

# UNIX MADE EASY

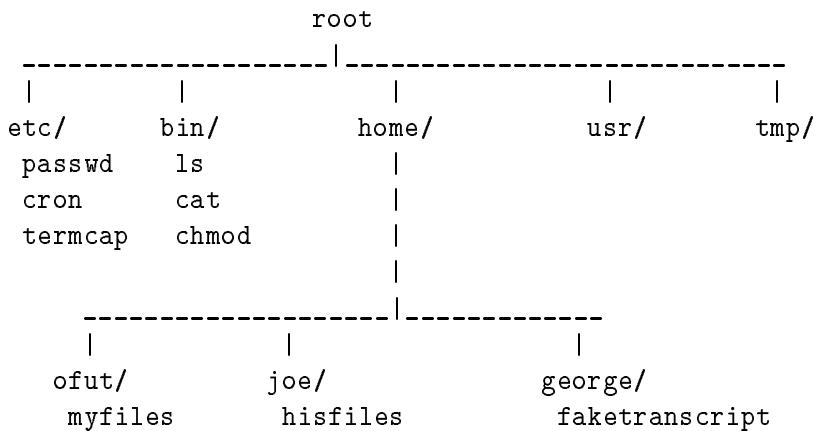
Jeff Offutt

## 1 DIRECTORIES

A major structural basis for UNIX is the tree format of the user directories. Files are stored as entries in the nodes of a large tree that encompasses the entire disk(s). Each user has a “home directory” that is a node on the tree. Your files are stored in that directory and sub-directories that you can create under your home directory.

The **COMMANDS** section will introduce a few commands that you can use with directories.

The overall directory structure looks something like this:



This is an abbreviated version of the directory structure. The directories **/ETC** and **/BIN** contain commands that we can execute. The directory **/home** contains home directories for users **OFUT**, **JOE**, and **GEORGE**. When Joe logs in, he finds himself in the directory **JOE/**, or **/home/JOE**. The long form “**/home/JOE**” gives the entire pathname starting at the root directory.

Joe has the ability to create files in his home directory. He cannot create files in the directories **OFUT/** and **GEORGE/**, though he can read files there (unless user **OFUT** expressly specifies that a file cannot be read by using the **chmod** command).

## 2 COMMANDS

This section is a brief introduction to some of the more important UNIX commands. For most commands, there are a variety of options that can be supplied to change what the operation does. These options are generally supplied with a dash: “`command -x`”.

An important operation is aborting a command or program. Type `<ctrl> -C` to abort.

**exit:** Log off of the system.

**man:** This is easily your most useful command. `man` is the help facility; a request to the online manual program. Call it with one argument, a command that you want information for. For example, if you type `man man`, you will get help for `man`. A useful option is the “keyword” option. `man -k topic` will give you a list commands related to `topic`.

### 2.1 File Handling Commands

**ls:** Lists the files in your current directory. Various options tell you more about the files. Try `ls -F` and `ls -l`. For convenience, files that begin with the character “.” do not appear when you type `ls`. Try `ls -a` to see these.

**chmod:** Changes the protection modes of a file. `chmod 600 <filename>` sets a file so that nobody but you can read or write it, `chmod 644 <filename>` allows everybody else to read it.

**cp:** Copies a file. `cp <oldname> <newname>` copies the file named `<oldname>` to `<newname>`.  
`cp /home/ofut/exam /home/george/innocent` copies the file `exam` from user `ofut`’s directory to user `george`’s.

**cat:** Prints a file onto your screen. `cat <filename>`.

**more:** Pages a file onto your screen — if the file is more than one screenfull, it pauses after each screen.

**vi:** The screen editor. See the companion document.

### 2.2 Directory Commands

**pwd:** Print working directory. Shows where you are in the directory tree.

**cd:** Change directory moves you from one place to another. `cd project` moves you to the local subdirectory `project`. `cd /home/joe` moves you to Joe’s home directory. `cd` moves you to your home directory. `cd ..` moves you one node up the directory tree.

**mkdir:** Make directory creates a new directory under your current directory. `mkdir project` creates the directory `project` mentioned above.

## 2.3 System Status Commands

**w:** Tells you who else is on the system.

**finger:** Gives you nosy information about other users. **finger** tells you information about all users currently logged on. **finger ofut** gives you information about user **ofut**.

**mail:** Electronic mail. **mail** to read your mail, **mail ofut** to send mail to your professor about how great the lectures are.

**passwd:** Change your password. DO THIS THE FIRST TIME YOU LOG IN! Just type **passwd** and it will prompt you.

**chsh:** Changes your login shell. Common shells include the Bourne shell (**sh**), the C shell (**csh**), (**tcsh**), the Korn shell (**ksh**), and the Bourne-again shell (**bash**).

## 3 Getting Started

One of the nice aspects of UNIX is that it is easy to tailor your work environment and even the command names to suit your tastes. Your environment includes your *shell*, your search path, your **cd** path, command and mail aliases, and editor defaults, among others.

The most important aspect of your environment is your login shell. On most systems, we have several choices, including *csh* (C-like shell), *sh* (Bourne shell), *bash* (Bourne-again shell), *tcsh* (C-like shell), and *ksh* (Korn shell). Use the **chsh** command to change your default shell: **chsh**, then type one of the above shells at the prompt. After running **chsh**, you must log off and log back on again to enter your new shell. THE REST OF THIS SECTION ASSUMES YOU ARE USING THE C SHELL.

Your environment is principally controlled by two files; **.login** and **.cshrc**. The **.login** file is executed every time you log in and the **.cshrc** file is executed every time you start a new shell. The **.login** file is a good place to put commands that set up your environment, terminal type, check your mail, etc. The **.cshrc** file is where you put commands that customize the shell to your own liking. It contains commands to set your prompt, command aliases, etc. The directory **~ofut/examples** (**~** is short for “the home directory of”) contains files called **login** and **cshrc** that have commands from my setup files. Use the **man** command to find out what the commands do.

Two other useful setup files are **.mailrc** and **.exrc**. The **.mailrc** file allows you to set up group mail aliases if you use the basic Unix **mail** command. The **.exrc** file contains commands that are executed each time you start **VI**. The **:ab** command sets an abbreviation; whenever the left-hand side is typed in input mode, the right-hand side is automatically substituted. The **:map** command sets up a mapping; whenever the left-hand side is typed in command mode, the right-hand side is automatically substituted. The **:se** command sets defaults; **:se all** lists all the defaults, **:se ic** tells the pattern matcher to ignore differences in case.

Copyright Jeff Offutt, 1995