.txt 25.bin
```

2.txt

23.sh

225.txt

225.sh

25.txt

55.txt

25.bin

522.sh

Filters : cut

- ▶ **cut** command
 - ▶ For **cutting out** the sections from **each line of files** and writing the result to standard output.
 - ▶ Can be used to **extract** parts of a line by
 - ▶ byte position
 - ▶ character
 - ▶ field
 - ▶ **cut OPTION... [FILE] ...**
 - ▶ Example
 - ▶ **cat /etc/passwd | cut -f 1 -d ":"**

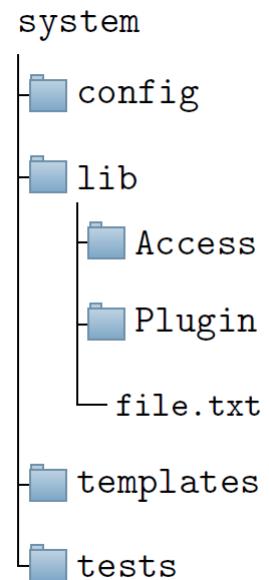
Filters : grep

- ▶ **grep** command
 - ▶ For **searching** plain-text data sets or files **for lines** that match a defined **pattern**
 - ▶ **grep [OPTIONS] PATTERN [FILE...]**
 - ▶ Example
 - ▶ **grep "nologin" /etc/passwd**
 - ▶ **Reverse: grep -v**

Archiving

► **tar** command

- ▶ For creating archive files (also known as tarballs)
- ▶ Historically, **tape archives** were used to backup files
- ▶ Packs multiple directories and their files into a **single file**
- ▶ May be compressed or otherwise
 - ▶ gzip
 - ▶ bzip2
- ▶ **tar [OPTION...] [FILE] ...**
- ▶ **tar -cvf archive.tar <source>** #create
- ▶ **tar -rvf archive.tar <source>** #append
- ▶ **tar -cvzf** #is similar to **tar** followed by **gzip**



END

