CompTIA Linux+

Search Text Files using Regular Expressions

Introduction

Welcome to the **Search Text Files using Regular Expressions** Practice Lab. In this module you will be provided with the instructions and devices needed to develop your hands-on skills.

Regular Expressions
Text Files
File system

Learning Outcomes

In this module, you will complete the following exercise:

- Exercise 1 - Search Text Files using Regular Expressions

After completing this lab, you will be able to:

- Search for specific text in a file
- Search for specific criteria through a file content or a filesystem

Exam Objectives

The following exam objectives are covered in this lab:

- **LPI:** 103.2 Process text streams using filters
- **LPI:** 103.7 Search text files using regular expressions
- **CompTIA:** 2.3 Given a scenario, create, modify, and redirect files.

Note: *Our main focus is to cover the practical, hands-on aspects of the exam objectives. We recommend referring to course material or a search engine to research theoretical topics in more detail.*

Lab Duration

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

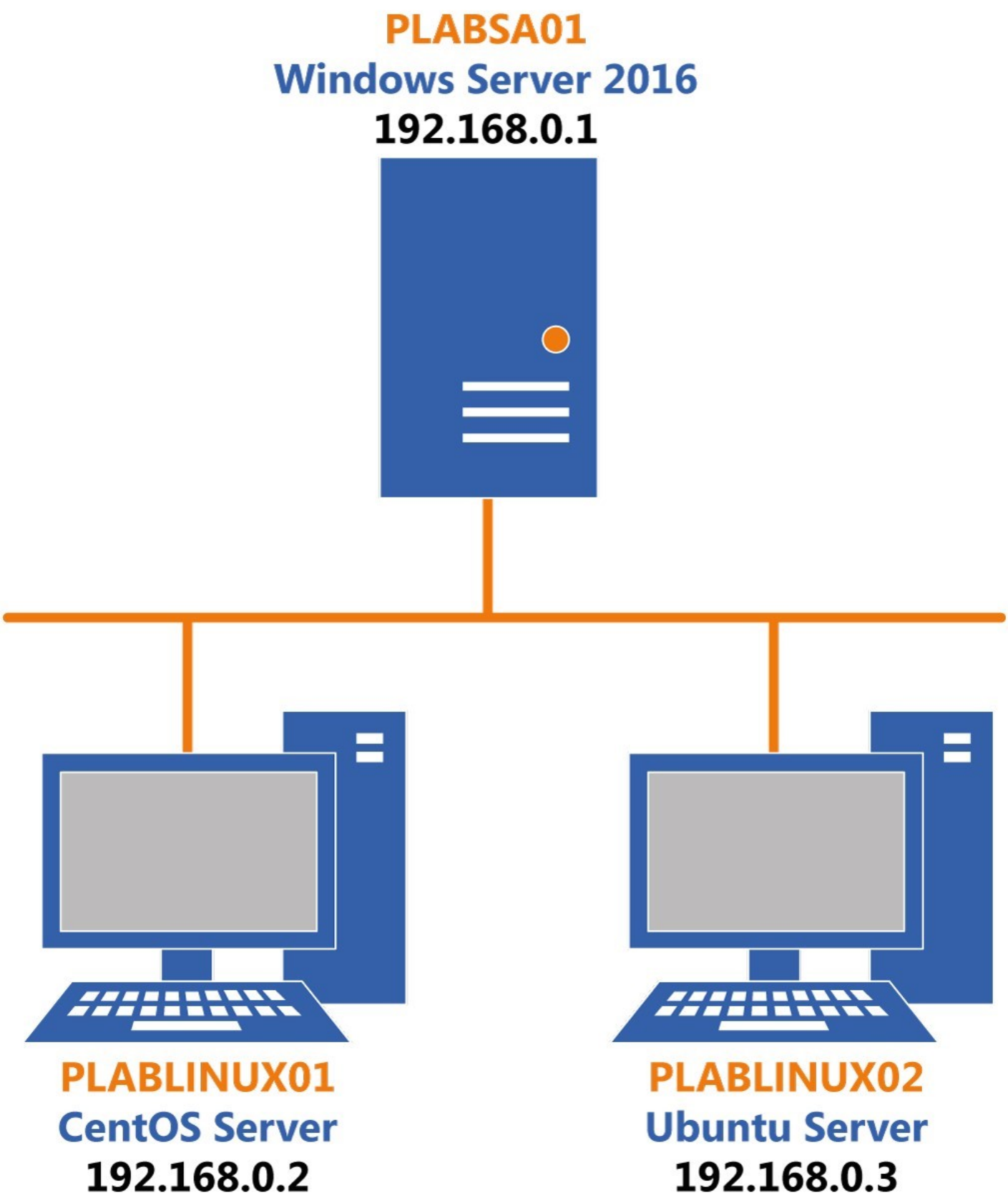
Help and Support

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.



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)

Click Next to proceed to the first exercise.

Exercise 1 - Search Text Files using Regular Expressions

Normally a system has a large number of text files. Searching for a specific text file can be a tedious task. You can search text files based on criteria specified using regular expressions.

In this exercise, you will understand how to search text files on a Fedora Linux system.

Learning Outcomes

After completing this exercise, you will be able to:

- Log into a Linux System
- Search for specific text in a file
- Search for specific criteria through a file content or a filesystem

Your Devices

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

- **PLABLINUX01** (CentOS Server)

Task 1 - Search for a Specific Text in a File

To search for specific text in a file, you create simple regular expressions containing several notational elements. In this task, you will determine the parameters related to a particular text in the **yum.conf** file.

To search for specific text in a file, perform the following steps:

Step 1

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

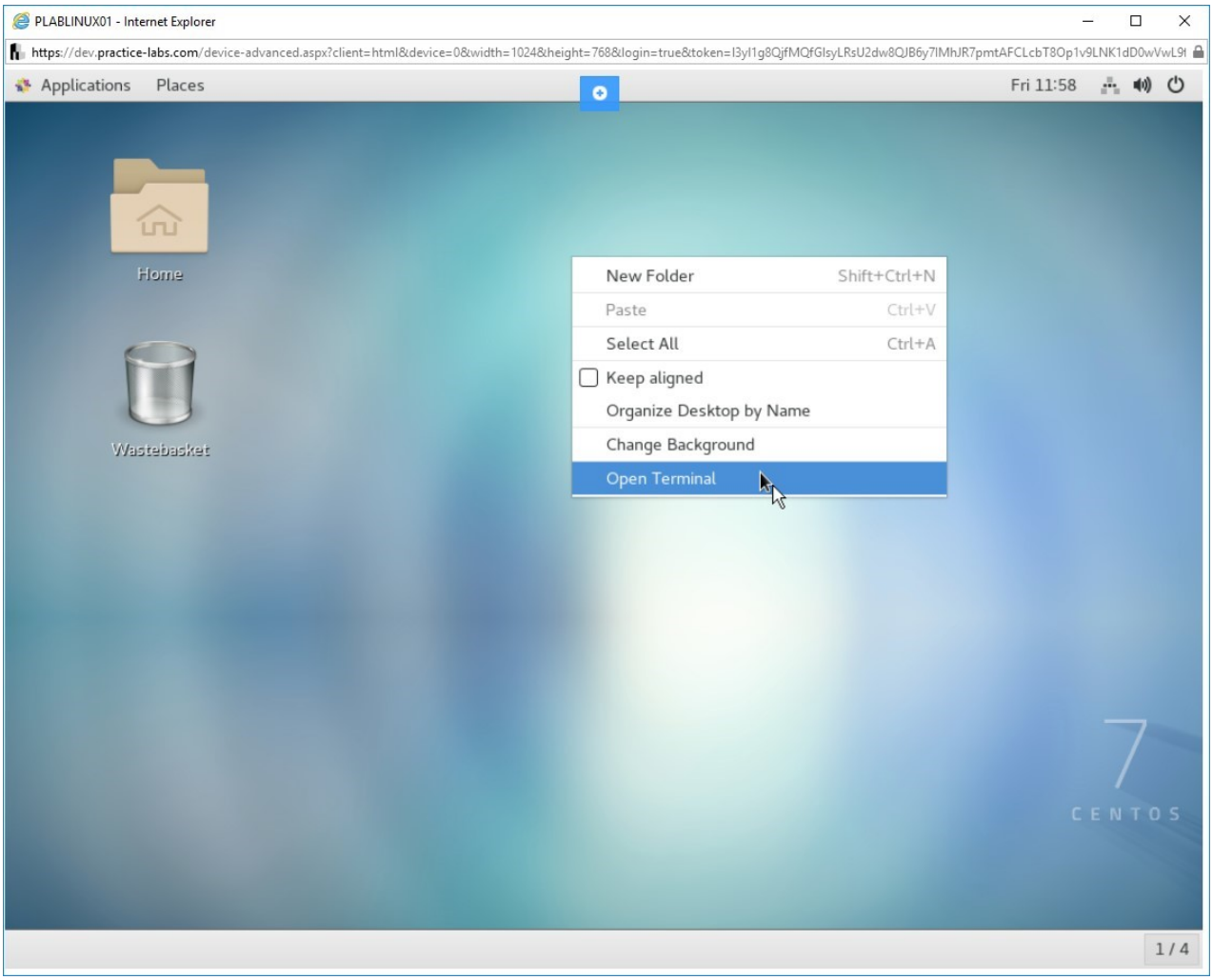


Figure 1.1 Screenshot of PLABLINUX01: Selecting the Open Terminal option from the context menu.

Step 2

The command prompt window is displayed. Type the following command:

```
su -
```

Press **Enter**.

At the **Password** prompt, type the following password:

```
Passw0rd
```

Press **Enter**.

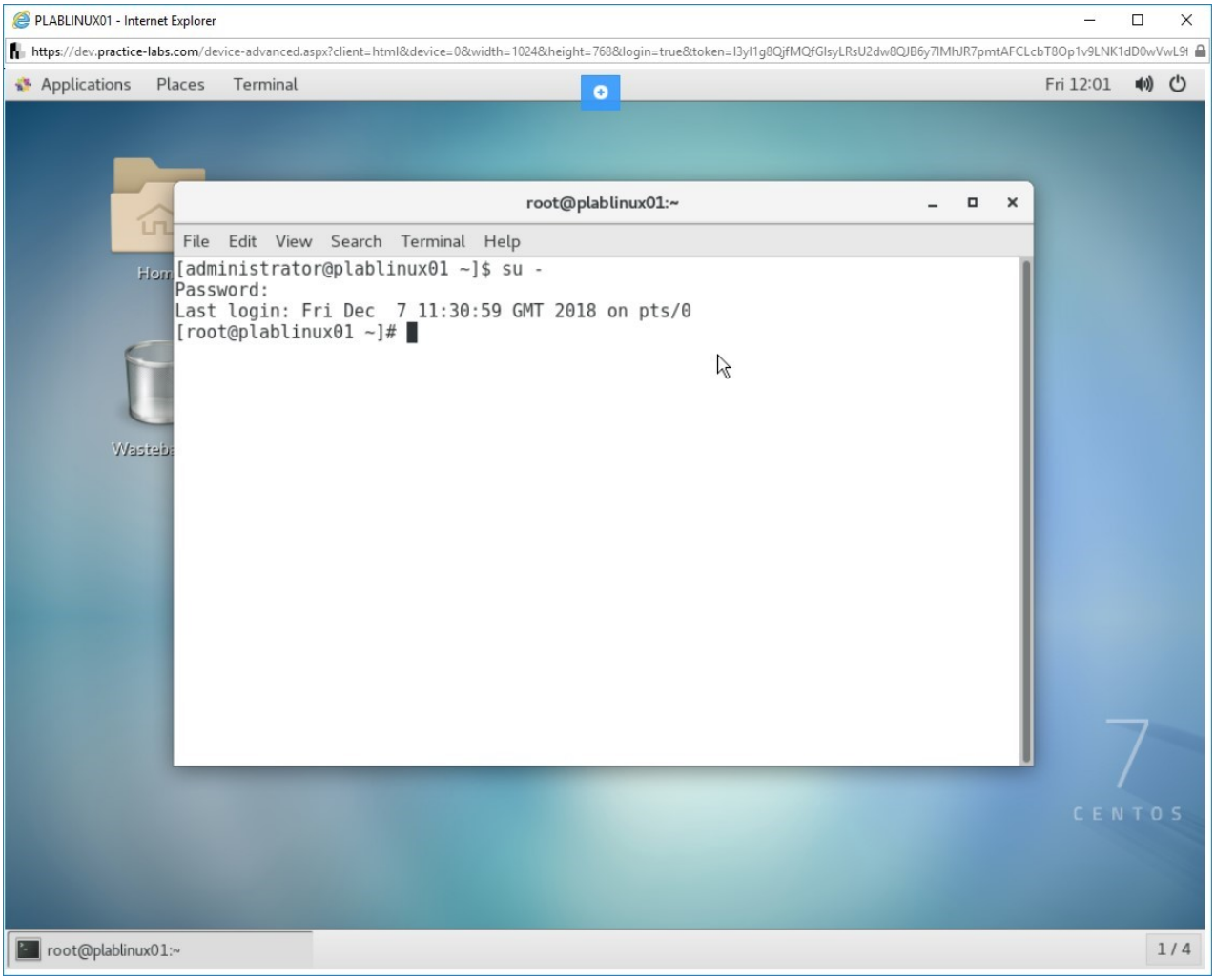


Figure 1.2 Screenshot of PLABLINUX01: Changing to the root account with the su command.

Step 3

Clear the screen by entering the following command:

```
clear
```

***Note**: The clear command is used before every step to enable the learners to get a clear view of the output of each command. Otherwise, it is not mandatory to use the clear command before every command.*

Using the grep command, you can find specific words in one or more files. If you specify more than one files, the file name is displayed in front of the line that contains the searched word.

For example, to find the word **old** in the **/etc/yum.conf** file, enter the following command:

```
grep -n "old" /etc/yum.conf
```

> ***Note**: There are various parameters that can be used with the grep command. The -n parameter displays the line number of the line that contains the searched text.*

Notice the number 20 at the beginning of the display. This is the line number of the line that contains the specified text.

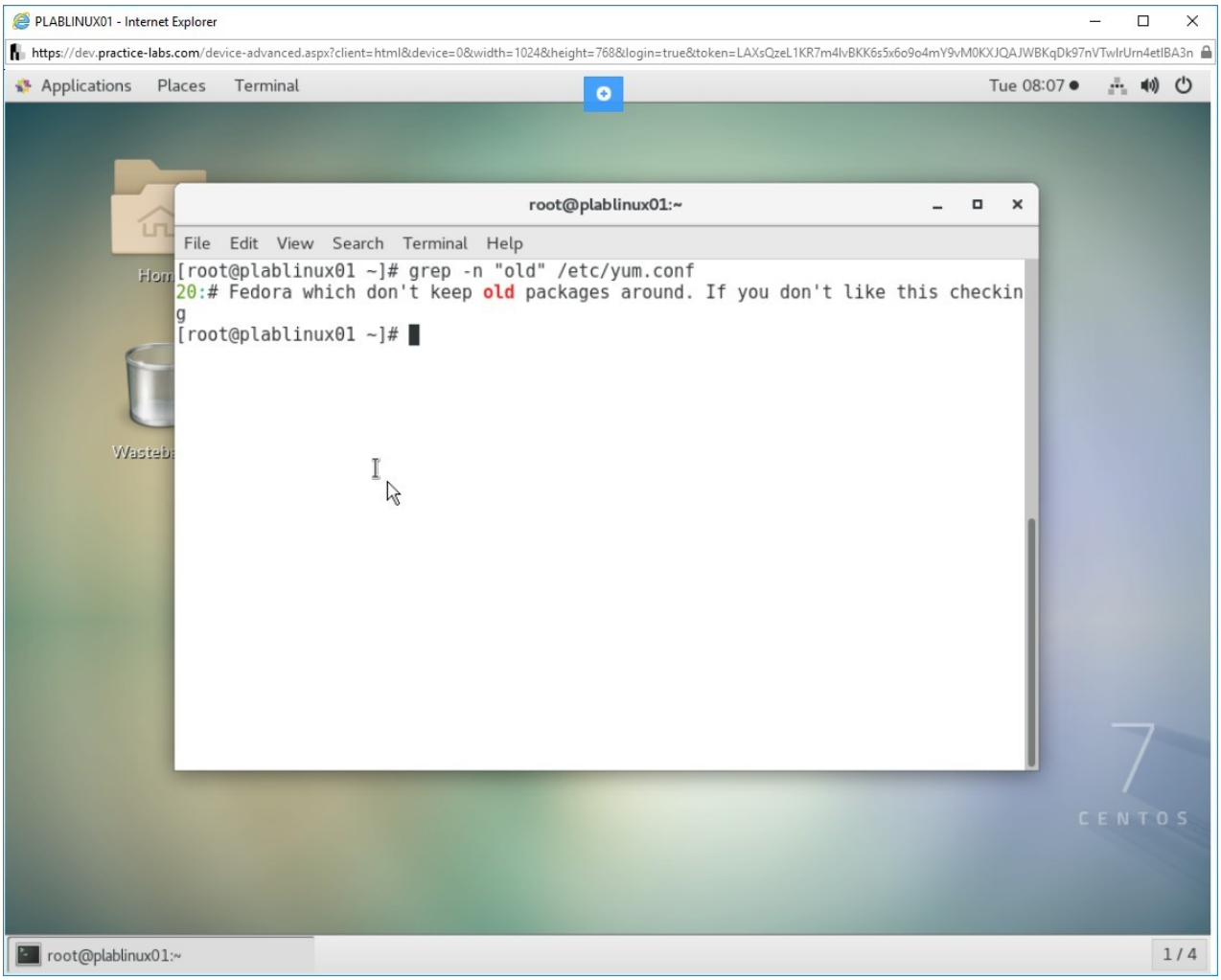


Figure 1.3 Screenshot of PLABLINUX01: Finding the word old in the /etc/yum.conf file.

Step 4

Instead of displaying the lines with the searched text, you can simply count the number of lines in which the searched text appears. You can count the number of lines by using the -c switch.

To use the -c switch, enter the following command:

```
grep -c "yum" /etc/yum.conf
```

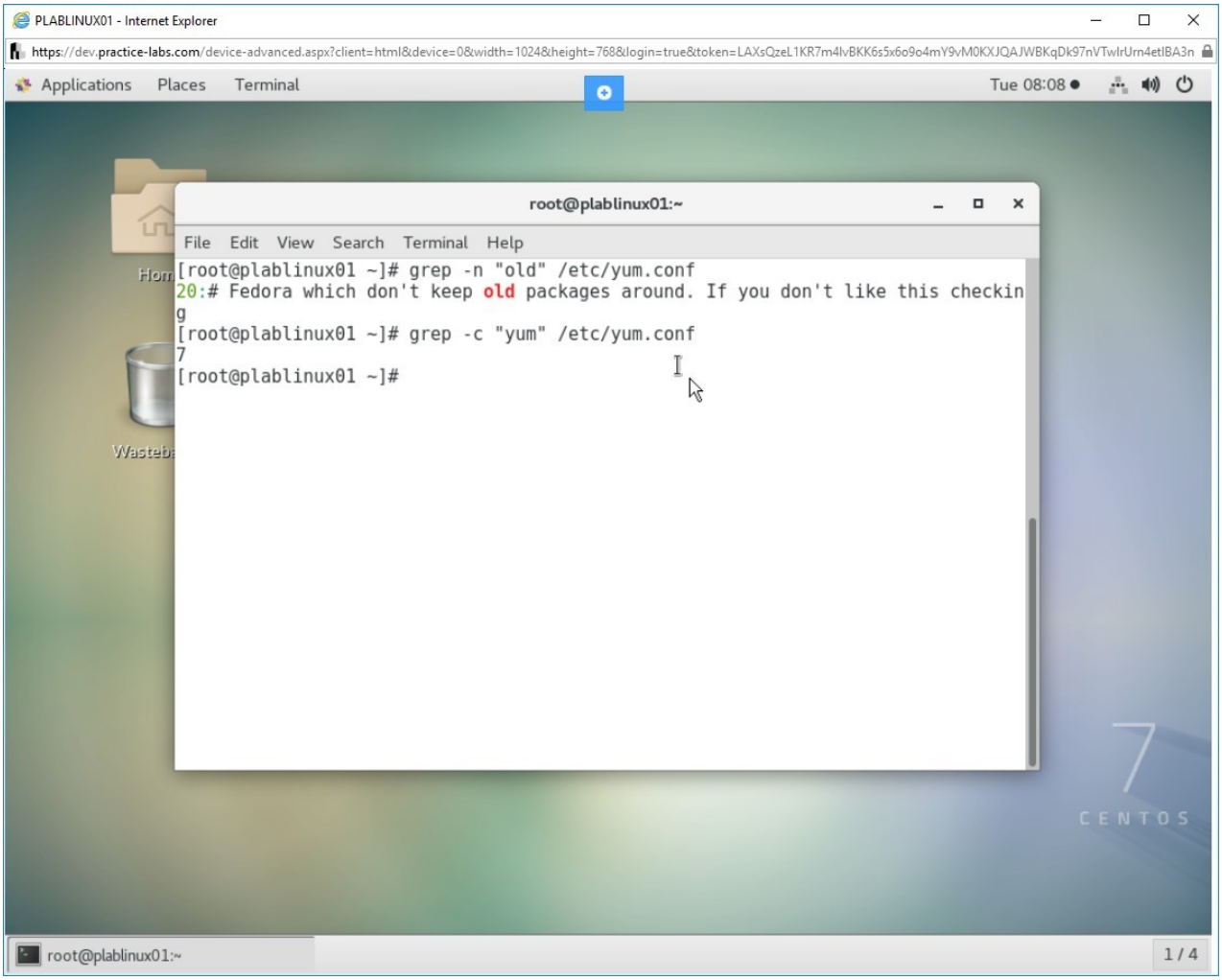


Figure 1.4 Screenshot of PLABLINUX01: Counting the number of lines in which the searched text appears.

Step 5

Let's try to find the number of words that start with y in the **yum.conf** file. Enter the following command:

```
grep -c "y*" /etc/yum.conf
```

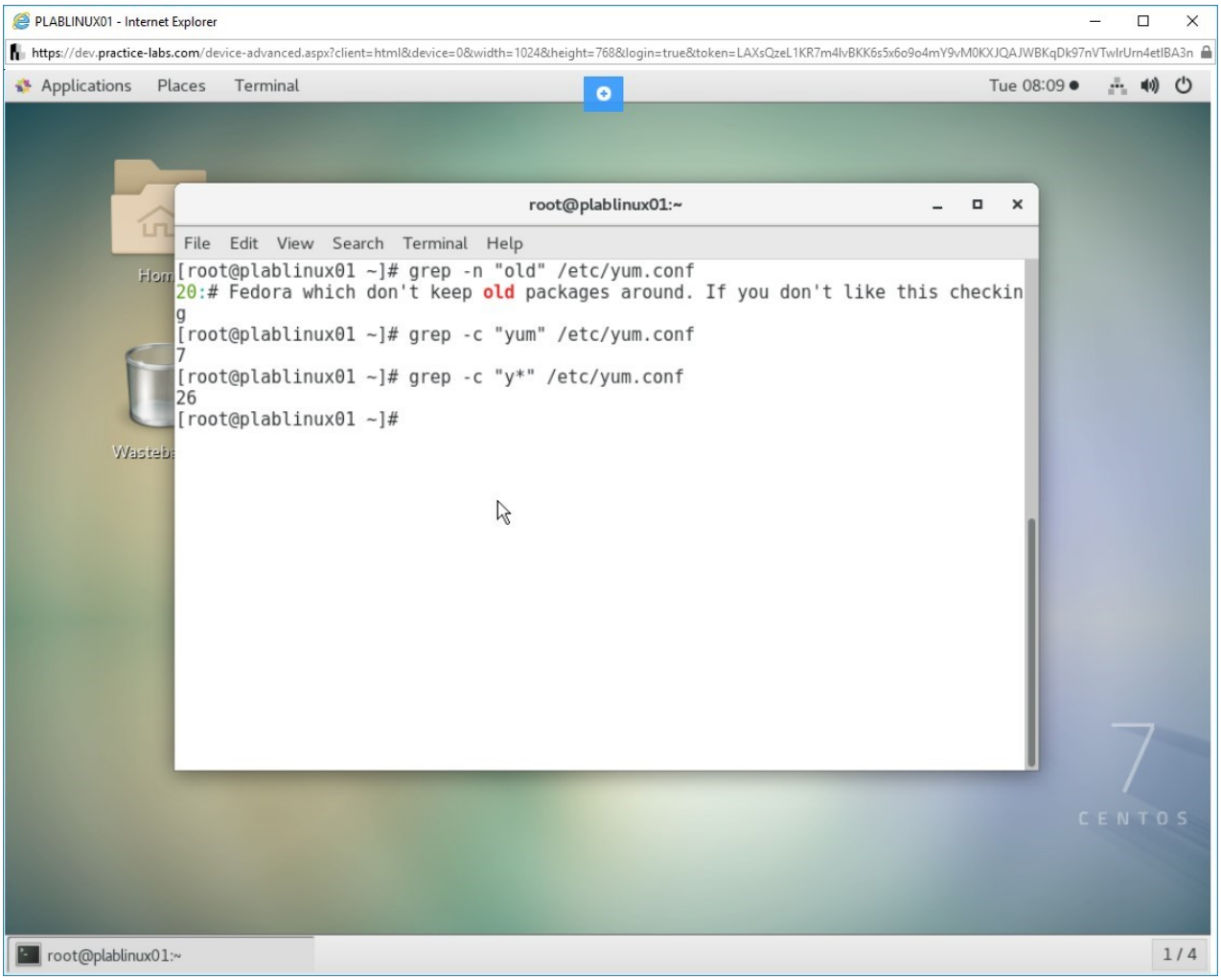


Figure 1.5 Screenshot of PLABLINUX01: Finding the number of words that start with y in the yum.conf file.

Step 6

Let's try to find the word "yum" in the **yum.conf** file that does not contain the **yum** word. Enter the following command:

```
grep -v "yum" /etc/yum.conf
```

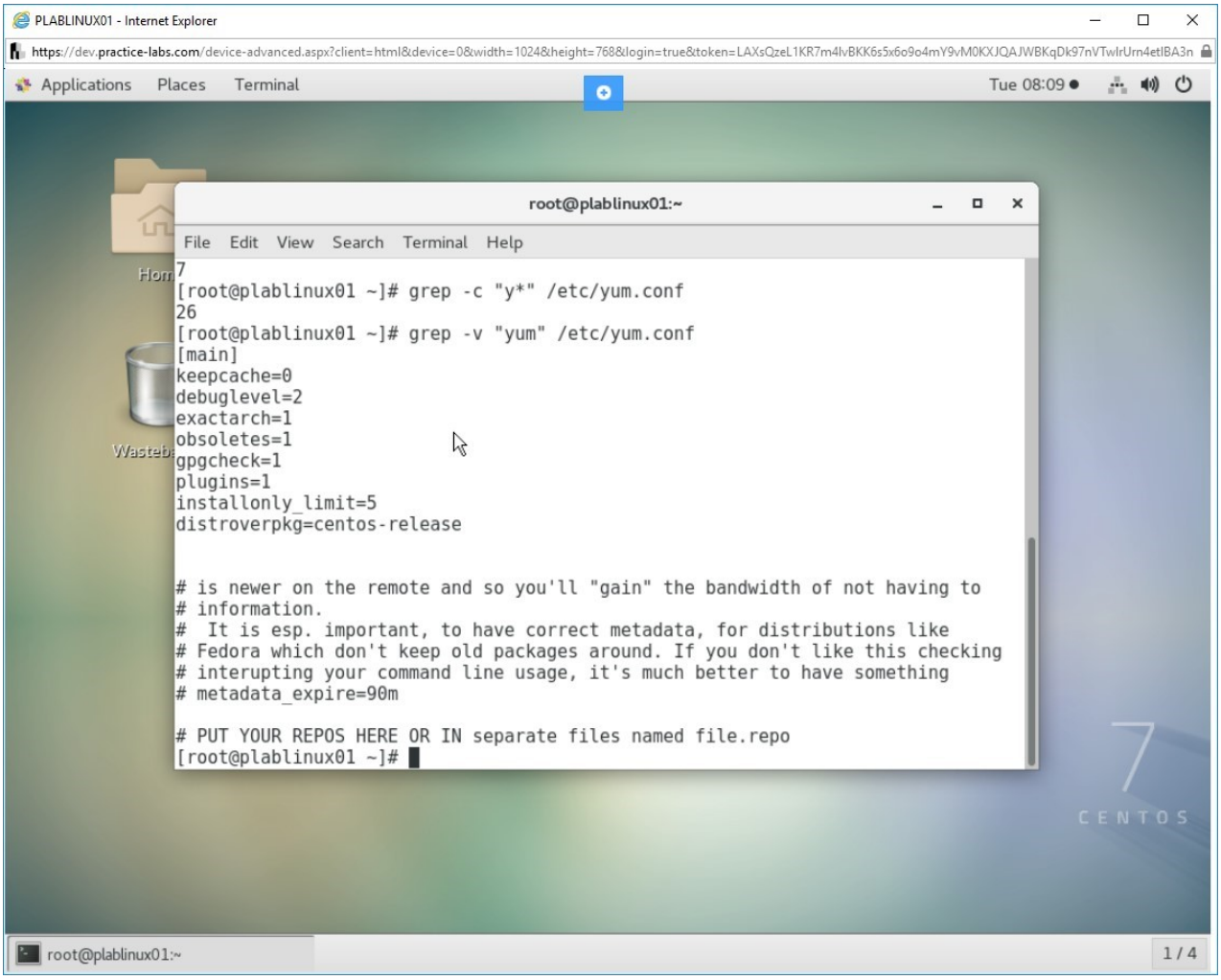


Figure 1.6 Screenshot of PLABLINUX01: Finding the word "yum" in the yum.conf file that does not contain the yum word.

Step 7

Clear the screen by entering the following command:

```
clear
```

To list all the blank lines with their line numbers, enter the following command:

```
grep -n "^$" /etc/yum.conf
```

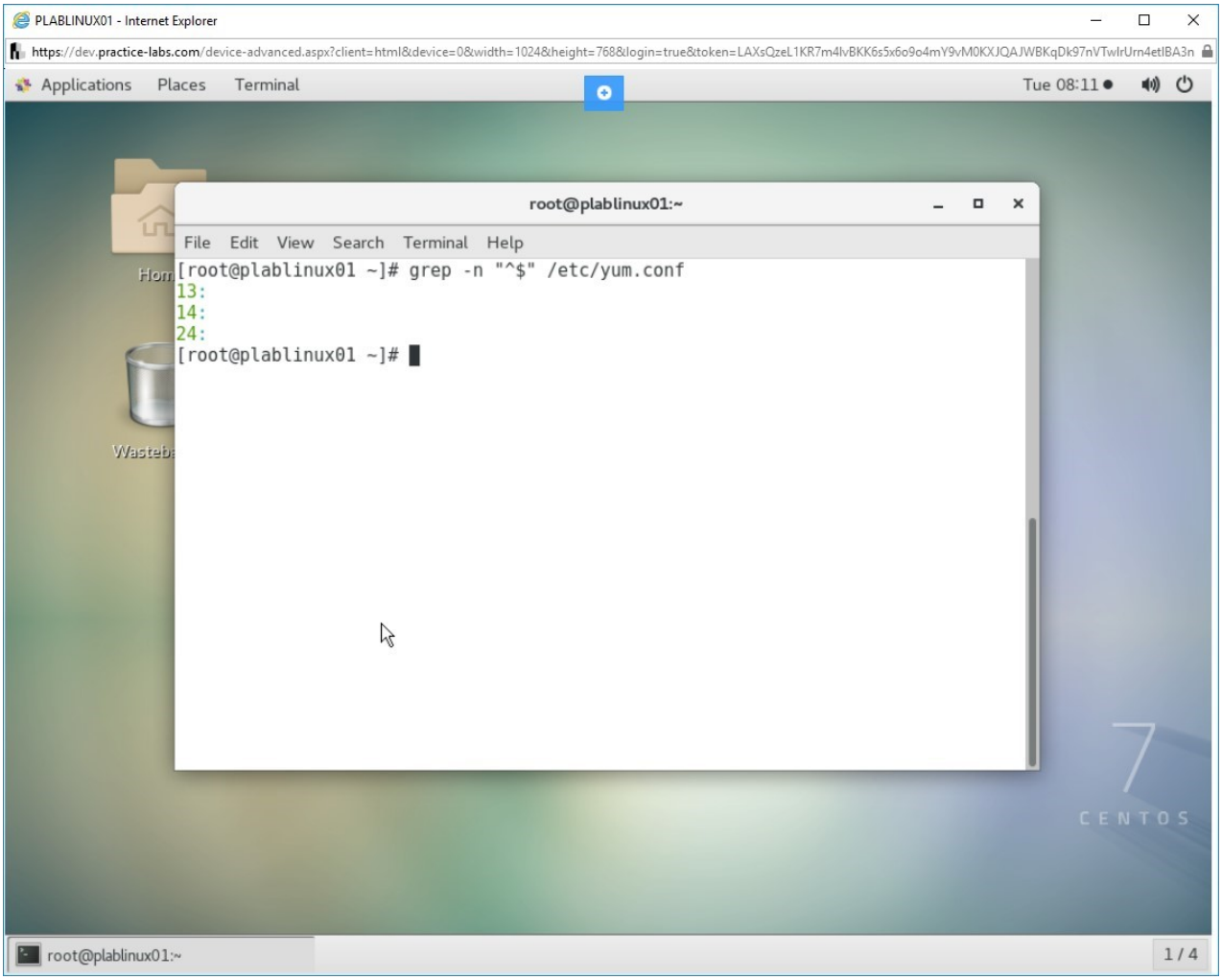


Figure 1.7 Screenshot of PLABLINUX01: Listing all the blank lines with their line numbers.

Step 8

You can also simply list the lines that contain a specific word. Enter the following command:

```
grep yum /etc/yum.conf
```

***Note**: In the above-given command, the word yum is being searched without quotes.*

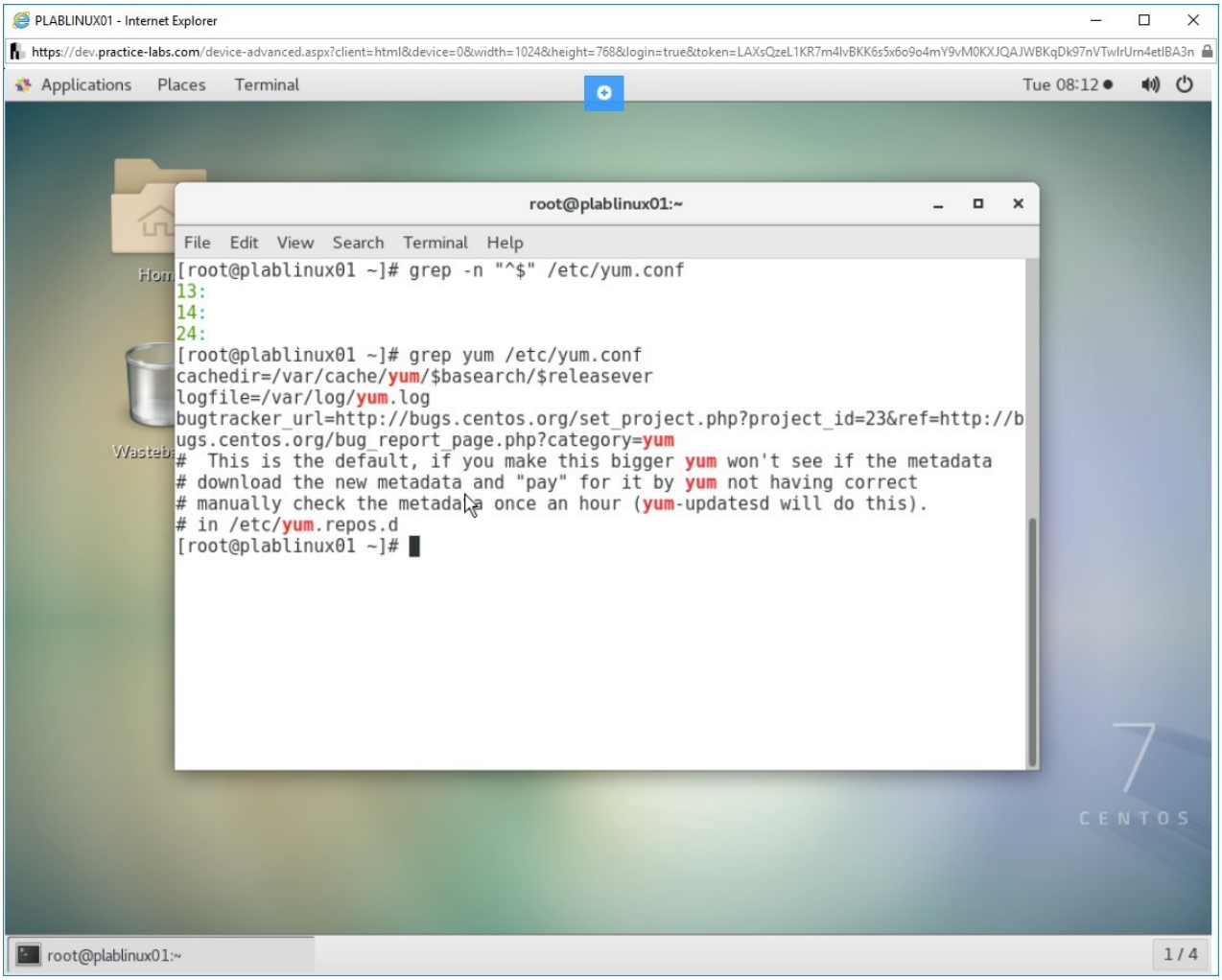


Figure 1.8 Screenshot of PLABLINUX01: Listing the lines that contain a specific word.

Task 2 - Search for Specific Criteria Through a File Content or a Filesystem

In the previous task, you used grep. In this task, you will use egrep. There is a distinct difference between grep and egrep. When you use grep, the special characters, such as ?, +, {, |, (, and), lose their meaning. To use the special characters as metacharacters, you need to use \?, \+, \{, \|, \(, and \). With egrep, you do not have to use the \?, \+, \{, \|, \(, and \). The special characters can be used as metacharacters.

To search for specific criteria through a file content or filesystem, perform the following steps:

Step 1

Clear the screen by entering the following command:

```
clear
```

You can list a specific set of files using the **pipe**. Enter the following command:

```
ls | egrep "yum" /etc/yum.conf
```

In this command, the pipe "|" is used to pass the results from egrep to the ls command.

The pipe character can also be used as an **OR** command.

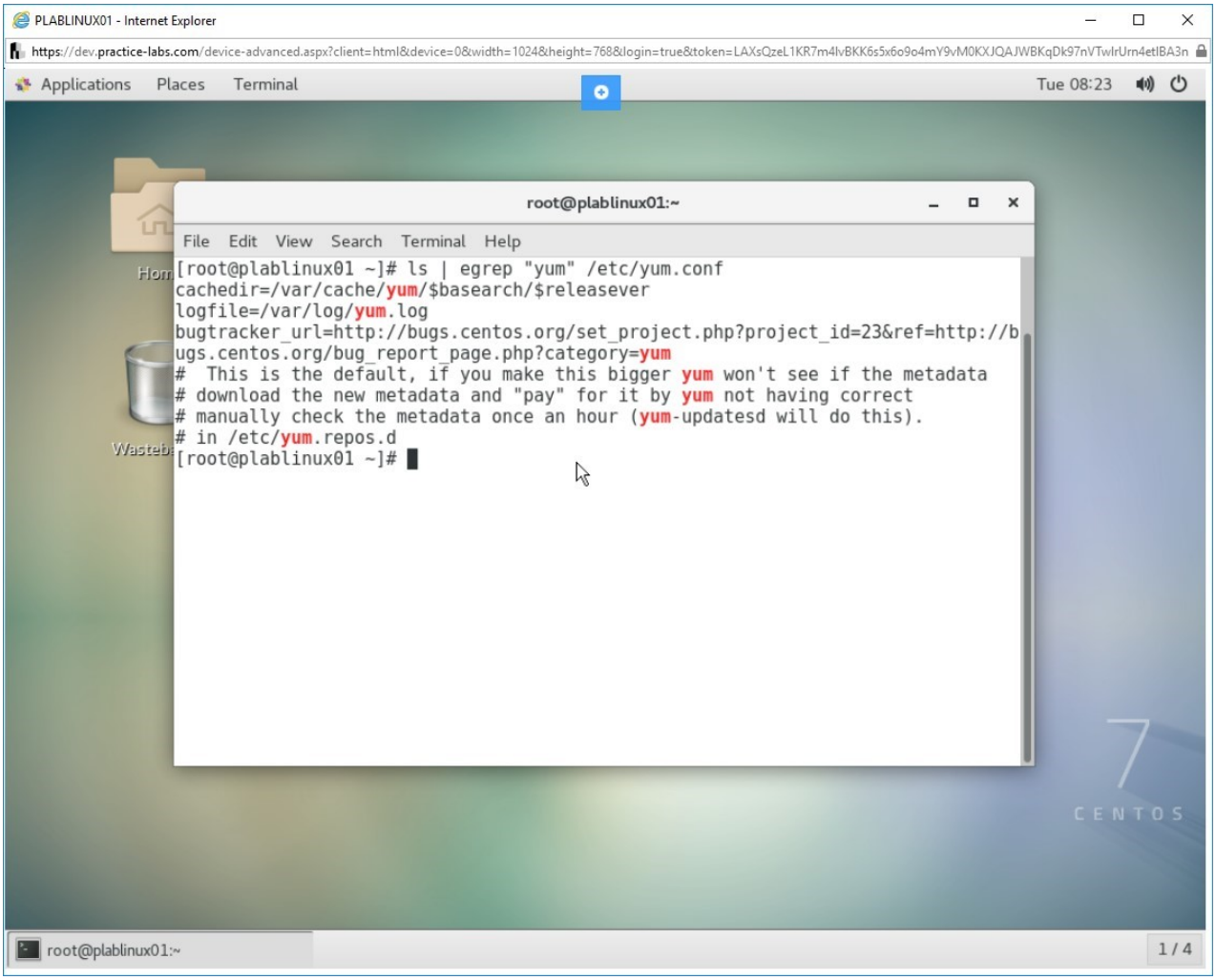


Figure 1.9 Screenshot of PLABLINUX01: Listing the occurrences that have yum in the name.

Step 2

Now, attempt the same command with **grep.** Enter the following command:

```
ls | grep "init|conf"
```

Note that no result is returned. **Grep** cannot interpret the pipe |.

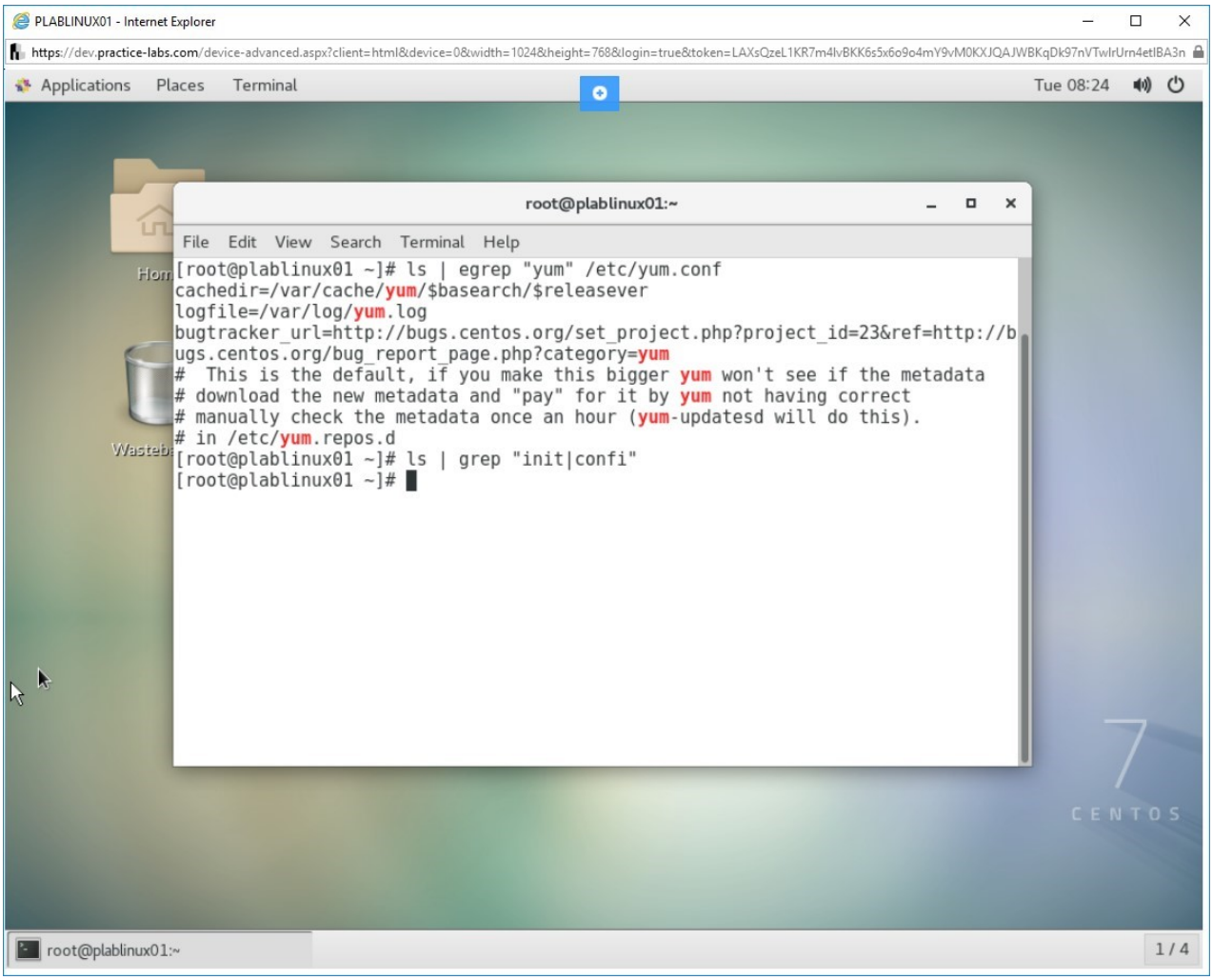


Figure 1.10 Screenshot of PLABLINUX01: Listing the occurrences that have yum in the name with the grep command.

Step 3

Clear the screen by entering the following command:

```
clear
```

To count the number of lines in **/etc/yum.conf** that either start with **1** or end with **01**, enter the following command:

```
egrep -c '^01|1$' /etc/yum.conf
```

Notice the use of single inverted commas (') in the command as compared to the use of double inverted commas (") in the earlier commands.

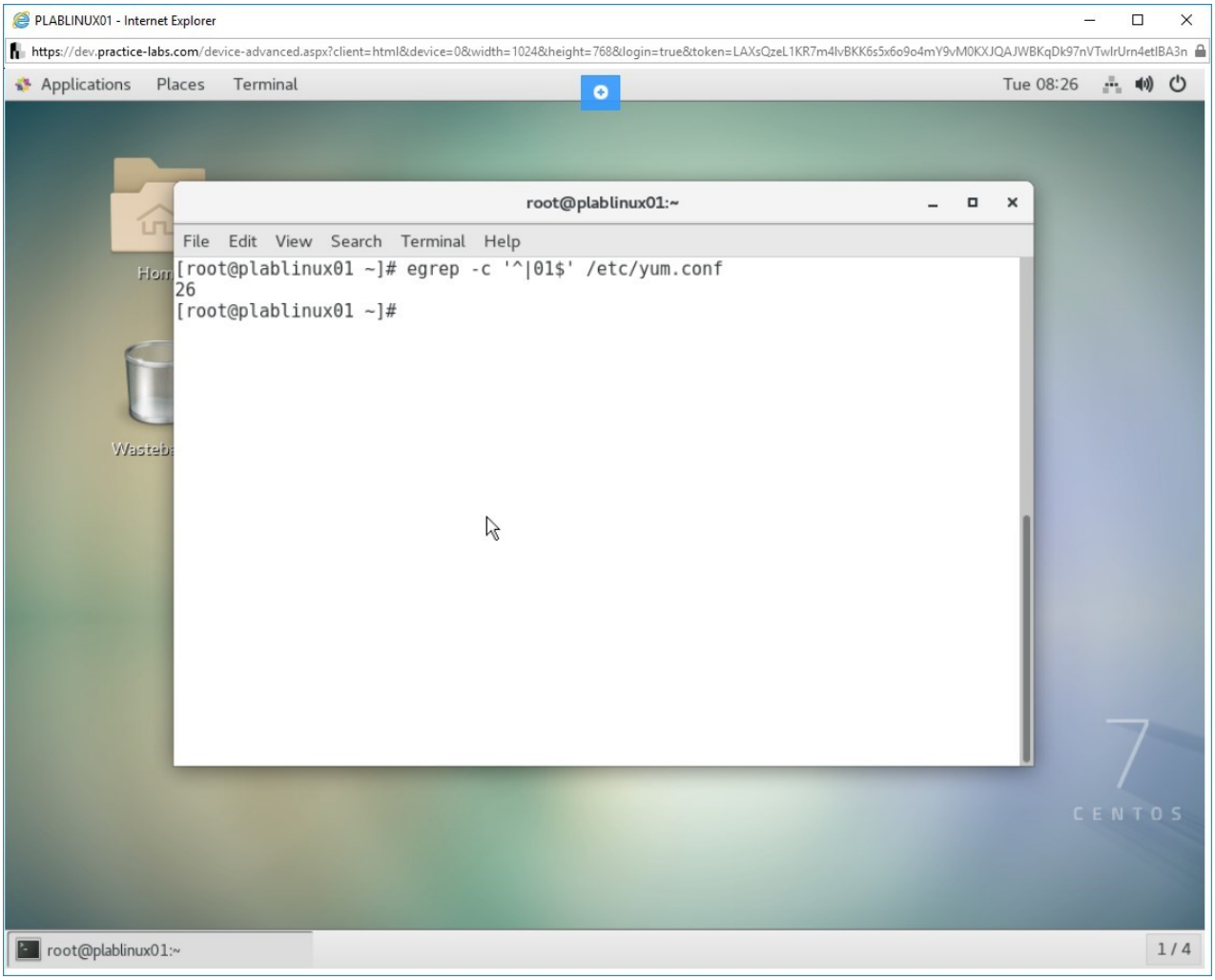


Figure 1.11 Screenshot of PLABLINUX01: Counting the number of lines in /etc/yum.conf that either start with 1 or end with 01.

Step 4

To find two words simultaneously in a file, enter the following command:

```
egrep 'Fedora|yum' /etc/yum.conf
```

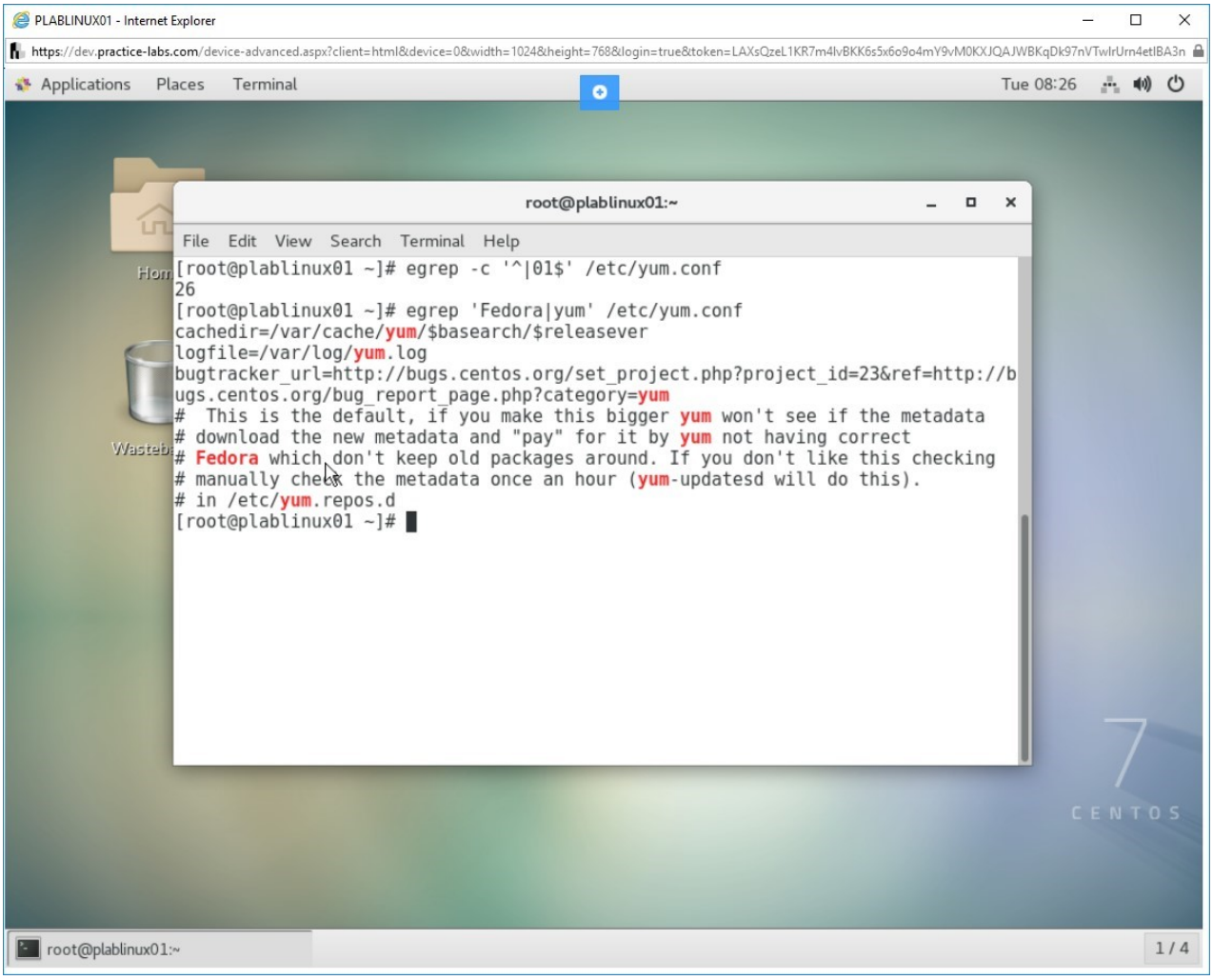


Figure 1.12 Screenshot of PLABLINUX01: Finding two words in a file simultaneously.

Step 5

Clear the screen by entering the following command:

```
clear
```

The **egrep** and **fgrep** commands are equivalent to grep with **-E** and **-F**.

The **fgrep** command is similar to **grep,** but it does not process any regular expression meta-characters as being special characters. To use the **fgrep** tool, enter the following command:

```
fgrep -c 'yum' /etc/yum.conf
```

"-c" stands for Count, this command counts how many times the word "yum" is in the file then outputs the total number.

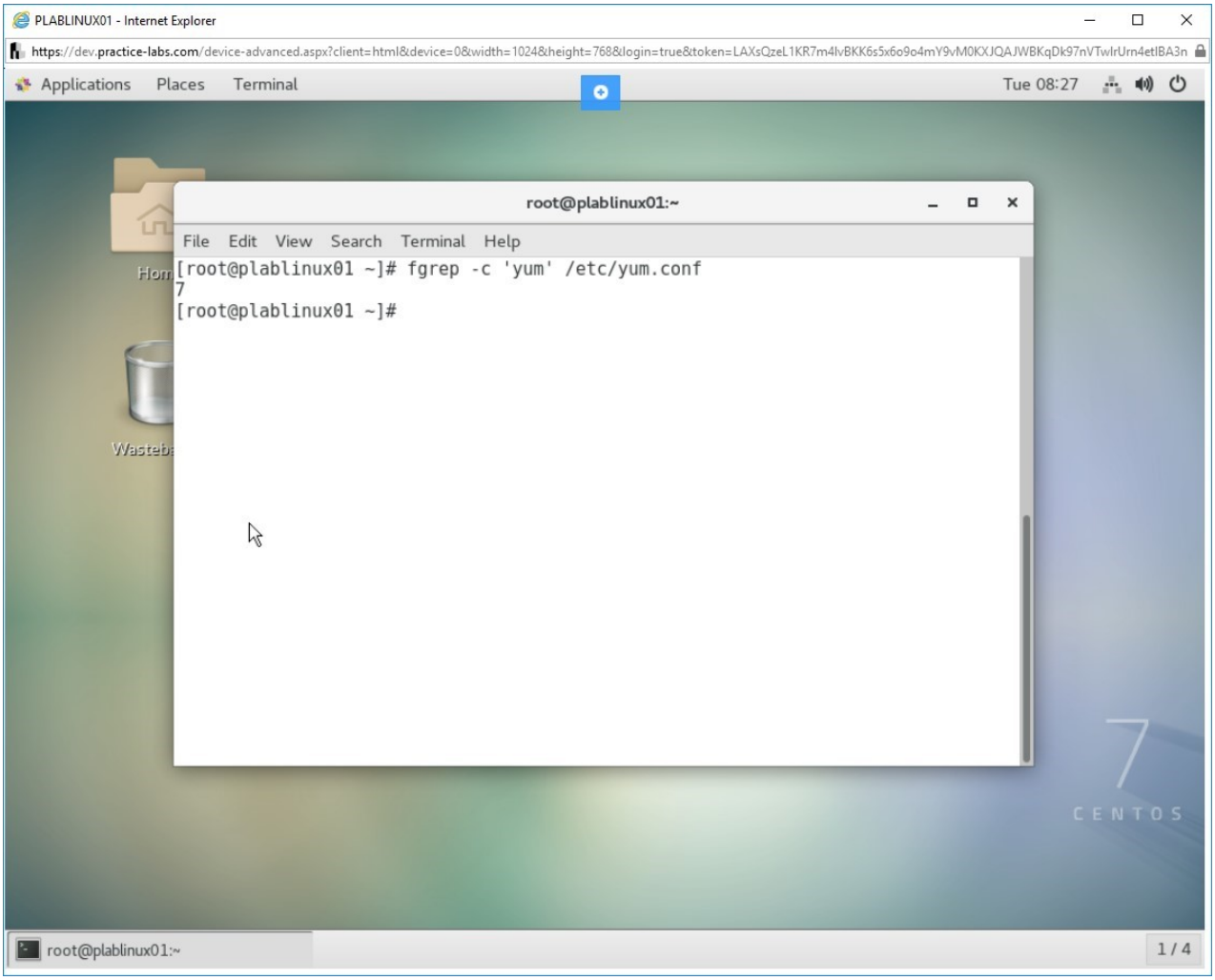


Figure 1.13 Screenshot of PLABLINUX01: Counting the occurrence of the word yum in the /etc/yum.conf file.

Step 6

To use the **egrep** tool to count two words, enter the following command:

```
egrep -c 'yum|Fedora' /etc/yum.conf
```

This command finds the total number of both words - **yum** as well as **Fedora** - in the **yum.conf** file.

You are unable to count multiple words using **fgrep**.

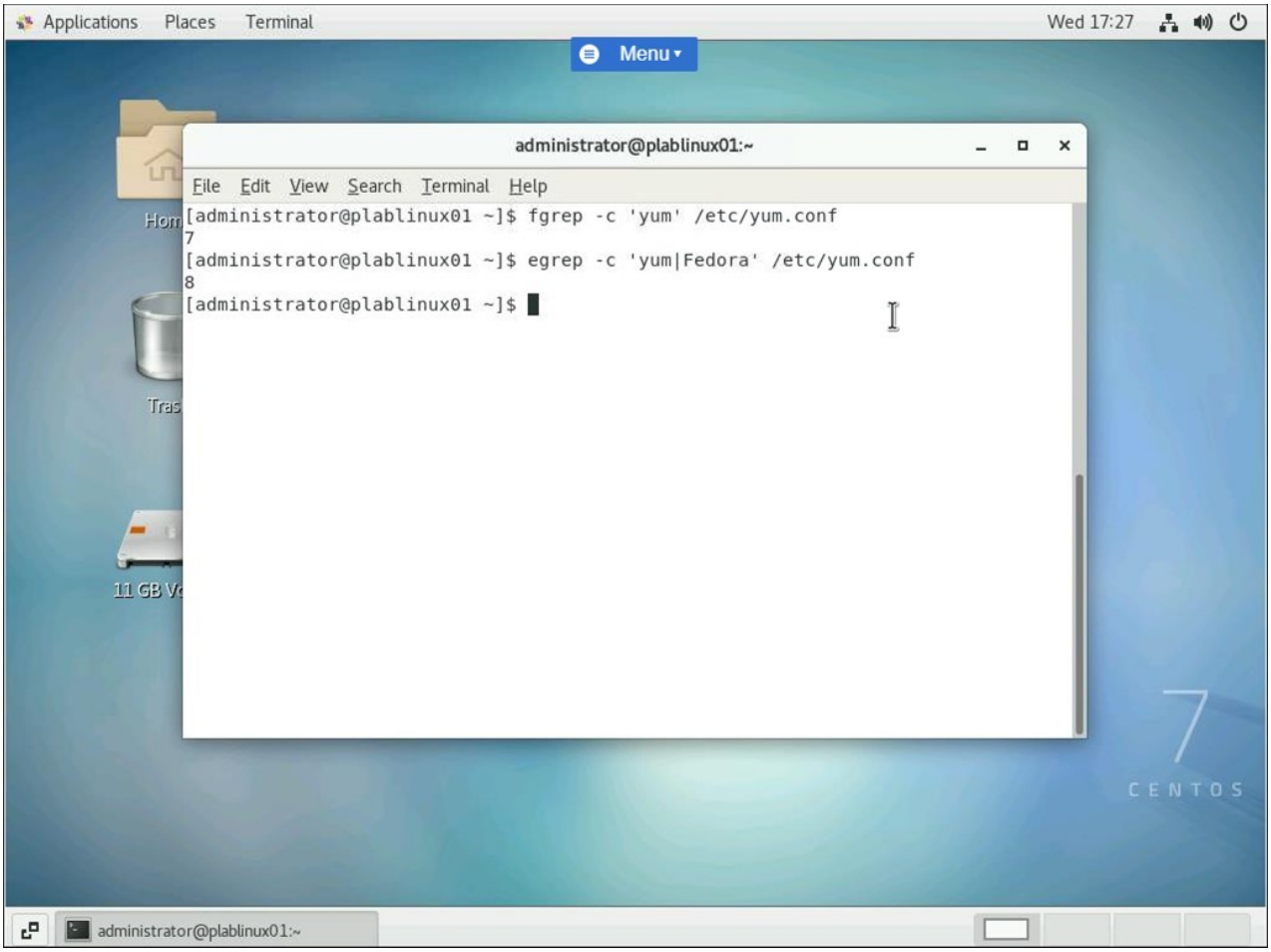


Figure 1.14 Screenshot of PLABLINUX01: Using fgrep to find two words in a file.

Step 7

Clear the screen with the **clear** command.

> ***Note**: You will be performing the following steps on the yum.conf file or any other file of your choice. You can also create a file with sample text and comments. However, it is recommended to make a copy of the file before running the below-listed commands to modify the file.*

To create a copy of the **yum.conf** file, enter the following command:

```
cp /etc/yum.conf yumtest.conf
```

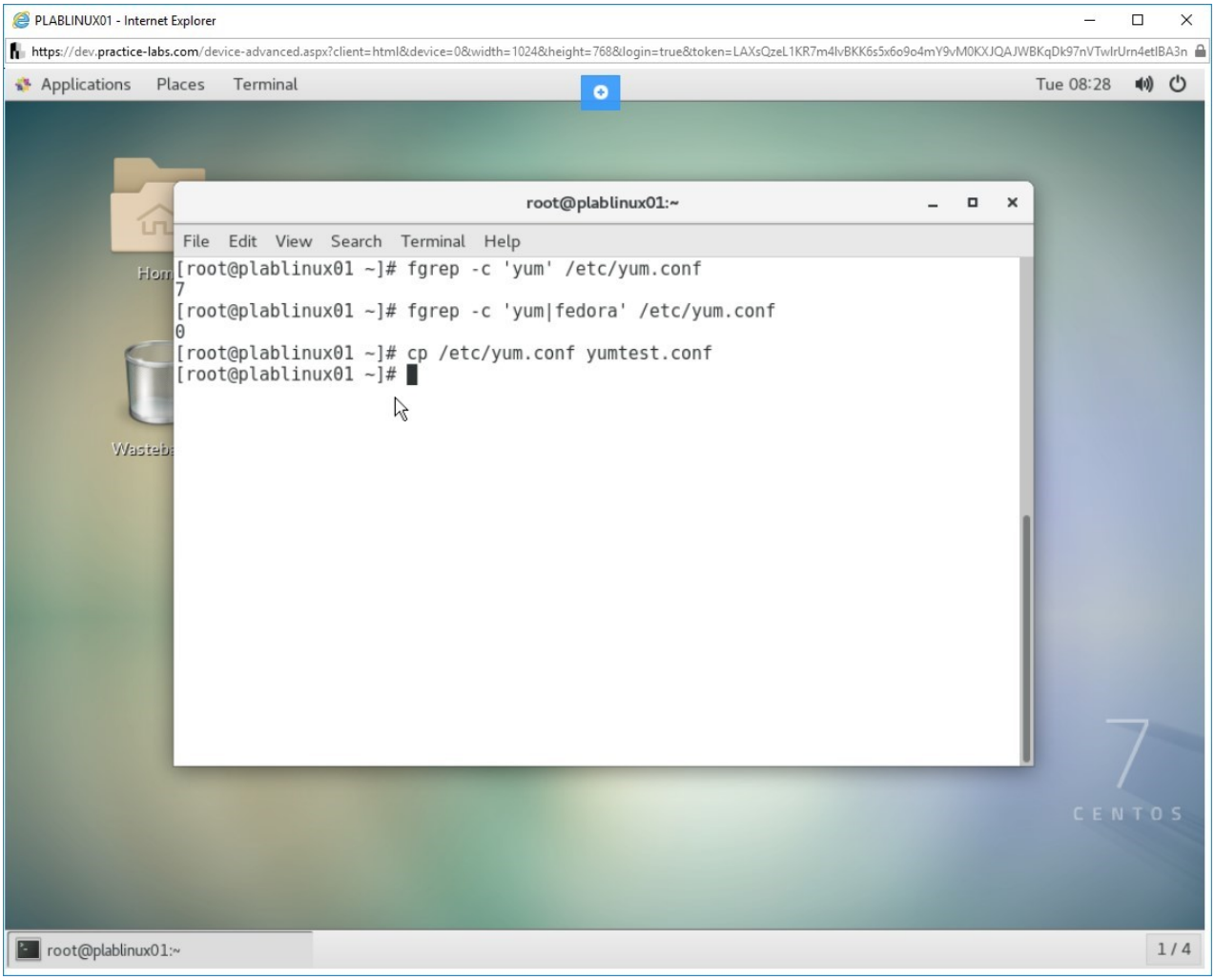


Figure 1.15 Screenshot of PLABLINUX01: Creating a copy of the yum.conf file.

Step 8

Clear the screen by entering the following command:

```
clear
```

The **sed** tool is used for performing automatic non-interactive editing of files. You can use most of the regular expressions with this tool to locate the required text.

For example, to delete all the commented lines in the **yumtest.conf** file, enter the following command:

```
sed '/^#/ d ' yumtest.conf
```

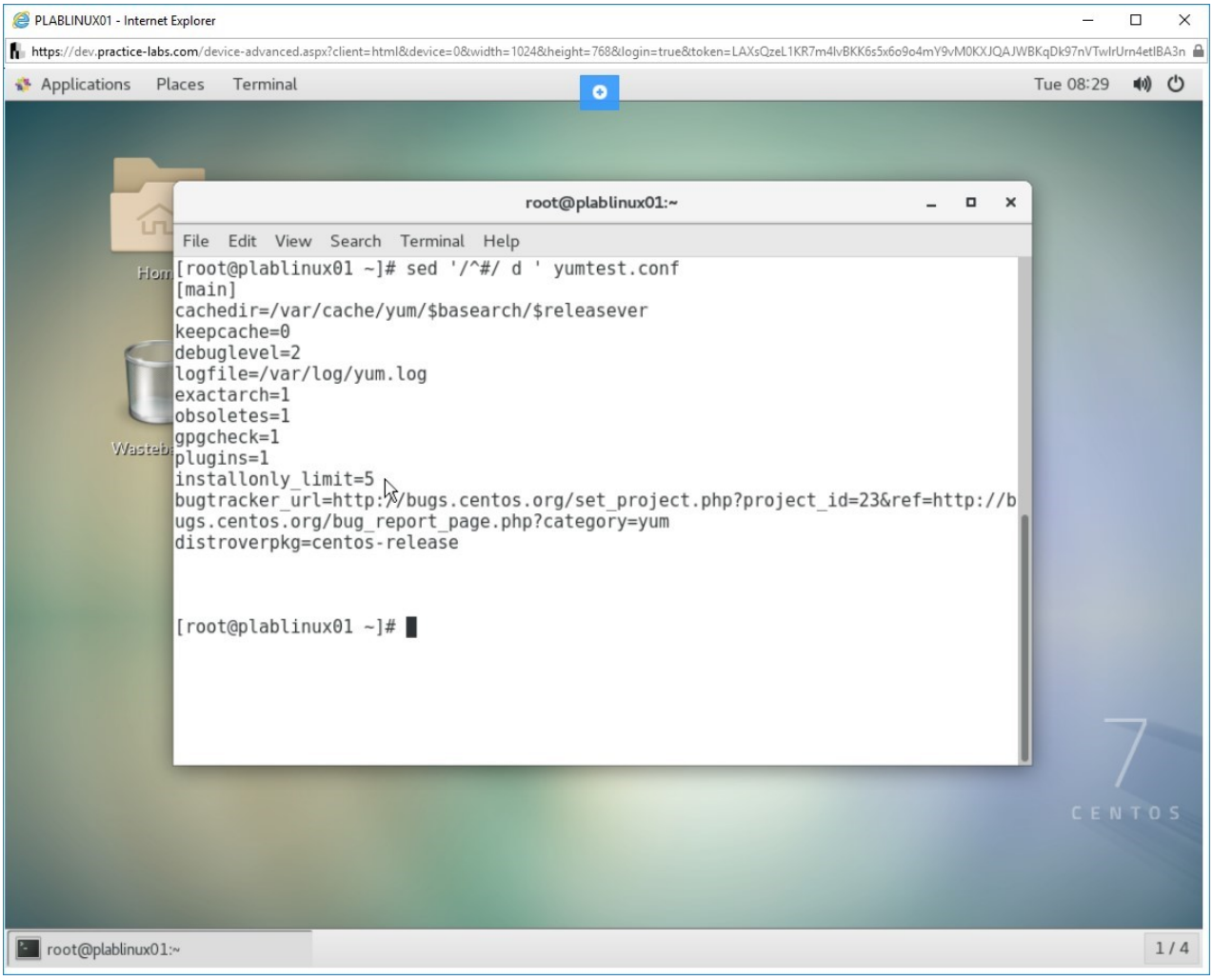


Figure 1.16 Screenshot of PLABLINUX01: Deleting all the commented lines in the yumtest.conf file.

Step 9

Clear the screen with the clear command.

To delete all the blank lines at the end of the file, enter the following command:

```
sed '/^  / d' yumtest.conf
```

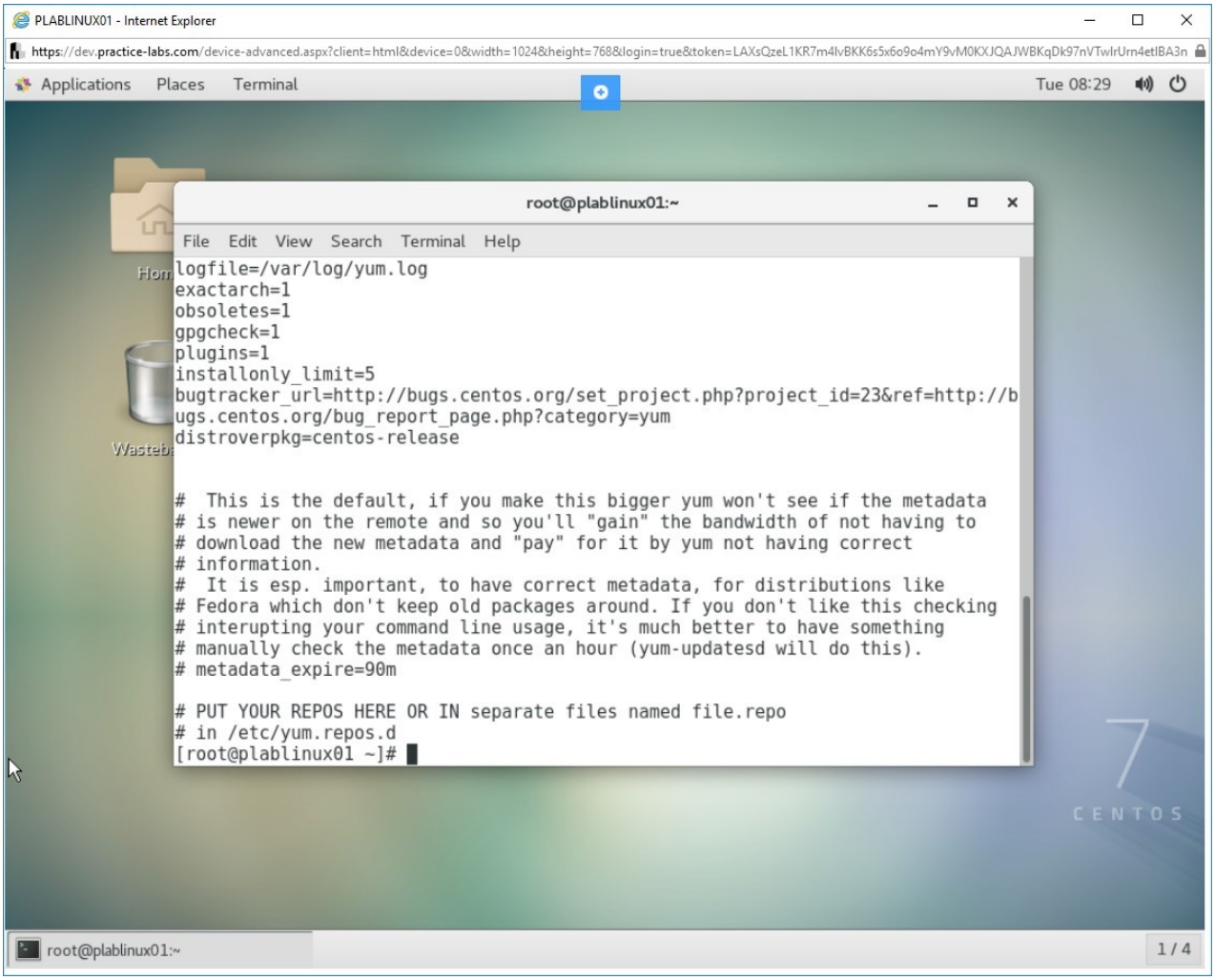


Figure 1.17 Screenshot of PLABLINUX01: Deleting all the blank lines at the end of the file.

Step 10

Clear the screen by entering the following command:

```
clear
```

You can also combine more than one **sed** commands. To combine more than one **sed** command, you use the **-e** switch. Enter the following command to delete all the blank lines and replace Fedora with Linux:

```
sed -e '/^$/ d'  -e 's/Fedora/Linux/g'  yumtest.conf
```

Notice the specified changes have been made to the file displayed in the output.

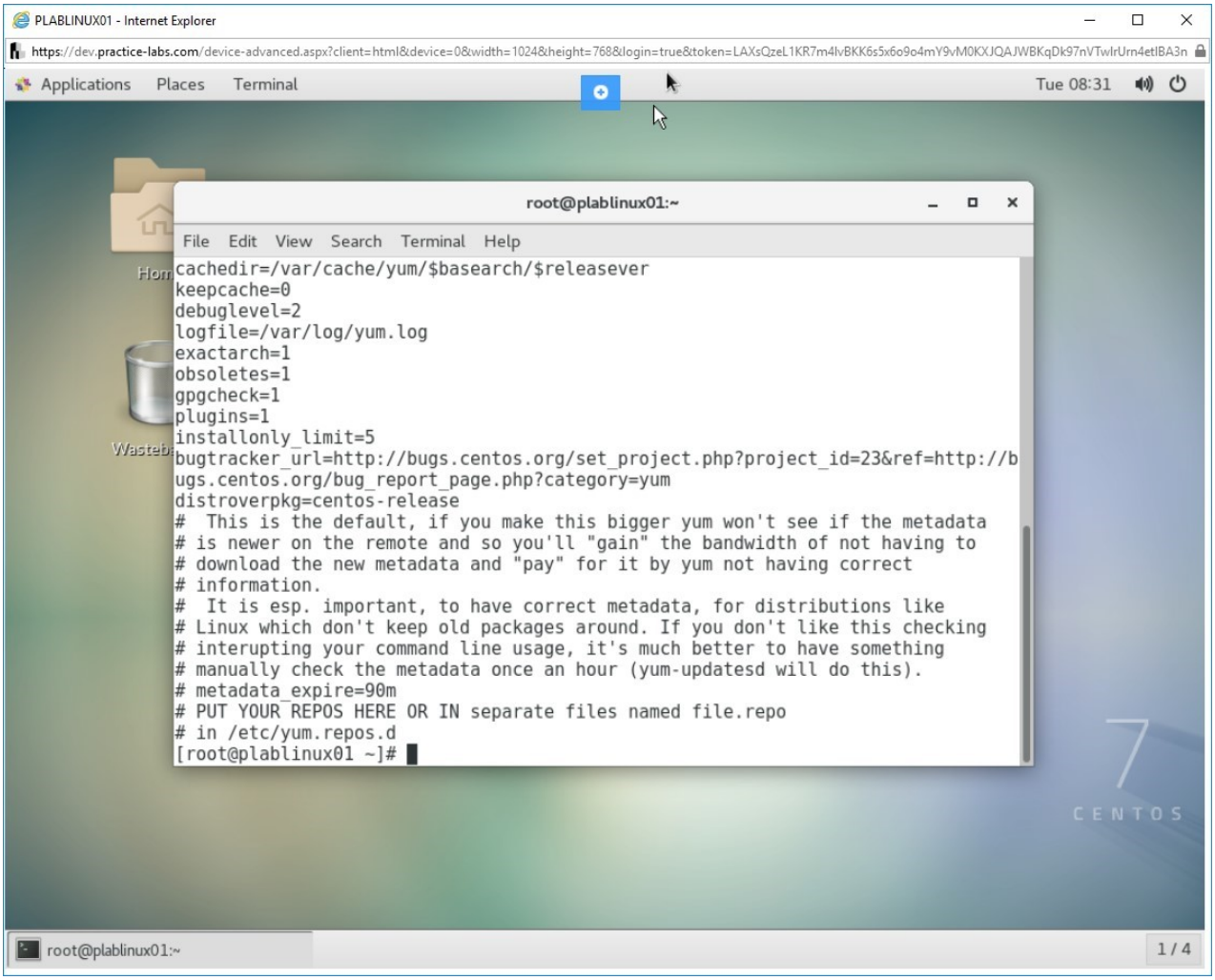


Figure 1.18 Screenshot of PLABLINUX01: Deleting all the blank lines and replacing Fedora with Linux.

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

Review

Well done, you have completed the **Search Text Files using Regular Expressions** Practice Lab.

Summary

You completed the following exercise:

- Exercise 1 - Search Text Files using Regular Expressions

You should now be able to:

- Search for specific text in a file
- Search for specific criteria through a file content or a filesystem

Feedback

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