CS246—Linux Command Summary

Fall 2016

Commands

cxit log ont passwd change your password clear clear screen clear clear screen man command show the manual page for command man -k word show a list of man pages that mention word show a list of man pages that mention word man -k word show a list of man pages that mention word show a list of man pages that mention word manual starting with c mention word ment	Command	Meaning	Options
clear screen	exit	log out	
clear screen	passwd	change your password	
Show all previously-issued commands Show all previously-issued commands Show all previously-issued command Show all previously-issued command Show and the part Show and the p		clear screen	
show all previously-issued commands !! execute most recently-issued command le execute most recently-issued command starting with c whoami display your login name date display current date and time pwd display current directory ls list contents of current directory ls ls -a show all files, including hidden files ls -l show in long format cp- dirl dir2 recursively copy dirl to dir2 cp- dirl dir2 recursively copy dirl to dir2 can be used to recursively remove a directory, if -r option is used can be used to create an empty file if file does not exist rent time cd dir change directory to dir create new directory dir in current directory mkdir dir create new directory dir in current directory rectory rmdit dir remove directory dir conly works if dir is empty; if not empty, use rm -r dir; can specify more than directory at once echo string display string to screen ching change personal info (name, address, etc.) on Unix system ching change personal info (name, address, etc.) on Unix system ching display current processes ps -A show ALL processes (incl. system processes) kill pid kill process with number pid show amount of time taken executing command fig bring background job to the fore- ground find dir-name "pattern" find all files whose names match pat-			
c	history	show all previously-issued commands	THOMES WAY TO
whoami display your login name date display current directory dir nume is a show all files, including hidden files is -a show all to dir2 recursively remove a directory and portion is used can be used to recursively remove a directory if -roption is used can be used to recursively remove a directory can specify more than one directory at once recurs and specify more than one directory at once and specify more than directory at once and specif	!!	· ·	
date pwd display current date and time pwd display current directory list contents of current directory move file to file2 (also use to remame) rm file remove file remove file can be used to recursively copy dir1 to dir2 rent time can be used to recursively remove a directory, if roption is used can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist rectory mkdir dir create new directory dir in current directory can specify more than one directory at once echo string display string to screen set permissions on file to perms change personal info (name, address, etc.) on Unix system change personal info (name, address, etc.) on Unix system change personal info (name, address, etc.) on Unix system show who is logged into this machine finger username show who is logged into this machine finger username show amount of time taken executing command show amount of time taken executing command find dir-name "pattern" find all files whose names match put-	!c		
Description Section Description Desc	whoami	display your login name	
list contents of current directory cp file1 file2 copy file1 to file2 move file1 to file2 (also use to rename) rm file remove file touch file update file's last modified time to current time change directory to dir create new directory dir remove directory dir can be used to recursively remove a directory, if -r option is used can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist rent time can specify more than one directory can specify more than one directory at once cecho string chomod perms file change personal info (name, address, etc.) on Unix system chsh change your login shell ps display current processes ps - A show all users' processes ps - A show ALL processes (incl. system processes) kill pid kill process with number pid show who is logged into this machine finger username show amount of time taken executing command fing bring background job to the foreground find dir -name "pattern" list a show all files, including hidden files show in long format cp - r dir1 dir2 recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory on the used to recursively remove a directory ont option is used can be used to recursively remove a directory on the used to recurs	date	display current date and time	
list contents of current directory cp file1 file2 copy file1 to file2 move file1 to file2 (also use to rename) rm file remove file touch file update file's last modified time to current time change directory to dir create new directory dir remove directory dir can be used to recursively remove a directory, if -r option is used can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist rent time can specify more than one directory can specify more than one directory at once cecho string chomod perms file change personal info (name, address, etc.) on Unix system chsh change your login shell ps display current processes ps - A show all users' processes ps - A show ALL processes (incl. system processes) kill pid kill process with number pid show who is logged into this machine finger username show amount of time taken executing command fing bring background job to the foreground find dir -name "pattern" list a show all files, including hidden files show in long format cp - r dir1 dir2 recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory, if -r option is used can be used to recursively remove a directory on the used to recursively remove a directory ont option is used can be used to recursively remove a directory on the used to recurs	pwd	display current directory	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			
rm file remove file can be used to recursively remove a directory, if -r option is used to return to most recent time can be used to create an empty file if file does not exist rent time change directory to dir can be used to create an empty file if file does not exist rent time can be used to create an empty file if file does not exist can be useful if you accidentally ran vi or emacs with an & ground find dir -name "pattern" find all files whose names match pat-	cp file1 file2	copy file1 to file2	cp -r dir1 dir2 recursively copy dir1 to dir2
touch file update file's last modified time to current time cd dir change directory to dir create new directory dir in current directory mkdir dir remove directory dir only works if dir is empty; if not empty, use rm -r dir; can specify more than directory at once echo string change personal info (name, address, etc.) on Unix system chsh change your login shell ps display current processes who show who is logged into this machine finger username time command show amount of time taken executing command find dir -name "pattern" find all files whose names match pat-	mv file1 file2	* *	
rent time cd dir change directory to dir create new directory dir in current directory mkdir dir create new directory dir in current directory rmdir dir remove directory dir only works if dir is empty; if not empty, use rm -r dir ; can specify more than directory at once echo $string$ display $string$ to screen chmod $perms$ $file$ set permissions on $file$ to $perms$ chfin change personal info (name, address, etc.) on Unix system chsh change your login shell ps display current processes $ps - a$ show all users' processes $ps - A$ show ALL processes (incl. system processes) kill pid kill process with number pid kill -9 pid more forceful kill, for stubborn processes who show who is logged into this machine finger $username$ show amount of time taken executing $command$ for $command$ show amount of time taken executing $command$ find dir -name " $pattern$ " find all files whose names match pat - find dir -name " $pattern$ " find all files whose names match pat -	rm file	remove file	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	touch file	- *	can be used to create an empty file if file does not exist
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	cd dir	change directory to dir	cd - return to most recently visited directory
rmdir dir remove directory dir only works if dir is empty; if not empty, use rm -r dir ; can specify more than directory at once echo $string$ display $string$ to screen chmod $perms$ file set permissions on file to $perms$ change personal info (name, address, etc.) on Unix system chsh change your login shell ps display current processes ps -a show all users' processes ps -A show ALL processes (incl. system processes) kill pid kill process with number pid kill -9 pid more forceful kill, for stubborn processes who show who is logged into this machine finger $username$ show amount of time taken executing $command$ fg bring background job to the foreground find dir -name " $pattern$ " find all files whose names match pat -	mkdir dir	· ·	can specify more than one directory at once
chmod perms file change personal info (name, address, etc.) on Unix system chsh change your login shell ps display current processes ps - A show ALL processes (incl. system processes) kill pid kill process with number pid kill -9 pid more forceful kill, for stubborn processes who show who is logged into this machine finger username time command show amount of time taken executing command fg bring background job to the foreground find dir -name "pattern" find all files whose names match pat-	rmdir dir	remove directory dir	
chfn change personal info (name, address, etc.) on Unix system chsh change your login shell ps display current processes ps -a show all users' processes kill pid kill process with number pid kill -9 pid more forceful kill, for stubborn processes who show who is logged into this machine finger username show personal info for username time command show amount of time taken executing command fg bring background job to the foreground find dir -name "pattern" find all files whose names match pat-	echo string	display string to screen	
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	chmod perms file	set permissions on file to perms	
ps display current processes ps -a show all users' processes ps -A show ALL processes (incl. system processes) kill pid kill process with number pid kill -9 pid more forceful kill, for stubborn processes who show who is logged into this machine finger username show personal info for username time command show amount of time taken executing command fg bring background job to the fore- ground find dir -name "pattern" find all files whose names match pat-	chfn		
kill pid kill process with number pid kill -9 pid more forceful kill, for stubborn processes who show who is logged into this machine finger username show personal info for username time command show amount of time taken executing command fg bring background job to the foreground find dir -name "pattern" find all files whose names match pat-	chsh	change your login shell	
kill pidkill process with number pidkill -9 pidmore forceful kill, for stubborn processeswhoshow who is logged into this machinefinger usernameshow personal info for usernametime commandshow amount of time taken executing commandfgbring background job to the foregrounduseful if you accidentally ran vi or emacs with an & groundfind dir -name "pattern"find all files whose names match pat-	ps	display current processes	
who show who is logged into this machine finger username show personal info for username time command show amount of time taken executing command bring background job to the foreground find dir -name "pattern" find all files whose names match pat-	kill pid	kill process with number pid	kill -9 pid more forceful kill, for stubborn processes
time command show amount of time taken executing command fg bring background job to the foreground find dir -name "pattern" find all files whose names match pat-	who	show who is logged into this machine	
show amount of time taken executing command fg bring background job to the foreground find dir -name "pattern" find all files whose names match pat-	finger username		
ground find dir -name "pattern" find all files whose names match pat-	time command show amount of time taken executing		
find dir -name "pattern" find all files whose names match pat-	fg	bring background job to the fore-	useful if you accidentally ran vi or emacs with an &
	find dir -name "pattern"	find all files whose names match pat-	

Tools

Tool	Purpose	Options
cat <i>f1 f2</i>	display files $f1, f2, \dots$ one after the	cat -n f1 f2 attaches line numbers
	other	
more file	display file one screen at a time	
$\operatorname{diff} f1 \ f2$	compare files $f1$ and $f2$; outputs in-	diff -w f1 f2 ignores whitespace
	structions for converting $f1$ to $f2$	
cmp <i>f1 f2</i>	compare files $f1$ and $f2$; outputs the	
	first position where they differ	
wc file	count the number of words, lines, and	wc -c file show just the number of characters
	characters in file	wc -l file show just the number of lines
		wc -w file show just the number of words
egrep pat file	print all lines in <i>file</i> that contain pat-	egrep -n pat file print matching lines with line numbers
	tern pat	egrep -v pat file print lines that do not match pat
head file	print first 10 lines of file	-num prints num lines (e.g. head -5 file)
tail file	like head, but prints last 10 lines of	
	file	
sort file	sorts the lines of file	sort -n file sorts strings of digits in numerical order
uniq file	removes consecutive duplicate lines	removes all duplicates if file is sorted
	from file	

Programs

Program	Purpose	Options
vi file	invoke vi text editor on file	
emacs file	invoke emacs text editor on file	
pico file	invoke pico text editor on file	
pine (or alpine)	read email	
wget url	fetch file from the web at <i>url</i>	
xpdf file	display pdf file	
lpr file	print file to printer	lpr -Pljp_3016 file specifies the printers in
		MC3016
lpq	checks the print queue	
lprm jobno	removes job jobno (must belong	
	to you!) from print queue	
ssh machine	make SSH connection to ma-	ssh -Y (or -X) machine enable X forwarding
	chine; opens a secure shell on re-	(must have X server running on local machine)
	mote machine; type exit to end	
	SSH connection	
scp mach1:file1 mach2:file2	securely copy file1 on mach1 to	can omit <i>mach1</i> if it is the local machine; similarly
	file2 on mach2	for mach2

Variables

Variable	Meaning
\${PWD}	present working directory (equivalent to executing pwd)
\${HOME}	your home directory (equivalent to \sim)
\${SHELL}	your default shell
\${PRINTER}	your default printer
\${PATH}	your default search path for commands
\${\$}	current script's process ID
\${0}	name of currently-running script
$\$\{1\}, \$\{2\}, \dots$	arguments 1, 2, of current script/function
\${#}	number of args supplied to currently-running script/function (not including script name)
\${@}	all args supplied to currently-running script/function as separate strings (not including script name)
\${?}	return code of most recently-executed command/function

Permissions

Symbol	Meaning
u	file's owner
g	members of the file's group, other than the owner
0	other users
a	all users (equivalent to ugo)
+	add permission bit
_	revoke permission bit
=	set permission bits exactly
r	read permission. for files—file's contents can be read. for directories—directory's contents can be listed
W	write permission. for files—file's contents can be modified. for directories—files can be added/renamed/removed in the directory
X	execute permission. for files—file may be executed as a program or script. for directories—directory can be traversed (i.e. can cd into the directory)

Script Conditional Operators

Operator	Meaning
=	string equality
!=	string inequality
-eq	integer equality
-ne	integer inequality
-gt	integer greater than
-ge	integer greater than or equal to
-lt	integer less than
-le	integer less than or equal to
-a	and
-O	or
!	not
\(, \)	parentheses for grouping
-d	file exists and is a directory
-e	file exists
-f	file exists and is a regular file
-r	file exists and is readable
-W	file exists and is writable
-X	file exists and is executable

Globbing Patterns

Operator	Meaning
*	matches 0 or more characters
?	matches one character
[abxy]	matches exactly one of the characters in brackets
[!abxy]	matches any character except the ones in the brackets
[a-z]	matches any character in the given range
{pat1,pat2}	matches either pat1 or pat2 (technically not a glob; note no spaces)

Directories

Directory	Meaning
	Current directory
	Parent of current directory
~	Your home directory
/	Root directory
Starts with / or ~	Absolute path
Does not start with / or ~	Relative path