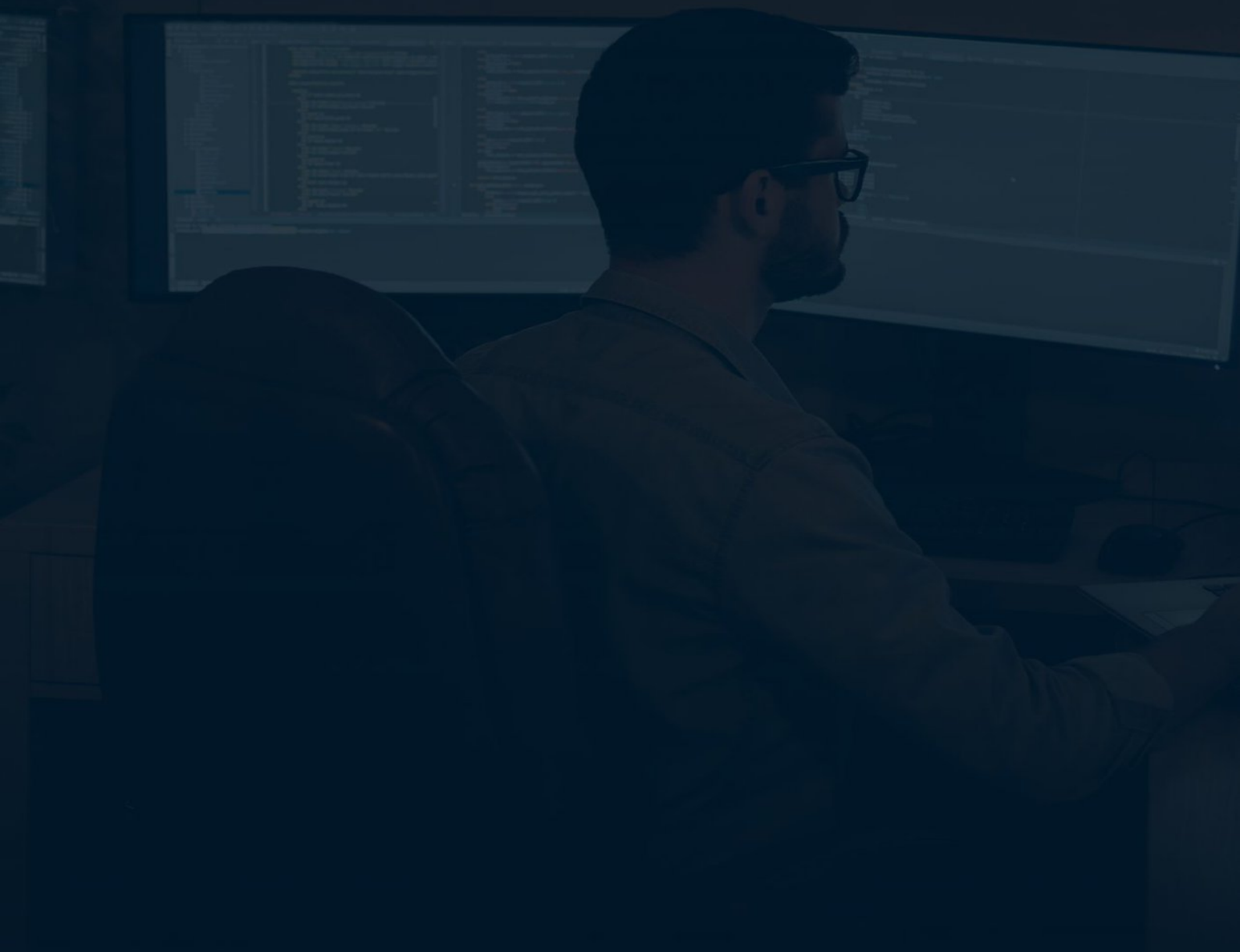
OPEN-SOURCE EBOOK

++101 Linux Commands



BOBBY ILIEV

Table of Contents

The `who` command

The `who` command lets you print out a list of logged-in users, the current run level of the system and the time of last system boot.

Examples

1. Print out all details of currently logged-in users

```
who -a
```

2. Print out the list of all dead processes

```
who -d -H
```

Syntax:

```
who [options] [filename]
```

Additional Flags and their Functionalities

Short Flag	Description
`-r`	prints all the current runlevel
`-d`	print all the dead processes
`-q`	print all the login names and total number of logged on users
`-h`	print the heading of the columns displayed

Short Flag	Description
`-b`	print the time of last system boot

018-the-free-command.md

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: --- Outputted values are not human-readable (are in bytes)

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
```

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.

The `man` command

The `man` command is used to display the manual of any command that we can run on the terminal. It provides information like: DESCRIPTION, OPTIONS, AUTHORS and more.

Examples:

1. Man page for printf:

```
man printf
```

2. Man page section 2 for intro:

```
man 2 intro
```

Syntax:

```
man [SECTION-NUM] [COMMAND NAME]
```

Additional Flags and their Functionalities:

Short Flag	Long Flag	Description
`-f`	-	Return the sections of an command
`-a`	-	Display all the manual pages of an command

Short Flag	Long Flag	Description
`-k`	-	Searches the given command with RegEx in all man pages
`-w`	-	Returns the location of a given command man page
`-I`	-	Searches the command manual case sensitive

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

For more information, Click here.