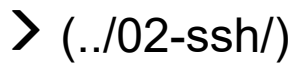


Extra Unix Shell Material (../)



Manual Pages

? Overview

Teaching: 10 min

Exercises: 0 min

Questions

- How to use man pages?

Objectives

- Use `man` to display the manual page for a given command.
- Explain how to read the synopsis of a given command while using `man`.
- Search for specific options or flags in the manual page for a given command.

We can get help for any Unix command with the `man` (short for manual) command. For example, here is the command to look up information on `cp`:

```
$ man cp
```

The output displayed is referred to as the “man page”.

The man page will be displayed in the default file viewer for our shell, which usually a program called `more`. When `more` displays a colon ‘:’, we can press the space bar to get the next page, the letter ‘h’ to get help, or the letter ‘q’ to quit.

`man`’s output is typically complete but concise, as it is designed to be used as a reference rather than a tutorial. Most man pages are divided into sections:

- **NAME:** gives the name of the command and a brief description
- **SYNOPSIS:** how to run the command, including optional and mandatory parameters. (We will explain the syntax later.)
- **DESCRIPTION:** a fuller description than the synopsis, including a description of all the options to the command. This section may also include example usage or details about how the command works.
- **EXAMPLES:** self-explanatory.
- **SEE ALSO:** list other commands that we might find useful or other sources of information that might help us.

Other sections we might see include **AUTHOR**, **REPORTING BUGS**, **COPYRIGHT**, **HISTORY**, **(known) BUGS**, and **COMPATIBILITY**.

How to Read the Synopsis

Here is the synopsis for the `cp` command on Ubuntu Linux:

SYNOPSIS

```
cp [OPTION]... [-T] SOURCE DEST
cp [OPTION]... SOURCE... DIRECTORY
cp [OPTION]... -t DIRECTORY SOURCE...
```

This tells the reader that there are three ways to use the command. Let's look at the first usage:

```
cp [OPTION]... [-T] SOURCE DEST
```

[OPTION] means the `cp` command can be followed by one or more optional flags (`./reference/#command`). We can tell they're optional because of the square brackets, and we can tell that one or more are welcome because of the ellipsis (...). For example, the fact that `[-T]` is in square brackets, but after the ellipsis, means that it's optional, but if it's used, it must come after all the other options.

`SOURCE` refers to the source file or directory, and `DEST` to the destination file or directory. Their precise meanings are explained at the top of the `DESCRIPTION` section.

The other two usage examples can be read in similar ways. Note that to use the last one, the `-t` option is mandatory (because it isn't shown in square brackets).

The `DESCRIPTION` section starts with a few paragraphs explaining the command and its use, then expands on the possible options one by one:

The following options are available:

```
-a    Same as -pPR options. Preserves structure and attributes of
      files but not directory structure.

-f    If the destination file cannot be opened, remove it and create
      a new file, without prompting for confirmation regardless of
      its permissions. (The -f option overrides any previous -n
      option.)
```

The target file is not unlinked before the copy. Thus, any existing access rights will be retained.

... ..

Finding Help on Specific Options

If we want to skip ahead to the option you're interested in, we can search for it using the slash key `/`. (This isn't part of the `man` command: it's a feature of `more`.) For example, to find out about `-t`, we can type `/-t` and press return. After that, we can use the `n` key to navigate to the next match until we find the detailed information we need:

```
-t, --target-directory=DIRECTORY
    copy all SOURCE arguments into DIRECTORY
```

This means that this option has the short form `-t` and the long form `--target-directory` and that it takes an argument. Its meaning is to copy all the `SOURCE` arguments into `DIRECTORY`. Thus, we can give the destination explicitly instead of relying on having to place the directory at the end.

Limitations of Man Pages

Man pages can be useful for a quick confirmation of how to run a command, but they are not famous for being readable. If you can't find what you need in the man page— or you can't understand what you've found— try entering “unix command copy file” into your favorite search engine: it will often produce more helpful results.

✈ You May Also Enjoy...

The explainshell.com (<http://explainshell.com/>) site does a great job of breaking complex Unix commands into parts and explaining what each does. Sadly, it doesn't work in reverse...

! Key Points

- `man command` displays the manual page for a given command.
- `[OPTION]...` means the given command can be followed by one or more optional flags.
- Flags specified after ellipsis are still optional but must come after all other flags.
- While inside the manual page, use `/` followed by your pattern to do interactive searching.

^ (../)

> (../02-ssh/)

Copyright © 2016 - 2019 Software Carpentry Foundation (<https://software-carpentry.org>)

Edit on GitHub (https://github.com/swcarpentry/shell-extras/edit/gh-pages/_episodes/01-man-pages.md) / Source (<https://github.com/swcarpentry/shell-extras/>) / Contributing ([../contributing/](https://github.com/swcarpentry/shell-extras/blob/master/CONTRIBUTING.md)) / Cite ([../citation/](https://github.com/swcarpentry/shell-extras/blob/master/CITATION.md)) / Contact (<mailto:lessons@software-carpentry.org>)