

August 5, 1996

Unix is a multi-user operating system originally designed for program development and scientific computing. At Kansas State University, Unix is available on several central servers and public Sun workstations.

This is a summary of the most common Unix commands. Consult the Sun Microsystems manuals for complete reference on these and other Unix commands and command options. Printed documentation is available in the Computing Information Center, 9A Faurchild Hall; online help is available by using the **man** command.

Literal text, shown in courier, must be entered unchanged. Parameters, shown in *italics*, represent user-supplied text. Options shown in brackets ([]) may be omitted, and mutually exclusive options are separated by a vertical bar (|).

General

login *userid* login to Unix with *userid*
logout logout if at login shell
exit exit from current command shell
passwd change user ID password
chfn change your full name
date show current date and time
man *command* show manual page (help) for *command*
man -k *keyword* show manual page descriptions that contain *keyword*
describe [*section*] describe local commands and policies
cal show the current month's calendar
clear clear terminal screen

File & Directory Parameters

filename one file may be used
filenames multiple files may be used. Separate file names with spaces. Wildcards are allowed.
directory one directory may be used
filedir one *filename* or *directory* may be used
filedirs multiple *filenames* or *directories* may be used

File and Directory Management

ls [*-options*] [*filedirs*]
list files in the current directory. Optionally, specify which file names to list. Options:
-a all entries (include dot files)
-t order by last modification time
-l long format
-r reverse order of listing
-s give the size in blocks
-X multi-column output, sort entries across page
-F append directories with a / and executables with a *
-R recursively list files in subdirectories

dir alias for **ls -la**
pwd show current directory
cd [*directory*]
move to *directory*. **cd** without specifying a *directory* moves you to your home directory. ~ (tilde) also represents your home directory and ~*userid* represents *userid*'s home directory.

mkdir *directory*
make new *directory*
rmdir *directory*
remove *directory*. **-i** interactively prompts for file removal, **-r** recursively removes subdirectories and files in *directory*.

mv *filedir1* *filedir2*
move and/or rename file or *directory* *filedir1* to *filedir2*
cp *filename1* *filename2*
copy the contents of *filename1* to *filename2*
cp *filenames* *directory*
copy the contents of *filenames* into *directory*

rm [*-f*] *filename* | [*-r*][*-f*] *directory*
remove *filename* from current *directory*. If **-r** is specified, *directory* and its contents will be removed. **-f** removes the file regardless of file permissions.
du [*-s*] [*directory*]
display number of kilobytes contained in *directory*. Use **-s** to only show grand total summary.

quota [*-v*]
check your disk space quota. **-v** shows your quota.
chmod *users* *operations* *permissions* *filedirs*
change file permission modes on files and *directories* *filedirs*. Parameters:

Users: **u** you, **g** your group, **o** all users
Operations: **+** (plus) grant, **-** (minus) deny, **=** (equal) set
Permissions: **r** read, **w** write, **x** execution
Example: **chmod og-rwx** **personal** removes read, write, and execution permissions for your group and all other users for file **personal**.

file

filename determine the type of contents of *filename*
compress *filename*
compress *filename* to save disk space. Successfully compressed files are appended with **.z**. Uncompress files with **uncompress**.

uuencode *filename*
decode a uuencoded file sent via mail or Usenet. Creates new file, the contents of input file *filename*. Encode a binary file with **uuencode**.

Program and Shell Commands

A shell is the user interface for Unix: it accepts commands and controls your terminal. The most commonly used shell is **csh**.

Ctrl-C stop execution of current program
Ctrl-D signal end of file
Ctrl-Z suspend execution of current program
bg place the last suspended job in the background and resume its execution
fg bring a program previously suspended or put in the background into foreground execution
jobs show all background processes running under the current shell. Displays job number, job status, and job name.

ps display process status report for your processes. Displays process ID number (PID) used for killing processes
kill **-9** *pid* kill execution of process with process ID number *pid*
kill **-9** **-1** kill execution of all processes except your current shell
time [*command*]
show shell's CPU and real time used. If *command* is given, *command* is executed and a report of real and CPU time is given.

alias [*abbreviation*] [*command*]
show all aliases. If *abbreviation* is included, only its alias is shown. If *command* and *abbreviation* are included, *abbreviation* will act like *command*. To make permanent, add this command to your **.cshrc** file.
Example: **alias dir 'ls -la'** makes alias **dir** act like the command **ls -la**.

set [*shellvar*] [= *value*]
show or set shell variables used for current shell. Shell variables will not be passed to other programs. In general, shell variables are not capitalized.
setenv [*SHELLVAR*] [= *value*]
show or set environment variables for current shell and programs started under the current shell. In general, environment variables are capitalized.

Input/Output Redirection

Pipes and Input/Output redirection allow one program to send its output to a second program or file. *command* can be nearly any Unix command which produces output.

command1 | *command2*

pipe (direct) the standard output of *command1* as the standard input to *command2*. In this command, the vertical bar (|) must be entered

command > *filename*

redirect the standard output of *command* to *filename*. *filename* will not be replaced. To allow overwriting, comment out the **set noclobber** line in your **.cshrc** file.

command >> *filename*

redirect the standard output of *command* to *filename*, appending to the end of *filename*.

command < *filename*

use *filename* as the standard input of *command*

Text File Processing Utilities

cat *filenames*

display contents of *filenames* on screen

more *filenames*

displays contents of *filenames* to screen, one page at a time. ? displays **more** commands.

head [-n] *filename*

type the first 10 lines of *filename* to the screen. Optionally, type the first *n* lines.

tail [+n | -n] *filename*

type the last 10 lines of *filename* to the screen. Optionally, type from line *n* with +*n* and the last *n* lines with -*n*.

grep [-i][**-c**][**-l**] [**-e**]*string* *filenames*

show occurrences of regular expression *string* in *filenames*

Options:

-i ignore case of *string* while searching

-c count the number of occurrences of *string* without showing found text

-l list only the filenames of the files which contain *string*

-e indicate start of string. Useful if *string* contains a dash (-).

diff *filename1 filename2*

display differences between text files *filename1* and *filename2*

pico *filename*

vi *filename*

emacs *filename*

edit *filename* using the **pico**, **vi**, or **emacs** full-screen system editor. See the *vi Reference Card* and the *emacs Reference Card* for details.

Communications

pine [*userid* [*@hostname*]]

read and send electronic mail. If *userid* is specified, **pine** sends mail to *userid*. If *userid* is on a remote computer, include *@hostname*. For details see the *Pine Reference Card*.

elm [*userid* [*@hostname*]]

read and send electronic mail, similar to **pine**.

nn [*newsgroup*]

read Usenet news. Optionally, read only subscribed group *newsgroup*. Run **setup-news** to setup **.newsrc** file for the first time.

finger [*userid*][*@hostname*]

display information about users. If *userid* is included, information on *userid* is displayed. If *@hostname* is included, user information from *hostname* is displayed.

talk [*userid* [*@hostname*]]

respond to or start an interactive talk session with *userid*. If *userid* is on another computer, include the remote host *@hostname*. Exit with **Ctrl-C**.

mesg [-y | -n]

permit (-y) or deny (-n) messages from write or talk from appearing on your screen

ftp *hostname*

GET and PUT files to and from remote computer *hostname*

telnet *hostname*

login to remote computer *hostname*

rlogin [-l *userid*] *hostname*

login to remote machine *hostname*. Optionally, specify *userid* to login as.

kermitt [-s *filename* | -r]

transfer files between Unix and another computer system. Send files from Unix with -s *filename*. Receive files from the remote computer with -r. If no options are given, **kermitt** enters interactive mode; type ? for help.

Printing

For a complete list of printer destinations, enter **describe printing**. Some common printer destinations are:

s231lx455 Seaton Hall, room 23

flmix Farrell Library, room 2

fl85mix Farrell Library, room 2, 8.5x11 paper

n126lx455 Nichols Hall, room 126

lpr [-P*destination*] *filename*

print *filename* to printer *destination*

setenv PRINTER *destination*

set default printer destination

tprint *filename*

print *filename* to a printer attached to your microcomputer

Languages

The **cc**, **gcc**, **g++**, and **f77** language compilers use many of the same options:

-o *executable*

name executable output program *executable* instead of **a.out**

-l*lib*

link program with object library *lib*. Allows special functions and procedures to be used. For example, **-lm** links with the math library.

-I *ipath*

include directories *ipath* in the list of directories to search for **#include** files

cc [-o *executable*][-I *ipath*] *sourcefiles* [-l*lib*]

compile C program *sourcefiles* with Sun's C compiler

gcc [-o *executable*][-I *ipath*][-l*lib*] *sourcefiles*

compile C program *sourcefiles* with GNU's C compiler

g++ [-o *executable*][-I *ipath*][-l*lib*] *sourcefiles*

compile C++ program *sourcefiles* with GNU's C++ compiler

f77 [-o *executable*][-I *ipath*][-l*lib*] *sourcefiles*

compile FORTRAN program *sourcefiles* with Sun's FORTRAN 77 compiler

sas [*sasprogram*]

run *sasprogram*. If *sasprogram* is not specified, **sas** enters interactive mode.

make [-f *makefile*] [*actiontarget*]

maintain, update, and regenerate related programs and files. If *makefile* is not specified, **make** executes from a file named **makefile**. If *actiontarget* is not found, **Makefile** is executed. If *actiontarget* is not specified, the first target is executed.

X Window System

xterm [*&*]

start a new X terminal. Often run in the background with **&** symbol.

xlock

lock workstation running X