Introduction to Linux
Session 2 – Files / Filesystems / Data

Peter Ruprecht
peter.ruprecht@colorado.edu

www.rc.colorado.edu

Outline

- Filesystem layout
- Permissions
- Pattern matching (regular expressions)
- Stream editing and column operations
- Links
- Finding files
- How full is a disk?

Slides available at

https://github.com/ResearchComputing/USGS_2014-07

Pipes and redirection

- Input and output redirection
  - Send output from a command to a new file with >
  - Append output to an existing file with >>
  - Use a file as input to a command with <
- Pipes: | sends output of one command to another command

```
ps –ef | grep ruprech
```

File- and directory-related commands

pwd – prints full path to current directory

cd – changes directory; can use full or relative path as target

mkdir – creates a subdirectory in the current directory

rmdir – removes an empty directory

rm – removes a file (`rm –r` removes a directory and all of its contents)

cp – copies a file

mv – moves (or renames) a file

ls – lists the contents of a directory (`ls –l` gives detailed listing)

chmod/chown – change permissions or ownership

df – displays filesystems and their sizes

du – shows disk usage (`du –sk` shows size of a directory and all of its contents in KB)

File-viewing commands

less – displays a file one screen at a time

cat – prints entire file to the screen

head – prints the first few lines of a file

tail – prints the last few lines of a file (with –f shows in real time the end of a file that may be changing)

diff – shows differences between two files

grep – prints lines containing a string or other regular expression

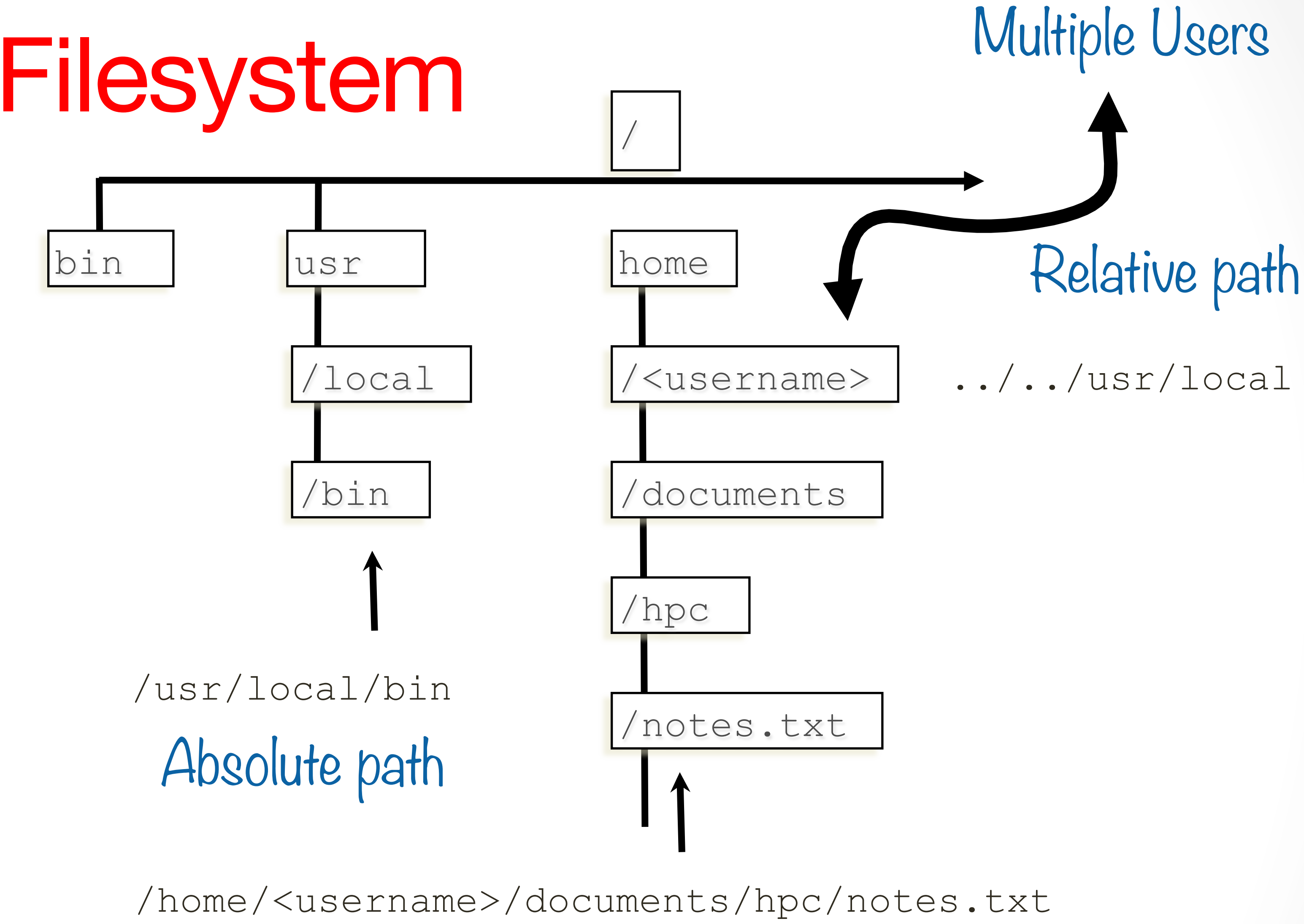
tee – prints the output of a command and also copies the output to a file

sort – sorts lines in a file

find – searches for files that meet specified criteria

wc – count words, lines, or characters in a file

Filesystem



/

bin

usr

home

Multiple Users

Relative path

/local

/bin

/<username>

/documents

/hpc

/notes.txt

../../usr/local

/usr/local/bin

Absolute path

/home/<username>/documents/hpc/notes.txt

Research Computing storage

Please take a look at

https://www.rc.colorado.edu/services/storage/filesystemstorage

for details on using different types of storage systems in the RC environment.

Modes (permissions)

- 3 classes of users:
  - User (u)   *aka "owner"*
  - Group (g)
  - Other (o)
- 3 types of permissions:
  - Read (r)
  - Write (w)
  - Execute (x)

Modes (continued)

- `chmod` changes modes:

To add write and execute permission for your group:

```
chmod g+wx filename
```

To set only read and execute for your group and others:

```
chmod go=rx filename
```

Regular expressions

`string` match string exactly

`.` Match single character

`19.3` (matches 1903, 1913, 19A3)

`*` Match zero or more of preceding character

`‘bugs*’` (matches bug, bugs, bugsss)

`^` Match beginning of line

`‘^data’` (line starts with `data`)

`$` Match end of line

`‘^…$’` (line with exactly 3 chars)

`[ ]` Match from set

`‘Jun[0-9]*_201[01]’` (`Jun` followed by any number of integers followed by `_2010` or `_2011`)

Stream editing (with sed)

```
sed ‘s/Kr/krypton/g’ < input.txt > output.txt
```

(global find-and-replace of Kr with krypton)

```
cat input.txt | sed ‘/^$/d’ > output.txt
```

(remove all blank lines)

```
sed —e ‘s/^/   /’ input.txt > output.txt
```

(add 3 spaces to beginning of each line)

Column operations (with `awk`)

```
awk ‘{print $3}’
```

(print 3rd field or column)

```
awk –F: ‘{print $1,$3}’
```

(print 1st and 3rd fields; fields delimited by :)

```
awk ‘{print $NF}’
```

(print last column; NF means number of fields)

```
awk ‘{print NF}’
```

(print number of fields)

More with awk

```
awk ‘{total = total + $1}END{print total}’
```

(sums the first column)

```
grep ‘^[0-9]’ data.txt | \
  awk ‘{print $2, 3.14*$1}’
```

(for lines beginning with a number, print the 2nd column followed by the 1st column times pi)

How full is a disk?

- `df` displays filesystem information
  - Check if your disk is filling
  - Find where a filesystem is physically located
  - The “-h” flag gives “human readable” units

- `du` shows disk usage
  - `du –sk * | sort –n` is useful for finding large directories

Finding files (with `find`)

```
find /somedir –name “*.pdf”
```

(find files ending in .pdf in /somedir (& subdirs))

```
find ~ -mtime +3
```

(find files in homedir modified over 3 days ago)

```
find . –perm 644 –exec chmod g+w {} \;
```

(find files with rw-r- - r- - ; change to rw-rw-r- -)

```
Find . –name “*.ps” –a –mtime -3
```

(find .ps files modified less than 3 days ago)

Data transfer

See XSEDE site documentation about moving data

- Globus Online
  - Large file transfers with “drag and drop” interface to move data between Globus or Gridftp endpoints
- Utilities
  - scp, sftp, rsync
  - Work best with smaller files or smaller numbers of files

File editing

- **nano** – simple and intuitive to get started with; not powerful; keyboard driven
- **vi/vim** – universal; keyboard driven; powerful but some learning curve required
- **emacs** – keyboard or GUI versions; helpful extensions for programmers; well-documented
- **OpenOffice / LibreOffice** – for WYSIWYG

http://xkcd.com/378/

Thank you!

Slides available at:

https://github.com/ResearchComputing/USGS_2014-07