

Each of the commands included here must be entered at a UNIX prompt: **bash-3.2\$**

Changing Your HP UNIX Password

passwd

begins the Password Change Sequence:

1. Enter the **passwd** command.
2. The system responds:

Changing password for <your login ID>

and prompts for the

Old password:

3. Enter your current password (to confirm your identity).

For reasons of security, the password will not display on the screen. Don't be alarmed. The characters you enter are sent to the server.

If you mistype the password entry, you'll have to start over again.

Backspacing does not delete mistyped characters; it adds ASCII backspace characters to the line of entry.

4. The prompt **New password:** appears.

Please refer to *Rules and Suggestions for Passwords at CCSF* below.

The system will end the password change sequence if you take too much time, so decide upon your new password *before* you begin. Refer to *Rules and Suggestions* below.

5. The prompt **Re-enter new password:** appears.

Enter the new password a second time for confirmation.

6. If you get an error message such as **New password too short**, the password change sequence returns you to Step 4 and gives you another opportunity to enter a new password.

7. If you make repeated mistakes, the system will end the password sequence with the message:
Too many failures — try later.

CONTENTS: COMMAND SUMMARY

UNIX commands in this handout explore:

- The Optional Password Change.....page 1
- File Commands.....page 2
- Printing Commands.....pages 3-4
- Account Information.....page 5
- Script Command.....page 5
- Scripting a Session.....page 6
- Help Command.....page 6

Rules and Suggestions for Passwords at CCSF

Rules:

1. Passwords must be at least 6 characters (and no more than 12)
2. Passwords must contain at least 2 alpha and at least one numeric character.
3. *Don't use spaces or characters that carry special meaning in UNIX, like * ? / \ !*
4. Remember that UNIX is case sensitive. Uppercase letters may be used, but keep in mind that any uppercase letter(s) you use in a password must be entered in upper case *at every login*.

Suggestions:

1. Don't use obvious passwords like your first name; even words found in the dictionary are too predictable.
2. Think of a phrase that's easy to remember (or a favorite song title or lyric) and use the first letter of each of the words (and at least one number).
3. Example: From the sentence: "I don't want to do this one," you might create the password: idwtdt1

Using Files

Note: in the following sections, *<fn>* stands for the name of one of your saved files.

Displaying the Names of Saved Files

ls displays the names of files in the current directory that are not hidden (uses lowercase letter L)

ls -l or **ll** displays detailed information about non-hidden files in the current directory (All uses of lowercase letter L not number 1)

- Column 1: the file type and the permissions (read, write, and/or execute for owner, group, and other users)
- Column 2: the size of the file in little links
- Column 3: the login ID of the owner of the file
- Column 4: the group to which the owner of the file belongs
- Column 5: the size of the file in bytes
- Column 6: the date of the last modification
- Column 7: the time of the last modification
- Column 8: the filename

ls -a displays the names of all the files in the current directory (including hidden files)

Displaying the Contents of a Text File

cat <fn> displays the file named *<fn>* on the screen

Example: To see what is in the file called *prog1*, enter: **cat prog1**

Use *cat* on short files containing ordinary text created with a simple text editor like *pico*.

Note: if you try to display a compiled program you'll get strange characters on the screen and the responses on your terminal screen might be slow and unpredictable.

more <fn> displays the contents of the file *<fn>* one screen (about 20 lines) at a time

When the *more* command is in effect:

- Tap the *Spacebar* to advance from one screen of lines to the next.
- Press the *Enter* key to advance one line at a time.
- Tap the *q* key to quit before all the lines in the file have been displayed.

less <fn> allows scrolling both **up** and **down** through the file

Use the *Up* and *Down Arrow* keys (*w* and *y*) to scroll backward and forward one screen at a time.

Copying a File

cp -i <fn> <newfn> copies the contents of existing file *<fn>* into a new file *<newfn>*

Example: To copy the contents of the file called *oldfile* to a file called *newfile*, enter: **cp -i oldfile newfile**

This version of the copy command (with the *-i* option) will warn you if a file named ***newfile*** already exists and will ask if you want to overwrite the current contents of ***newfile*** before the copy process begins.

If you reply *y* for *yes*, the current contents of ***newfile*** will be replaced with the contents of ***oldfile***.

A response of *n* for *no* aborts the copy process so that nothing happens.

Adding the Contents of One File to the End of Another (Appending)

cat file1 >> file2 adds the contents of *file1* to the end of *file2*

This command does not change or remove the contents of ***file1***. For use only with text files, not with compiled code

Deleting a File

rm <fn> deletes a file and frees the disk space it occupied

Example: To require deletion confirmation (*y* or *n* for *yes* or *no*), use option *-i* and enter: **rm -i myfile**

Printing

Using the UNIX Printers in the ACRC

The UNIX printers in the ACRC are text printers exclusively.

NEVER print or attempt to print:

- any part of a log-in session using the *FILE* menu *PRINT* option
- compiled code (like an a.out file or a [filename].class file)
- a script file with a *pico*, *vi*, or *pine* session in it
- a formatted text file (like a document in MS Word)
- a binary file (like an image, a graphic, or a browser page)
- a GUI interface (like printing from a Linux station)

lp

The names of the UNIX printers in the lab are *iclpr1* and *iclpr2*

If you want to print to one of the UNIX printers, you have to specify one of these names as the destination of your print job, using the Unix command *lp* with the option *-d* (for destination) followed by the name of the printer you want to use. If you want to print a file called *UnixFile1* on printer *iclpr1*, enter

lp -d iclpr1 UnixFile1

Note that the space used after *-d* is optional; the other spaces in the command line, after *lp* and after the name of the printer, are required.

pr

The *lp* command prints the file without page breaks or interpretation of tab characters. In order to print page breaks and tabs, the *lp* command must be piped through the *pr* command with an *-e* option. Enter

pr -e UnixFile1 | lp -d iclpr2

expand

To avoid lines printing beyond the righthand perforated edge of the paper, use *expand* to expand tabs, fold (or wrap) long lines, and paginate.

expand Unixfile1 | fold | pr | lp -d iclpr1

You might prefer the next command: it makes the text more readable by expanding tabs to only four characters (instead of the default eight characters) and by folding (or wrapping) the line on spaces between words instead of in the middle of words.

expand -t 4 Unixfile1 | fold -s | pr | lp -d iclpr2

*Printing Long Jobs on Unix Printers***wc**

Unix printing in the ACRC is limited to 9 pages per job.

If you want to check the number of lines in the file to be printed, use the command *wc* with the line option:

wc -l UnixFile1

If the number of lines is greater than 594 (9 pages of 66 lines), then the file must be printed in smaller parts.

head

If the file is 750 lines long, for instance, you won't be able to print the file in one job in the ACRC. However, you can print the first 500 lines using the *head* command if you enter

head -500 UnixFile1 | lp -d iclpr1

tail

Then, you can print the last 250 lines using the *tail* command. Enter

tail -250 UnixFile1 | lp -d iclpr2

Using the Laser Printers


ASCII text files created in your HP Unix account may be printed on laser printers *Pay1*, *Pay2*, or *Pay3* after the files have been downloaded to your Windows home directory on H: drive.


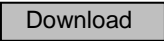

To Download the File:



Double-click the yellow *SSH Secure Shell Client for FTP* icon on the Windows Desktop.

Refer to the handout *Using SSH Secure Shell Client for FTP* to familiarize yourself with the details of its use.

- Login to your HP UNIX account. 

(Click the *Quick Connect* button or select the *Quick Connect* option from the *File* menu.)
- Your UNIX home directory should appear on the right (remote) side of the window. Move to the directory that has the file you want to download. Highlight the name of the file.
- If you just want to print a simple text file, double-click the filename you highlighted above to open it and then click on the *File* menu and select the *Print* command. Please note that this method of printing defaults to the *Notepad* application and does not expand tabs. If this method does not deliver the desired results, close the *Notepad* application, highlight the name of the file, and proceed to the next step to download the file.
- Click the  button to download the file.
- Go to your Windows home directory. (Select *H:* from the *Look in:* menu.) Find the destination path for the file.
- Click the *Download* button. 
- If the file is in (unformatted) ASCII text, rename the file so that it has a *.txt* extension.
- Click the *Disconnect* button  in the button bar below the menu, or select *Disconnect* from the *File* menu to log off your Unix account.
- Click the **X** in the red button in the upper right corner or select *Exit* from the *File* menu to close the *SSH Secure Shell Client FTP* window.

To Print the Downloaded File:

Double-click the *Computer* icon on the Windows Desktop . Open your Windows Intranet home directory on the H: drive shown in the *Network Location* section.

- Open the file you want to laser print.

Remember, unformatted text files must be renamed with a .txt extension.
- Select the *Print* option from the *File* menu.

Review the document in the *Print Preview* option to confirm the appearance of the document.

Click the *Print* icon in the upper left corner to print.

In the *Print* window, specify the page(s) you want to print, if not *All*.

Click the *OK* button or strike the *ENTER* key to accept the selections.
- When the blue *Print Control For The Desktop* window appears:

We recommend that you type your login ID or a simple sequence of numbers/letters in the *Client ID* box. The *Client ID* does not have to be your login or user id, but your own unique login will not be confused with anyone else's name or nickname. Be sure to **remember** what you enter for your *Client ID* because you'll have to enter it in the *User ID* box at the Print Release Station.

Note **the number of pages** and the **total cost of printing** for this job so you'll know how much money you'll need on your print card.

Click the *Print* button.

Please refer to the *Laser Printing from Windows* handout for further assistance with printing, paying for, and retrieving laser output.

Finding Out About Your Account

Check Your Disk Quotas

The space that you are given to save files is limited, and so is the number of files that you are allowed to save. In Fall of 2008, the respective limits were set to 200,000 kilobytes and 20,000 files (subject to change). If you have been using this HP UNIX account for some time, it's good practice to use the `quota -v` command regularly to check how close you are to your limits. Delete files if it looks as though you might fill your disk space soon or if you'll need the space to accommodate large files.

quota -v displays current file usage and limits

Displays the following format (Your *usage* and *files* numbers will vary.):

File system	usage	quota	limit	timeleft	files	quota	limit	timeleft
/students	223	200000	200500		32	20000	20000	

Columns 2 and 3 show how many kilobytes you have stored and how many you are allowed. If you have exceeded either of your limits, the column labeled "timeleft" will indicate how long you have to delete files before the system locks your account.

The last 4 columns display similar statistics for the number of files you have saved.

Both numbers will be higher than you expect because there are hidden files that are necessary for UNIX to configure programs like *mail* and your text editor. Don't delete these files or your UNIX account might not work reliably.

Scripting a Session

script <fn> copies the keystrokes you enter and the responses displayed

The `script` command captures whatever comes up on the screen between the time that you type in the `script` command and the time that you stop the process with `exit`. Script sessions can include capturing the interactive input and the screen responses of UNIX commands or of programs that display output to the screen.

The script session captured is written to the file you name in the `script` command or to a file called `typescript` if you choose not to customize the filename. `Script` command does not append to the end of an existing file with contents, it replaces it, so don't name the script with the name of a file you are still using.

Scripting procedure:

- Enter the `script` command, followed by a space and the name of the file to which you want to save the visual recording of the session.
- Start whatever you want to copy (a common example would be to display a program's source code onscreen with the `cat` command, and then compile and run the program).
- When you have finished, stop the scripting process with the `exit` command.
- You can then use the `lp` command to print the file which holds the results of the `script` command.

Warning: The `script` command does not work the same way when you are running multiple processes, especially spawned shells.

ps -u <your login id> shows all processes currently in progress.

PID	TTY	TIME	COMMAND
12158	ttyp5	0:00	bash
28524	ttyp4	0:00	script
29275	pts/28	0:00	bash
28525	ttyp4	0:00	bash
12157	pts/28	0:00	script
12161	ttyp5	0:00	ps

Enter an `exit` command at the UNIX prompt to discontinue each of the `script` commands in progress. The last script started is the first script discontinued.

Sample Script Command Procedure (Using a C++ Program)

script *hwk1*

Starts copying what will appear on the screen to a file named *hwk1* in your current directory.

NEVER name the script with the same name as the C++ program.
The script file immediately destroys the program by writing over it.

cat *prg1.cpp*

For java programs, use:

cat *prg1.java*

Displays the contents of the C++ source file *prg1.cpp* on the screen.

In this example, the script process records everything that appears on the screen to the file named *hwk1*.

When **script** command is in progress, commands like *pine* or editors like *pico* or *vi* are prohibited because the printers cannot interpret the changes these commands create in screen displays.

NEVER **cat** the script file while it is in progress.
It will duplicate itself in the script in progress and never find the end of the file.
Use *Ctrl/C* to stop such a file from listing endlessly.

aCC *prg1.cpp*

For java programs, use:

javac *prg1.java*

Compiles the source code in *prg1.cpp*

There are different arguments that your instructor might recommend.

Any error messages displayed on the screen would also be copied to the *hwk1* file by the script process.

a.out

For java programs, use:

java *prg1*

Runs the compiled C++ code. The *script* process captures output displayed on the screen to the *hwk1* file.

NEVER **cat** an *a.out* file. Object code is not a legible representation of your program.

exit

Ends the script (copying) process.

lp -d iclpr2 *hwk1*

Sends *hwk1* — which is everything the *script* command captured — to one of the UNIX printers.

Getting Help Online with a Unix Command

man *<cmd>*

You can read the pages of the UNIX manual on your screen by using the *man* command followed by a space and the name of the UNIX command for which you want information. Entries are often long and contain many options that you might not be interested in, but you can get a lot of information about the command, its options, and the system environment in this way. The manual pages are displayed like the *more* command, so you can:

- see the next page by pressing the *spacebar*
- scroll up and down the manual page by piping it with the *less* command: **man *<cmd>* | less**
- quit the *manual* command in progress and get back to the UNIX prompt by pressing *q*

man -k *<cmd>*

If you just want to find out whether there is a manual entry for a particular command, or if you want to see whether you have the correct spelling of a command, use this form of *man*. It gives a one-line response if the manual page for the command exists on the system, or an error message if it doesn't.