

OPEN-SOURCE EBOOK

++101 LINUX COMMANDS

BOBBY ILIEV

Table of Contents

101 Linux commands Open-source eBook	15
Hacktoberfest	16
About me	17
Ebook PDF Generation Tool	19
Book Cover	20
License	21
The ls command	22
The cd command	24
The cat command	26
The tac command	29
The head command	31
The tail command	33
The pwd command	36
The touch Command	38
The cal Command	41
The bc command	44

The df command	47
The help command	50
Syntax	51
Options	52
Example	53
The factor command	54
Syntax	55
Options	56
Examples	57
The uname command	58
Syntax:	59
Examples	60
Options	61
The mkdir command	62
Syntax	63
Examples	64
Options	65
The gzip command	66
Usage	67
Compress a file	68
Decompress a file	69
Compress multiple files:	70
Decompress multiple files:	71

Show memory usage in human-readable form

Action: --- Output the memory usage - available and used, as well as swap

Details: --- Outputted values ARE human-readable (are in GB / MB)

Command:

```
free -h
```

The `top/htop` command

`top` is the default command-line utility that comes pre-installed on Linux distributions and Unix-like operating systems. It is used for displaying information about the system and its top CPU-consuming processes as well as RAM usage.

`htop` is interactive process-viewer and process-manager for Linux and Unix-like operating system based on ncurses. If you take `top` and put it on steroids, you get `htop`.

Comparison between top and htop:

Feature	top	htop
Type	Interactive system-monitor, process-viewer and process-manager	Interactive system-monitor, process-viewer and process-manager
Operating System	Linux distributions, macOS	Linux distributions, macOS
Installation	Built-in and is always there. Also has more adoption due to this fact.	Doesn't come preinstalled on most Linux distros. Manual installation is needed
User Interface	Basic text only	Colorful and nicer text-graphics interface
Scrolling Support	No	Yes, supports horizontal and vertical scrolling
Mouse Support	No	Yes
Process utilization	Displays processes but not in tree format	Yes, including user and kernel threads
Scrolling Support	No	Yes, supports horizontal and vertical scrolling
Mouse Support	No	Yes
Process utilization	Displays processes but not in tree format	Yes, including user and kernel threads
Network Utilization	No	No
Disk Utilization	No	No
Comments	Has a learning curve for some advanced options like searching, sending messages to processes, etc. It is good to have some knowledge of top because it is the default process viewer on many systems.	Easier to use and supports vi like searching with <code>/</code> . Sending messages to processes (kill, renice) is easier and doesn't require typing in the process number like top.

Examples:

top

1. To display dynamic real-time information about running processes:

```
top
```

2. Sorting processes by internal memory size (default order - process ID):

```
top -o mem
```

3. Sorting processes first by CPU, then by running time:

```
top -o cpu -O time
```

4. Display only processes owned by given user:

```
top -user {user_name}
```

htop

1. Display dynamic real-time information about running processes. An enhanced version of **top**.

```
htop
```

2. displaying processes owned by a specific user:

```
htop --user {user_name}
```

3. Sort processes by a specified **sort_item** (use **htop --sort help** for

available options):

```
htop --sort {sort_item}
```


Syntax:

```
top [OPTIONS]
```

```
htop [OPTIONS]
```

The syntax of the `passwd` command is :

```
$ passwd [options] [LOGIN]
```

options

`-a, --all`
This option can be used only with `-S` and causes show status **for** all users.

`-d, --delete`
Delete a user's **password**.

`-e, --expire`
Immediately expire an account's password.

`-h, --help`
Display help message and exit.

`-i, --inactive`
This option is used to disable an account after the password has been expired **for** a number of days.

`-k, --keep-tokens`
Indicate password change should be performed only **for** expired authentication tokens (passwords).

`-l, --lock`
Lock the password of the named account.

`-q, --quiet`
Quiet mode.

`-r, --repository`
change password **in** repository.

`-S, --status`
Display account status information.

The **w** command

The **w** command displays information about the users that are currently active on the machine and their [processes](#).

Examples:

1. Running the **w** command without [arguments](#) shows a list of logged on users and their processes.

```
w
```

2. Show information for the user named *hope*.

```
w hope
```

Syntax:

```
finger [-l] [-m] [-p] [-s] [username]
```

Additional Flags and their Functionalities:

Short Flag	Long Flag	Description
-h	--no-header	Don't print the header.
-u	--no-current	Ignores the username while figuring out the current process and cpu times. <i>(To see an example of this, switch to the root user with su and then run both w and w -u.)</i>
-s	--short	Display abbreviated output <i>(don't print the login time, JCPU or PCPU times).</i>

Short Flag	Long Flag	Description
<code>-f</code>	<code>--from</code>	Toggle printing the from (<i>remote hostname</i>) field. The default as released is for the from field to not be printed, although your system administrator or distribution maintainer may have compiled a version where the from field is shown by default.
<code>--help</code>	-	Display a help message, and exit.
<code>-V</code>	<code>--version</code>	Display version information, and exit.
<code>-o</code>	<code>--old-style</code>	Old style output (<i>prints blank space for idle times less than one minute</i>).
<code>user</code>	-	Show information about the specified the user only.

Additional Information

The header of the output shows (in this order): the current time, how long the system has been running, how many users are currently logged on, and the system load averages for the past 1, 5, and 15 minutes.

The following entries are displayed for each user:

- login name the tty
- name the remote
- host they are
- logged in from the amount of time they are logged in their
- idle time JCPU
- PCPU
- command line of their current process

The JCPU time is the time used by all processes attached to the tty. It does not include past background jobs, but does include currently running background jobs.

The PCPU time is the time used by the current process, named in the "what" field.

This is a sample from "101 Linux Commands eBook" by Bobby Iliev the Hacktoberfest
community.

For more information, [Click here](#).