

Solaris Frequently Asked Questions

Composed by Ong Zhiyang (16/01/04)

Table of Contents

<u>1.</u>	<u>General</u>	3
<u>1.1.</u>	<u>How do I check the name and version of the operating system running on my machine?</u>	3
<u>1.2.</u>	<u>How do I check my disk utilisation?</u>	4
<u>1.3.</u>	<u>How do I find out more information about the processes/applications that I am running?</u>	5
<u>1.4.</u>	<u>How do I kill a process?</u>	5
<u>1.5.</u>	<u>How do I get rid of “core” files or temporary files?</u>	8
<u>1.6.</u>	<u>How do I export a display from a remote workstation?</u>	9
<u>1.7.</u>	<u>How do I find out the name of my workstation/PC or its IP address?</u> 10	
<u>1.8.</u>	<u>Command not found... What shall I do to remedy the problem?</u>	10
<u>1.9.</u>	<u>What is command piping?</u>	11
<u>1.10.</u>	<u>What is Input/Output redirection?</u>	11
<u>1.11.</u>	<u>What is a symbolic link?</u>	12
<u>1.12.</u>	<u>How do I find out the top 5 or 10 processes?</u>	14
<u>1.13.</u>	<u>What is the host status of remote machines?</u>	14
<u>2.</u>	<u>Login Problems</u>	16
<u>2.1.</u>	<u>I am unable to login. The message “Login incorrect” keeps appearing. What should I do?</u>	16
<u>2.2.</u>	<u>How can I change my password?</u>	16

2.3.	<u>What should I not change my password to?</u>	17
2.4.	<u>What do I do now that I forgotten my password?</u>	18
2.5.	<u>How do I logout?</u>	18
2.6.	<u>What do I do if my system hangs?</u>	18
<u>3.</u>	<u>Shell</u>	20
3.1.	<u>How do I know what UNIX shell am I using?</u>	20
3.2.	<u>How can I change to another UNIX shell?</u>	20
3.3.	<u>Are there any advantageous of using one shell compared to other shells?</u>	20
3.4.	<u>How do I set my shell variables?</u>	21
<u>4.</u>	<u>Directory and File Manipulation</u>	24
4.1.	<u>How do I find out what is my current working directory?</u>	24
4.2.	<u>How do I change directories?</u>	24
4.3.	<u>How do I create/remove directories?</u>	24
4.4.	<u>How do I list the contents of a directory?</u>	25
4.5.	<u>How do I copy, move or remove files?</u>	25
4.6.	<u>How do I copy files from one directory to another?</u>	26
4.7.	<u>How do I compress and uncompress a file?</u>	26
4.8.	<u>How do I change the permissions on my files?</u>	27
4.9.	<u>How do I create a file or change its access or modification time?</u>	27
4.10.	<u>How do I remove files that are bigger than a certain size?</u>	27
<u>5.</u>	<u>Security</u>	29
5.1.	<u>How should I select a password?</u>	29
5.2.	<u>How can I keep my password safe?</u>	29

<u>5.3.</u>	<u>How can I prevent others from accessing my account in my absence from the workstation without overloading the server?</u>	29
<u>5.4.</u>	<u>What shall I do if I suspect someone has been accessing the network using my password?</u>	29
<u>6.</u>	<u>Printing</u>	31
<u>6.1.</u>	<u>How do I print my documents in the UNIX xterm terminal?</u>	31
<u>6.2.</u>	<u>The printer is not printing my document, how can I check the status of my print job?</u>	32
<u>6.3.</u>	<u>How do I find out the name of the printer in the department's network?</u>	33
<u>6.4.</u>	<u>How do I remove a print job?</u>	33

1. General

1.1. *How do I check the name and version of the operating system running on my machine?*

On the command line, type:

```
uname -rs
```

The name of the machine's operating system as well as its version will be printed as standard output in your shell terminal.

For example:

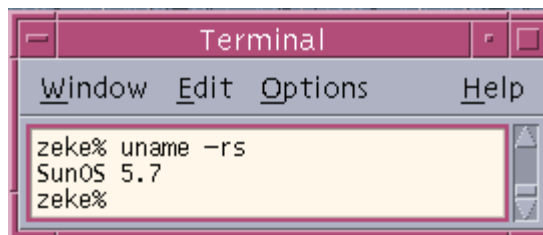


Figure 1.1 An example of the *uname* command usage to find out the name and version of the operating system.

The operating system of the example above is SunOS. Its version is 5.7.

1.2. *How do I check my disk utilisation?*

To check available disk space on partition, try typing:

- On SunOS 4.x system:

`df . [Enter]`

- On Solaris 2.x system:

`df -k . [Enter]`

To check disk space used by a directory:

`du -sk Directory-Name [Enter]`

To check disk space used from current directory as well as all of its subdirectories, try:

`du -sk [Enter]`

That is, it recursively performs the `du` command till it has been executed on all subdirectories of the current working directory.

Type `du * > filename` to find out the disk utilisation of files and directories that are accessible to group members and other users and place the result into a file with the name *filename*.

1.3. ***How do I find out more information about the processes/applications that I am running?***

To view all process information on the current system, type:

- On SunOS
ps-auxw
- On Solaris 2.x
ps -ef

This will result in displaying the process IDs as well as the name of the process on the shell terminal. Other information like the amount of CPU time consumed as well as the terminal (workstation) from which this process is being run will also be displayed.

In UNIX each application that is running is known as a process. For example, the UNIX shell terminal from which you type your commands at the command-line is a process. The ps command is a process, just like xedit as well as vi.

1.4. ***How do I kill a process?***

First, either find out the [process ID](#) or [name of the process](#), which is currently executing, that you want to kill (put an end to).

To kill a process:

- With its process ID

Type:

kill Process-ID,

where Process-ID is the process ID of the process that you want to kill.

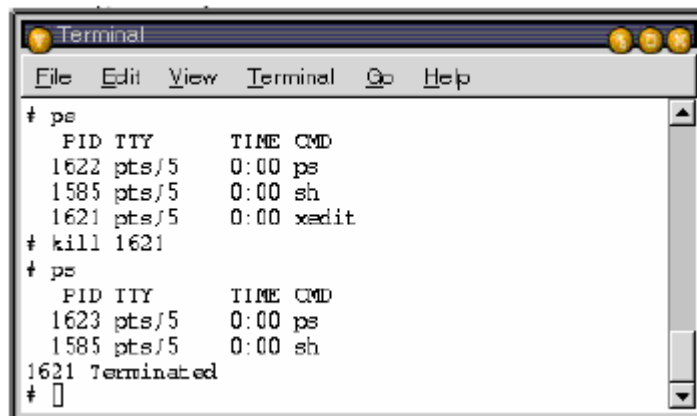
- With its name

Type:

`kill Process-Name,`

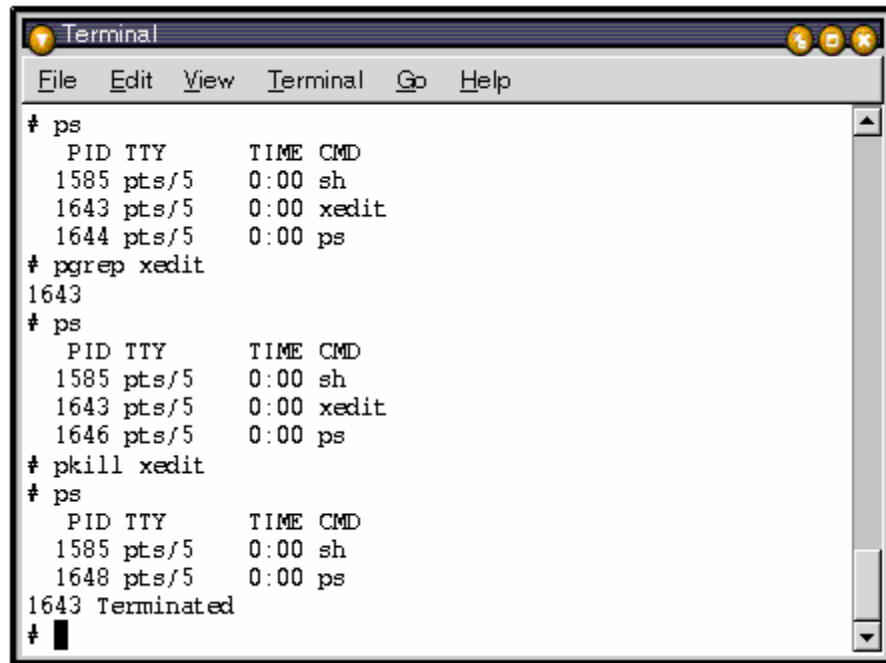
where Process-Name is the name of the process that you want to kill.

For example, in the example indicated by Fig. 1.4a, the process `xedit` has a process ID (PID), found with the aid of the `ps` command, of 1621 and the `kill` command is used to kill it.



```
Terminal
File Edit View Terminal Go Help
# ps
  PID TTY      TIME CMD
  1622 pts/5    0:00 ps
  1585 pts/5    0:00 sh
  1621 pts/5    0:00 xedit
# kill 1621
# ps
  PID TTY      TIME CMD
  1623 pts/5    0:00 ps
  1585 pts/5    0:00 sh
1621 Terminated
#
```

Figure 1.4a An example of how to find out the processes that are running on the system and kill the desired process using its process ID



```
Terminal
File Edit View Terminal Go Help
# ps
  PID TTY          TIME CMD
 1585 pts/5        0:00 sh
 1643 pts/5        0:00 xedit
 1644 pts/5        0:00 ps
# pgrep xedit
1643
# ps
  PID TTY          TIME CMD
 1585 pts/5        0:00 sh
 1643 pts/5        0:00 xedit
 1646 pts/5        0:00 ps
# pkill xedit
# ps
  PID TTY          TIME CMD
 1585 pts/5        0:00 sh
 1648 pts/5        0:00 ps
1643 Terminated
#
```

Figure 1.4b An example of how to find out the processes that are running on the system and kill the desired process using its name

In Figure 1.4b, having used to *ps* command to list the processes that are running, the *pkill* command is used to kill the desired process by its command name (CMD).

Sometimes the kill command may not kill a process. This is because the process may be deemed to be vital to the running of the system. Hence, if the user is certain about killing that process, the user can use the -9 option.

That is,

```
kill -9 Process-ID
```

will end the process, regardless of what it is doing.

Please duly note that using the -9 option to kill a shell process, like csh or sh (C- shell and Bourne shell processes), will close the shell terminal.

To kill a number of processes with a common feature on the SunOS operating system, try:

```
ps -auxw | grep "FIELD" | grep -v "grep" | awk '{ print $2 }' | xargs -i -t kill -9 {}
```

where "FIELD" is a common string of all the process to be killed.

For example, to use the command to kill all processes run by the user with the username johann, try:

```
ps -auxw | grep johann | grep -v "grep" | awk '{ print $2 }' | xargs -i -t kill -9 {}
```

To find out the name of the processes that are currently executing, you can use the ps command.

Else, you can type:

```
pgrep Process_Name
```

to find out if a process with the name "Process_Name" is currently executing. If there is at least one process with the name "Process_Name" in execution, a set of numbers corresponding to the process IDs of these processes will be displayed on the shell terminal. Else, only the shell prompt is displayed to indicate that there are not processes named "Process_Name" that are currently executing.

1.5. **How do I get rid of "core" files or temporary files?**

Type:

```
find . -type f -name filename -exec rm -i {} \;
```

Note that *filename* is the filename of the files that you want to get rid of or the pattern expression of the type of files you want to look for.

For example, use the following expressions:

core	Get rid of all core files
'*.nfs'	Get rid of all <i>.nfs</i> temporary files
'*.pdv'	Get rid of all <i>.pdv</i> temporary files
'LogFile*.txt'	Get rid of all text files that begin with LogFile and end with the <i>.txt</i> extension

This command recursively searches its current working directory as well as its all of its subdirectories for regular files that match the pattern expression of *filename*.

Once each of these files is found, the user is prompted at the shell terminal requesting for confirmation of the removal of that file.

1.6. ***How do I export a display from a remote workstation?***

At the shell prompt, type `echo $DISPLAY`.

This cause the name of the workstation/PC you are working on to appear on the shell terminal. If it does, type `xhost +` before you telnet into a server.

If it does not, type `xhost +` before you telnet into a server followed by `setenv DISPLAY machine_name/IP_address`.

[What is *machine_name/IP_address*? See Section 1.7.](#)

This will allow the telnet/rlogin session to open up Graphical User Interface (GUI) programs from your shell terminal and execute them as you would on the machines that you have rlogin-ed or telnet-ed to.

Typing `setenv DISPLAY machine_name/IP_address` will allow GUI programs of the server to be exported to your workstation/PC.

1.7. ***How do I find out the name of my workstation/PC or its IP address?***

Type `nslookup workstation/PC` to find out the name as well as the Internet Protocol (IP) address of the workstation or PC that you are working on.

The name of the workstation/PC you are working on may be found on a sticker adhered to your monitor.

1.8. ***Command not found... What shall I do to remedy the problem?***

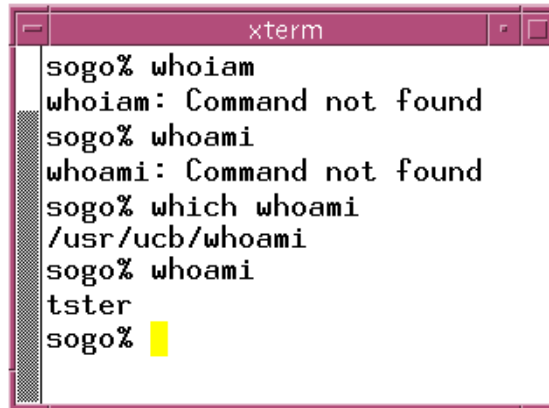
When a *command not found* message appears, check that you have entered the command correctly (common typographical mistakes includes usage of the wrong case or spelling errors) and enter the correct command if an incorrect command was entered.

If the same message appears again, type *which command-name* to determine the path of the command. If the path of the command is incorrect, change it before entering the command again.

Else, if the command not found message still appears again, it means that this command does not exist.

For example, the command *whoami* was entered after correcting a typographical mistake upon receipt of the *command not found* message. When the same message appears, the *which* command was entered to

determine the path of the *whoami* command. The path was found to be wrong and it was subsequently corrected. The *whoami* command was run again in another session and was executed successfully.



```
sogo% whoiam
whoiam: Command not found
sogo% whoami
whoami: Command not found
sogo% which whoami
/usr/ucb/whoami
sogo% whoami
tster
sogo% █
```

1.9. *What is command piping?*

Piping filters the output of one command to the input of another command. The | symbol is used to pipe output from one command to another command.

Its syntax is: command1 | command2

For example, more ReadMe | grep NEdit prints each line of the ReadMe file that contains the word “*NEdit*” on the UNIX shell terminal.

1.10. *What is Input/Output redirection?*

Input redirection causes the shell terminal to take its input from a file instead of the UNIX standard input in the shell terminal from the user. That is, input to the command is read from a file instead of from the user. Output redirection causes the output of the command to be placed in a file, which name is given as part of output redirection, instead of the standard output at the UNIX console.

Its general syntax is: `command < input_file > output_file`

The `<` symbol is used to redirect input whilst the `>` symbol is used to redirect output. The `>>` symbol is used to append the output of the command to the named file.

For example: `java < input_file MazeBot > output_file`

In the above example, the Java Class file *MazeBot.class* is executed whilst taking its input from the file named *input_file*; it redirects the output to the file *output_file*.

`java LogManager >> Log_file`

The output of the execution of the object file *LogManager.class* is appended to the existing file named *Log_file*.

1.11. **What is a symbolic link?**

A symbolic link is a special file that points to another file on the system. When you access a symbolic link file, you access the pathname stored inside it. The symbolic link uses this pathname to advance to the file or directory on the system represented by the pathname stored in this symbolic link. In that sense, it is similar to an alias or a shortcut for users familiar with the Mac OS and Microsoft Windows operating systems.

Since symbolic links can point to any type of file or directory, you can use symbolic links to make a file appear as though it is located in many different places or has many different names in the file system.

Its syntax is: `ln -s source destination`

source is the relative or absolute directory path of where the directory that you wish to create a symbolic link for is. destination is the name, of your choice, for the symbolic link file in this current directory.

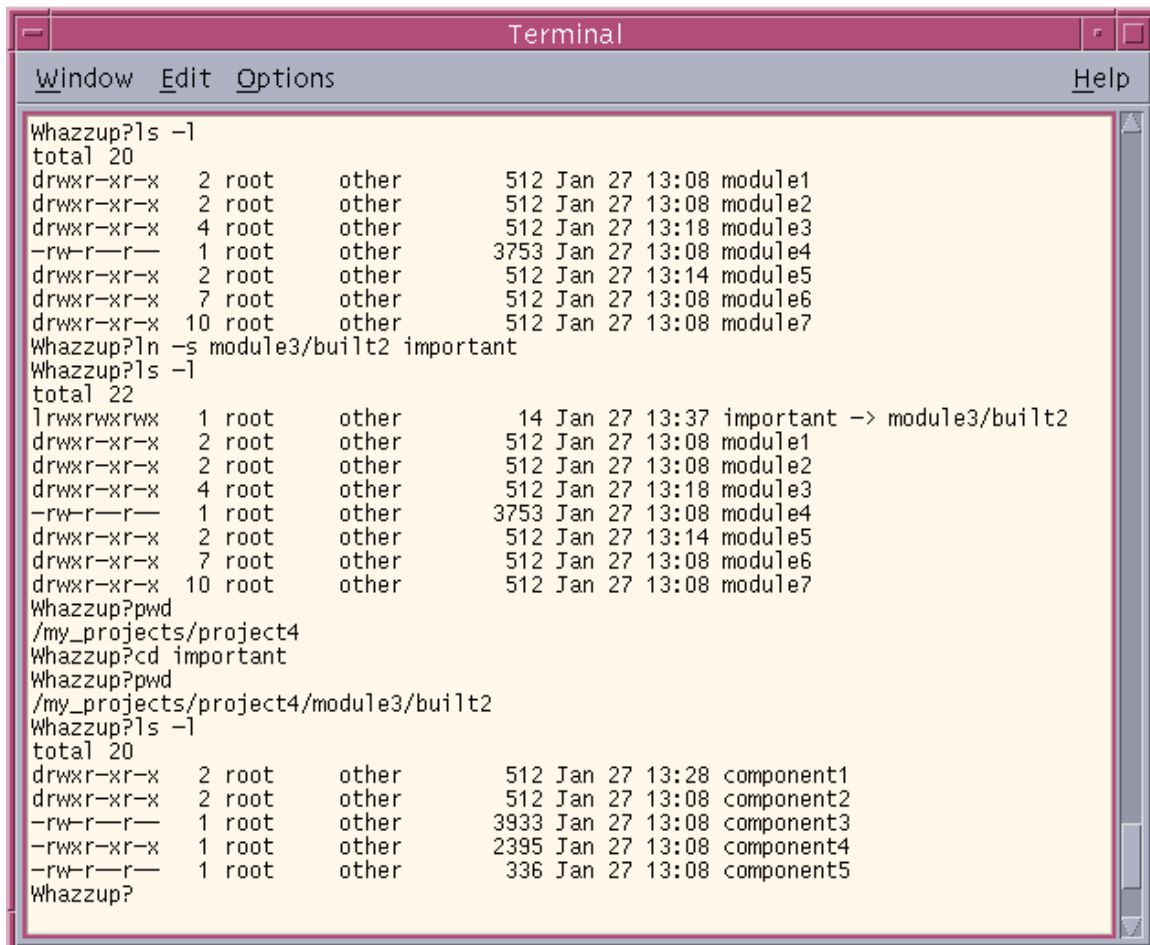
Figure 1.11 shows you how to create a symbolic link by using the *ln* command with the *-s* option. The *ls -l* command used after the symbolic link is created indicates that the *important* file is a symbolic link since its file type is indicated by *l*, which is the first character of the line, for symbolic links. It is an alias/shortcut to its directory “*module3/built2*” as indicated by the arrow from *important* to *module3/built2*.

When the directory is changed to the *important* subdirectory, the current working directory becomes:

/my_projects/project4/module3/built2

Instead of:

/my_projects/project4/



```
Terminal
Window Edit Options Help
Whazzup?ls -l
total 20
drwxr-xr-x  2 root  other    512 Jan 27 13:08 module1
drwxr-xr-x  2 root  other    512 Jan 27 13:08 module2
drwxr-xr-x  4 root  other    512 Jan 27 13:18 module3
-rw-r--r--  1 root  other   3753 Jan 27 13:08 module4
drwxr-xr-x  2 root  other    512 Jan 27 13:14 module5
drwxr-xr-x  7 root  other    512 Jan 27 13:08 module6
drwxr-xr-x 10 root  other    512 Jan 27 13:08 module7
Whazzup?ln -s module3/built2 important
Whazzup?ls -l
total 22
lrwxrwxrwx  1 root  other    14 Jan 27 13:37 important -> module3/built2
drwxr-xr-x  2 root  other    512 Jan 27 13:08 module1
drwxr-xr-x  2 root  other    512 Jan 27 13:08 module2
drwxr-xr-x  4 root  other    512 Jan 27 13:18 module3
-rw-r--r--  1 root  other   3753 Jan 27 13:08 module4
drwxr-xr-x  2 root  other    512 Jan 27 13:14 module5
drwxr-xr-x  7 root  other    512 Jan 27 13:08 module6
drwxr-xr-x 10 root  other    512 Jan 27 13:08 module7
Whazzup?pwd
/my_projects/project4
Whazzup?cd important
Whazzup?pwd
/my_projects/project4/module3/built2
Whazzup?ls -l
total 20
drwxr-xr-x  2 root  other    512 Jan 27 13:28 component1
drwxr-xr-x  2 root  other    512 Jan 27 13:08 component2
-rw-r--r--  1 root  other   3933 Jan 27 13:08 component3
-rwxr-xr-x  1 root  other   2395 Jan 27 13:08 component4
-rw-r--r--  1 root  other    336 Jan 27 13:08 component5
Whazzup?
```

Figure 1.11 An example of how to create a symbolic link as well as how to recognise one.

1.12. *How do I find out the top 5 or 10 processes?*

On the command line, type:

```
/usr/ucb/ps -auwx | head -6
```

Or

```
/usr/ucb/ps -auwx | head -11
```

This command will print out the top 5 or 10 processes that are currently in use as well as a header to explain the significant of the data presented by this command. Hence, the command `/usr/ucb/ps -auwx | head -n`, where `n` is an integer, requires you to increment `n` by one to view the top `n` processes.

That is, if you want to view the top `m` – where `m` is an integer – processes, substitute the number `(m+1)` into the number `n`.

1.13. *What is the host status of remote machines?*

The *rup* command is used to show the host status of remote machines.

```
ca Telnet ernesto
ernesto# hostname
ernesto
ernesto# uname
SunOS
ernesto# rmp
  fe80::203:baff: up 3 days, 21:49, load average: 0.00, 0.01, 0.01
ernesto.ime.org up 3 days, 21:49, load average: 0.00, 0.01, 0.01
  fe80::a00:20ff: up 40 days, 27 mins, load average: 0.00, 0.00, 0.01
  fe80::a00:20ff: up 13 days, 1 min, load average: 0.00, 0.00, 0.01
  fe80::a00:20ff: up 13 days, 52 mins, load average: 0.00, 0.01, 0.01
  fe80::a00:20ff: up 13 days, 24 mins, load average: 0.00, 0.00, 0.01
  fe80::a00:20ff: up 13 days, 34 mins, load average: 0.00, 0.00, 0.01
  fe80::a00:20ff: up 9 days, 17:02, load average: 0.00, 0.00, 0.01
  fe80::a00:20ff: up 16 days, 20:15, load average: 0.01, 0.02, 0.02
jamp7000.ime.or up 70 days, 23:50, load average: 0.42, 0.38, 0.37
tianjin.ime.org up 9 days, 17:02, load average: 0.00, 0.00, 0.01
hangzhou.ime.or up 16 days, 20:15, load average: 0.01, 0.02, 0.02
ernesto# _
```

It indicates how long has the server been up and running, in terms of days, hours and minutes. It also indicates the load of the server as a percentage of the total amount of amount of processing time available.

2. Login Problems

2.1. *I am unable to login. The message “Login incorrect” keeps appearing. What should I do?*

Ensure that the “CAPS LOCK” is not on! UNIX is case sensitive. Entering your username and password with “CAPS LOCK” on will not allow you to be logged in to the network. Login again whilst ensuring that you have entered the correct username and password.

[See Section 2.4 if you have forgotten your password.](#)

If a “Disk quota exceeded” message appears, you will not be allowed to login because you have used up too much disk space. To be able to login again, [speak to your friendly system administrator to help you delete some unwanted files.](#)

If you have any problems, please kindly seek help from your friendly system administrators.

This has the following implication: You must not exceed your disk quota if you want to login to the network. That is, [you must remove “core” files as well as temporary files and other unwanted files to keep to your disk quota.](#)

Do not interchange the order of your login username and password entries.

2.2. *How can I change my password?*

Use the **passwd** command.

You will be asked to key in your old password for authentication purposes, followed by your new password.

When you are asked to re-enter your new password, ensure that the case of both entries of the new password are the same ([UNIX is case sensitive](#)). If they do not match, the user's password is not changed.

2.3. *What should I not change my password to?*

You should not change your password to:

- A word (or words) in any language
- A proper name

Do not use information:

- That can be found in your wallet
- Commonly known about you (car license number, mobile phone number, pet name, email addresses, etc)
- Control characters, such as '*', '[', ']', and '?'

Please kindly select password that is memorable and unobvious to avoid hacker break-ins. Passwords should be kept confidential. They should not be recorded anywhere except in your memory.

Include a mixture of character types (alphanumeric, numeric, and special), uppercase and lower case in your password that includes at least six characters so that it will be hard to crack. For example, try "cH#i54.orLA".

Do change your password often... so that if hackers manage to gain access to your password, they will not be able to access your account or the company network for long.

2.4. ***What do I do now that I forgotten my password?***

Please kindly pay your friendly department system administrator a visit.

Your friendly department system administrator will reset your password to a temporary password, which will be issued to you. Once you have logged into to any server on the network with that temporary password, [change your password](#).

This will allow your password to be kept private and confidential.

2.5. ***How do I logout?***

If you have connected to a server using *rlogin* or *telnet*, type exit to close the Bourne Shell and logout. If you are in the C Shell, type *logout* to logout. On some machines, this command has been disabled. Hence, you should type exit to close the C Shell and enter the Bourne Shell. From the Bourne Shell, you should type exit to close the terminal

Most of users are logged into to a UNIX Operating (/Desktop) Environment, such as Common Desktop Environment (CDE) or GNOME, there is an exit icon available on the toolbar to allow you to logout of the network from your workstation.

2.6. ***What do I do if my system hangs?***

Try to telnet into the server from another workstation and use the [aforementioned command “ps -ef | grep my-name”](#) to find out which process is taking up a lot of CPU time and is causing the system to hang for you, where your username is *my-name*.

Once you have identified the processes that are causing the system to hang, terminate them with the [*kill* command](#).

If your system that hung is not restored to its normal service, contact your system administrator for help.

3. Shell

3.1. *How do I know what UNIX shell am I using?*

If your shell prompt is a dollar-sign symbol, \$, or ends with it, you are using the Bourne-type Shell. That is, it can be a Bourne Shell, **sh**, a Bourne Again Shell, **bash**, a Korn Shell, **ksh**, or a POSIX Shell, **sh**.

If it ends with the percentage sign, it is a C-type Shell. That is, it can be a C Shell, **csh**, TENEX/TOPS Shell, **tcsh**.

3.2. *How can I change to another UNIX shell?*

At the command prompt, type the abbreviation for the UNIX shell that you want to switch to.

For example, if you want to switch to a Bourne Shell, type sh at the UNIX shell prompt. If you want to switch to the C Shell, type csh at the command prompt to switch to the C Shell.

3.3. *Are there any advantageous of using one shell compared to other shells?*

Some shells, for example, the Bourne Again Shell, the Korn Shell, the C Shell and the TENEX/TOPS Shell, have the following properties:

- File name completion
- Command history as well as command editing
- Ease in executing multiple background processes.
- Aliases, which are short mnemonic names, can used to run long commands.

Note that the original Bourne Shell does not include these facilities.

3.4. *How do I set my shell variables?*

Open up a text editor and save the file as a *.cshrc* file with all the shell environment variables set to those that you desire.

Aliases can be created for long commands and/or a series of commands. This reduces the amount of typing needed to save time as well as repetitive finger and wrist motion.

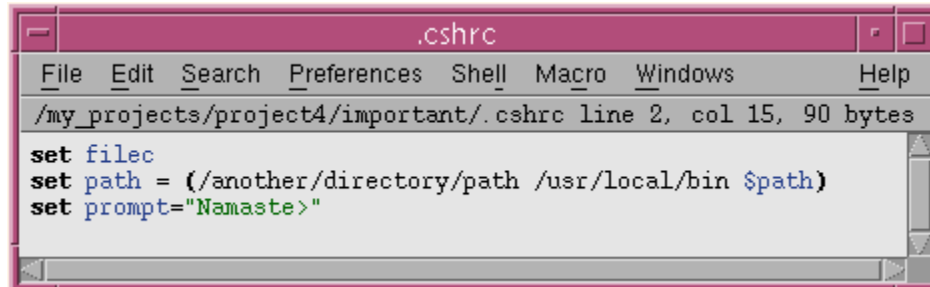
The *path* environment variable is a shell environment variable that defines which directories to search as well as the order in which to search for them. Amend this path to allow commands to run different processes at any directory.

The *prompt* environment variable is a shell variable that you can modify to change the shell prompt. For example, instead of having murray\$ for a Bourne Shell prompt or zelda\$ for a C Shell prompt, you can set *prompt* to something of your preference.

To set up temporary shell variables, set them up in the shell terminal that you are using. That way, you can modify shell variables temporarily. Their effects last as long as this session of shell terminal, which variables you are modifying, is in use. When this shell is closed, the changes will cease to occur.

This allows users to experiment with what type of shell environments they would like to have before making them permanent by putting them in the *.cshrc* file.

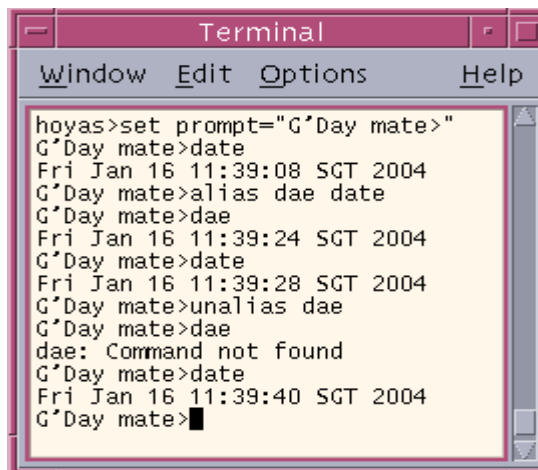
For example, Figure 3.4a shows an example of how to include another directory to search for commands, change the shell prompt to “Namaste>” as well as to enable auto-completion of directories and files.



```
.cshrc
File Edit Search Preferences Shell Macro Windows Help
/my_projects/project4/important/.cshrc line 2, col 15, 90 bytes
set filec
set path = (/another/directory/path /usr/local/bin $path)
set prompt="Namaste>"
```

Figure 3.4a An example of a .cshrc file.

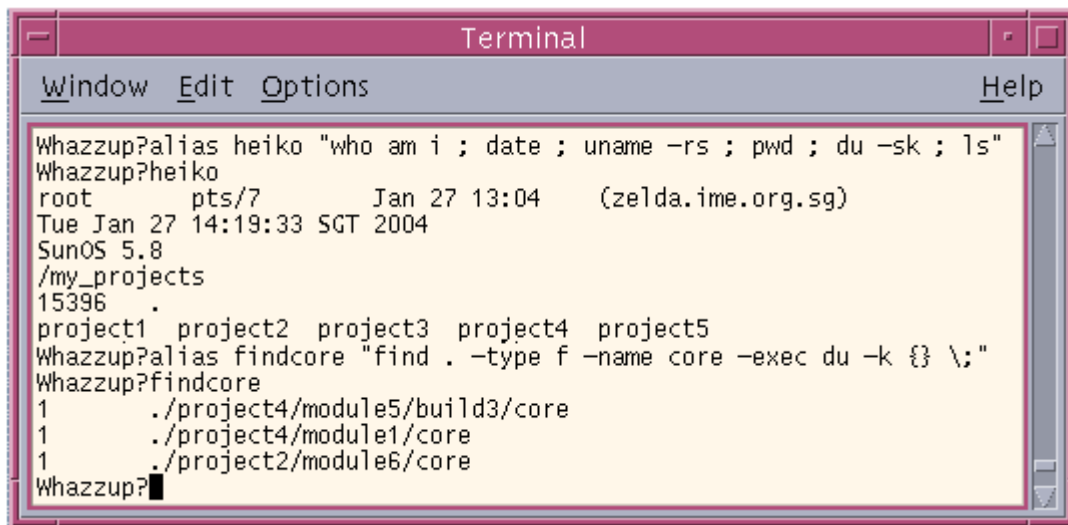
Figure 3.4b shows an example of modifying shell variables temporary. They are to change the shell prompt from “*hoyas>*” to “*G’Day mate>*” as well as to set an alias “*dae*” for the command *date*. This allows me to type *dae* or *date* to find out today’s date and time. The alias *dae* was no longer made an alias due to its confusing command name. Hence, *dae* can no longer be used as a command alias for the command *date*.



```
Terminal
Window Edit Options Help
hoyas>set prompt="G'Day mate>"
G'Day mate>date
Fri Jan 16 11:39:08 SGT 2004
G'Day mate>alias dae date
G'Day mate>dae
Fri Jan 16 11:39:24 SGT 2004
G'Day mate>date
Fri Jan 16 11:39:28 SGT 2004
G'Day mate>unalias dae
G'Day mate>dae
dae: Command not found
G'Day mate>date
Fri Jan 16 11:39:40 SGT 2004
G'Day mate>
```

Figure 3.4b An example of modifying environment variables only for this terminal session.

An example of using aliases as command shortcuts are shown in Figure 3.4c as heiko is made an alias of several UNIX commands through the use of commands separators whilst the alias findcore is a shortcut for a long command used to search for core files and yield their disk usage.



```
Terminal
Window Edit Options Help
Whazzup?alias heiko "who am i ; date ; uname -rs ; pwd ; du -sk ; ls"
Whazzup?heiko
root      pts/7          Jan 27 13:04    (zelda.ime.org.sg)
Tue Jan 27 14:19:33 SGT 2004
SunOS 5.8
/my_projects
15396    .
project1 project2 project3 project4 project5
Whazzup?alias findcore "find . -type f -name core -exec du -k {} \;"
Whazzup?findcore
1        ./project4/module5/build3/core
1        ./project4/module1/core
1        ./project2/module6/core
Whazzup?
```

Figure 3.4c Example of using aliases as shortcuts for long or multiple commands

4. Directory and File Manipulation

4.1. *How do I find out what is my current working directory?*

Type *pwd* to print the current working directory on the UNIX shell terminal.

4.2. *How do I change directories?*

Use the *cd* command to change directories.

Type *cd /* to access the user directory.

Enter *cd ..* to enter the parent directory of the current working directory.

To go up two directories, try *cd ../../*.

Enter a subdirectory with the name “subdirectory” with the command *cd subdirectory*

Enter a subdirectory “subDir” of the current working directory’s parent

4.3. *How do I create/remove directories?*

Use the *mkdir* command to create a directory and the *rmdir* command to remove an empty subdirectory.

To create a subdirectory named “subdirectory” in the current working directory, type *mkdir subdirectory*

If the subdirectory “subdirectory” is empty, use this command to remove it: *rmdir subdirectory*

To create a subdirectory “subDir” in the current working directory’s parent, try *mkdir ../subDir*

The *rm* command cannot be used to remove a subdirectory that has some files in it. The subdirectory needs to be emptied first.

The *rm* command also cannot be used to empty the current working directory. It can only be used to remove a subdirectory.

4.4. **How do I list the contents of a directory?**

Use the *ls* command to list the visible contents for a directory. Files beginning with a dot (.) are not listed since they are “hidden” files.

To view all the files in a directory in the long format, type *ls -a -l* (or *ls -al*). The hidden files will be displayed as well.

4.5. **How do I copy, move or remove files?**

The command for copying a file is *cp* whilst the command for moving a file is *mv* and the command for removing a file is *rm*.

For example:

The command *cp subDir/file1 ../file2* copies the file named “file1” from the subdirectory subDir into the parent directory as a file named “file2”.

The command *mv file3 file4* moves the file named “file3”, in the current working directory, into the file named “file4” into the current working directory. That is, “file3” is renamed as “file4”.

The command *rm -r -f subDir* removes the subdirectory “subDir” as well as all of its contents without prompting the user to confirm the removal of those files.

The command ***rm *.java*** removes all Java files (files with the .java extension) in the current working directory.

4.6. ***How do I copy files from one directory to another?***

To copy files from the “olddir” subdirectory to the “newdir” subdirectory, try one of the following:

cp -r olddir newdir

Or

cd olddir ; find . -print | cpio -pdm newdir

Or

cd newdir followed by ***tar -cvf -C olddir | tar -xvpf -***

4.7. ***How do I compress and uncompress a file?***

To compress a list of files (file1, file2 and file3), try one of the following methods:

compress -fv file1 file2 file3

Or

gzip *.jpg

Or

tar cvf filename.tar file1 file2 file3

To uncompress a file, try one of the following methods, corresponding to the type of compressed file that you receive.

For compressed files with a .Z extension, try:

uncompress file1 file2 file3

For compressed file with a .gz extension, try:

gunzip *.jpg.gz

For a compressed file with a .tar extension, try:

tar xvf filename.tar

4.8. ***How do I change the permissions on my files?***

To change the permissions of several files named file1, file2 and file3, try:
`chmod abc file1 file2 file3`

The letter a, b and c are octal numbers. They represent the permissions of the owner (a), group (b) and other users (c) for the list of files and/or directories in concern. Each octal number is a result of adding the respective numbers used to indicate read (4), write (2) and execute (1) permissions.

For example, the command `chmod 750 tempDel findCore` allows the owner to read, write to and execute the files tempDel as well as findCore. Group users are allowed to read and execute these files whilst other users have no access to these files

4.9. ***How do I create a file or change its access or modification time?***

The command `touch file1` will create a file named “file1” if file1 does not exist in the current working directory.

To change the access and modification times of a file, try:

`touch -acm -t 03152357 myfile`

This change the access and modification times of the file “myfile” to 2357 hrs on the 15th of March, 2003.

4.10. ***How do I remove files that are bigger than a certain size?***

Use the following command:

`find . -type f -size +n -print -exec rm {} \;`

where n is an integer indicating the size of the files than you want to search for in half kilobytes (KB) This will allow users to search for files that are bigger than (n/2) KB.

That is, to search for and remove files greater than 1 MB, substitute 2000 into n. That is, *find . -type f -size +2000 -print*.

5. Security

5.1. *How should I select a password?*

[See Section 2.3](#)

5.2. *How can I keep my password safe?*

Remember your password. Do not keep a copy of your password anywhere else. That is, do not write your password on a piece of paper that is kept in your wallet or underneath your keyboard.

5.3. *How can I prevent others from accessing my account in my absence from the workstation without overloading the server?*

If you are leaving your workstation for a while (5-15 minutes), lock your workstation.

Else, logout of your workstation so that others can use the workstation in your prolonged absence. This will also reduce the server load and allow the machine to be operating at a faster speed.

Always logout after use.

5.4. *What shall I do if I suspect someone has been accessing the network using my password?*

Take note of the date and time as well as the machine of your previous login whenever you login to a server from a workstation. This information is

display in the shell terminal when you login. If you notice that they do not tally with your previous logins, take extreme caution of logins at odd hours from mysterious network terminals/workstations that you do not access the network from. It is highly likely in these cases that someone has gained illegal access to the network using your account.

Please kindly inform your system administrator about this intruder access as soon as possible.

Use the history command to determine if any suspicious commands have been entered to gain unauthorized access to your/the company's documents as well as to conduct poor attempts to gain passwords of other users on the network, change your password or access and make use of proprietary software illegally.

6. Printing

6.1. *How do I print my documents in the UNIX xterm terminal?*

If the user wants to print in monochrome, type:

```
lp -d Printer-Name File-Name [Enter]
```

If the user wants to print in colour, try:

```
lp -o A@ -d Printer-Name File-Name
```

where the symbol @ indicates the size of the paper to be printed on. For example, replace @ with the number 4 to print on sheets of A4 paper.

OR

```
lp -o "-tag=A@" -d Printer-Name File-Name
```

where the symbol @ indicates the size of the paper to be printed on. For example, replace @ with the number 4 to print on sheets of A4 paper.

If you have entered a printer-name that is not connected to the network, the following message should appear:

```
Printer-Name: Unknown Printer
```

If this happens, please kindly check that you have correctly spelt the name of the printer.

If you do not know the name of the printer in the department's network, see [Section 6.3](#) for more details.

6.2. *The printer is not printing my document, how can I check the status of my print job?*

Firstly, check the status of the printer that you sent your print job to:

```
lpstat -pPrinter-Name [Enter]
```

Secondly, find out the status of print jobs on printers. Try either of the following:

```
lpq -Pprinter-name [Enter]
```

OR

```
lpstat -o Printer-Name [Enter]
```

OR

```
lpstat -u all [Enter]
```

A list of print jobs as well as their print job IDs will be displayed.

When you have entered the correct command to print a file, the printer may take some time before you can collect your printout as your print job joins the printer queue. That is, the printer may be busy printing other files due to print jobs requested before yours.

Hence, you can check the status of your print jobs by using the aforementioned commands. They will allow you to find out if it is being printed or queued.

If your print job does not appear using these commands, please kindly contact your system administrators.

You should check the status of the printer that you send the print job to because the printer may not have been powered on or connected to the network.

6.3. *How do I find out the name of the printer in the department's network?*

On the command line, type:

```
admintool &
```

Go to its toolbar, click on the “Browse” tab and select the “Printers” option.

A list of Printer-Names, the servers that they are connected to as well as a brief description of the printer will be displayed.

6.4. *How do I remove a print job?*

First, use the `lpstat` command to determine the job ID of the print job that you wish to remove from the printer's queue. See [Section 6.2](#) for more information.

Secondly, type either of the following to remove a print job with the job ID.

```
lprm jobID [Enter]
```

OR

```
Cancel jobID [Enter]
```

Sometimes the cancel command may not flush the buffer in the printer of your print job. Hence, either use the preferable lprm command or take the following steps:

- Disconnect the printer from the network
- Cancel the print job as aforementioned
- Cancel or reset the printer