

Shell

Manpage

Most of Unix systems are managed by using Shell. Just as you need to know a minimum number of words to have a discussion in a language, you need to know a minimum number of commands to be able to easily interact with a system. Unix systems all have, sometimes with slight differences, the same set of commands. While it is not too hard to remember commands, it might be hard to remember all of their options and how exactly to use them. The solution to this is the `man` command. Let's go through a part of the `ssh` one, as there are few elements to know to be able to read a man page:

NAME

`ssh` – OpenSSH SSH client (remote login program)

SYNOPSIS

```
ssh [-1246AaCfGkKMNnqsTtVvXxYy] [-b bind_address] [-c cipher_spec] [-D [bind_address:]port] [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11] [-i identity_file] [-L [bind_address:]port:host:hostport] [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port] [-Q cipher | cipher-auth | mac | kex | key] [-R [bind_address:]port:host:hostport] [-S ctl_path] [-W host:port] [-w local_tun[:remote_tun]] [user@]hostname [command]
```

DESCRIPTION

`ssh` (SSH client) is a program for logging into a remote machine and for executing commands on a remote machine. It is intended to replace `rlogin` and `rsh`, and provide secure encrypted communications between two untrusted hosts over an insecure network. X11 connections and arbitrary TCP ports can also be forwarded over the secure channel.

Some tips:

- The `NAME` will summarize what the command is doing. As it is usually super short, you might want to look at `DESCRIPTION` (bellow) if ever it does not gives clear enough information
- The `SYNOPSIS` will help you to understand the structure of the command:
 - A shell command usually have this format: `command options parameters`
 - Options inside `[]` are optional
 - The string without `[]` are mandatory
- `ssh [-1246AaCfGkKMNnqsTtVvXxYy] [-D [bind_address:]port]`
 - `ssh` is mandatory
 - `-1246AaCfGkKMNnqsTtVvXxYy` is optional
 - `-D [bind_address:]port` is optional (with `bind_address:` being itself optional within `-D [bind_address:]port`)

Commands

Here is the (non-exhaustive) list of commands & concepts you should master to be verbose with Unix systems:

awk # pattern scanning and processing language
basename # strip directory and suffix from filenames
bg # resumes suspended jobs without bringing them to the foreground
cat # print files
cd # change the shell working directory.
chmod # change file mode
chown # change file owner and group
crontab # maintain crontab files
curl # transfer a URL
cut # remove sections from each line of files
date # display or set date and time
dig # DNS lookup utility
df # report file system disk space usage
diff # compare files line by line
du # estimate file space usage
echo # display a line of text
find # search for files in a directory hierarchy
fg # resumes suspended jobs and bring them to the foreground
grep # print lines matching a pattern
kill # send a signal to a process
less # read file with pagination
ln # create links
ls # list directory contents
lsb_release # print distribution-specific information
lsof # list open files
mkdir # create
mv # move files
nc # arbitrary TCP and UDP connections and listens
netstat # print network connections, routing tables, interface statistics...
nice # execute a utility with an altered scheduling priority
nproc # print the number of processing units available
passwd # change user password
pgrep # look up processes based on name and other attributes
pkill # send signal to processes based on name and other attributes
printenv # print all or part of environment
pwd # print name of current/working directory
top # display Linux processes
tr # translate or delete characters
ps # report a snapshot of the current processes
rm # remove files or directories
rmdir # remove directories
rsync # remote file copy
scp # secure copy (remote file copy program)
sed # stream editor for filtering and transforming text
sleep # suspend execution for an interval of time
sort # sort lines of text file
ssh # OpenSSH SSH client (remote login program)
ssh-keygen # SSH key generation, management and conversion
su # substitute user identity
sudo # execute a command as another user
tail # output the last part of files
tar # manipulate archives files
tr # translate or delete characters
uname # Print operating system name
uniq # report or omit repeated lines
uptime # show how long system has been running
w # Show who is logged on and what they are doing
whereis # locate the binary, source, and manual page files for a command
which # locate a command

```
wc # print newline, word, and byte counts for each file
xargs # build and execute command lines from standard input
| # redirect standard output to another command
> # redirect standard output
< # redirect standard input
& # send process to background
```

Shortcuts

Some handy shortcuts:

```
CTRL+A # go to beginning of line
CTRL+B # moves backward one character
CTRL+C # stops the current command
CTRL+D # deletes one character backward or logs out of current session
CTRL+E # go to end of line
CTRL+F # moves forward one character
CTRL+G # aborts the current editing command and ring the terminal bell
CTRL+K # deletes (kill) forward to end of line
CTRL+L # clears screen and redisplay the line
CTRL+N # next line in command history
CTRL+R # searches in your command history
CTRL+T # transposes two characters
CTRL+U # kills backward to the beginning of line
CTRL+W # kills the word behind the cursor
CTRL+Y # retrieves last deleted string
CTRL+Z # stops the current command, resume with fg in the foreground or bg in the background
```