

# Saul Coval Computers

http://www.coval.net

## Basic Solaris Commands

### Quick Reference Card

#### Conventions

**<CR>** RETURN key  
**<ESC>** ESCAPE key  
**<DEL>** DELETE key  
**<Ctrl-X>** press <Control> key and type x  
*italics* items to be replaced by your own requirements

#### Shell Commands

`passwd` Change password  
`logout` End terminal session

#### File Hierarchy

`cd dir1` Change to directory *dir1*  
`ls` List files in directory  
`ls -l` List files in detail  
`mkdir dir1` Create new directory *dir1*  
`rmdir dir1` Remove directory *dir1*  
`cp f1 f2` Copy file *f1* to *f2*  
`mv f1 [f2...] dir1` Move files *f1* to *fn* to directory *dir1*  
`mv dir1 dir2` Rename directory *dir1* as *dir2*  
`rm filename` Delete (remove) file *filename*  
`ln file1 name` Create a hard link to *file1* called *name*  
`ln -s file1 name` Create a soft link to *file1* called *name*  
`pwd` Show path to current directory

#### Getting Help

`man name` Show man page for command *name*  
`man -k subject` Show man pages relating to *subject*  
`man -s# subject` Show man page relating to *subject* in section number #  
`man -s# Intro` Show introductory man page for section #

#### File types and Listing

`file filename` Classify the file  
`strings filename` Show any ASCII strings in a file  
`cat filename` Display contents of file to STDOUT  
`more filename` Display contents of file one screenful at a time  
`head filename` Display first 10 lines of file  
`head -n filename` Display first *n* lines of file  
`tail filename` Display last 10 lines of file  
`tail -n filename` Display last *n* lines of file  
`tail -f filename` Recursively display last 10 lines of file  
`cut` Extract character or fields from text  
`wc filename` Count lines, words and characters in file  
`diff f1 f2` Find differences between two files  
`diff3 f1 f2 f3` Find differences between 3 files  
`sort filename` Sort file alphabetically by first letter  
`uniq` Report or filter out repeated lines

#### Redirection

`STDIN` Standard Input, typically the keyboard  
`STDOUT` Standard Output, typically the screen  
`STDERR` Standard Error, where errors are sent. Typically the screen  
`comm > file` Output of `comm` creates `file`  
`comm < file` Contents of `file` used as input to `comm`  
`comm 2> filename` Error messages from `comm` sent to `file`  
`comm >> filename` Output from `comm` appended to `file`  
`cat file <<EOF data...` Create a document called `file` containing `data`  
`EOF`  
`comm1 | comm2` Output from `comm1` used as input to `comm2`  
`mkfifo name` Create a named pipe called `name`  
`mknod name p` Create a named pipe called `name`

#### File Security

`chmod mode filename` Change security settings on file  
`chown user [:group] filename` Change owner [and owning group] of file  
`chgrp group filename` Change owning group of file  
`umask mode` Set default creation permissions

#### Symbolic modes

Mode	Meaning	Mode	Meaning
u	user	r	read
g	group	w	write
o	other	x	execute
a	all	u+s	setuid
=	assign	g+s	setgid
+	add	+t	sticky bit
-	remove		

#### Octal modes

Base directory mode is 777. Base file mode is 666

user			group			other		
r	w	x	r	w	x	r	w	x
4	2	1	4	2	1	4	2	1
<b>7</b>			<b>7</b>			<b>7</b>		

#### Shell Wildcards

Meta-character	Meaning
*	Any character
?	Any single character
[ ]	A range of characters

## Translations and Searching

tr *set1 set2* Translates *set1* to *set2*  
 sed Powerful text manipulation tool  
 grep *pattern filename* Finds lines containing *pattern* in file  
 grep -v *pattern filename* Finds lines **NOT** containing *pattern* in file  
 grep -i *pattern filename* Finds all lines containing *pattern* in file ignoring case  
 find *path condition* Finds files matching *condition* from *path* downwards  
 find *path -inum n* Finds hard links, i.e. All files with the same i-node number  
 who Displays users on system  
 who am i Shows real user id  
 w Displays users on system  
 id Shows effective username & UID, and group membership  
 look *word* Searches /usr/dict/words for *word*

## Regular Expressions

.	Any character
^	Start of line
\$	End of line
*	Any number of the preceding characters
?	A single character
[ ]	Holds a range
[^ ]	Holds a negated range
\ ( \)	Creates a submatch
\ 1-9	Reference a submatch
\	Escapes special character meanings

## Networking

telnet *hostname [port]* Connects to host and opens a shell. Optionally on specified port.  
 ftp *hostname* Connects to a remote host to transfer files  
 ssh *hostname [command]* Makes a secure connection to host and opens a shell.

## Processes and Process Control

ps Displays processes running on a host  
 prstat Displays iterating list of processes by CPU usage  
 command & Run *command* in background  
 jobs Print list of jobs  
 fg [%n] Resume foreground job *n*  
 bg [%n] Resume background job *n*  
 stop %n Suspend background job *n*  
 kill [%n] Kill job *n*  
 <Ctrl-c> Interrupt process  
 <Ctrl-z> Suspend current process  
 kill *n* Kill process *n*  
 kill -9 *n* Terminate process *n*  
 <Ctrl-s> Stop screen scrolling  
 <Ctrl-q> Resume screen output  
 sleep *n* Sleep for *n* seconds

## Shells and Variables

*variable=value* Create local variable *variable* with value *value*  
 export *variable* Make *variable* an environmental variable  
 unset *variable* Remove environment variable  
 set Show local variables  
 env Show environmental variables  
 alias *name1 name2* Create command alias  
 alias Show command aliases  
 unalias *name1* Remove command alias *name1*  
 history Display recent commands  
 ! *n* Submit recent command *n*  
 set -o vi Recall commands, edit and re-execute using vi commands

set -o emacs Recall commands, edit and re-execute using emacs commands

## Shell Initialization

**/etc/profile (\$HOME/.profile)**  
 sh, bash, ksh system wide (per user) init (system wide init has no effect in CDE environment).  
**/etc/.login (\$HOME/.login)**  
 csh, tcsh system wide (per user) init (system wide init has no effect in CDE environment).

## \$HOME/.cshrc

per user csh, tcsh init  
 (order: /etc/.login → \$HOME/.cshrc → \$HOME/.login)

## \$HOME/.kshrc

per user ksh init  
 (order: /etc/profile → \$HOME/.profile → \$HOME/.kshrc)

## \$HOME/.bashrc

per user bash init  
 (order: /etc/profile → \$HOME/.profile → \$HOME/.bashrc)

## Shell Programming

#!*path/to/shell* 'sh-bang' is a special string which indicates that the path following contains the location of the command to run the script

## Flow Control

	sh;bash;ksh	csh;tcsh
if-then-else	if [ <i>condition</i> ]; then <i>actions_1</i> ; elif [ <i>condition</i> ]; then <i>actions_2</i> ; else <i>actions_3</i> ; fi	if ( <i>condition</i> ) then <i>action_1</i> ; else if ( <i>condition</i> ) then <i>action_2</i> ; else <i>action_3</i> ; endif
do while	while [ <i>condition</i> ]; do <i>actions</i> ; done	while ( <i>condition</i> ) <i>actions</i> end
until	until [ <i>condition</i> ]; do <i>actions</i> ; done	<b>Not available</b>
for	for <i>arg</i> in <i>list...</i> ; do <i>actions</i> ; done	foreach <i>arg</i> ( <i>list</i> ) <i>actions</i> end