Use Streams Pipes and Redirects

Introduction

Welcome to the **Use Streams Pipes and Redirects** Practice Lab. In this module you will be provided with the instructions and devices needed to develop your hands-on skills.

Stream Pipes
Redirects
Linux System

Learning Outcomes

In this module, you will complete the following exercise:

- Exercise 1 - Use Streams Pipes and Redirects

After completing this lab, you will be able to:

- Redirect standard input, standard output, and standard error
- Pipe the output of one command to the input of the other command
- Use the output of one command as the input to another command
- Send the output to both stdout and a file

Exam Objectives

The following exam objectives are covered in this lab:

- **LPI:** 103.4 Use streams, pipes, and redirects
- **CompTIA:** 5.1 Given a scenario, deploy and execute basic BASH scripts.

Lab Duration

It will take approximately **1 hour** to complete this lab.

Help and Support

For more information on using Practice Labs, please see our **Help and Support** page. You can also raise a technical support ticket from this page.

Click Next to view the Lab topology used in this module.

Lab Topology

During your session, you will have access to the following lab configuration.

PLABSA01
Windows Server 2016
192.168.0.1

PLABLINUX01
CentOS Server
192.168.0.2

PLABLINUX02
Ubuntu Server
192.168.0.3

Depending on the exercises you may or may not use all of the devices, but they are shown here in the layout to get an overall understanding of the topology of the lab.

- **PLABSA01** (Windows Server 2016)
- **PLABLINUX01** (CentOS Server)
- **PLABLINUX02** (Ubuntu Server)

Exercise 1 - Use Streams, Pipes, and Redirects

The Linux shells use pipes (|) as an operator between two commands, where the output of one command is the input of the second command. Redirects are meant to direct the output to a specific file or take the input from a specific file. In this exercise, you will understand how use streams, pipes, and redirects.

Learning Outcomes

After completing this exercise, you will be able to:

- Log into a Linux System
- Redirect standard input, standard output, and standard error
- Pipe the output of one command to the input of the other command
- Use the output of one command as the input to another command
- Send the output to both stdout and a file

Your Devices

You will be using the following device in this lab. Please power these on now.

- **PLABLINUX02** (Ubuntu Server)

PLABLINUX02
Ubuntu Server
192.168.0.3

Task 1 - Redirect Standard Input, Standard Output, and Standard Error

For each command that you execute in a terminal window, different streams, which are sequences or streams of characters, are used by Linux to get input, output, or to

generate an error. Each of these streams is identified by a unique value associated with it:

- A standard input value 0, which is the default keyboard
- A standard output value 1, which is the default terminal
- Standard output for errors value 2, which is the default terminal

In this task, you will redirect the standard output and the standard error message to a file.

To redirect standard input, output, and error, perform the following steps:

Step 1

On the desktop, right-click and select **Open In Terminal**.

Step 2

To redirect the output to a file named **find.txt** for this, enter the following command:
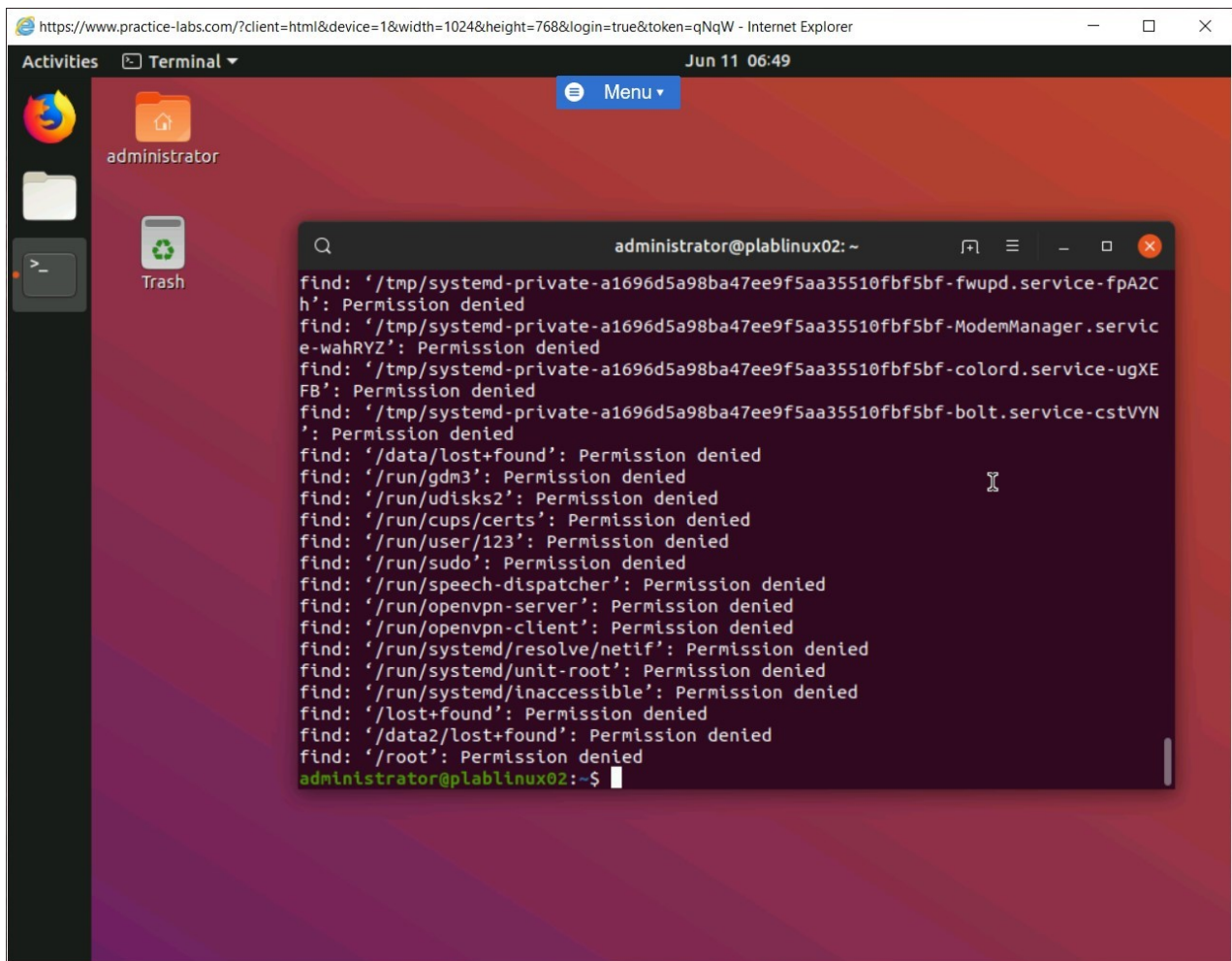
```
find / -iname *.txt > find.txt
```

Figure 1.2 Screenshot of PLABLINUX02: Redirecting the output to a file.

Step 3

Clear the screen by entering the following command:

```
clear
```

Let's verify if the **find.txt** is created. Enter the following command:
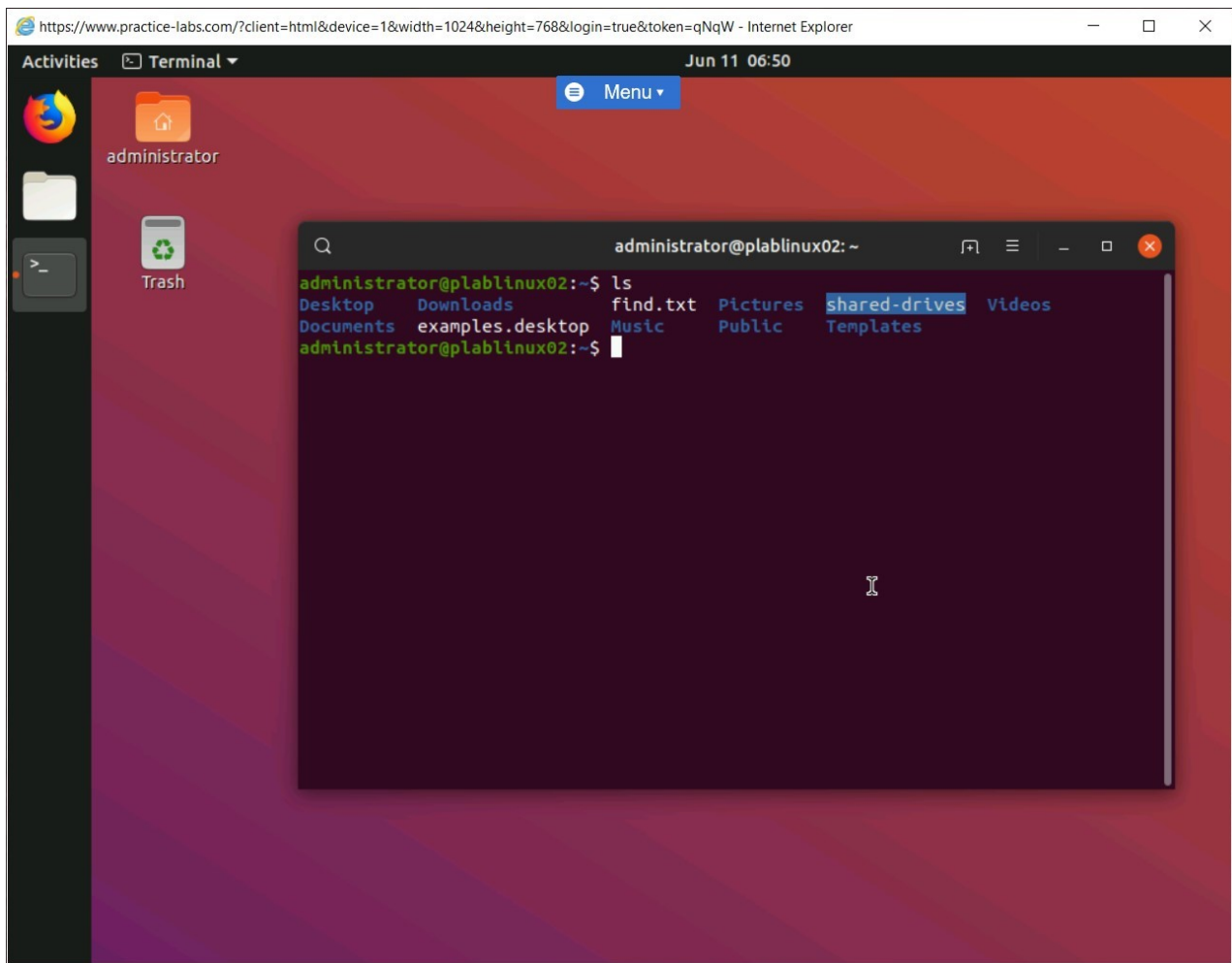
```
ls
```

Note that the file is generated.

Figure 1.3 Screenshot of PLABLINUX02: Verifying whether the file is created.

Step 4

You will now need to verify the contents of the file, which should have the output that you redirected earlier. To verify the contents of the file, enter the following command:
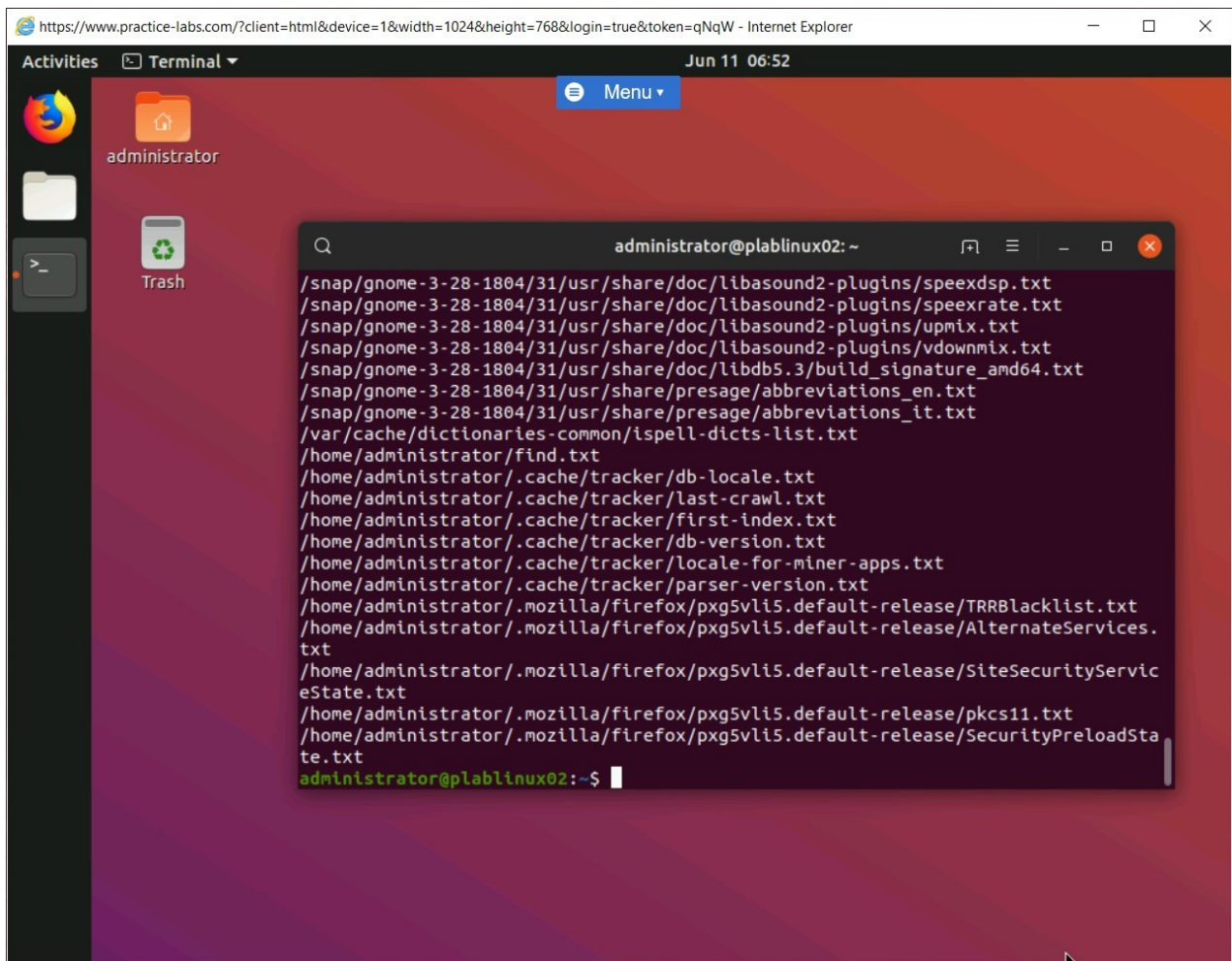
```
cat find.txt
```

Figure 1.4 Screenshot of PLABLINUX02: Verifying the contents of the file.

Step 5

Clear the screen by entering the following command:

```
clear
```

You can append more information into the same file if you do not want to create a new file. To do this, enter the following command:

```
find / -iname *.conf >> find.txt
```

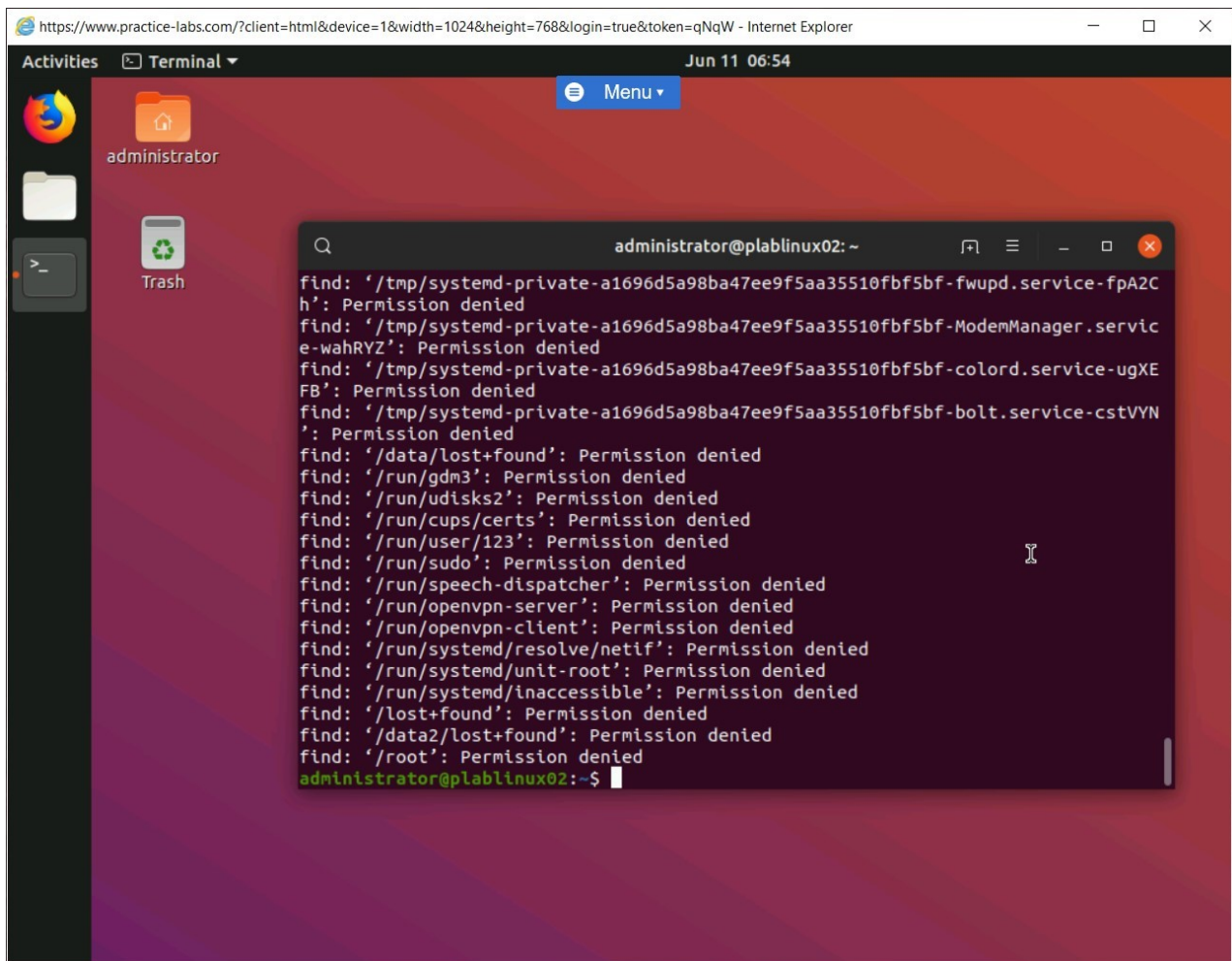Notice the use of a double redirector in this command.

Figure 1.5 Screenshot of PLABLINUX02: Appending more information into the same file.

Step 6

Clear the screen by entering the following command:

```
clear
```

Again, view the contents of the file by using the cat command.

```
cat find.txt
```

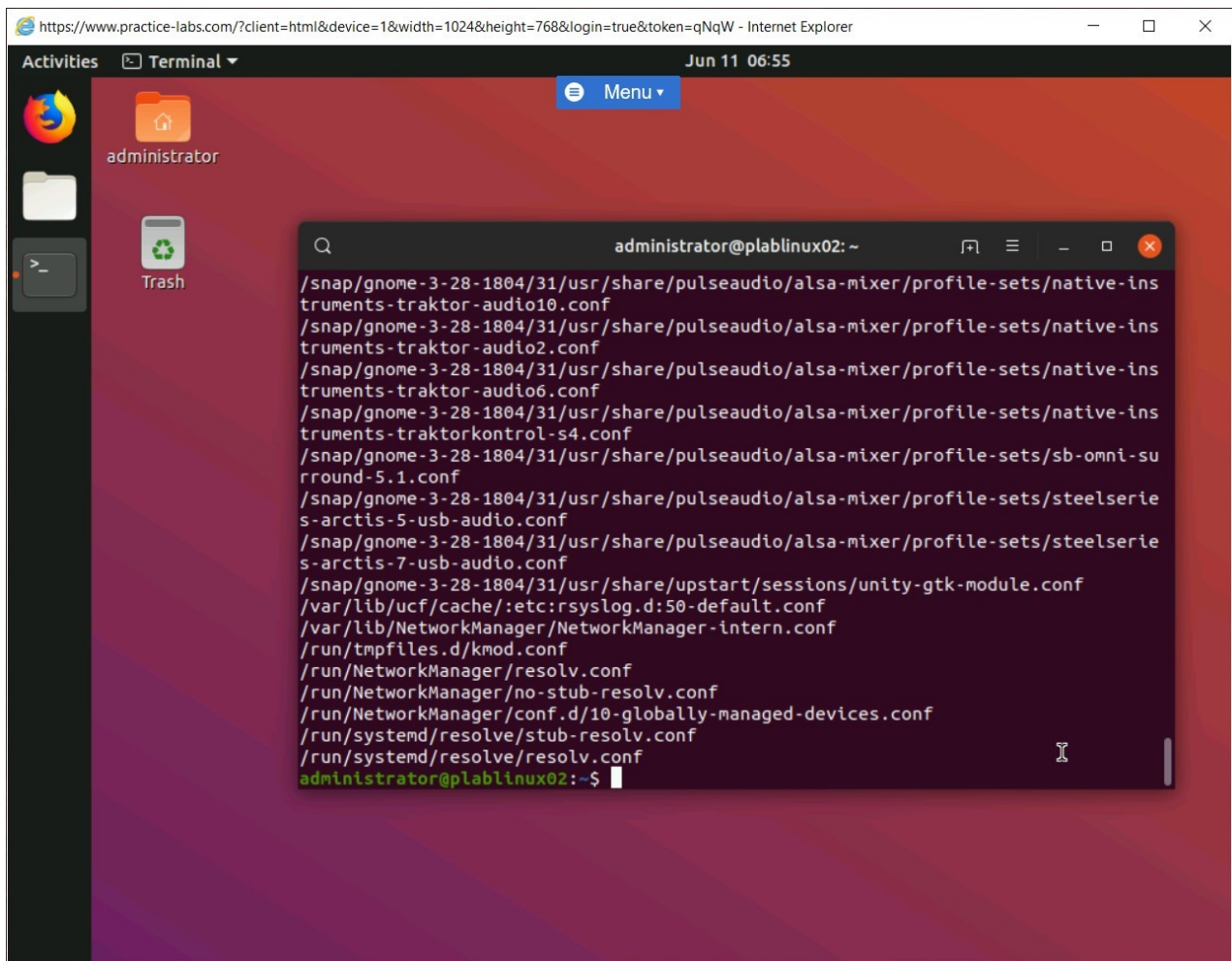Notice the additional contents appended to the file.

Figure 1.6 Screenshot of PLABLINUX02: Viewing the contents of the file.

Step 7

Clear the screen by entering the following command:

```
clear
```

You can redirect the errors to a file just like you redirect the output.

In this task, you will use a command that does not exist so that the error can be redirected to a file. To do this, enter the following command:

```
list 2>test.txt
```

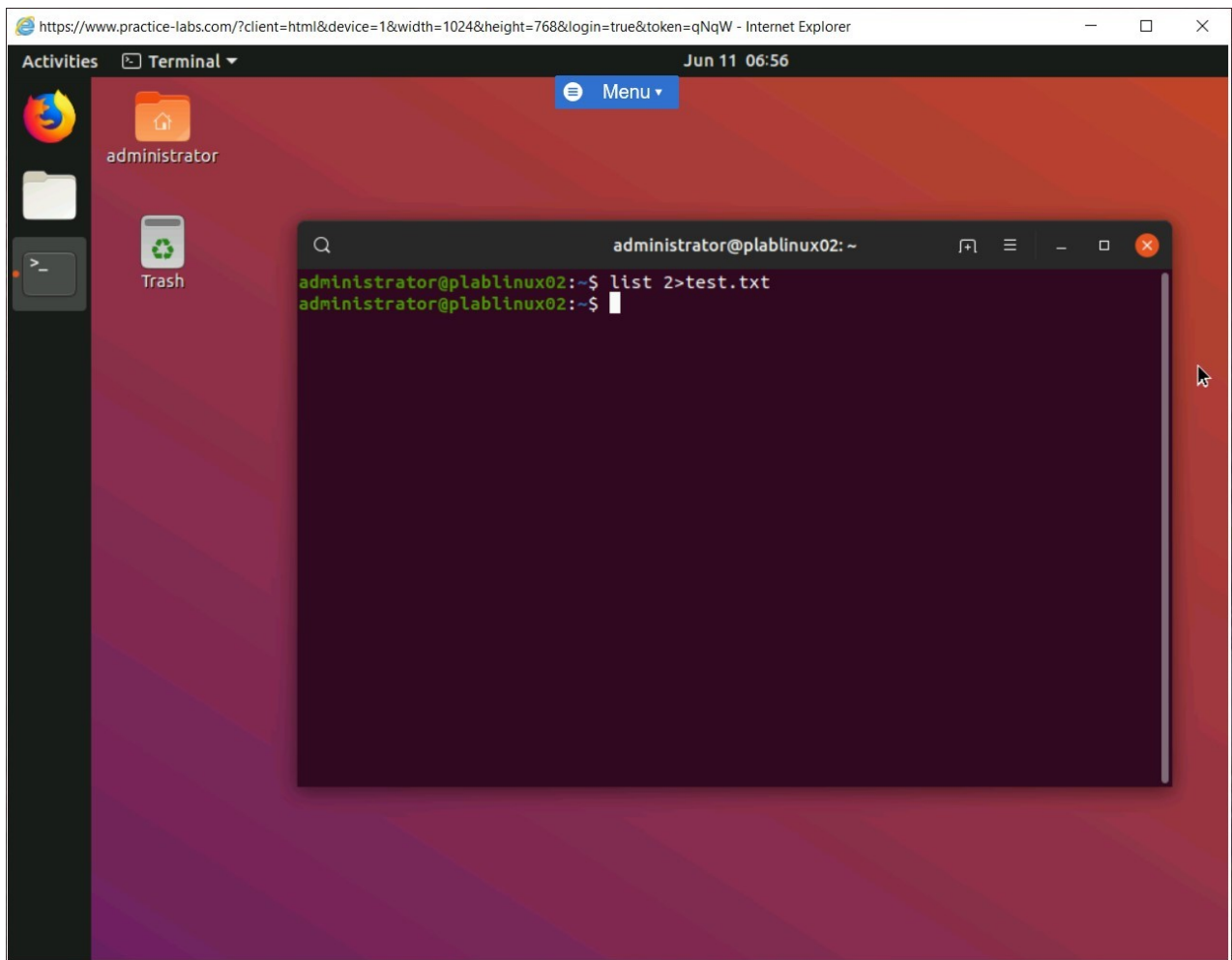Notice the use of a single redirector here.

Figure 1.7 Screenshot of PLABLINUX02: Using a command that does not exist so that the error can be redirected to a file.

Step 8

To view the **test.txt** file, type the following command:

```
cat test.txt
```
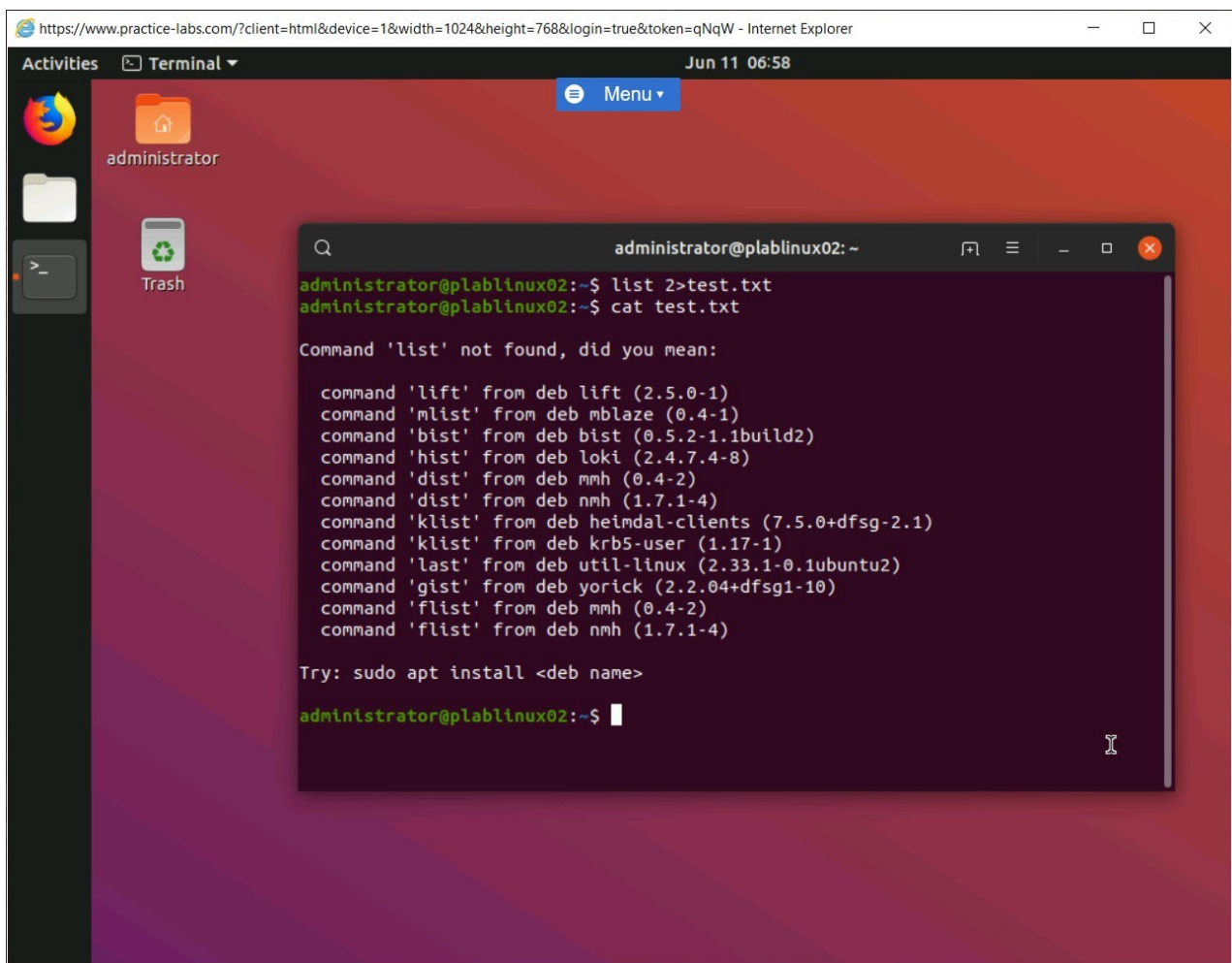
Press **Enter**.

Note that this file contains the error.

Figure 1.8 Screenshot of PLABLINUX02: Viewing the contents of the file.

Step 9

Clear the screen by entering the following command:

```
clear
```

You can also combine the output and error or simply two outputs using &.

This example demonstrates showing one output on the screen and redirecting another to the file. To do this, type the following command:
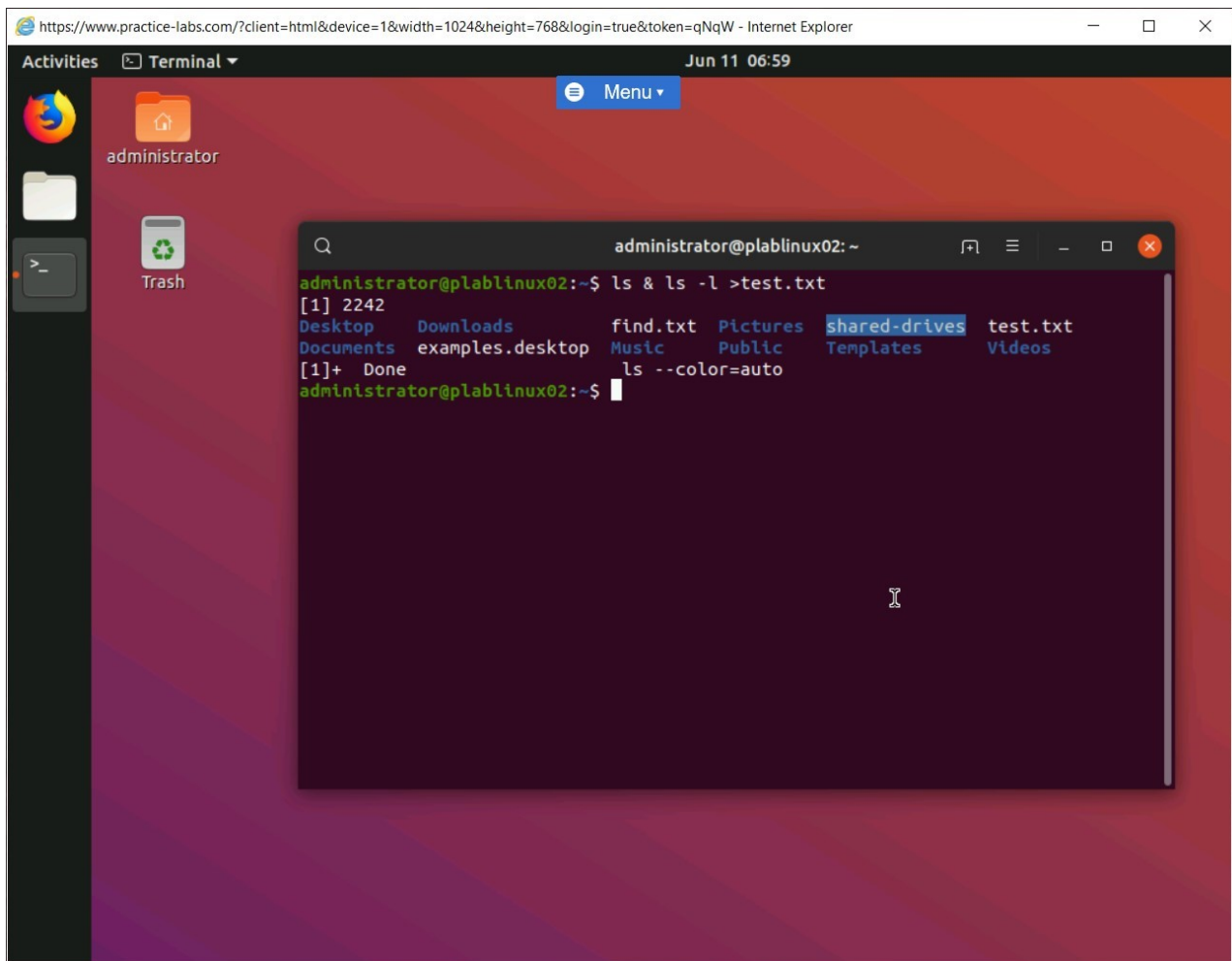
```
ls & ls -l >test.txt
```

Press **Enter**.

Figure 1.9 Screenshot of PLABLINUX02: Showing one output on the screen and redirecting another to the file.

Task 2 - Pipe the Output of one Command to the Input of another Command

Pipes provide a method to apply multiple commands concurrently. This process operates by using the output of one command as the input to the other. In this task, you will pipe the output of one command to the input of the other.

To pipe the output of one command to the input of another command, perform the following steps:

Step 1

Clear the screen by entering the following command:

```
clear
```

To redirect the standard output of the first command to the standard input of the second command, enter the following command:

```
ls -l /dev | more
```

The more parameter is used for pagination. You will need to press **Enter** to continue reading further.
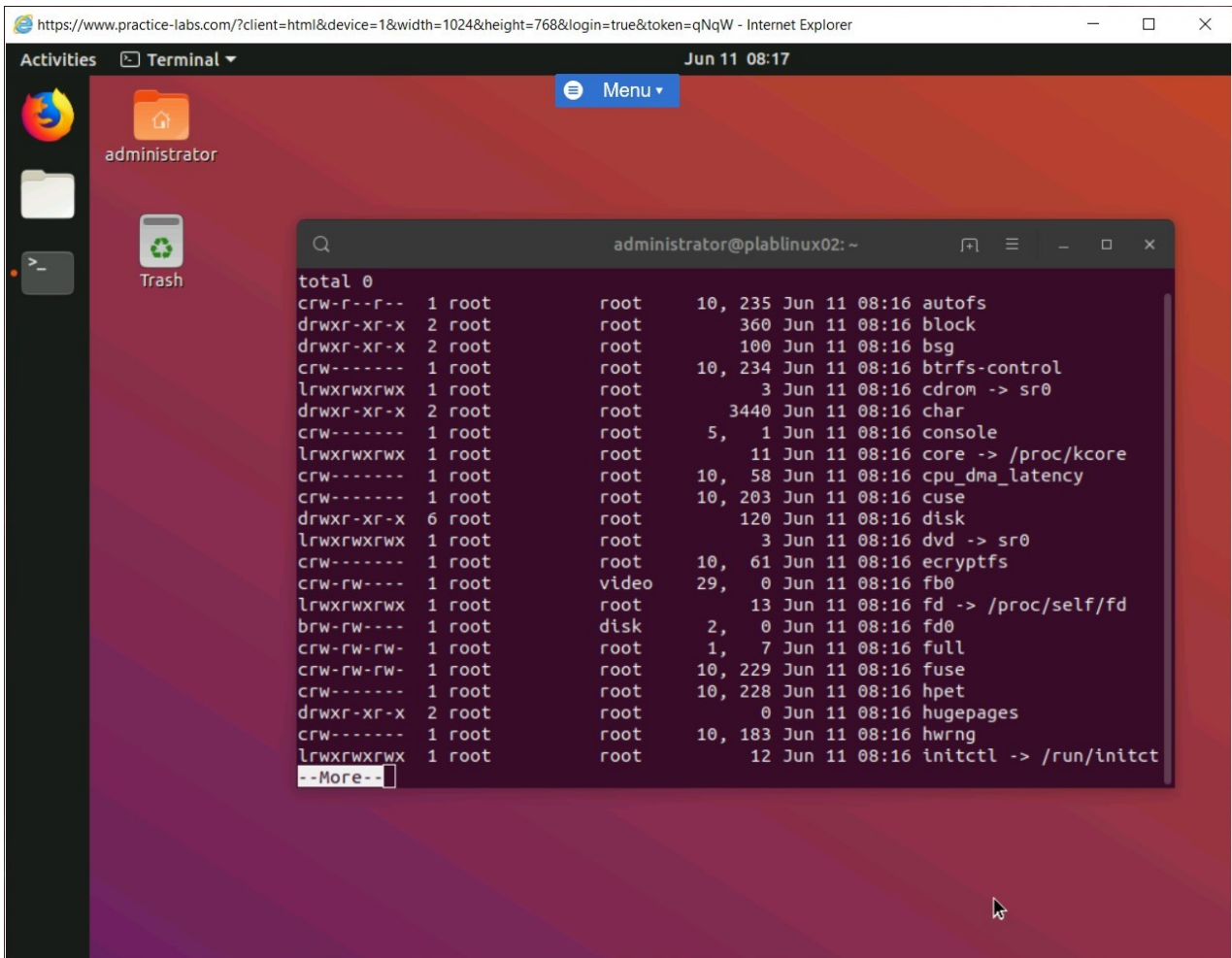

Figure 1.10 Screenshot of PLABLINUX02: Redirecting the standard output of the first command to the standard input of the second command.

Step 2

You can continue to press **Enter** to view the complete output or terminate **CTRL + c** to terminate the output.

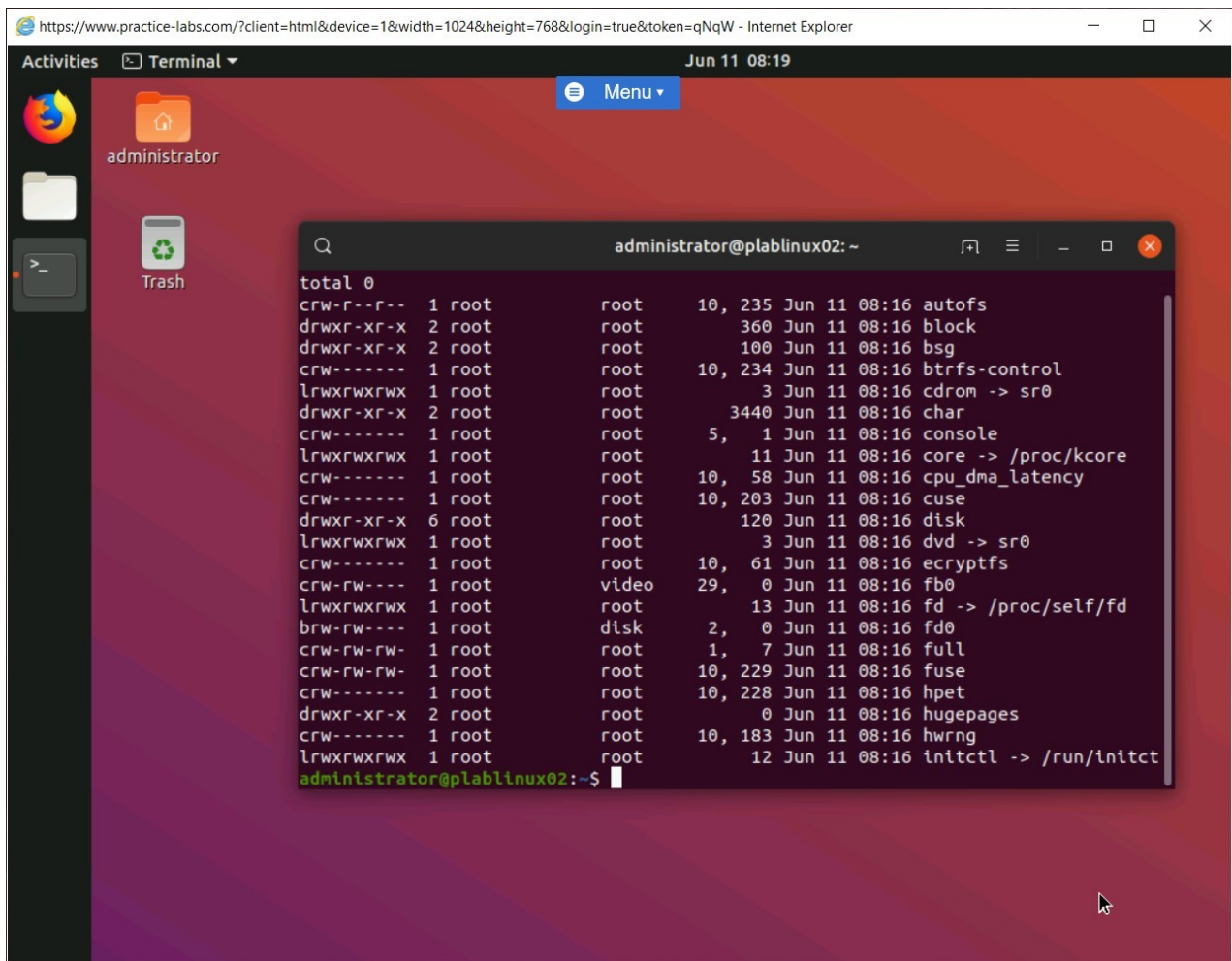You are back at the command prompt.

Figure 1.11 Screenshot of PLABLINUX02: Terminating the command.

Task 3 - Use the Output of One Command as Argument to another Command

To use the output of one command as an argument to the other command, perform the following steps:

Step 1

Clear the screen by entering the following command:

```
clear
```

Type the following command:

```
ls -l /etc/*.conf | grep user
```

Press **Enter**.

This command uses piping with the grep command. The command will find any file that has the word user in the name.
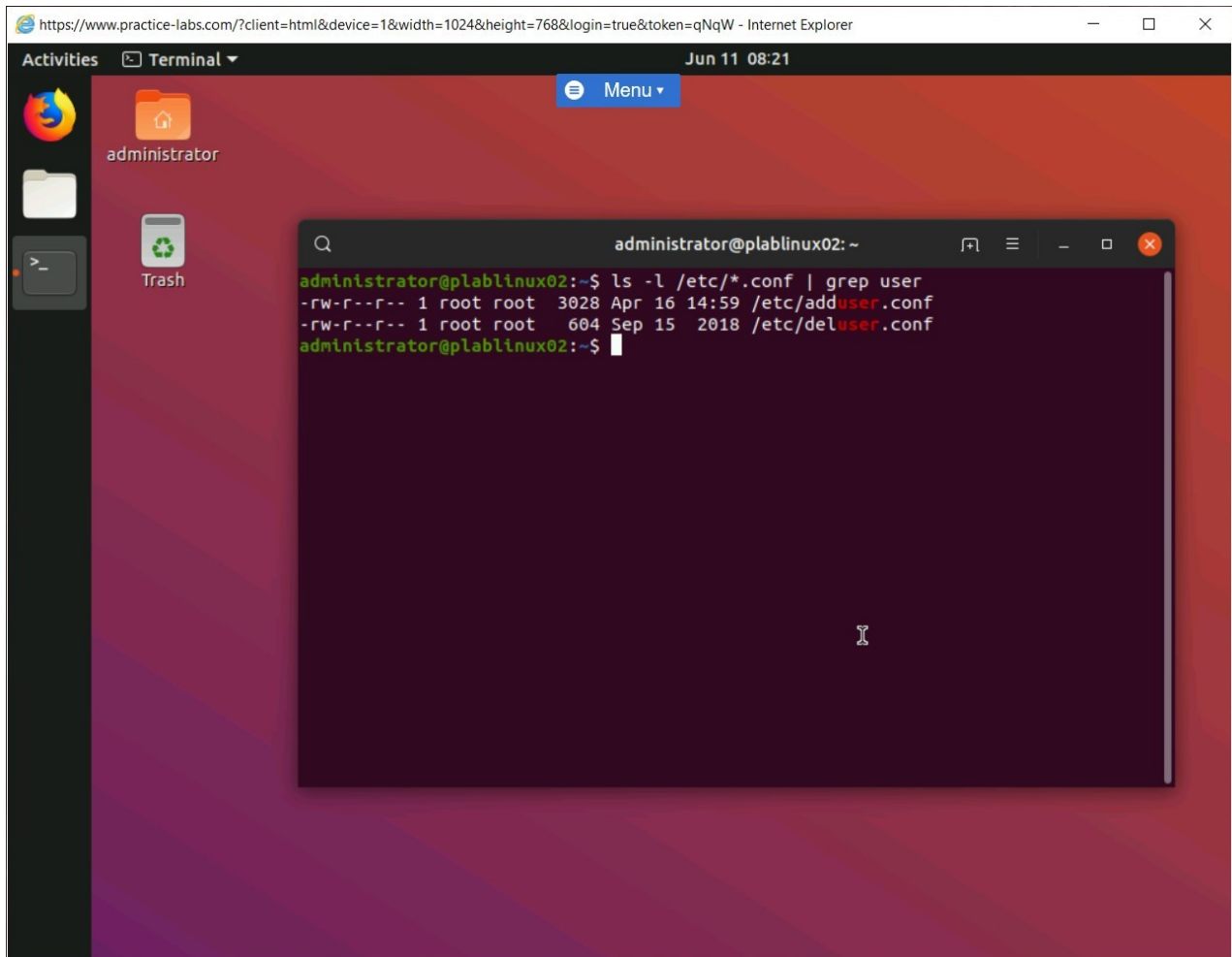

Figure 1.12 Screenshot of PLABLINUX02: Using piping with the grep command.

Step 2

The xargs command generates an argument list using standard input for a command. The xargs command is typically used with a pipe | operator.

To use the xargs command, enter the following command:

```
ls | xargs cat
```

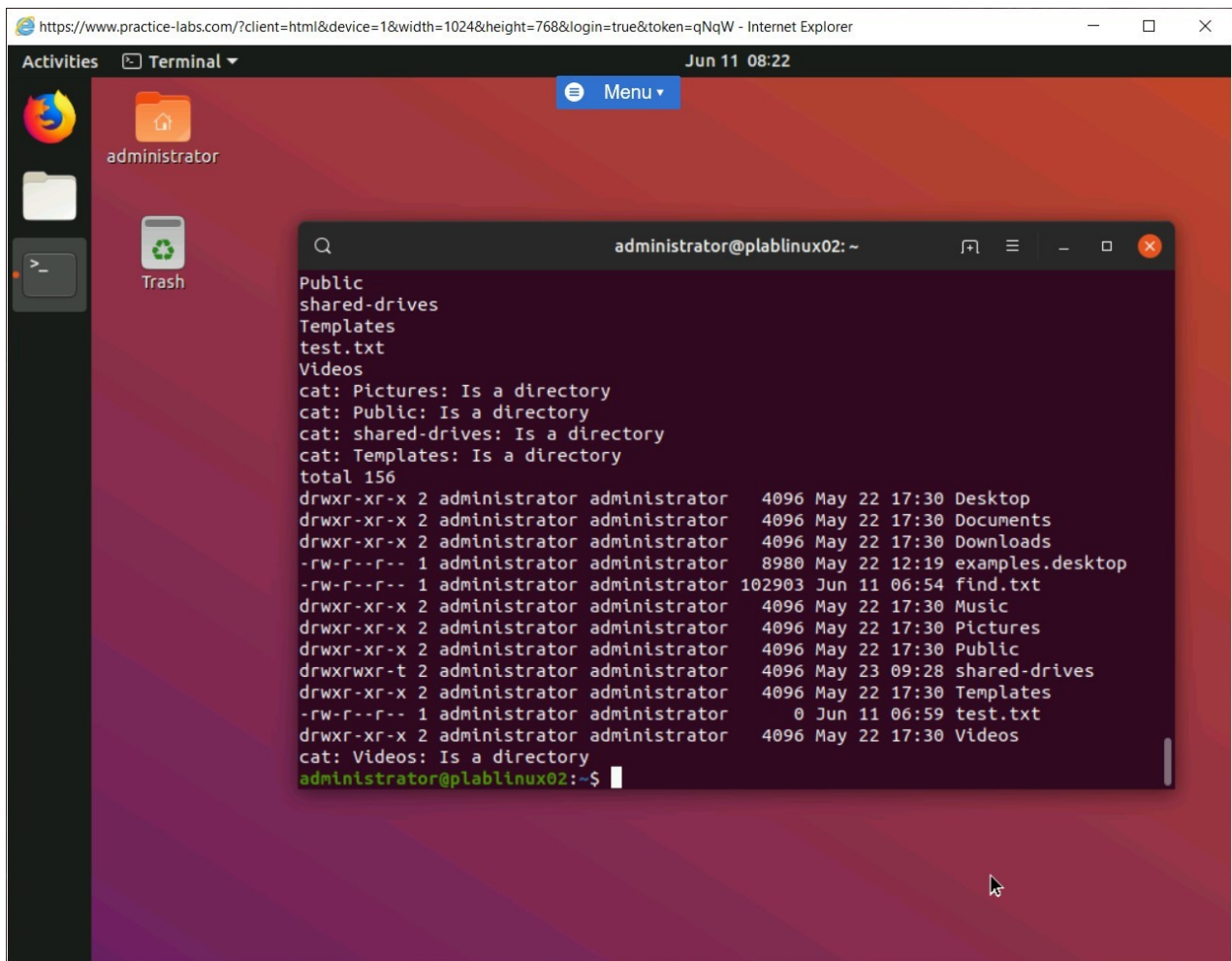The cat parameter reads the contents of the files displayed in the ls parameter.

Figure 1.13 Screenshot of PLABLINUX02: Using the xargs command.

Step 3

Clear the screen by entering the following command:

```
clear
```

Type the following command to create a file:
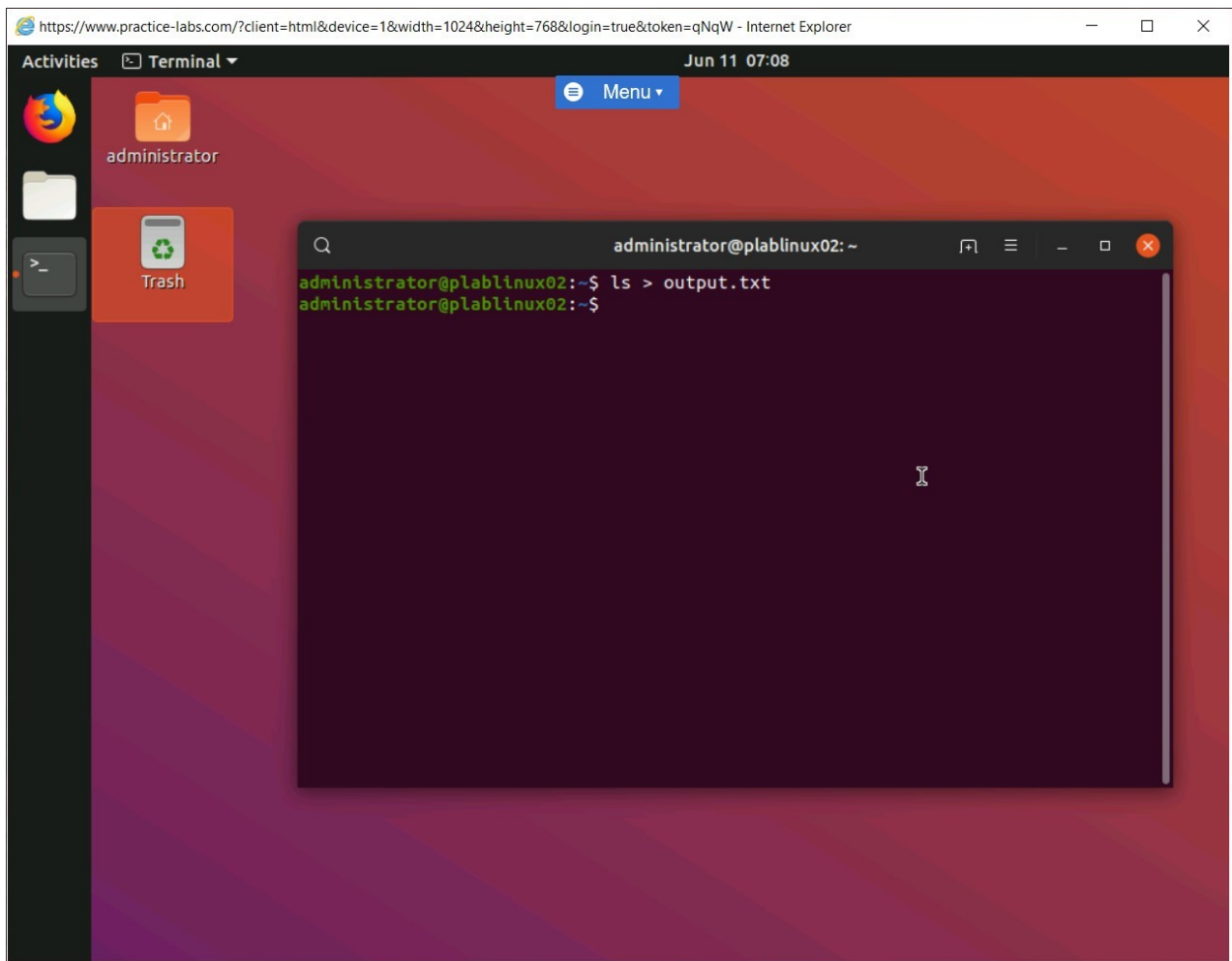
```
ls > output.txt
```

Press **Enter**.

Figure 1.14 Screenshot of PLABLINUX02: Using the ls command to create a file.

Step 4

Use the xargs command to view the output:
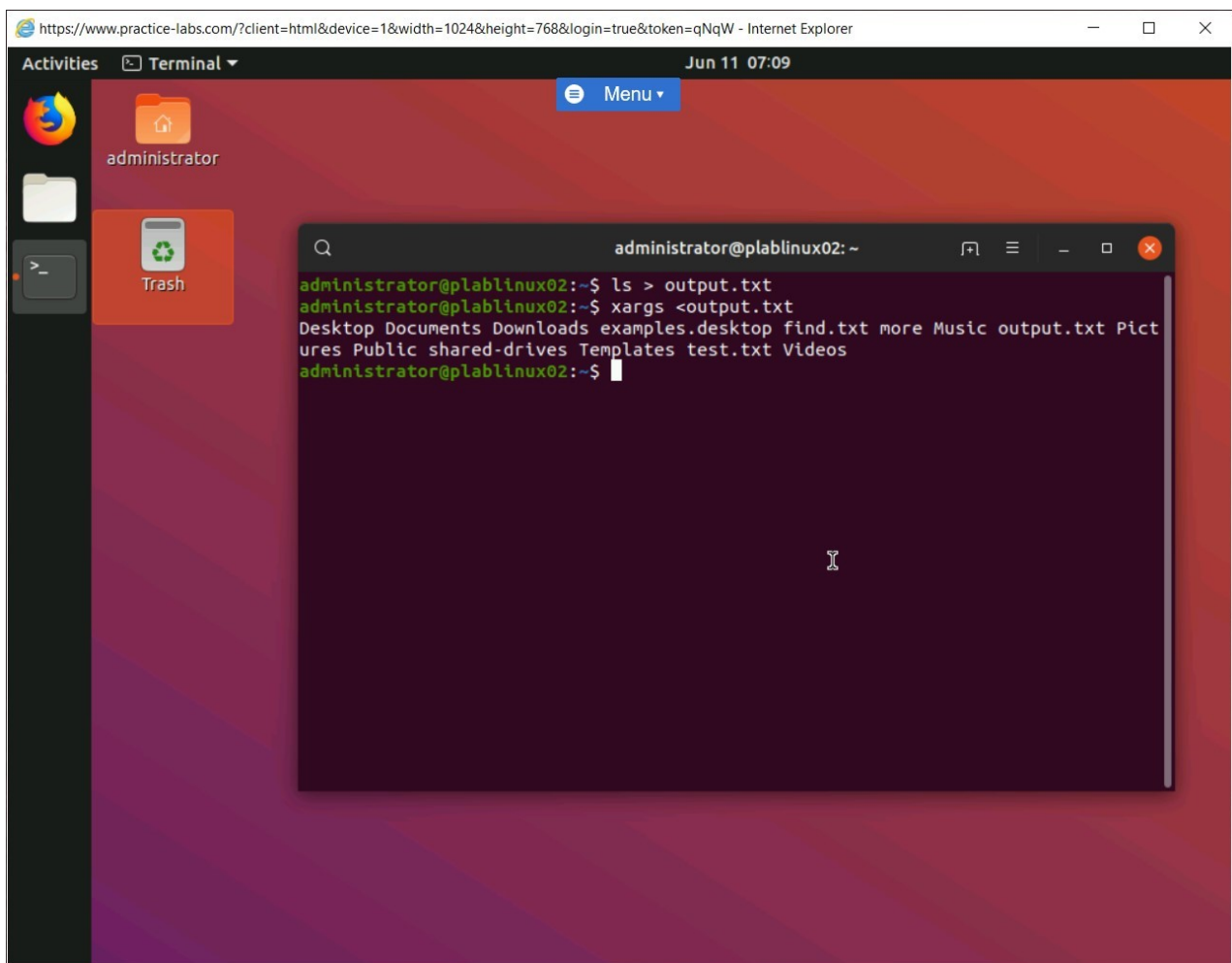
```
xargs <output.txt
```

Figure 1.15 Screenshot of PLABLINUX02: Using the xargs command to view the output.

Step 5

Clear the screen by entering the following command:

```
clear
```

You can also use xargs to search a large number of files for a specific keyword. The xargs command will search each word once in each of the files. Type the following command:

```
find / -type f | xargs -n 1 grep -H linux
```

Press **Enter**.

The word **linux** will be searched. Press Ctrl + C to break this command.

> *Note: This command takes quite a long time to search the **linux** word throughout the system.*
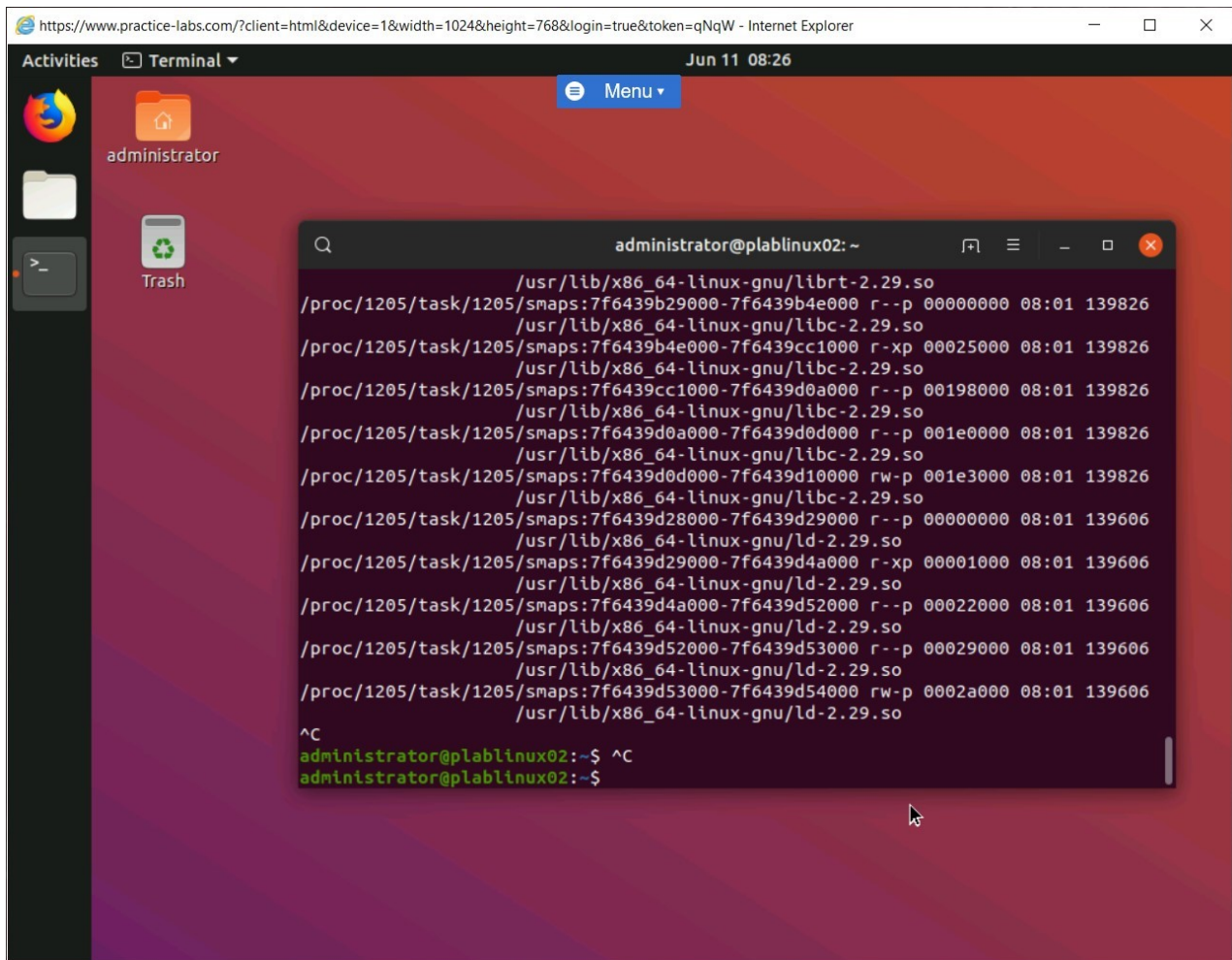

Figure 1.16 Screenshot of PLABLINUX02: Using the xargs to search a large number of files for a specific keyword.

Task 4 - Send Output to Both Stdout and a File

Linux offers command that enables you to send output to both - the standard output file (stdout) and any other file.

To send the output to both - stdout and a file, perform the following steps:

Step 1

Clear the screen by entering the following command:

```
clear
```

You can redirect the standard output to a file and on the screen at the same time. Enter the following command:
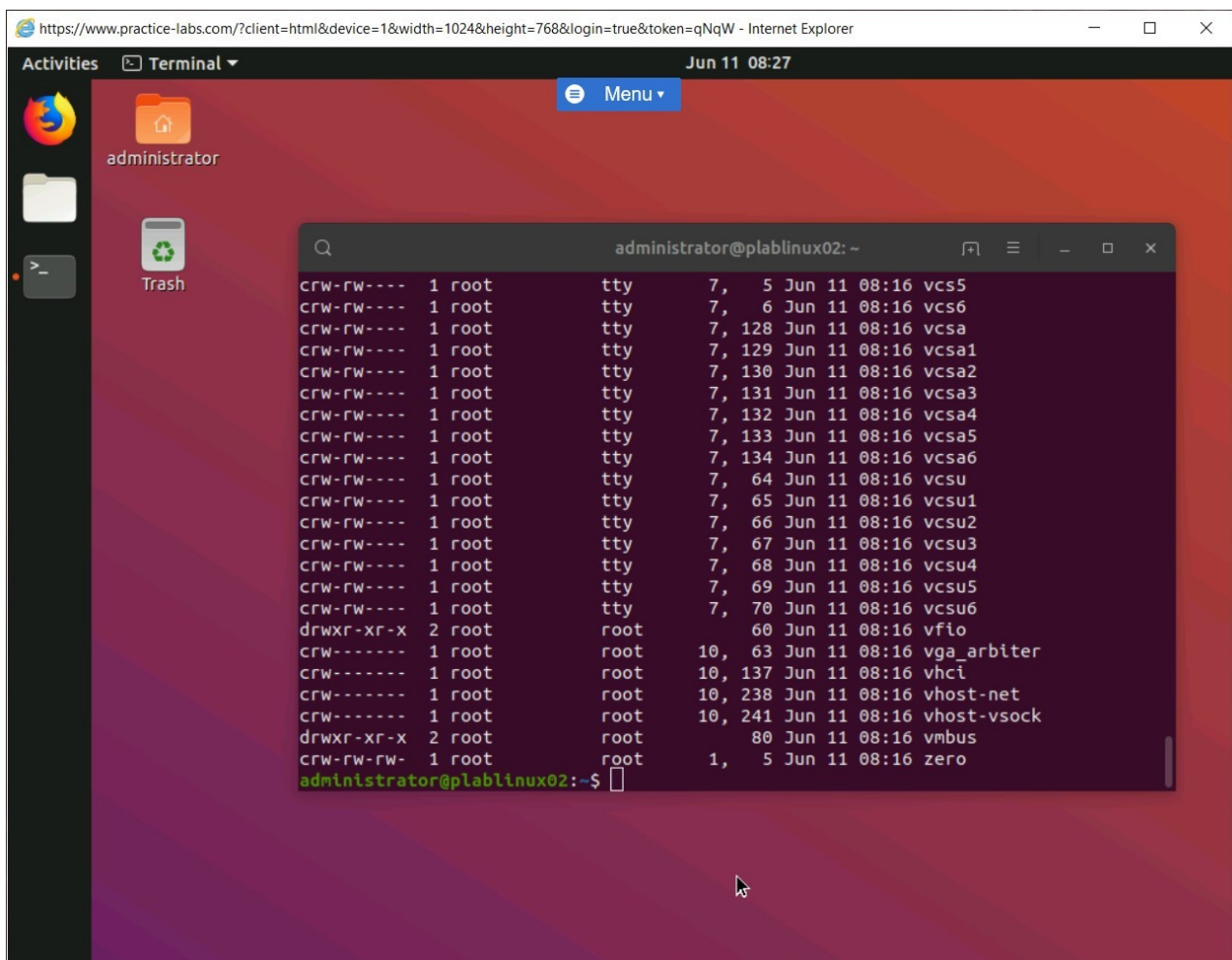
```
ls -l /dev | tee output.txt
```


Figure 1.17 Screenshot of PLABLINUX02: Redirecting the standard output to a file and on the screen at the same time.

Step 2

Clear the screen by entering the following command:

```
clear
```

Type the following command to verify that the **output.txt** file exists:
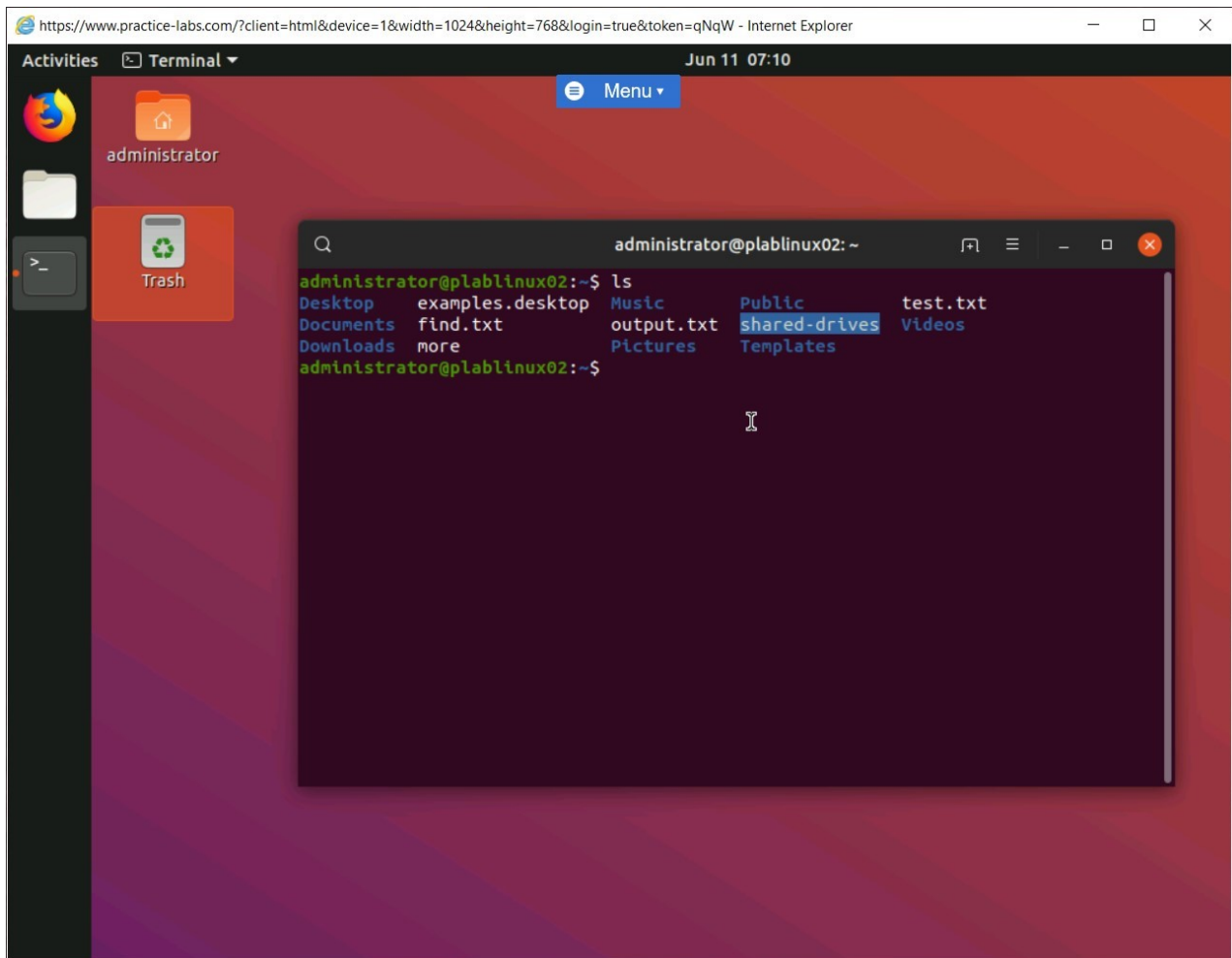
```
ls
```

Press **Enter**.

Figure 1.18 Screenshot of PLABLINUX02: Verifying that the output.txt file exists.

Step 3

To view the contents of the **output.txt** file, type the following command:

```
cat output.txt
```
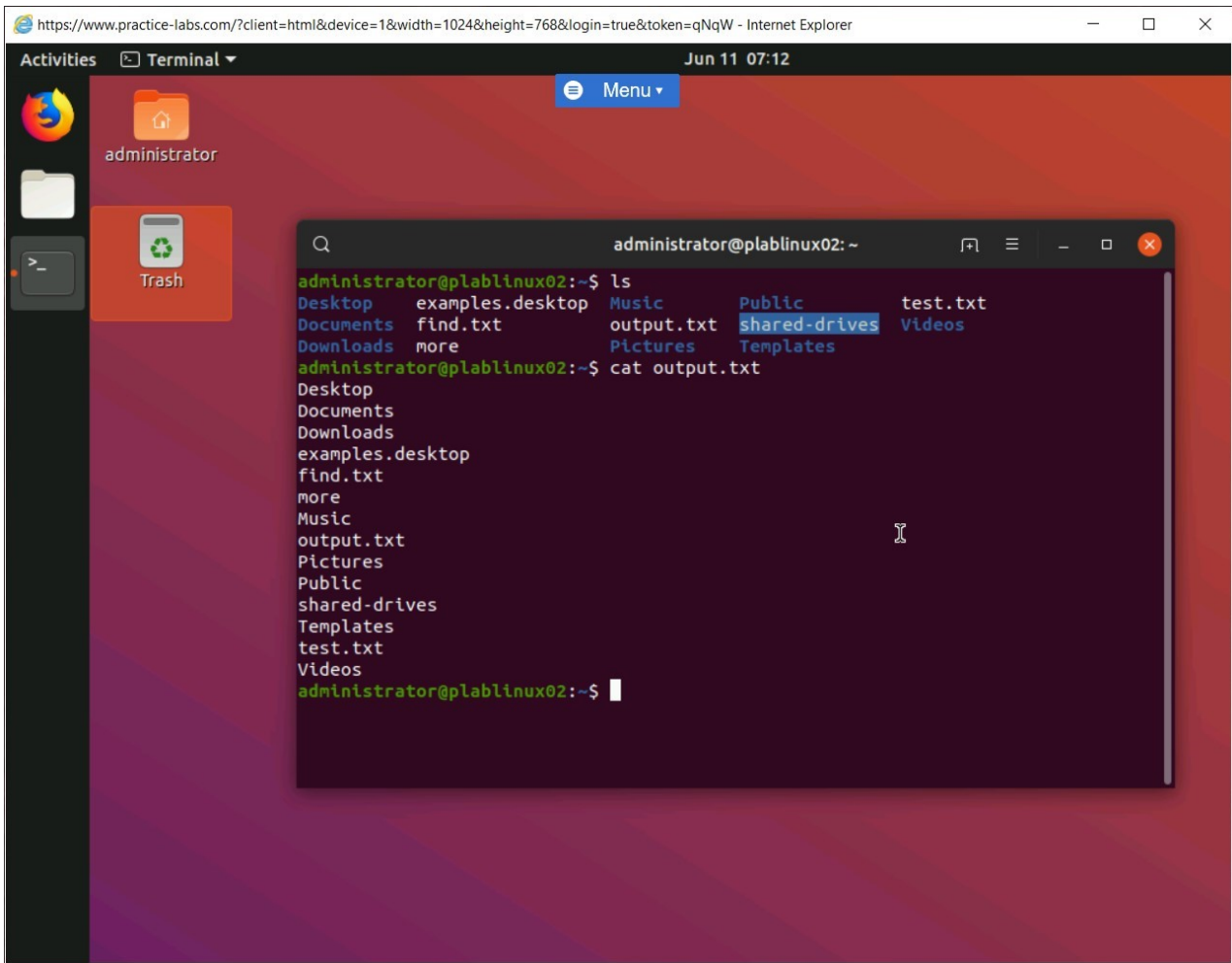
Press **Enter**.

Figure 1.19 Screenshot of PLABLINUX02: Viewing the contents of the output.txt file.

Keep all devices in their current state and proceed to the next exercise.

Review

Well done, you have completed the **Use Streams Pipes and Redirects** Practice Lab.

Summary

You completed the following exercise:

- Exercise 1 - Use Stream Pipes and Redirects

You should now be able to:

- Redirect standard input, standard output, and standard error
- Pipe the output of one command to the input of the other command
- Use the output of one command as the input to another command
- Send the output to both stdout and a file

Feedback

Shutdown all virtual machines used in this lab. Alternatively, you can log out of the lab platform.