cat (takes files (or stdin) as input and copies them, sending result to stdout

cat >filename (creates a file)

cat filename (shows you what is in file)

cat file1 file2 >file3 (creates file3 and puts contents of file1 and 2 into it)

mkdir x (make directory named x)

rmdir x (removes directory named x)

rm f (removes files named f)

more (display files page by page)

date (display current date)

id (display information about current user)

# (comment 🡪 comp will ignore anything after it

man (get info from Unix manual 🡪 man x (output first manual pg found that is named x)

man –k yy (output the names of pgs whose terse description contains yy

script (record a terminal session in a file named \_\_\_) 🡪 script yyy names the file yyy

mkdir –P x/y/z

path x/y/z/z (last z is a file created) – real application is to update a file (time)

rm –r r (erases everything (files + dir); -r is recursive cmd; r at end is name of top lvl dir

ls –R (recursive listing); how to go to top of tree structure and look down

ls –a (or –A) see hidden files 🡪 ls –A (more useful because it suppresses the . and ..)

-C (forces to list everything out in a row/line)

-1 (forces everything to be listed own)

ls | more (pager.. can scroll though pages)

who (who else is online)

id (who am i)

wc (word count 🡪 count lines (wc -l), words (wc -w), and bytes (wc -c). default is –lwc

ls | more (lists down in a column and you can scroll down using space bar)

ls –C | more (lists down in rows and columns and you can scroll down using space) 🡪-C flag forces multiple columns

ls -1 (lists down in a column all the way to the bottom and allows you to enter next cmd)

rmbs < typescript > typescript1

sftp (secure file transfer protocol) 🡪 sftp [user@]host connect to host as user

sftp cmds:

ls, pwd, mkdir, cd operate on remote system

use 1 prefix to each cmd to operate on the local system (eg. 1pwd)

get x (transfers file x from current directory on the remote system to the current dir on local system)

put x (transfers file x from current dir on local system to current dir on remote system)

put – transfer file x from the current dir on the local system to the current dir on the remote system

get – t/f file x from the current dir on the remote system to the current dir on the local system

If I am on Linux; want to copy script from current dir to hills; placing it in ~ (use put)

$ sftp hills.ccsf.edu’

Pw

Sftp> ls (remote) vs lls (local)

Sftp> put typescript

Msg

Sftp> pwd (remote) vs lpwd (local)

If on hills; T/F file to linux

$sftp IPaddress

Pw

Sftp>pwd

Sftp> (from dir your in)

Sftp> put (filename)

scp (secure copy program. Execute a single copy cmd between two hosts

scp xxx [user@]hostx:[path] copy xxx from the current host to hostx, logging in as user, and placing the result at path. If user is missing, the current login is used. If path is missing, the user’s home dir is used. \*\*\*\*

$scp

Copy local 🡪 remote

$scp (path local) user@host:(path remote – relative to home dir)

Eg. scp typescript hills.ccsf.edu:typescript1

Copy from remote 🡪 local

$scp user@host:(path remote) (path local)

If copying a dir: (from local to remote)

$scp –r (path local) user@host:(path remote)

Eg. scp –r (dir) (remote):dir

If copying a dir: from remote to local

$scp –r user@host:(path remote) (path local)