Fundamental UNIX Commands

SYNOPSIS

This section describes fundamental concepts and commands for using the UNIX operating system.

INTRODUCTION

The most distinguishing characteristic of the UNIX operating system is the file system structure. The file system structure resembles an inverted tree, with the user's files as leaves along the bottom and the root at the top. The user's home directory (the directory you are placed in when you log in) is placed at a specific point in the structure but the user is not limited to that directory. It is possible to move around in the tree - up, down, and sideways - into directories belonging to other users and/or the system. Users can protect their files and directories from the prying eyes of others by changing access permissions.

Each file and directory has a path name which uniquely identifies it. The path is described, starting with the root, down through the branches of the tree, to the directory containing the file. Filenames must be unique within a directory. On Central UNIX, this tree structure is spread over all of the sites that exist in the cluster

DIRECTORY MANIPULATION COMMANDS

cd path<CR>

The change directory command moves the user from the current working directory to the directory specified. If ***path*** is defined as a simple name such as ***docs***, then ***docs*** is located directly below the current directory and the user is moved into it. The command ***cd***, by itself on a line, returns the user to the login directory. Examples of uses of ***cd*** are

cd docs<CR>	Move down to the directory named ***docs***.
cd ..<CR>	Move up to the parent directory of the current directory.
cd ../data<CR>	Move up to the parent directory of the current directory, then down to the directory named ***data***.
cd /usr/local/bin<CR>	Move to the explicit directory location.
cd ~login_ID<CR>	Move to the home directory of the user specified by ***login_ID*** (C Shell only).

ls<CR>

Lists the contents of the current directory. ***ls*** has many parameters. You will probably find the following to be the most useful:

ls -a	Lists all files, including invisible files (files with a leading dot (.)).
ls -l	Lists all visible files and some attributes.
ls -F	Places a slash (/) after directory files and an asterisk (*) after executable files, and places an at-sign (@) after symbolic links.

ls -R	Recursively lists all files, including those in sub-directories from the current sub-directory to all sub-directories below.
ls -s	Lists all visible files and their file size in blocks.
ls -slagF	Command for a full directory listing (all attributes).

mkdir directory<CR>
This command is used to create a new directory within the current directory.

pwd<CR>
The ***pwd*** command (print working directory) allows you to determine the path name of the directory in which you are presently working.

rmdir directory<CR>
The ***rmdir*** (remove directory) command is used to delete a directory. You can only delete an empty directory.

pilot<CR>
The "***pilot*** " command allows you to browse your files in a manner very similar to the way that "pine" browses files. "***pilot*** " is only available on systems that have pine installed on them. "***pilot*** " will allow a user to list, delete, edit, and view files as well as move in and out of directories.

CHANGING FILE ACCESS PERMISSIONS

Access permissions fall into two categories, base permissions that exist on most UNIX systems, and extended permissions, available on most POSIX based systems such as AIX. This section addresses those forms for setting permissions.

NOTE: Systems using AFS (Andrew File System) and DFS (Distributed File System) *DO NOT* use base permissions except for the user level only, group and other permissions must be set through extended permissions.

A. BASE PERMISSIONS (ALL SYSTEMS)

Base permissions are set via the "***chmod*** " command as in the following examples:

chmod
Allows the owner of a file or directory to change the access permissions. Use of pneumonics is preferred over the octal numbers as pneumonics are additive and subtractive, octal numbers are absolute and may change something that you didn't want to.

chmod go-rwx filename<CR>
chmod 600 filename<CR> Makes a file private.

chmod a+r filename<CR>
chmod 644 filename<CR> Makes a file public in read mode.

chmod go-rwx,u-wx+r filename<CR>
chmod 400 filename<CR> Makes a file private and protects it from accidental change/deletion.

chmod u+x filename<CR>
chmod 700 filename<CR> Makes a file executable by the owner. This file can then be executed by simply entering its name.

chmod a+rx dirname<CR>
chmod 755 dirname<CR> Makes a directory public in read mode. Note: Directories must be "executable" in order to be searched.

B. EXTENDED PERMISSIONS (AIX FILESYSTEMS ONLY)

Extended permissions are set via Access Control Lists and the three commands that manipulate them. Extended permissions allow access to be controlled to the user or group level within the UNIX accounting structure. These commands are "***aclget*** ", "***aclput*** ", and "***acledit*** " and are described below. A fourth command developed at Cal Poly "***aclmod*** " is also described.

NOTE: While the examples given indicate filenames, these commands may also be used on directories as well.

1. ACLGET

"***aclget*** " is used to obtain the permissions of a file or directory and can be used as follows:

aclget -o outfile filename<CR>
Writes the file permissions for "***filename*** " out to the text file "***outfile*** " where it may be viewed or edited (see "ACLEDIT").

aclget filename > outfile<CR>
Same effect as the previous command.

2. ACLPUT

"***aclput*** " is used to apply a set of base and extended permissions contained in a text file to a specified file or directory. Some sample uses are as follows:

aclput filename < infile<CR>
aclput -i infile filename<CR>
Appies the base and extended permissions contained in the text file "***infile*** " to the file "***filename*** ".

aclget filename1 | aclput filename2<CR>
Applies the base and extended permissions from "***filename1*** " to "***filename2*** ".

3. ACLEDIT

"***acledit*** " is used to edit the base and extended permissions of a specific file or directory. It requires that the environmental variable EDITOR be set to a valid text editor on the system. For example

setenv EDITOR /usr/local/bin/pico
for the C and T Shells

EDITOR=/usr/local/bin/pico;export EDITOR
for the Bourne and Korn Shells.

An example of the command invocation of "***acledit*** " is

acledit filename<CR>
Initiates editing of the base and extended permissions of the file or directory "***filename*** " with the editor specified in the environmental variable "EDITOR".

An example of what would show up on your editor's screen when you edit the Access Control List for the first time is:

```
attributes:
base permissions
    owner(juser):  rwx
    group(student):  r-x
    others:  r-x
extended permissions
    disabled
```

This indicates that the file is world readable and executable and owner writable. To edit base permissions, the user edits the owner, group and others lines by manipulating the rwx settings for each level. To enable extended permissions, the user edits the last line of the file and adds information to the file as follows:

```
attributes:
base permissions
    owner(juser): rwx
    group(student):  r-x
    others:  r-x
extended permissions
    enabled
    keyword  rwx     c:name [c:name ... c:name]
     ...
    keyword  rwx     c:name [c:name ... c:name]
```

where "keyword" is either "permit", "deny", or "specify"; "rwx" is a the logical addition of "r--" for read, "-w-" for write, and "--x" for execute ("r-x" would indicate read and execute permissions); " c" is either "u" for a individual user or "g" for an account group (account groups may be listed by entering the system command "ypcat group" at the system prompt); and "name" is either an individual's login id (if " c" is "u"), or an account group (if "c" is "g"). You can have multiple entries for each permission for a type of permission, separated from each other by a space as follows:

```
permit  r-x     u:juser u:jdoe g:faculty
```

You may also have a line for each permission type for each keyword, thus

```
permit  r-x     g:student g:faculty
deny    r--     u:juser
```

would result in the file being readable and executable by all students and faculty, but not readable by the user " juser", even if they are a student or a faculty member.

"permit" allows you to say what users and/or groups are granted a specified permission, "deny" allows you to say what users and/or groups are denied a specified permission, and "specify" allows you to specify a specific

user's and/or group's permissions despite previous user and/or group specifications.

NOTE: The system space for any given file or directory access list cannot exceed 4096 bytes. While the file size of the access control list output does not strictly correspond to this, it does provide a safe guideline.

4. ACLMOD

aclmod is a command written at Cal Poly which uses the above commands combined with an easy-to-use syntax for easier use. The format of the command is

% ***aclmod {+|-}{r|w|x} {p|d}:{u|g}:{loginid|groupid} File|Dir [File|Dir...] ...<CR>***

where multiple files and/or directories may be specified for each permission set and the whole group may be repeated for additional permission sets.

The various options and flags are:

+	Add the specified permission to the ACL for the file(s) and/or subdirectory(s).
-	Remove the existing permission from the ACL for the file(s) and/or subdirectory(s).
r	Read mode.
w	Write mode.
x	Execute (file) or passthrough mode (directory).
p	Permit the user or group.
d	Deny the user or group.
u	Defines the following ID as a user.
g	Defines the following ID as a group.
loginid	A login ID the permission is being assigned for.
groupid	The group ID the permission is being assigned for.
File	A file at the current location or including the relative or absolute path (multiple files and/or directories may be specified).
Dir	A subdirectory at the current location or including the relative or absolute path (multiple files and/or directories may be specified).

For example,

% ***aclmod +rx p:u:juser mydir -r p:u:jdoe fil1 file2<CR>***

would ***p***ermit the ***u***ser ***juser*** to read and pass

through the subdirectory ***mydir*** which is located in the current directory as well as remove the ***r***ead ***p***ermissions for the ***p***ermitted user ***jdoe*** from files ***file1*** and ***file2***.

FILE MANIPULATION COMMANDS

cat file1 [file2...]<CR>
This is an easy way to list one or more files to the screen.

cp file1 file2<CR>
The ***cp*** (copy) command takes the contents of one file (***file1***) and duplicates it to another file (***file2***). If ***file2*** already exists, it is deleted before ***file1*** is renamed.

grep pattern filename<CR>
The ***grep*** command searches one or more files for a pattern. For example, if the command line reads:

grep Abc myfile<CR>

grep would display each line in file myfile which contains an uppercase ***A*** followed by a lowercase ***b*** and ***c***. ***grep*** is very versatile and has many options. See the ***man*** pages for more detailed information.

head filename<CR>
head -nn filename<CR>
The first form displays the first 10 lines of the file. The second form displays the first "***nn***" lines of the file specified.

more filename<CR>
more lists a file to the screen. It pauses automatically every 23 lines to allow you to read a screen full at a time. Press the space bar for another screen -full, "***q*** " to quit and ***<CR>*** to advance one line. On Central UNIX you may also use the similar command ***pg***.

mv file1 file2<CR>
Move (***mv***) renames and moves files. If ***file2*** already exists, it is deleted before ***file1*** is renamed.

rm filename<CR>
The ***rm*** (remove) command is used to delete a file.

sort filename<CR>
Sorts a file in ascending order beginning with the first column of data. The ***sort*** command has many parameters and is very powerful and versatile. You will need to check the manual for more information.

tail filename***<CR>***
tail -nn filename<CR>
The first form causes the last 10 lines of the file to be displayed. The second form displays the last "***nn***" lines of the file.

PRINTING FILES

A. TO THE SYSTEM PRINTER ON CENTRAL UNIX

The user may print files on the system high-speed printer by use of the ***lp*** or ***lpr*** (AIX only) commands. The general format of the ***lp*** command is

lp -tii_boxnn filename<CR>

where "***ii***" is the user's first and last initials, "***nn***" is the user's selected box number for output distribution, and "***filename***" is the name of the file to be printed. **NOTE:** If the string ***ii_boxnn*** exceeds 8 characters, it will be truncated to the first 8 characters.

The general format of the ***lpr*** command is

lpr -J ii_boxnn filename<CR> (not on HP-UX).

where "***ii***" is the user's first and last initials, "***nn***" is the user's selected box number for output distribution, and "***filename***" is the name of the file to be printed. **NOTE:** If the string ***ii_boxnn*** exceeds 8 characters, it will be truncated to the first 8 characters.

B. PRINTING TO THE SYSTEM PRINTER ON THE SUN SYSTEM

The user may print files on the Sun system high-speed printer by use of the ***lpr*** command. The general format of the ***lpr*** command is

lpr filename<CR>

C. PRINTING POSTSCRIPT FILES TO LASER PRINTER FROM THE SUNS

The user may print files on laser printers from the Suns by use of the ***lpr*** command. The general form of the ***lpr*** command in this instance is

lpr -Pps filename<CR>

where "***filename***" is the name of a PostScript format file to be printed and "***ps*** " is the name of the printer.

D. TO THE TERMINAL PRINTER

The user may print files to the printer by use of either the ***cat*** command or the ***pr*** command. Please refer to the documentation for your terminal or terminal emulation software for more information on enabling a local printer. When the printer is enabled, the ***cat*** command will copy the specified file to the terminal and the printer. The ***pr*** command does the same as ***cat*** but adds page headers. In either case (***cat*** or ***pr***), the terminal printer should be correctly enabled immediately before the user types the ***<CR>*** on the command line. The printer should be released when the file has finished printing.

pr -lnn filename<CR>

Where "***nn*** " is the number of lines per page (e.g., 68 should be used for 8.5 inches at 8 lines per inch), and "***filename***" is the name of the file to be formatted for the terminal printer.

WARNING: The following command may cause your terminal or terminal emulation to hang if the printer is not configured properly, the software is not configured properly to support an ANSI vt100 (vt102) completely, or the software itself doesn't support an ANSI vt100 completely. An example of software which does this properly is the DOS version of Kermit. Please refer to your software documentation for further information.

To print a file to a terminal in one of the ITS open user terminal labs, use the command

% ***pcprint filename<CR>*** (not on HP-UX).

where "***filename*** " is the name of the file to be printed on the locally attached ANSI printer.

SITE MANIPULATION COMMANDS

Central UNIX allows users to interact with as many sites as there are in the cluster. The cluster comes equipped with several commands to assist in this. The following sub-sections describe the commands and uses.

A. YOUR HOME SITE VERSUS OTHER SITES

Your home directory (the directory that you are placed in when you log on) is where your files are stored. In most cases, the cluster will perform better when you are logged on to the site which contains that directory. To determine your home site, perform the following steps:

1. Log on to the cluster selecting any site.
2. At the system prompt, enter the command

   whereshome<CR>

   if you are logged onto your home site, no output will be received. If you are logged onto a site which is not your home site, the system will respond with the name of your home site.

B. OTHER SITE MANIPULATION COMMANDS (RISC/6000)

The "***on*** " command is no longer available on the RISC/6000 systems.

C. DETERMINING SITE AVAILABILITY

At times it may be necessary to remove one or more sites from the cluster due to either hardware and/or software maintenance. When this occurs, some of the files will become unavailable. It is not uncommon to see such errors as

`Command not found.` — If the command is a valid command that has been executed before, the site it is located on may be temporarily unavailable.

`Unable to change directory to "/u_site/login_ID", You are in "/" instead.` — Your home site or the site serving your files is temporarily unavailable. Until it is available again, your files are not available and you cannot save or write files.

UNIX EDITORS

There are many editors on Central UNIX. The following are recommended.

pico filename<CR>
This is a full screen editor which is also used with the " ***pine*** " mail program. See the chapter, "Using PICO" for more information.

vi filename<CR>
This is the full-screen editor. See the chapter, "vi - Full Screen Editor" for more information.

ONLINE HELP COMMANDS

man -k keyword<CR>
apropos keyword<CR> (not HP-UX)
Lists all UNIX commands or C functions with this keyword in their description. On HP-UX you must use "`man -k`" and not "`apropos`".

man command<CR>
On-line Unix manual. Enter "***man command<CR>***" at the terminal and the portion of the manual pertaining to that command will be printed on the screen. ***man csh*** describes the C Shell command environment in detail. ***man ksh*** describes the Korn Shell in detail.

USER INTERACTION COMMANDS

The Unix system allows you to interact with other users. The following commands facilitate user communication.

pine username<CR>
Allows you to send electronic mail to another user. (See the chapter for "Using PINE" for more information on this command.)

pine<CR>
Allows the user to read incoming mail messages. For more information, see the chapter, "Using PINE".

mailx username<CR> HP-UX
mail username<CR>
Allows you to send electronic mail to another user. (See the chapter for "Unix Mail" for more information on this command. PINE is recommended instead of mail for new users.)

mailx<CR> HP-UX
mail<CR>
Allows the user to read incoming mail messages. For more information, see the chapter, "Unix Mail". (PINE is recommended instead of mail for new users.)

ph real_name<CR>
Allows the user to lookup another user's email address based on their real name (see chapter "Using ph" for more information).

talk username[@site_name]<CR>
Allows users to communicate back and forth using a split screen. It is an upgraded version of "***write***". End the conversation with ***CTRL-C***. **NOTE:** The site name may be required when using ***talk*** to communicate with a user who is on another site.

users<CR>
Lists the usernames of everyone presently logged onto the cluster by site.

w<CR>
Lists the usernames, activity, and login information of everyone presently logged onto the current site.

who<CR>
Lists the usernames and login information of everyone presently logged onto the cluster.

write username[@site_name]<CR>
Allows users to communicate with each other while logged onto the cluster. End the conversation with ***CTRL-D***. **NOTE:** The site name may be required when using ***write*** to communicate with a user who is on another site.

COMBINING UNIX COMMANDS

UNIX allows the user a great deal of flexibility in nesting commands.

> Redirect standard output to a file.

cat file1 file2 > file3<CR>

This example concatenates two files and redirects the output to a third file (file3) instead of to the screen (default).

< Redirect standard input to a command

mailx freddie < filename<CR>

This example mails a file (filename) to user freddie.

>> Append to the end of a file.

cat tailfile >> oldfile<CR>

"***oldfile***" now contains its previous contents plus the contents of " ***tailfile***".

| Output from one command can be passed to another by means of a pipe, which is represented in the command-line as a vertical bar.

ls | sort > newfile<CR>

This example sorts the contents of a directory, then redirects output to a file.

SETTING UP YOUR ENVIRONMENT

new.dots<CR>

Updates your "dot" files (described below) to the current system defaults. Your old "dot" files are renamed to prevent the loss of custom information you have added to them. You can determine if your "dot" files are out of date by entering

% ***ls -la /usr/local/skel/.??*<CR>***

for the system default files and

% ***ls -la .??*<CR>***

in your home directory and comparing the dates on the listings.

.cshrc

You have a default ***.cshrc*** file in your account. "***cat .cshrc<CR>***" will allow you to view its contents. ***.cshrc*** is one of the files executed when you log onto UNIX under the C shell.

.exrc

You have a default ***.exrc*** file in your account. "***cat .exrc<CR>***" will allow you to view its contents. ***.exrc*** provides default options for the full screen editor, ***vi***.

.login

You have a default ***.login*** file in your account. "***cat .login<CR>***" will allow you to view its contents. When you log onto UNIX under the C shell, the commands in your ***.login*** file are executed immediately. This is a good place to store ***setenv*** commands.

.logout

You have a default ***.logout*** file in your account. "***cat .logout<CR>***" will allow you to view its contents. ***.logout*** is executed when you log off a UNIX system under the C shell.

.mailrc

You have a default ***.mailrc*** file in your account. "***cat .mailrc<CR>***" will allow you to view its contents. This file provides default options for the system mail utility. Mail aliases may be added to this file to allow you to customize your mail environment.

.pinerc

Pine creates this file the first time you run it. "***cat .pinerc<CR>***" will allows you to view its contents. This file provides default options for the "***pine***" program and may be changed via the "setup" option from the pine main menu.

.profile

You have a default ***.profile*** file in your account. "***cat .profile<CR>***" will allow you to view its contents. When you log onto UNIX under the Bourne shell, the commands in your ***.profile*** file are executed immediately. This is a good place to store commands which define your environment.

printenv

Displays the environmental variables which are currently set for C and T-shells.

env

Displays the environmental variables which are currently set for Bourne and Korn shells.

setenv TERM terminal-type<CR>

If TERM is not already defined when you log in, it is necessary to issue this command before you enter vi or pico, so that UNIX will know what kind of terminal you are using. Substitute the terminal-type of your terminal in the command above. This works for the C shell only. See ***TERM*** below for the Bourne shell implementation. Please refer to the section in the "UNIX Login/Logout Procedures" User Guide for a list of supported terminal types.

TERM=terminal-type; export TERM<CR>

Bourne shell version of ***setenv TERM***. See ***setenv*** above for more information.

MISCELLANEOUS COMMANDS

alias newname command<CR>
alias newname pathname<CR>

Allows you to create an easy-to-remember substitution for a complex command or a pathname. A good place for alias commands is in your .cshrc file which is executed every time you log in.

assist<CR>

Access on-line assistance files. We will be adding to this area on an on-going basis with how-to's and other on-line information. "***assist***" uses ***gopher*** as a tool to provide the information.

bugs<CR>

Access the bugs database for Central UNIX. This is a useful tool in trying to solve a problem. Always check the "bugs" database before mailing a problem to bugs. "***bugs***" uses ***gopher*** as a tool to access the database.

date<CR>

Displays the current date and time.

gopher<CR>

Accesses Cal Poly's Gopher services. An information retrieval system containing information about broad areas from computers to local activities.

Gopher is a information gathering program that works with other machines around the Internet.

logout<CR>
Log off the Unix system (C shell).

exit<CR>
Exit from the current shell. If the current shell is the login shell, you will be logged off the system.

finger login_name[@site_name]<CR>
Provides information about ***login_name*** such as the user's real name. If ***@site_name*** is specified, the site you are logged into will query the indicated site.

lynx<CR>
Allows the user to access the World Wide Web (WWW) via the ASCII terminal client ***lynx***. WWW documents are hyper text oriented documents that can also contain embedded objects. **NOTE:** ***lynx*** can only view text objects.

passwd<CR>
Allows the user to change your account password. Your password should be at least four characters in length, up to a maximum of eight. It is also desirable to include numbers and mixed case characters within the password for additional security. Develop a habit of changing your password on a regular basis

policy<CR>
Access the system, network, and computing policy files for Cal Poly. Users should make themselves familiar with these files and check them when new policies are announced in system news. "***policy***" uses ***gopher*** as a tool to access the system policy files.

sysaliases<CR>
Access a database of system aliases and their contents. A system alias may be used as an address for an electronic mail message. "***sysaliases***" uses ***gopher*** to access the system alias files.

sysnews<CR>
Read current system announcements. Displays a menu which provides several options. Among these options are: "***catchup*** " which makes the system think that the system news articles have all been read; "***startover*** " which make the system think that all of the articles are unread. The main option, " ***read*** ", presents a list of numbered system news article titles, preceding each unread article with a "+". The user then selects any article by entering its menu number from the "***read*** " sub-menu.

whatson<CR>
Provides a list of software applications available on the cluster. The ***whatson*** command uses the ***more*** command to display the information on the user's terminal.

CTRL-D
Indicates an end-of-file to the operating system. This is very useful with such system utilities as ***mail*** and ***cat*** where you may be required to indicate the end of a message or file. When entered outside of a system command, ***CTRL-D*** has the effect of indicating end-of-file to your shell and disconnecting you from the system.

CTRL-Q
Resume output to the terminal.

CTRL-S
Suspend output to the terminal.

MANAGING YOUR DISK SPACE

Every user is issued a disk space quota. You can keep track of the space your are using with these commands.

du<CR>
du -k<CR>
Displays the user's disk utilization for the current directory and all the directories below the current directory. When used in the home directory, ***du*** will give you the total of all the disk space in use by your account. On most systems the numbers indicate 1K of disk storage (1024 Bytes). The ***du*** command may not be accurate if you are logged into a system other than where your files are stored. When the ***-k*** parameter is used on some systems, it returns the utilization in 1K byte units instead of blocks.

quota -v<CR>
Displays the disk quotas across all sites in the cluster. The information provided is for both file space and inodes (numbers of files or links to files). For each type, the current value is displayed along with the soft and hard limits. Soft limits may be exceeded temporarily, hard limits may not be exceeded. If you have exceeded your quota, messages will appear for your home site with your UID. Messages without your UID for other sites should be ignored. **NOTE:** Quotas are displayed in 1K increments (2000 is equivalent to 2 megabytes), the in-use amount on some system are displayed in blocks.

If you find that you have exceeded your quota, and have a real need for additional disk space, you may obtain a Status Change form for Central UNIX from the Computer Account Clerk's office. Estimate the amount of space that you need for your academic projects; have it signed by your advisor (faculty sign their own forms, supervisors sign staff forms); then return the form to the Computer Account Clerk's office for processing.

JOB CONTROL (C SHELL AND T SHELL)

A. SUBMITTING AND MONITORING BATCHJOBS

Within the C and T shells, a user may suspend one or more jobs, bring them back to the foreground, and kill them. The letters job may be replaced with the job number obtained from the ***jobs*** command, a command name (if there is only one command of that name being run as a job), a plus (+) refers to the current job, a minus (-) refers to the previous job, and a second **%** is also a synonym for the current job.

batch < filename<CR> On AIX
batch filename<CR> On the Suns
Start a batch job when the systems loads permit. The job consists of a series of commands contained in the file ***filename***. Output from the job is mailed on Central UNIX unless output is redirected. On the Suns, output is mailed only if there is an error.

bg<CR>
bg %job<CR>
Puts the current or specified job into the background, continuing it if it was stopped.

command [parameters] &<CR>
Start a command and place the process into the background.

CTRL-C
Abort a current job.

CTRL-Z
Suspend current process and place in background as a stopped job. **NOTE:** This does NOT kill the process.

fg %job<CR>
Bring the specified job into the foreground, continuing it if it was stopped.

jobs<CR>
jobs -l<CR>
lists the active jobs, the ***-l*** option also displays process ID information.

kill %job<CR>
kill -9 %job<CR>
Sends the TERM (terminate) signal to the specified job. Use of the ***-9*** option causes a KILL signal to be sent to the job.

nice +nn command [parameters][&]<CR>
Run the command at a lower priority where "***nn***" may range from ***0*** for normal execution to ***20*** for only when the machine is idle. On HP-UX the range is from 0 for normal execution to 39 for obly when the machine is idle.

nohup command [parameters] &<CR>
Run the command in the background and continue execution even after the user has logged off the system.

notify<CR>
notify %job<CR>
Causes the shell to notify the user when the status of the current or specified job changes.

stop<CR>
stop %job<CR>
Stops the current or specified job which is executing in the background.

%job<CR>
Brings the specified job into the foreground.

%job &<CR>
Continues the specified job in the background.

B. KILLING UNWANTED PROCESSES

There may come a time when you have processes running on Central UNIX that you didn't exit properly. If you become disconnected from the cluster for some reason, there is a chance that the processes started by the previous sessions could continue after the disconnect. To remove these processes, the user should become familiar with the following commands:

ps<CR>

Displays information about the processes with your effective user ID on the current site.

ps -fu login_ID<CR>

Displays information about the processes running under your user ID on the current site. This provides more detailed information than a simple ***ps***.

killall<CR> (not on HP-UX).

Sends a signal 9 (SIGKILL) to all processes except the calling process.

killall -<CR> (not on HP-UX).

Sends a signal 15 (SIGTERM) to all processes except the calling process, waits 30 seconds, and then sends a signal 9.

kill 0<CR>

Sends a signal to all background processes with a real or effective user ID that matches the real or effective user ID of the sender.

kill -15 -1<CR>

Sends a signal to all processes with a real or effective user ID of the account you are in, even when those processes are on another site. Shells are not stopped.

kill -9 0<CR>

Stop all processes that you own and logout.

kill -9 -1<CR>

Stop all processes with a real or effective user ID of the account you are in and logout.

COMMAND SUMMARY

The following is a summary of commands and functions discussed within this User Guide.

acledit filename<CR>	AIX Only edit the Access Control List for the file or directory specified.
aclget filename<CR>	AIX Only display the Access Control List for the specified file or directory.
aclget filename > control_file<CR>	AIX Only store the Access Control List for the file or directory specified in the file "control_file ".
aclmod +rx p:u:www .. ../public_html +r p:u:www index.html<CR>	AIX Only permit the parent (home) directory and the ***public_html*** (current) directories to the ***www*** account as well as the ***index.html*** file in the current subdirectory (***public_html***).
aclput filename < control_file<CR>	AIX Only apply the Access Control List contain in the file "***control_file*** " against the file or directory specified.
alias newname command_or_pathname<CR>	substitute command or pathname wherever newname is used.
assist<CR>	Access on-line documentation and how- to's on Central UNIX.
apropos keyword<CR>	AIX Only List all commands with this keyword in their description.
batch < filename<CR>	On Central UNIX, runs the script contained in "***filename***" when system loads permit.
batch filename<CR>	On the Suns, runs the script contained in "***filename***" when system loads permit.
bg<CR>	Puts the current job into the background.
bugs<CR>	Access the bugs on-line database of reported problems on Central UNIX.
cat file1 [file2...]<CR>	Copy one or more files to standard output (the screen).
cd /usr/students/stname<CR>	Make the specified directory the current working directory.
chmod mode filename<CR>	Change the access permissions on filename.
command [parameters] &<CR>	Start a command and place the process into the background.

cp file1 file2<CR>	Duplicates the contents of ***file1*** into ***file2***.
CTRL-C	Abort the current executing task.
CTRL-Z	Suspend the current executing task.
date<CR>	Display the current date and time.
du<CR>	Displays the user's current disk utilization for the current location and all directories below.
exit<CR>	Leave current shell. If current shell is login shell, log off the system (Bourne shell).
fg %job<CR>	Bring the job into the foreground.
finger login_name[@site_name]<CR>	Provides information about login_name such as the user's real name.
grep pattern filename<CR>	Search one or more files for the pattern.
gopher<CR>	Start the gopher information retrieval program.
head filename<CR>	Display the first 10 lines of a file.
jobs<CR>	List the current jobs.
killall<CR>	Kill all processes except for the calling process (not on HP-UX).
kill 0<CR>	Stop all of your background processes.
kill process-id<CR>	Stop the process indicated by process-id.
kill %job<CR>	Send a TERM signal to the specified job.
logout<CR>	Log off the system (C shell).
lp -tii_boxnn filename<CR>	Central UNIX Send the named file to the system printer.
lpr -J ii_boxnn filename<CR>	AIX Only Send the named file to the system printer.
lpr -Jii_boxnn filename<CR>	Send the named file to the system printer (Sun system).
lpr -Pps filename<CR>	Send the named file to the Postscript laser printer (Sun system).
ls<CR>	Lists the contents of the current working directory.
lynx<CR>	access the World Wide Web client ***lynx***.
mailx username<CR> ***mail username<CR>***	HP-UX Send a mail message to the user ID username.
man command<CR>	Access the on-line UNIX manual.
man -k keyword<CR>	List all commands with this keyword in their description.

mkdir directory<CR>	Create a new directory under the current working directory.
more filename<CR>	List a file to the screen, pausing after each screen. Press the space bar to continue.
mv file1 file2<CR>	Rename file1 to file2.
new.dots<CR>	Update the current "dot" files in the user's home directory to the system standard set.
nice +nn command [parameters][&]<CR>	Run the indicated command at a lower relative priority (***0***=normal ***20***=lowest on most systems, ***0***=normal ***39***=lowest on HP-UX).
nohup command [parameters] &<CR>	Run the command in the background and don't stop when the user logs out.
notify<CR>	Have the shell notify the user when a job's status changes.
on site_name command [parameters]<CR>	Execute ***command*** on the site indicated by ***site_name***. Currently unavailable on most systems.
passwd<CR>	Change the password for the current user ID.
ph real_name<CR>	Allows the user to lookup another user's email address based on their real name (see chapter "Using ph" for more information).
pilot<CR>	The "***pilot*** " command allows you to browse your files in a manner very similar to the way that "pine" browses files. "***pilot*** " is only available on systems that have pine installed on them. "***pilot*** " will allow a user to list, delete, edit, and view files as well as move in and out of directories.
policy<CR>	Access the system policy files on Central UNIX.
pr -lnn filename<CR>	Copy filename to the screen with page headers at the top of each page of length nn.
ps<CR>	Displays process information with your effective user ID.
pwd<CR>	Display the path of the current working directory.
quota -v<CR>	Display the user's current quota and disk usage in blocks.
rm filename<CR>	Delete ***filename***.
rmdir directory<CR>	Delete the directory ***directory*** if it is empty.
setenv TERM terminal-type<CR>	Defines the terminal you are on to the system (C shell).

sort filename<CR>	Sorts a file into ascending order beginning with the first column of data.
stop<CR>	Stops the current job which is executing in the background.
sysaliases<CR>	Access database of system e-mail aliases and their contents.
sysnews<CR>	Start the System News browser.
tail filename<CR>	Display the last 10 lines of a file.
talk username<CR>	Allows users to communicate with each other interactively with a split screen.
TERM=terminal-type;export TERM<CR>	Defines the terminal you are on to the system (Bourne shell).
vi filename<CR>	Execute the full-screen editor with filename.
w<CR>	Lists the usernames of everyone presently logged onto the current site.
whatson<CR>	Provides a list of software available on the system.
whereshome<CR>	Central UNIX Returns the name of your home site.
who<CR>	List the user IDs of everyone logged in.
write username<CR>	Allows users to communicate with each other interactively.
%job<CR>	Brings the job into the foreground.
%job &<CR>	Continues the specified job in the background.

NOTES