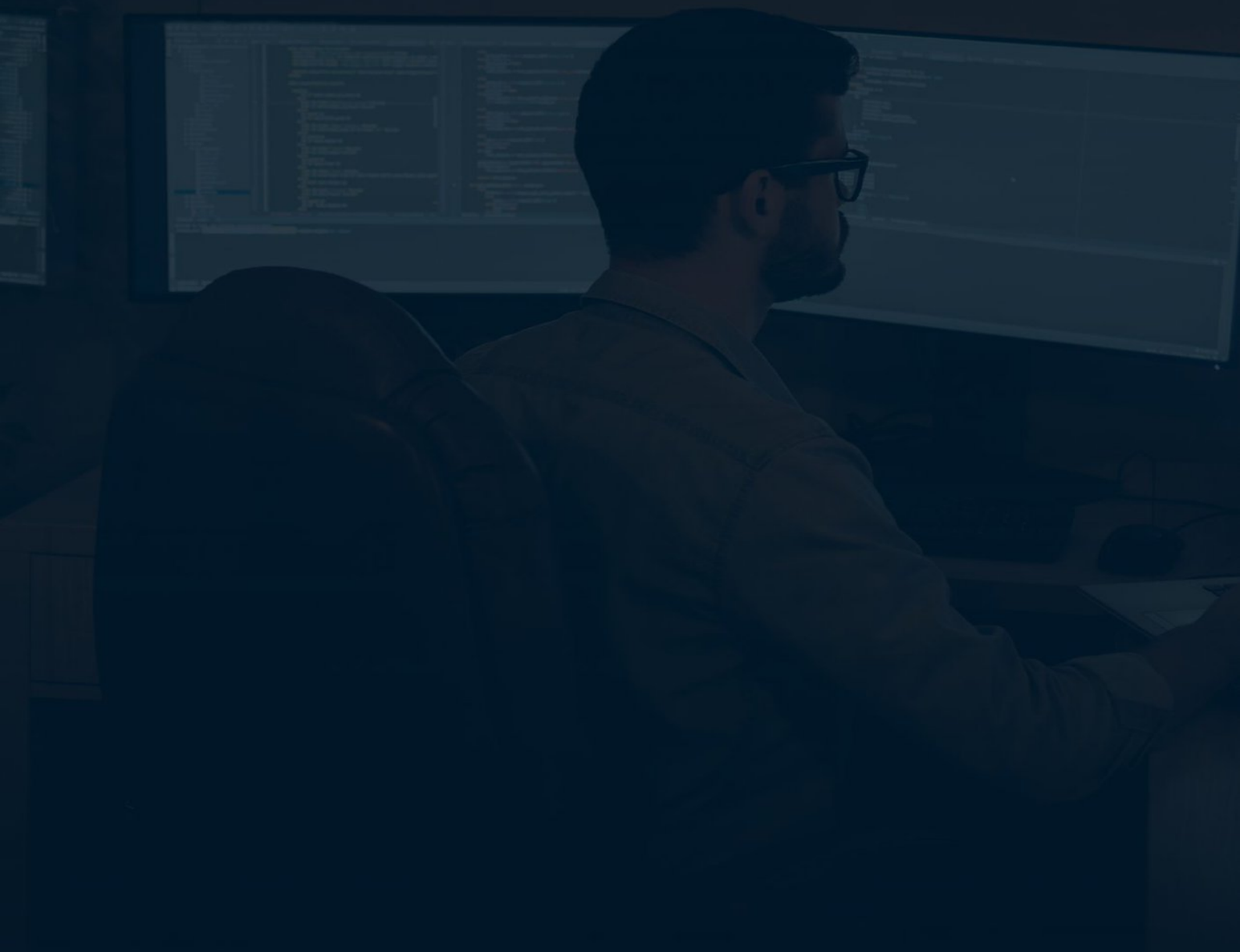
OPEN-SOURCE EBOOK

++101 Linux Commands



BOBBY ILIEV

Table of Contents

101 Linux commands Open-source eBook

This is an open-source eBook with 101 Linux commands that everyone should know. No matter if you are a DevOps/SysOps engineer, developer, or just a Linux enthusiast, you will most likely have to use the terminal at some point in your career.

The `free` command

The `free` command in Linux/Unix is used to show memory (RAM/SWAP) information.

Usage

Show memory usage

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

Details: --- The values are shown in kibibytes by default.

Command:

```
free
```

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
```

Show memory usage with a total line

Action: --- Output the memory usage and also add a summary line with the total.

Details: --- The `-t` flag is useful for seeing the combined total of memory and swap.

Command:

```
free -t
```

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.

The `in.fingerd` Service

It’s important to distinguish between the `finger` command and the **`in.fingerd` service**. The `finger` command is local, while `in.fingerd` is a network daemon that allows remote queries of user information. This service is typically disabled by default in modern systems due to potential security risks.

If enabled, the `in.fingerd` service can expose user information over the network, which could be exploited by attackers. To mitigate this risk, system administrators should ensure the service is disabled if it is not needed.

Disabling the `in.fingerd` Service:

If you are concerned about remote queries, you can disable the `in.fingerd` service:

```
sudo systemctl disable in.fingerd
sudo systemctl stop in.fingerd
```

By disabling the `in.fingerd` service, you prevent remote querying of user information, enhancing system security.

The `groups` command

In Linux, there can be multiple users (those who use/operate the system), and groups (a collection of users). Groups make it easy to manage users with the same security and access privileges. A user can be part of different groups.

Important Points:

The `groups` command prints the names of the primary and any supplementary groups for each given username, or the current process if no names are given. If more than one name is given, the name of each user is printed before the list of that user's groups and the username is separated from the group list by a colon.

Syntax:

```
groups [username]
```

Example 1

Provided with a username

```
groups demon
```

In this example, username demon is passed with groups command and the output shows the groups in which the user demon is present, separated by a colon.

Example 2

When no username is passed then this will display the group membership for the current user:

```
groups
```

Here the current user is demon . So when we run the `groups` command without arguments we get the groups in which demon is a user.

Example 3

Passing root with groups command:

```
$demon# groups
```

Note: Primary and supplementary groups for a process are normally inherited from its parent and are usually unchanged since login. This means that if you change the group database after logging in, groups will not reflect your changes within your existing login session. The only options are –help and –version.

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

For more information, Click here.