

# Perform CPU Monitoring and Configuration

- **Introduction**
  - **Lab Topology**
  - **Exercise 1 - Perform CPU Monitoring**
  - **Review**
- 

## Introduction

Welcome to the **Perform CPU Monitoring and Configuration** Practice Lab. In this module you will be provided with the instructions and devices needed to develop your hands-on skills.

CPU Monitoring  
Configuration  
CPU  
Perf

## Learning Outcomes

In this module, you will complete the following exercise:

- Exercise 1 - Perform CPU Monitoring

After completing this lab, you will be able to:

- Use various commands to monitor a CPU
- Install and use perf

## Exam Objectives

The following exam objectives are covered in this lab:

- **LPI: 108.2 System logging**
- **CompTIA: 3.4 Given a scenario, implement logging services.**

**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.

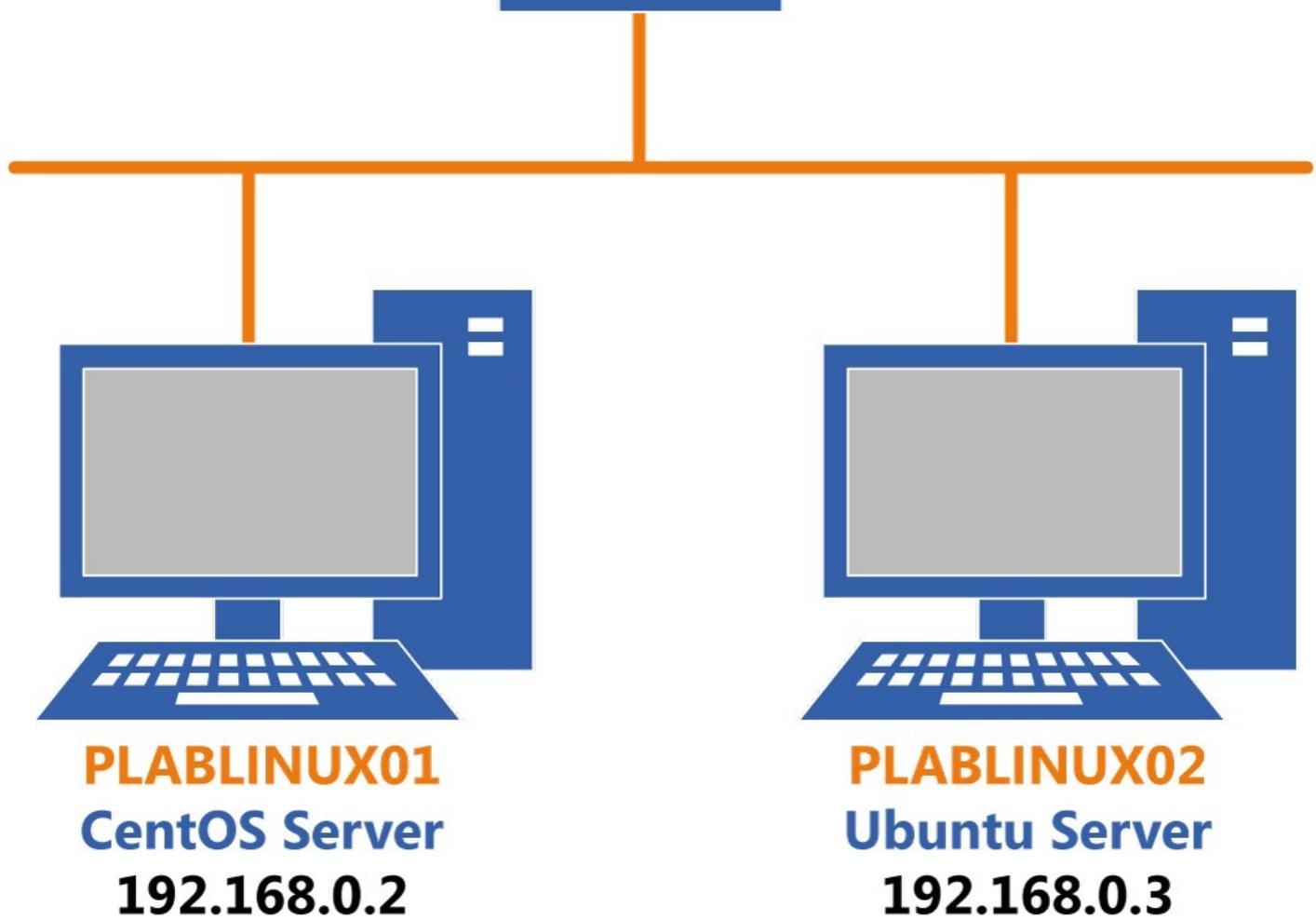
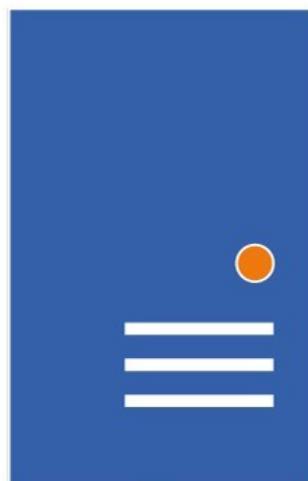
Click Next to view the Lab topology used in this module.

---

## Lab Topology

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

**PLABSA01**  
**Windows Server 2016**  
**192.168.0.1**



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 - Perform CPU Monitoring

Monitoring CPU utilization is one of the key system administration tasks. Each server can have various applications installed. The administrator must run various commands to monitor the CPU utilization and then fine-tune the system performance if required.

In this exercise, you will learn to perform CPU monitoring.

## Learning Outcomes

After completing this exercise, you will be able to:

- Log into a Linux System
- Use various commands to monitor a CPU
- Install and use perf

## Your Devices

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

- **PLABLINUX01** (CentOS Server)



## Task 1 - Use Various Commands to Monitor a CPU

There are various commands that you can use to monitor one or more CPUs in a server. Some commands allow you to capture static information and some allow you to capture real-time information. Some of the key commands are used in the exercise. In

this task, you will learn to use various commands to monitor a CPU. To do this, perform the following steps:

## Step 1

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

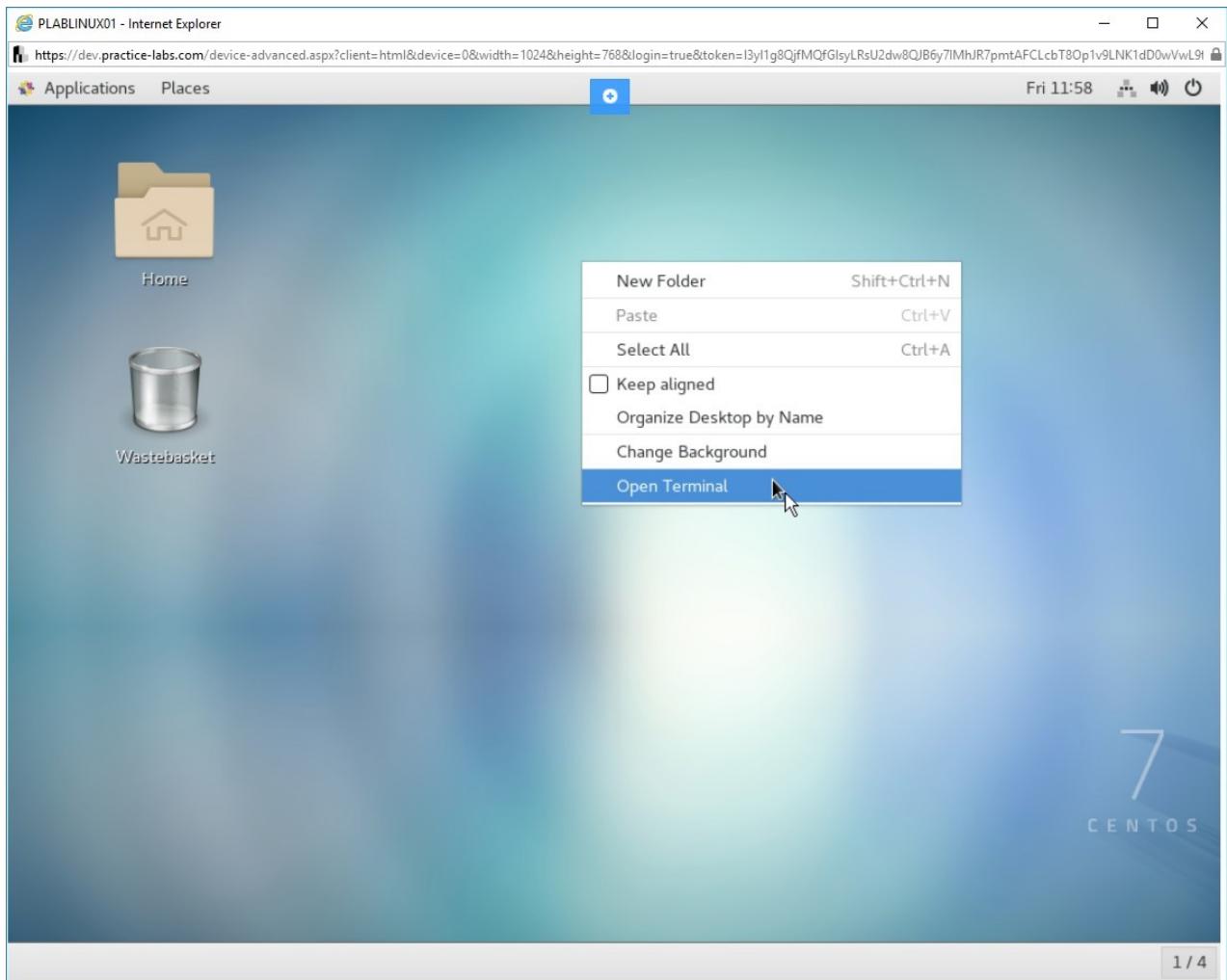


Figure 1.1 Screenshot of PLABLINUX01: Selecting the Open Terminal option from the context menu.

## Step 2

The terminal window is displayed. The top command displays the CPU, memory, tasks, and swap information. Type the following command:

```
top
```

Press **Enter**.

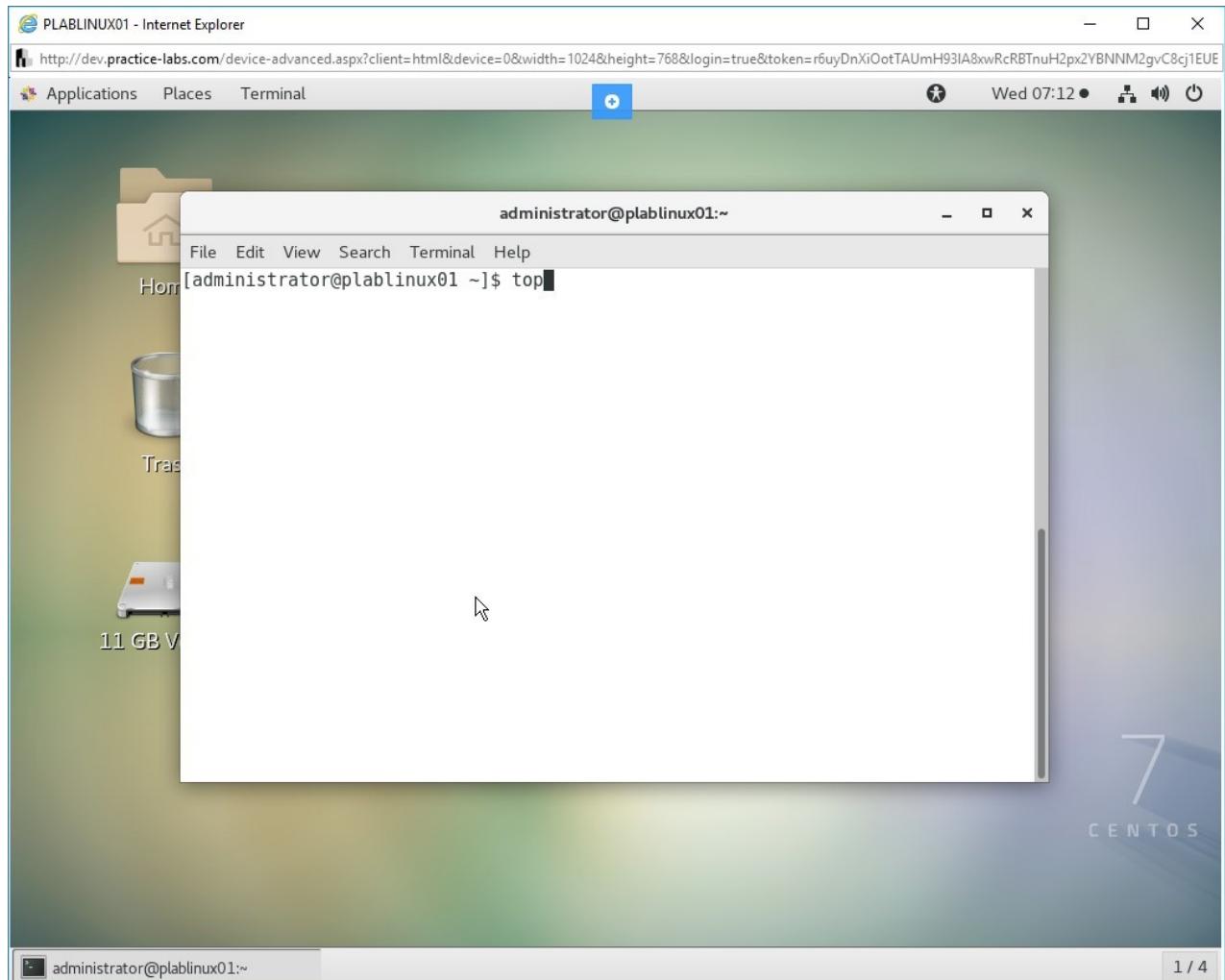


Figure 1.2 Screenshot of PLABLINUX01: Executing the top command.

## Step 3

The output of the **top** command is displayed. Notice the CPU column, which is the fourth column from the right.

**Note:** To break the command, press *Ctrl + C*.

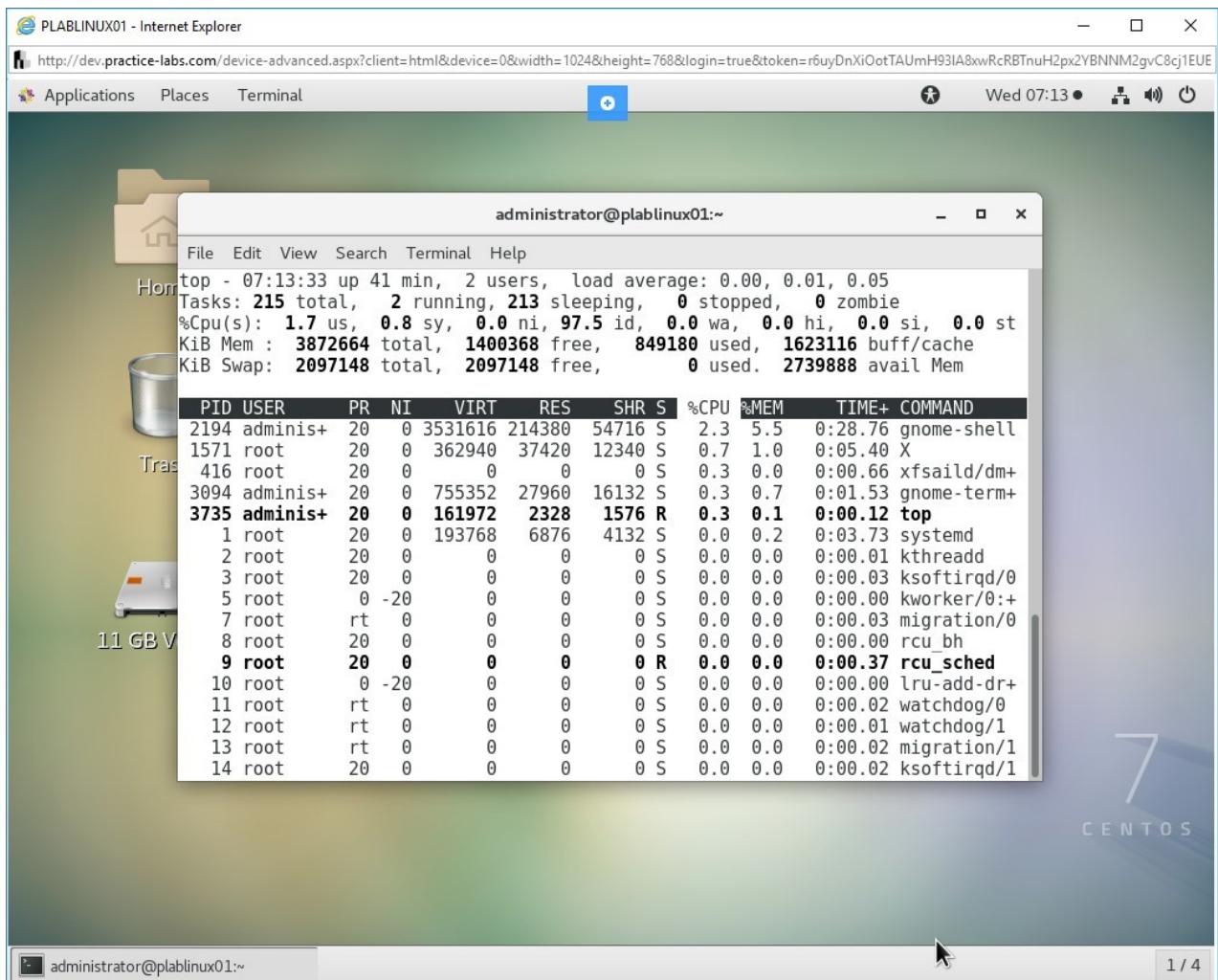


Figure 1.3 Screenshot of PLABLINUX01: Displaying the result of the top command.

## Step 4

Clear the screen by entering the following command:

```
clear
```

You can also display the processes and the CPU utilization by a specific user. Type the following command:

```
top -u root
```

Press **Enter**.

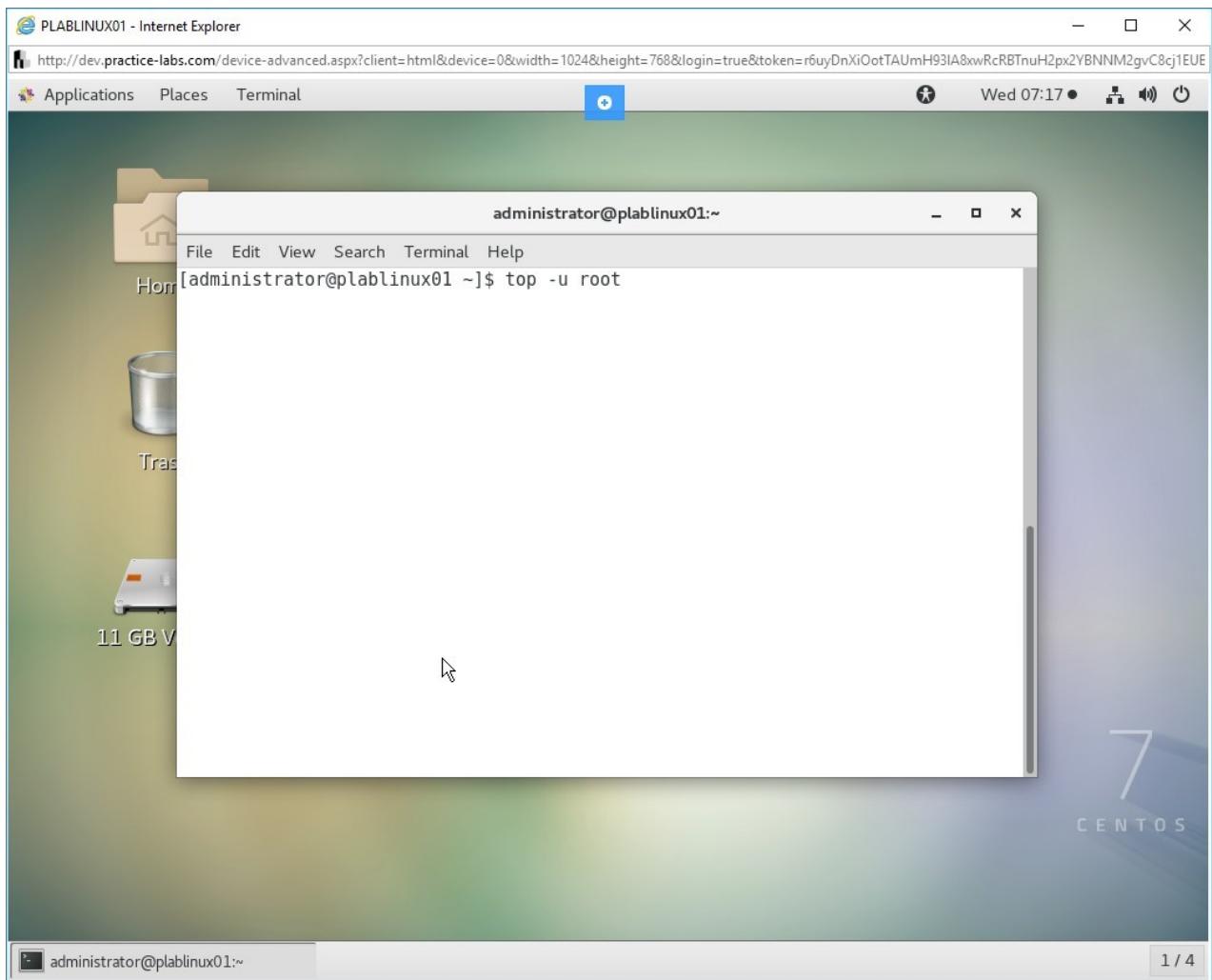


Figure 1.4 Screenshot of PLABLINUX01: Displaying the processes and the CPU utilization by a specific user.

## Step 5

The output of the **top -u root** command is displayed.

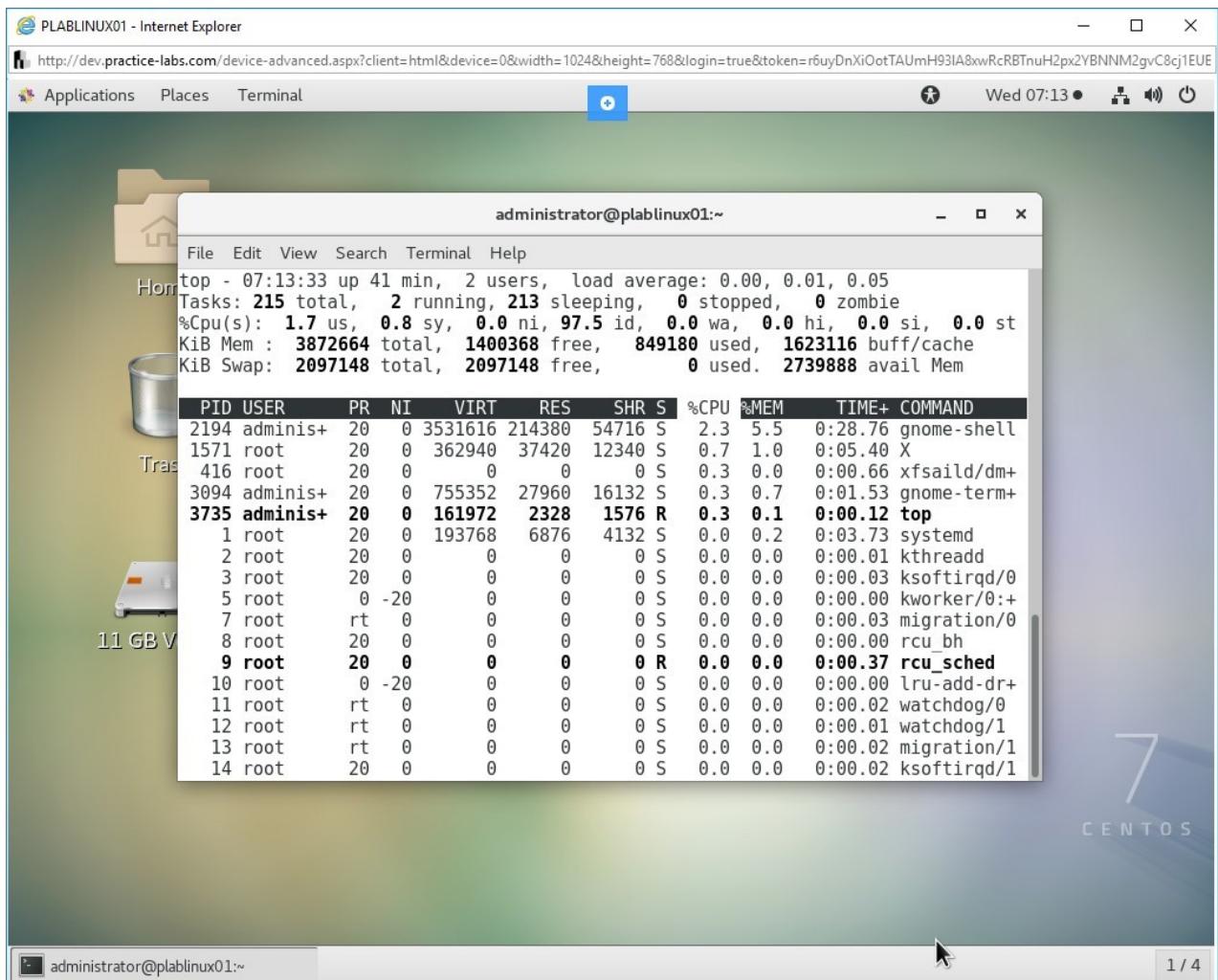


Figure 1.5 Screenshot of PLABLINUX01: Displaying the output of the top -u root command.

## Step 6

You can also sort the output by CPU usage. Press **Shift + P** to sort the processes by CPU utilization. Notice that the sorting is now by CPU utilization.

**Note:** To break the command, press **Ctrl + C**.

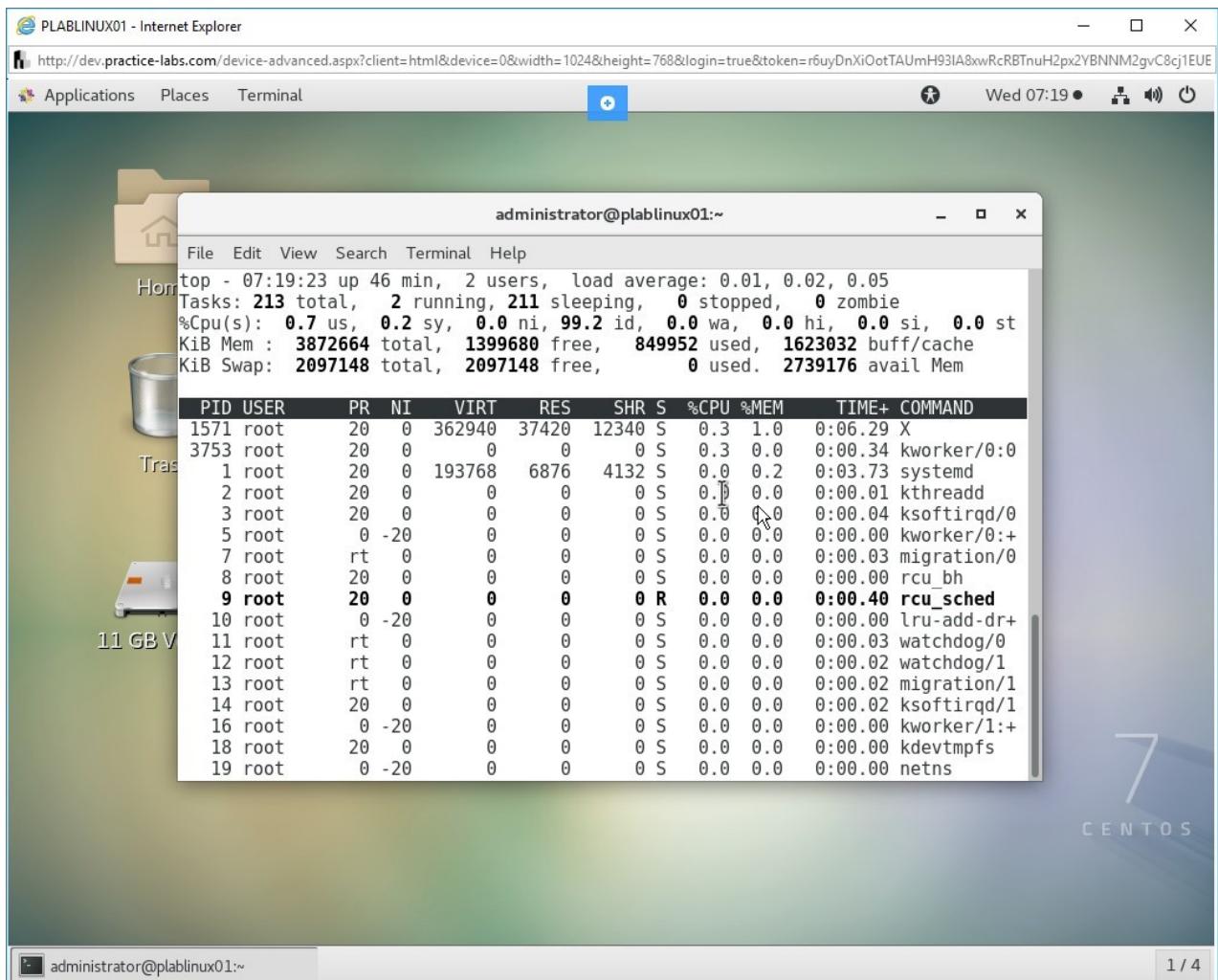


Figure 1.6 Screenshot of PLABLINUX01: Sorting the top command output.

## Step 7

Clear the screen by entering the following command:

```
clear
```

You can use the **mpstat** command to display the CPU information. This command is useful when you have a system with multiple CPUs. Type the following command:

```
mpstat
```

Press **Enter**. Notice that this command shows all under the CPU column. This means that it is showing the consolidated performance of two CPUs.

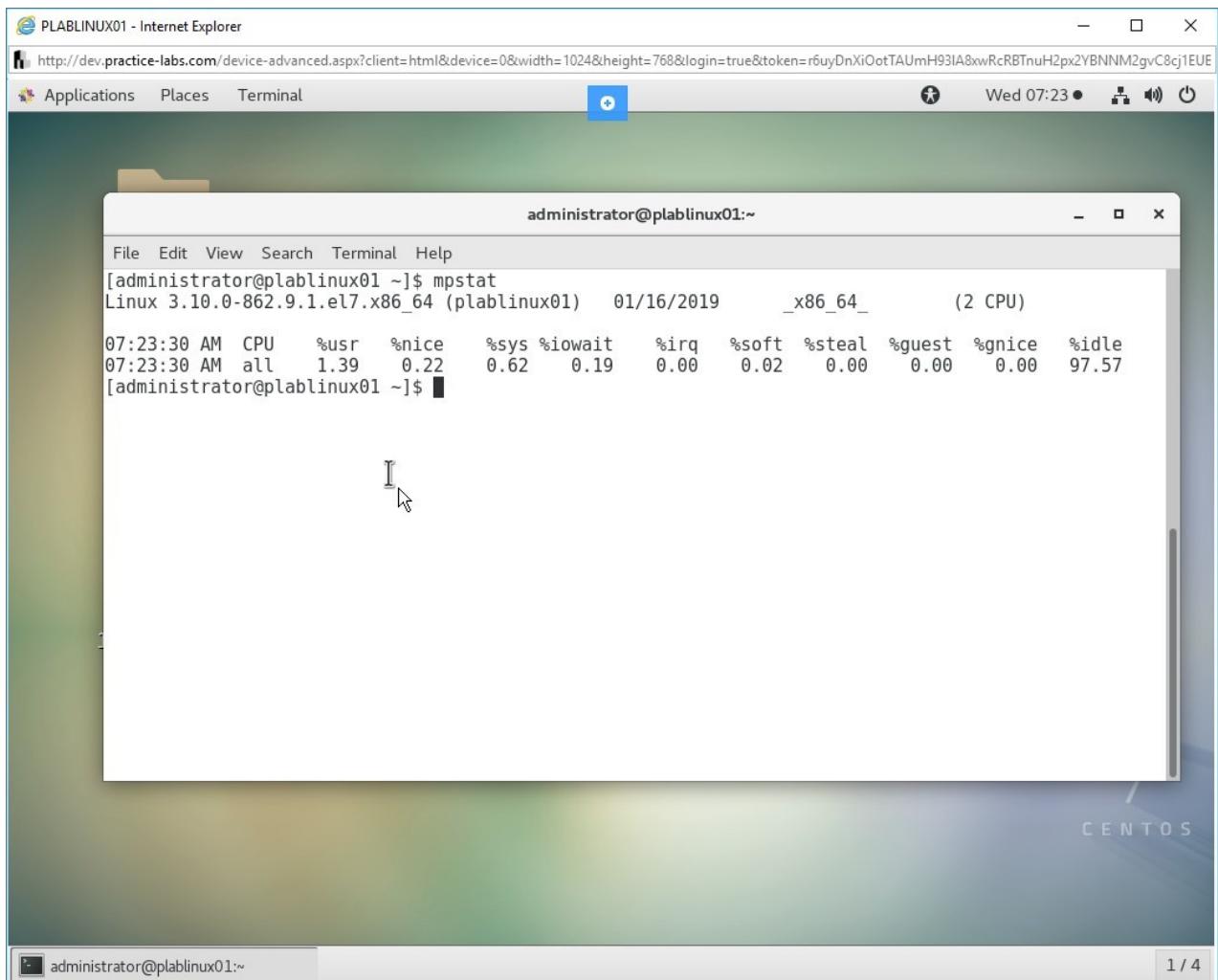


Figure 1.7 Screenshot of PLABLINUX01: Executing the mpstat command.

## Step 8

Using the mpstat command, you can also show the individual performance of the CPUs. Type the following command:

```
mpstat -P ALL
```

Press **Enter**. Notice that this command now shows the individual performance of each CPU.

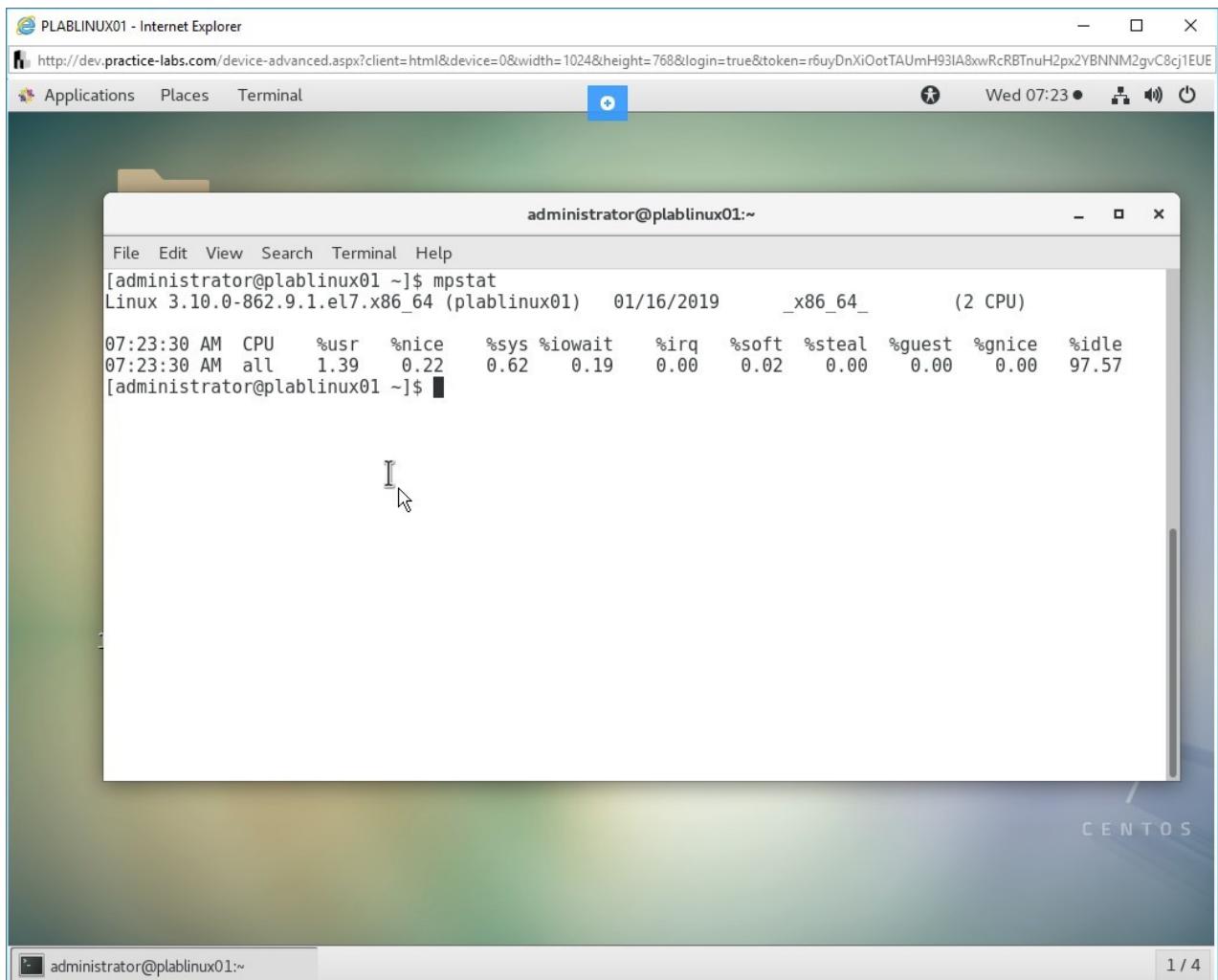


Figure 1.8 Screenshot of PLABLINUX01: Showing the individual performance of the CPUs.

## Step 9

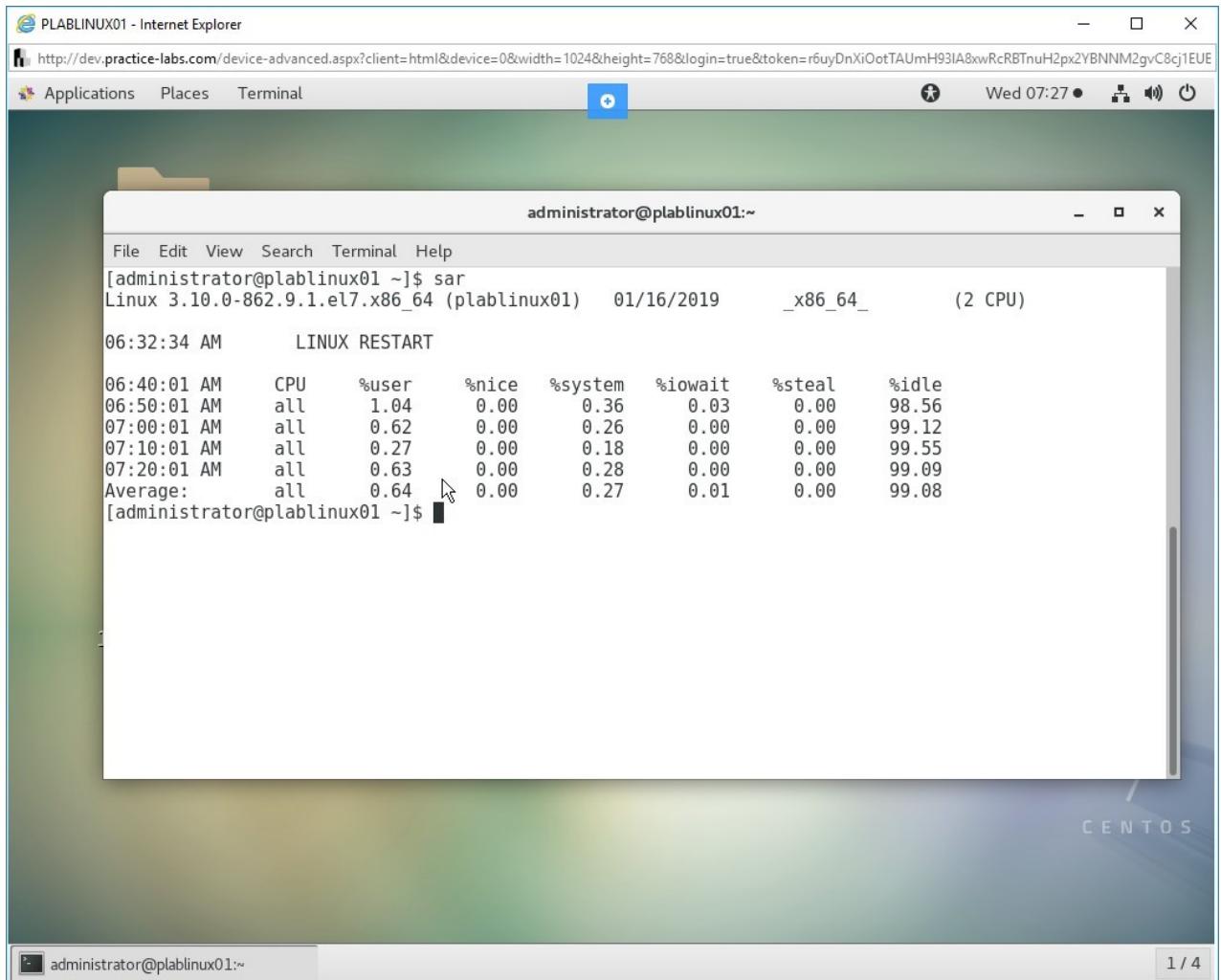
Clear the screen by entering the following command:

```
clear
```

Using the **sar** command, you can display CPU activity for today. Type the following command:

```
sar
```

Press **Enter**. Notice that this command shows all under the CPU column. This means that it is showing the consolidated performance of two CPUs.



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "administrator@plablinux01:~". The window content displays the output of the "sar" command. The output shows system activity over time, including CPU usage statistics. The terminal window is part of a desktop interface with other windows like "PLABLINUX01 - Internet Explorer" and "Applications Places Terminal" visible in the background.

```
administrator@plablinux01:~$ sar
Linux 3.10.0-862.9.1.el7.x86_64 (plablinux01) 01/16/2019 _x86_64_ (2 CPU)

06:32:34 AM      LINUX RESTART

06:40:01 AM      CPU      %user    %nice   %system   %iowait   %steal    %idle
06:50:01 AM      all      1.04     0.00     0.36      0.03     0.00     98.56
07:00:01 AM      all      0.62     0.00     0.26      0.00     0.00     99.12
07:10:01 AM      all      0.27     0.00     0.18      0.00     0.00     99.55
07:20:01 AM      all      0.63     0.00     0.28      0.00     0.00     99.09
Average:         all      0.64     0.00     0.27      0.01     0.00     99.08
```

Figure 1.9 Screenshot of PLABLINUX01: Displaying the output of the sar command.

## Step 10

Using the **sar** command, you can also display the CPU activities with a difference of a few seconds. For example, you can display the CPU utilization every two seconds 10 times. Type the following command:

```
sar -u 2 10
```

Press **Enter**. Notice that the CPU utilization is displayed 10 times with a difference of 2 seconds.

The screenshot shows a terminal window titled "administrator@plablinux01:~". The window displays a series of CPU utilization statistics over time, starting at 06:32:34 AM and ending at 07:30:46 AM. The data is presented in a table format with columns for time, CPU type, user percentage, nice percentage, system percentage, iowait percentage, steal percentage, and idle percentage. The terminal also shows the command used to generate the report: [administrator@plablinux01 ~]\$ sar -u 2 10. The background of the desktop shows a green gradient and the word "CENTOS".

Time	CPU	%user	%nice	%system	%iowait	%steal	%idle
06:40:01 AM	CPU	1.04	0.00	0.36	0.03	0.00	98.56
06:50:01 AM	all	0.62	0.00	0.26	0.00	0.00	99.12
07:00:01 AM	all	0.27	0.00	0.18	0.00	0.00	99.55
07:10:01 AM	all	0.63	0.00	0.28	0.00	0.00	99.09
Average:	all	0.64	0.00	0.27	0.01	0.00	99.08
[administrator@plablinux01 ~]\$ sar -u 2 10							
Linux 3.10.0-862.9.1.el7.x86_64 (plablinux01) 01/16/2019 _x86_64_ (2 CPU)							
07:30:30 AM	CPU	2.26	0.00	0.50	0.00	0.00	97.24
07:30:32 AM	all	2.01	0.00	0.75	0.00	0.00	97.24
07:30:34 AM	all	2.26	0.00	0.50	0.00	0.00	97.24
07:30:36 AM	all	2.26	0.00	0.50	0.00	0.00	97.24
07:30:38 AM	all	3.02	0.00	1.76	0.00	0.00	95.23
07:30:40 AM	all	2.26	0.00	0.50	0.00	0.00	97.24
07:30:42 AM	all	1.75	0.00	0.25	0.00	0.00	97.99
07:30:44 AM	all	2.26	0.00	0.25	0.00	0.00	97.49
07:30:46 AM	all	2.01	0.00	0.50	0.00	0.00	97.49
07:30:48 AM	all	2.26	0.00	0.25	0.00	0.00	97.49
07:30:50 AM	all	3.27	0.00	0.50	0.00	0.00	96.23
Average:	all	2.33	0.00	0.58	0.00	0.00	97.09
[administrator@plablinux01 ~]\$							

Figure 1.10 Screenshot of PLABLINUX01: Displaying the CPU utilization every two seconds 10 times using the sar command.

## Step 11

Clear the screen by entering the following command:

```
clear
```

You can use the **sar** command to get multiple samples and multiple reports using an output file. Also, you can make this process run in the background. Type the following command:

```
sar -o plab.file 10 10 >/dev/null 2>&1 &
```

Press **Enter**. Notice that only a process id is displayed.

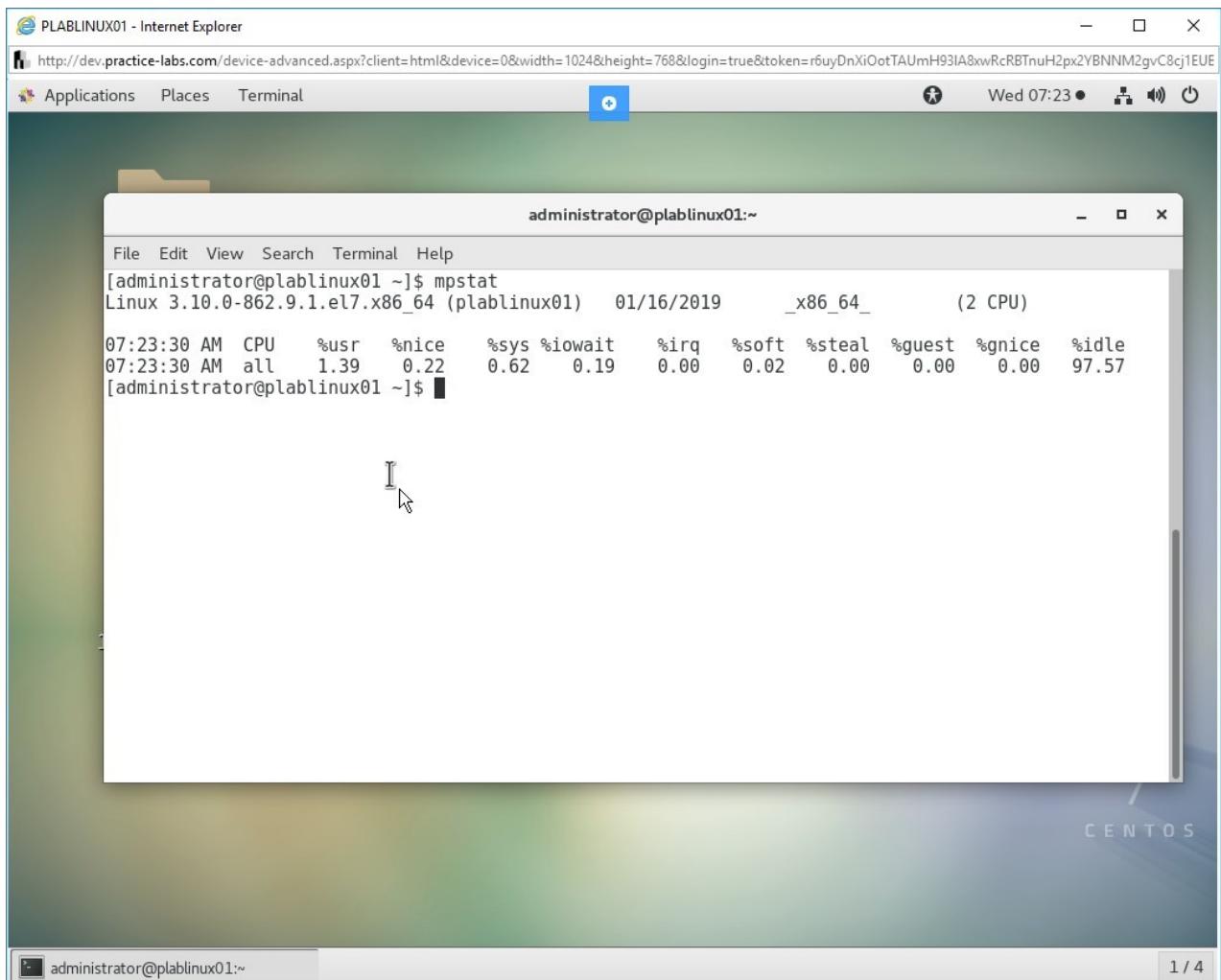


Figure 1.11 Screenshot of PLABLINUX01: Getting multiple samples and multiple reports using an output file using the sar command.

## Step 12

After the file is generated with the output, you can use the file to display the CPU statistics.

```
sar -f plab.file
```

Press **Enter**. Notice the captured information is displayed.

The screenshot shows a terminal window titled "administrator@plablinux01:~". The window displays the output of the "sar" command, specifically the CPU statistics. The output includes a header with system information (Linux 3.10.0-862.9.1.el7.x86\_64), the date (01/16/2019), and the architecture (\_x86\_64\_). Below this, there is a detailed table of CPU usage statistics for each minute from 07:33:18 AM to 07:34:58 AM. The columns represent time, CPU type (all), user percentage, nice percentage, system percentage, iowait percentage, steal percentage, and idle percentage. The "idle" column shows values ranging from 97.39% to 99.80%. An average row is shown at the bottom. The command "sar -o plab.file 10 10 >/dev/null 2>&1" is run at the end, followed by "[1]+ Done".

Time	CPU	%user	%nice	%system	%iowait	%steal	%idle
07:33:18 AM	all	1.81	0.05	0.40	0.00	0.00	97.74
07:33:28 AM	all	2.01	0.00	0.60	0.00	0.00	97.39
07:33:38 AM	all	1.00	0.00	0.30	0.00	0.00	98.70
07:33:48 AM	all	0.05	0.05	0.10	0.00	0.00	99.80
07:34:08 AM	all	0.15	0.00	0.15	0.00	0.00	99.70
07:34:18 AM	all	0.05	0.00	0.10	0.00	0.00	99.85
07:34:28 AM	all	0.10	0.05	0.10	0.00	0.00	99.75
07:34:38 AM	all	0.15	0.00	0.35	0.00	0.00	99.50
07:34:48 AM	all	0.05	0.00	0.10	0.00	0.00	99.85
07:34:58 AM	all	0.05	0.05	0.15	0.00	0.00	99.75
Average:	all	0.54	0.02	0.24	0.00	0.00	99.20

Figure 1.12 Screenshot of PLABLINUX01: Using the sar output file to display the CPU statistics.

## Step 13

Clear the screen by entering the following command:

```
clear
```

You can also find out the processes that are having high utilization or using more CPU than the other processes. You can sort them and also limit the entries to 10. Type the following command:

```
ps -eo pcpu,pid,user | sort -k 1 -r | head -10
```

Press **Enter**. Notice the captured information is displayed. Processes are sorted based on CPU utilization.

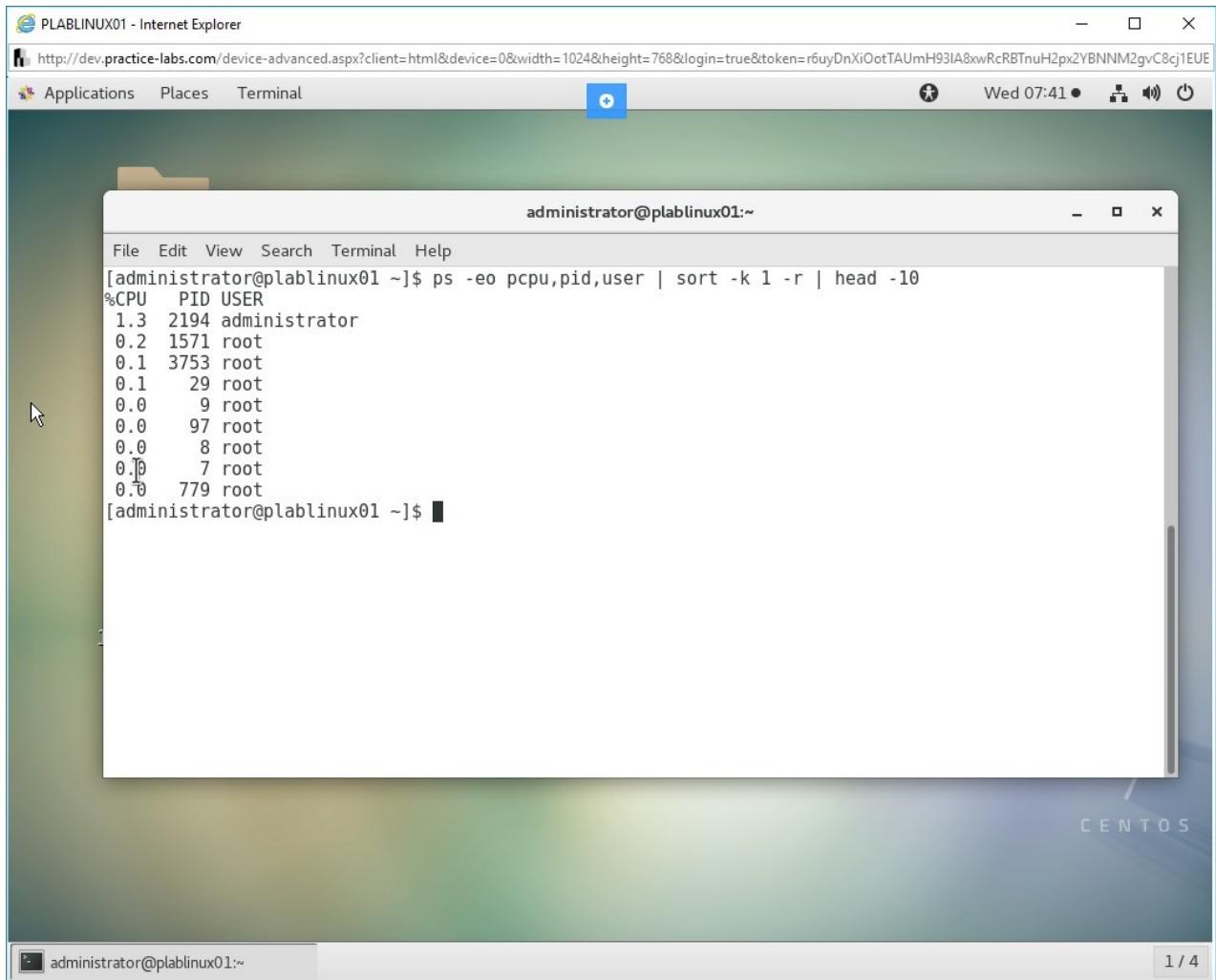


Figure 1.13 Screenshot of PLABLINUX01: Displaying processes with high utilization.

## Step 14

Clear the screen by entering the following command:

```
clear
```

The **iostat** command can display the CPU and input/output statistics for devices and partitions. Type the following command:

```
iostat
```

Press **Enter**.

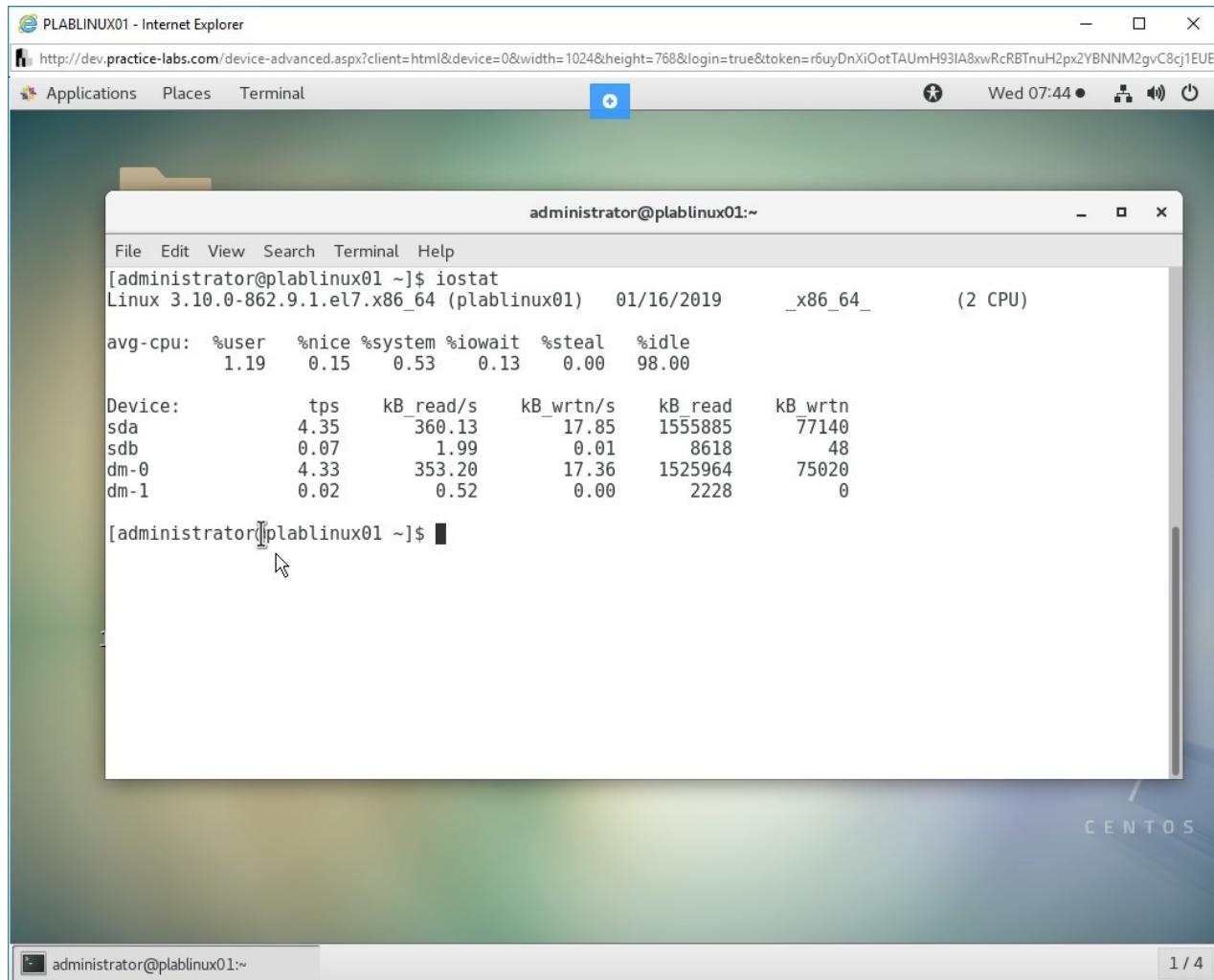


Figure 1.14 Screenshot of PLABLINUX01: Displaying the output of the iostat command.

## Step 15

Clear the screen by entering the following command:

```
clear
```

The **iostat** command can be used to display only the CPU utilization statistics. Type the following command:

```
iostat -c
```

Press **Enter**.

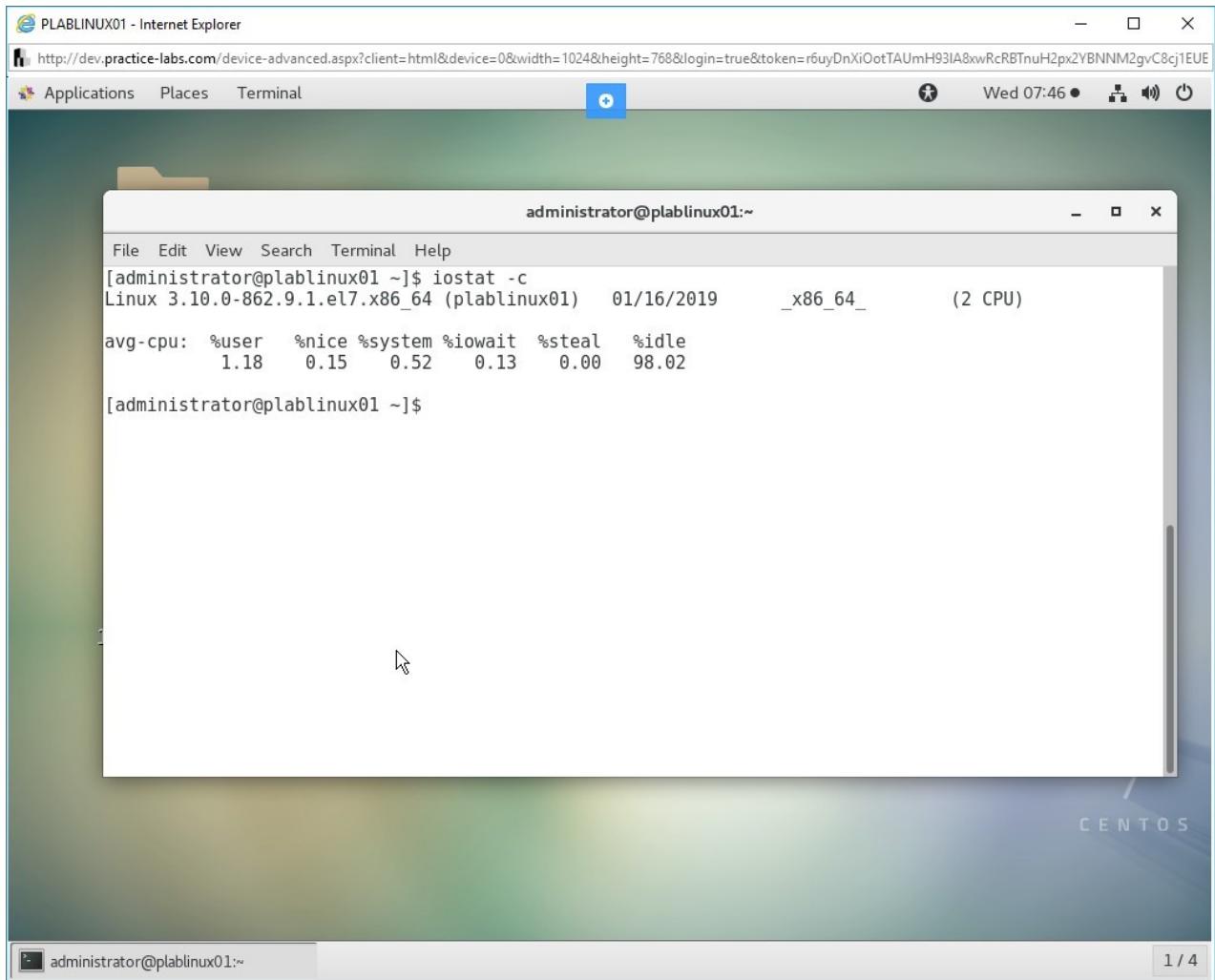


Figure 1.15 Screenshot of PLABLINUX01: Displaying only the CPU utilization statistics using the iostat -c command.

## Step 16

Clear the screen by entering the following command:

```
clear
```

You can also use the **vmstat** command to display various information, such as CPU, swap, and memory utilization. Type the following command:

```
vmstat
```

Press **Enter**. Notice that this command generates reasonably good information for CPU. It generates information for CPU user time, system time, idle time, and wait time.

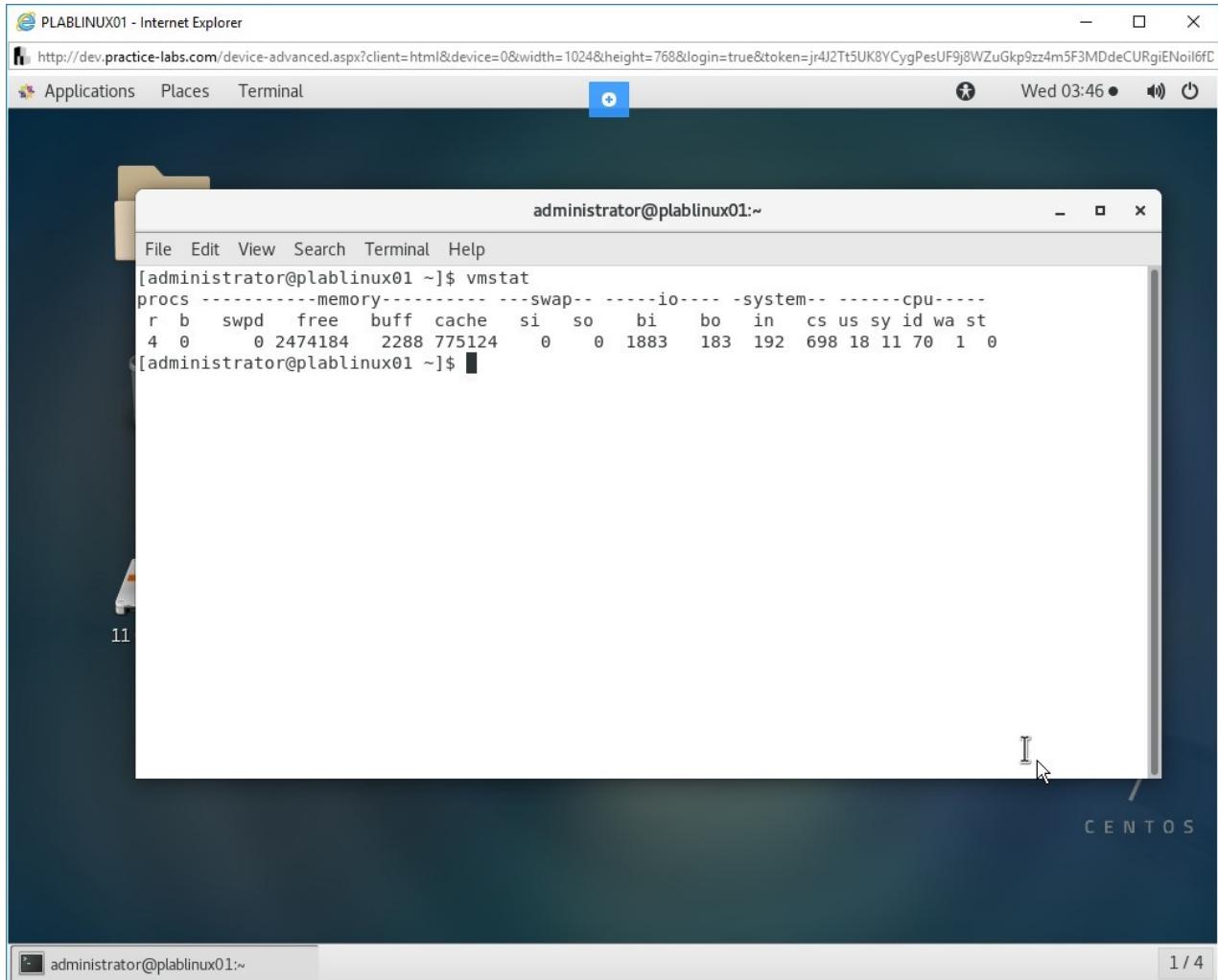


Figure 1.16 Screenshot of PLABLINUX01: Displaying the output of the **vmstat** command.

## Step 17

Clear the screen by entering the following command:

```
clear
```

You can also use the **vmstat** command to display various information, such as CPU, swap, and memory utilization. Type the following command:

```
vmstat -s
```

Press **Enter**. Notice that this command generates output for memory, CPU, and swap.

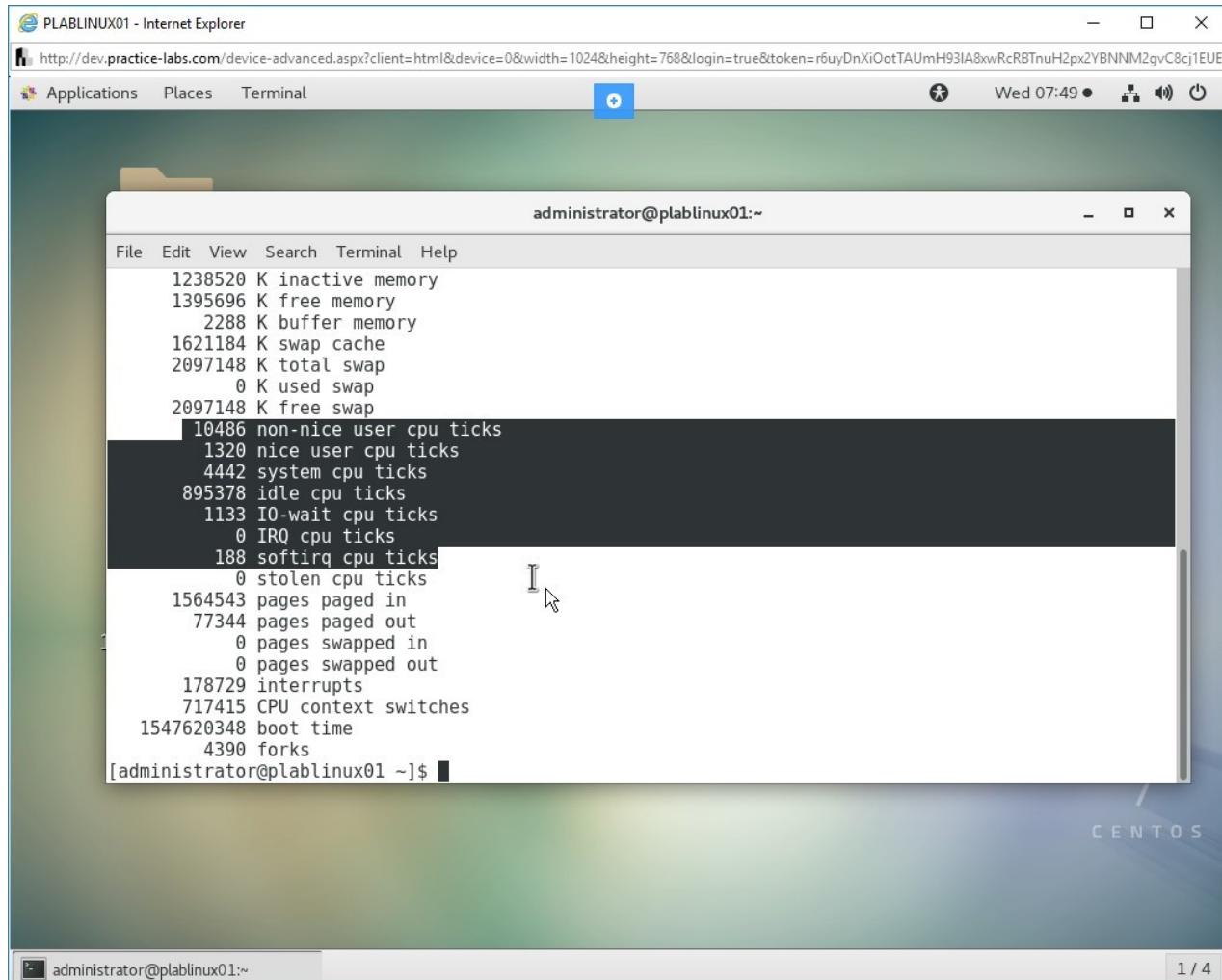


Figure 1.17 Screenshot of PLABLINUX01: Using the vmstat command to display various information, such as CPU, swap, and memory utilization.

## Task 2 - Install and Use Perf

Perf is a monitoring and performance analysis tool for Linux. You can use it on different flavors of Linux, such as Ubuntu and CentOS.

In this task, you will learn to install and use perf. To install and use perf, perform the following steps:

### Step 1

Clear the screen by entering the following command:

clear

The terminal window is displayed. You will install perf, which is a performance monitoring and analysis tool. To do this, type the following command:

`sudo yum install perf`

Press **Enter**.

When prompted, type the following password:

**Passw0rd**

Press **Enter**.

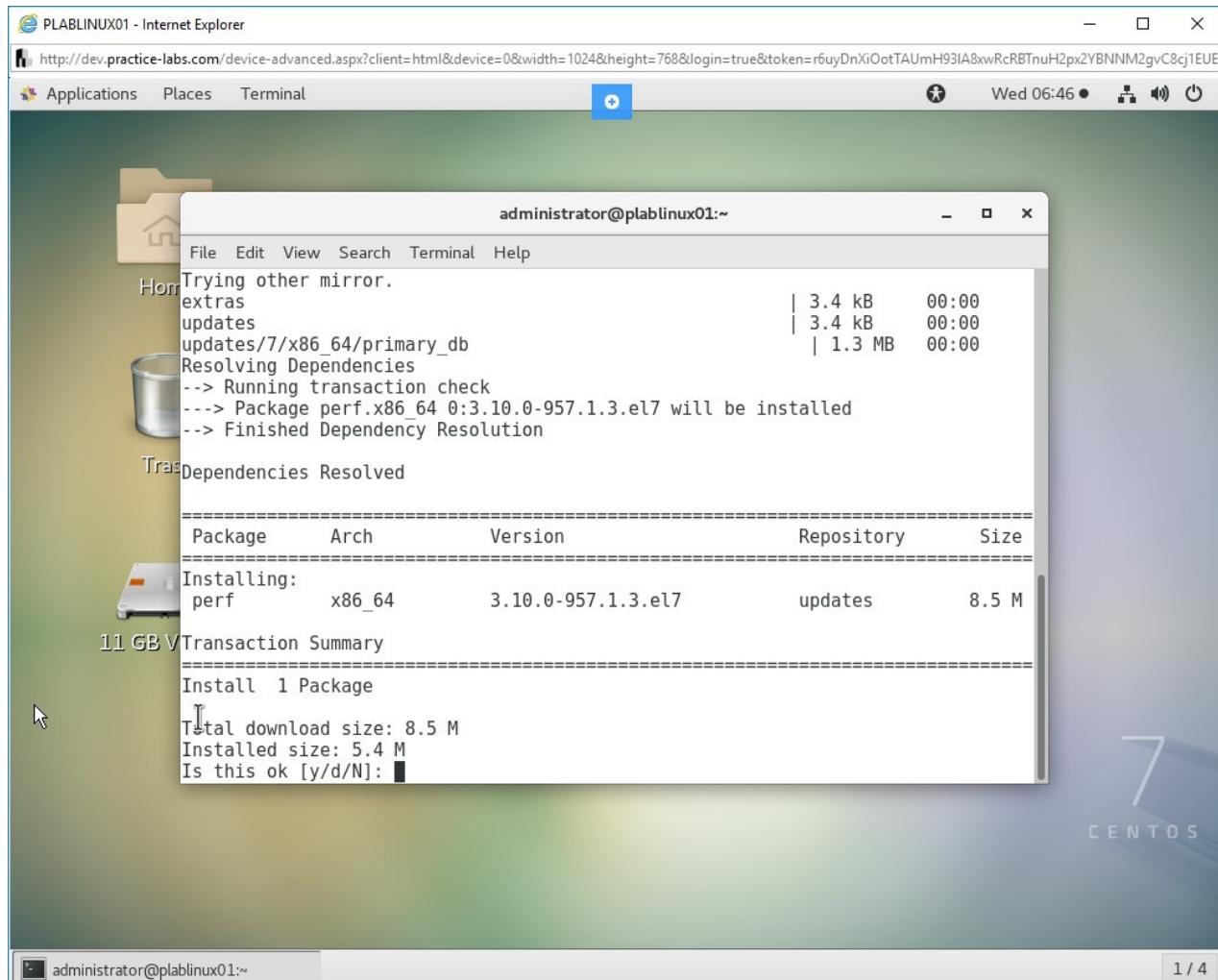


Figure 1.18 Screenshot of PLABLINUX01: Downloading and installing perf.

## Step 2

To continue with the installation, type the following

y

Press **Enter**.

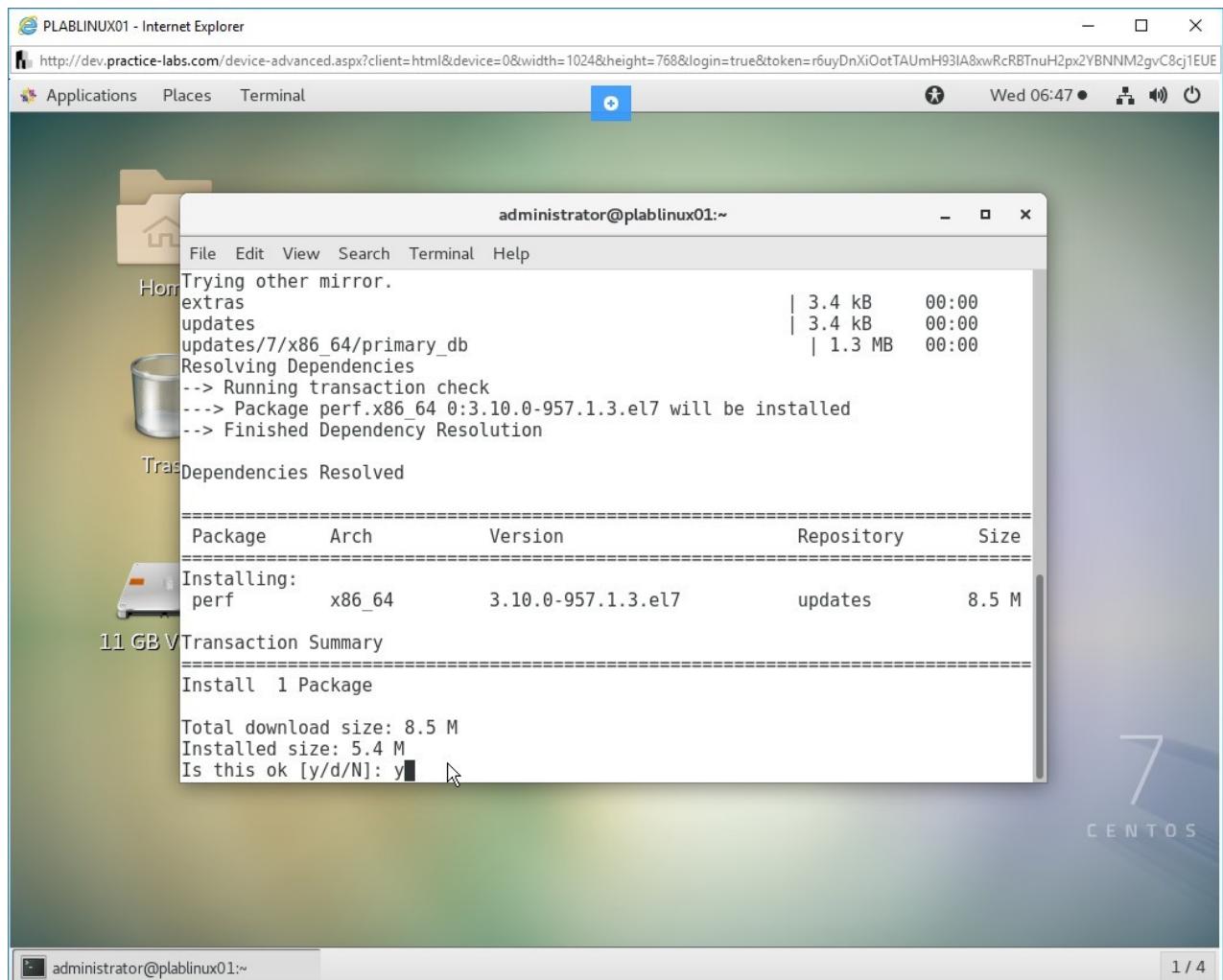


Figure 1.19 Screenshot of PLABLINUX01: Confirming the installation of perf.

## Step 3

When the installation is completed, you are prompted with the **Complete!** message.

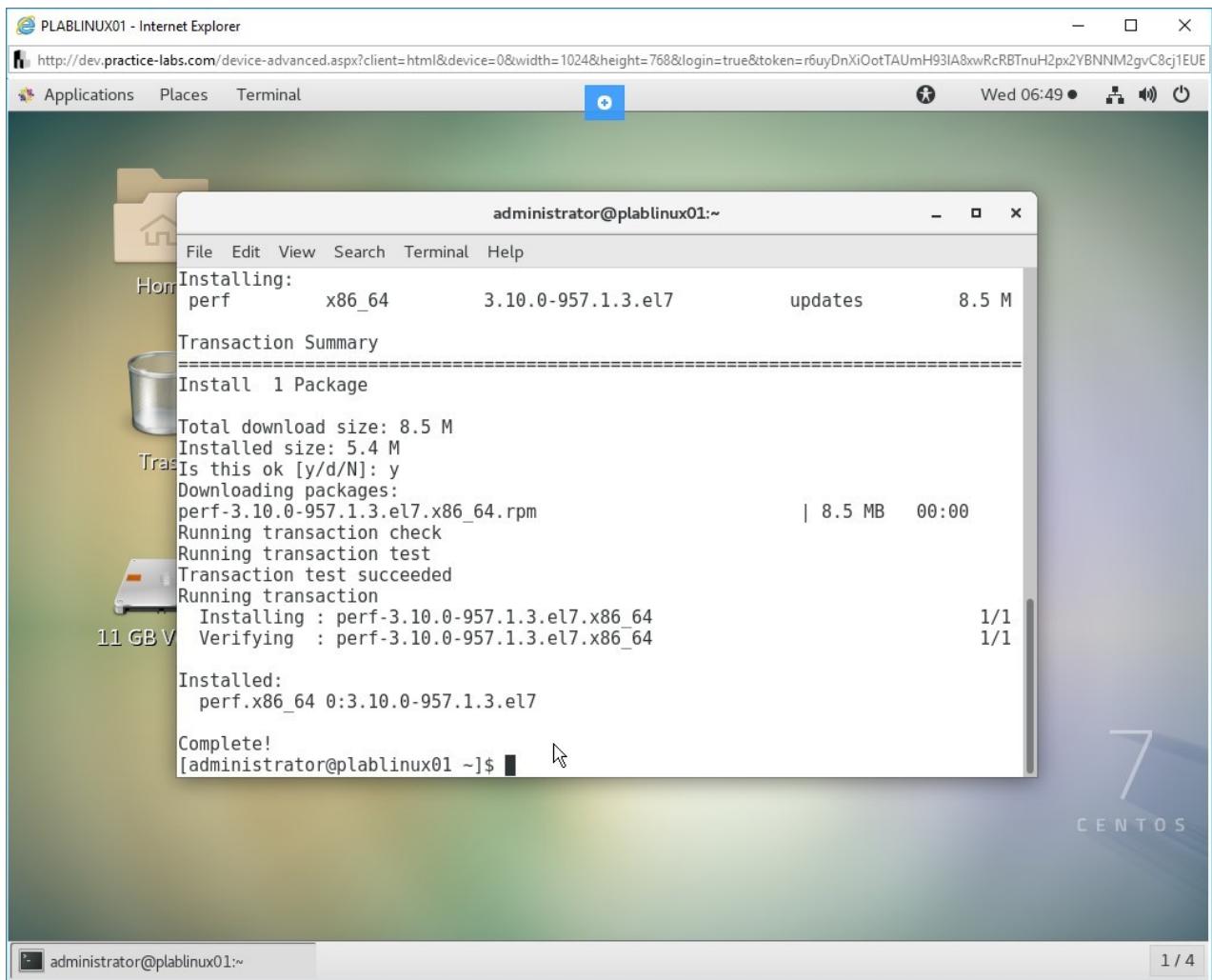


Figure 1.20 Screenshot of PLABLINUX01: Showing the installation completion message.

## Step 4

Using the **perf** command, you can measure the CPU performance of specific commands. For example, you can profile CPU while **Documents/** directory is being copied to the **PLAB** directory.

Type the following command:

```
perf stat cp -r Documents/ PLAB
```

Press **Enter**. Notice that the CPU profiling is done only for the **cp** command and results are generated.

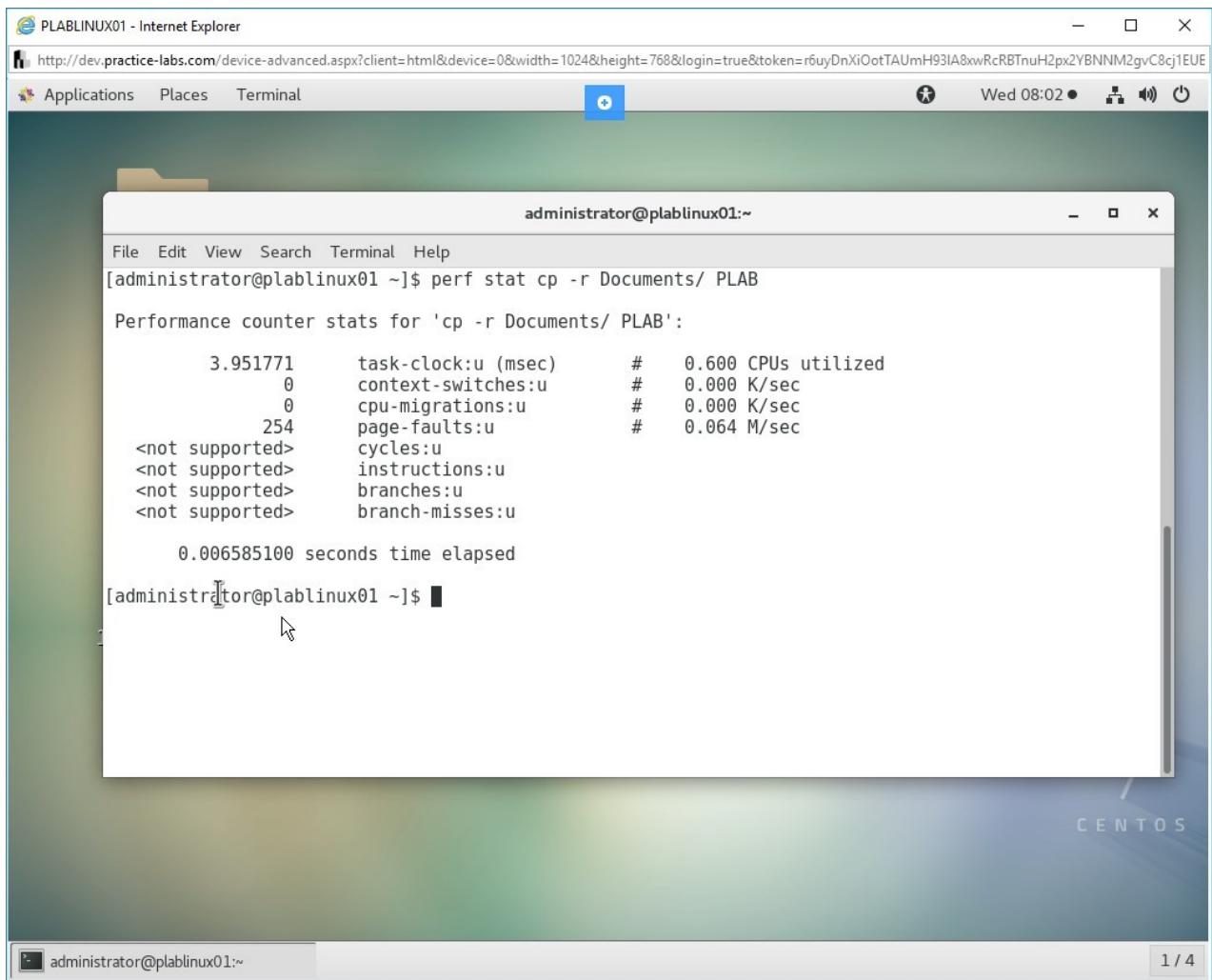


Figure 1.21 Screenshot of PLABLINUX01: Measuring the CPU performance of a specific command

## Step 5

Clear the screen by entering the following command:

```
clear
```

Using the **perf** command, you can also measure the CPU performance for a specific time period. For example, you can profile CPU for 10 seconds only. Type the following command:

```
sudo perf stat -a -- sleep 10
```

Press **Enter**. Notice that **sudo** command is used. For collecting system-wide profiling information, you will need to use administrative privileges.

If prompted, type the following password:

**Passw0rd**

Press **Enter**. Notice that the perf command does not display any information instantly.

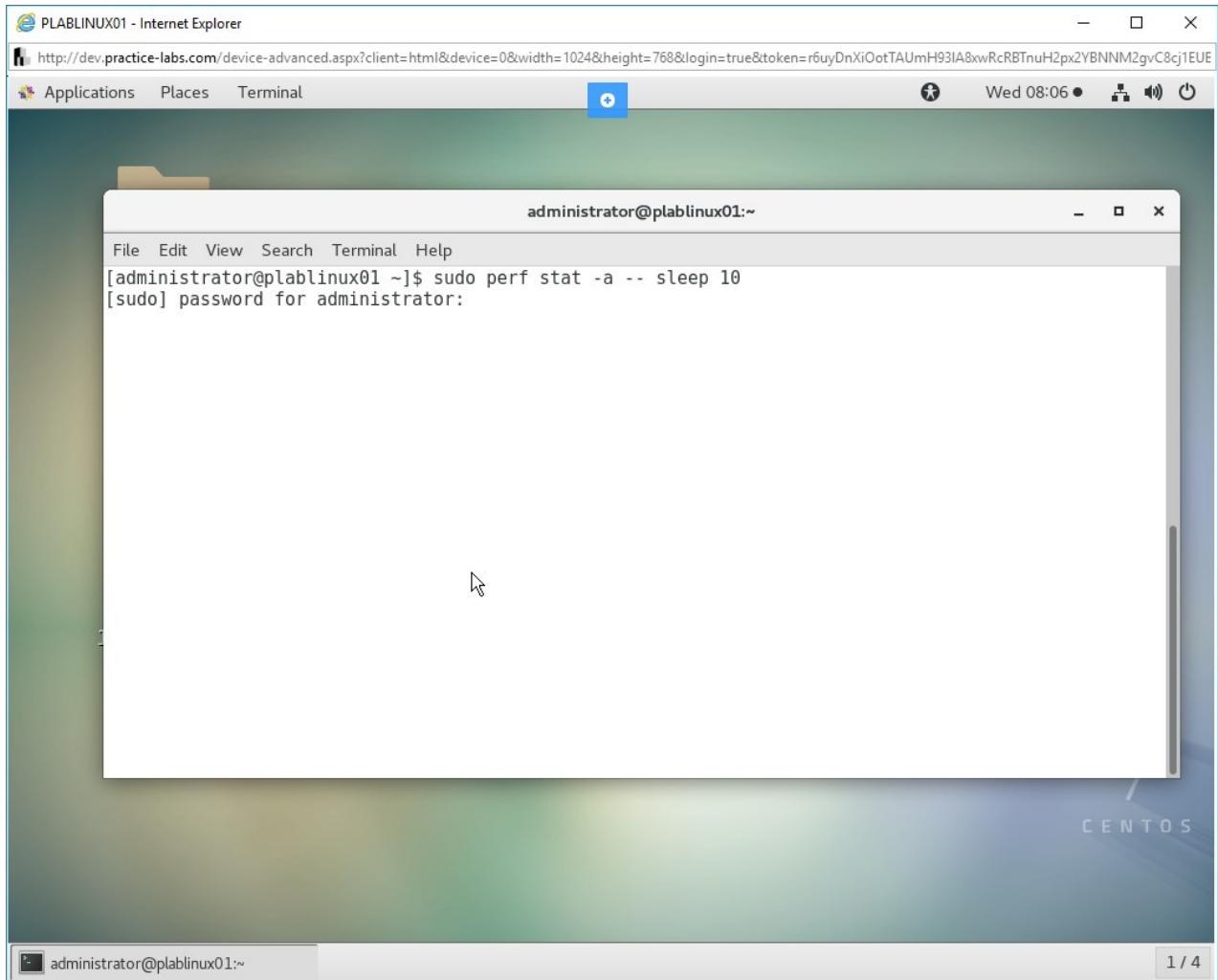


Figure 1.22 Screenshot of PLABLINUX01: Measuring the CPU performance for a specific time period using the perf command.

## Step 6

After **10** seconds, the profiling information is displayed.

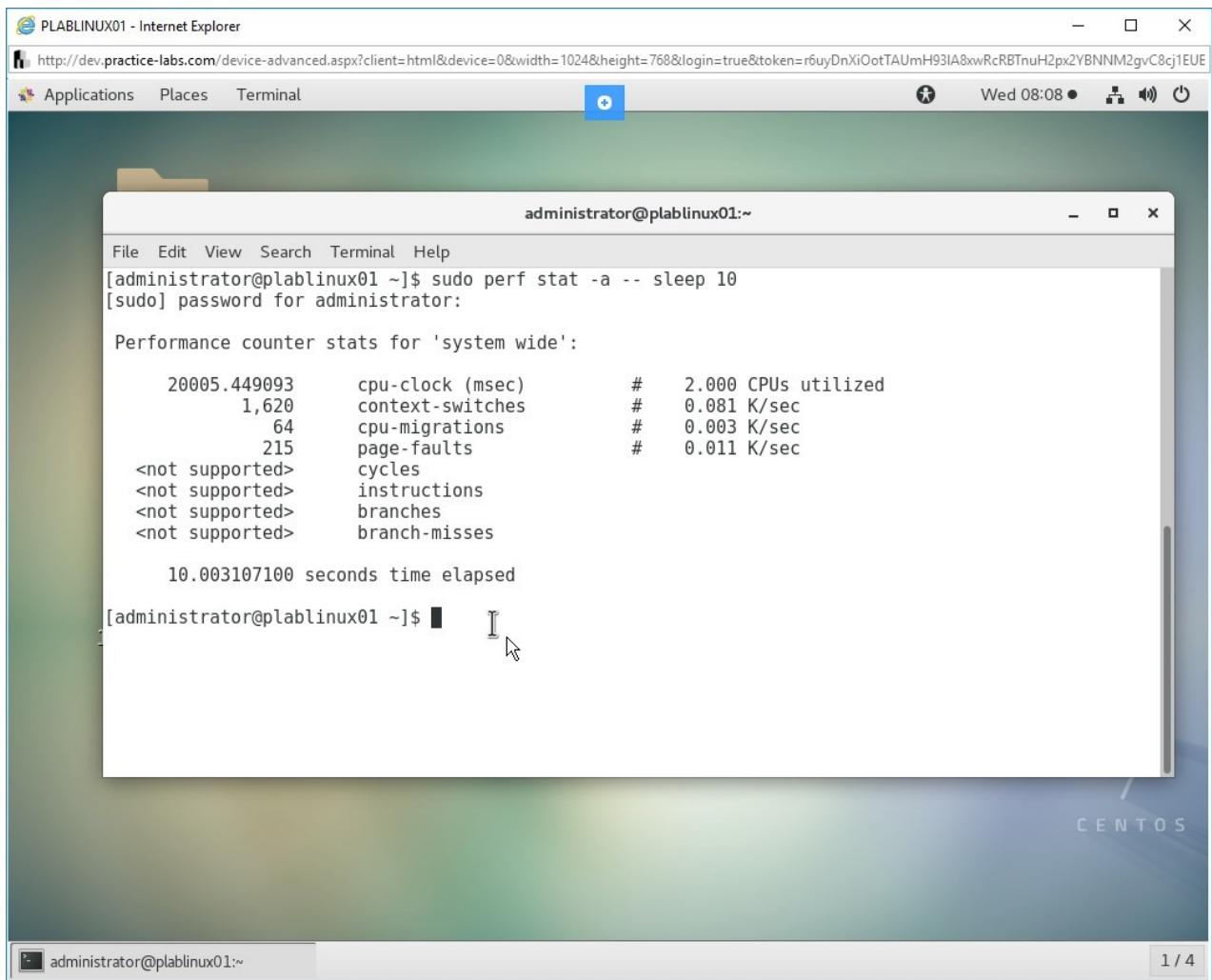


Figure 1.23 Screenshot of PLABLINUX01: Selecting the Open Terminal option from the context menu.

## Step 7

Clear the screen by entering the following command:

```
clear
```

Using the **perf** command, you can also measure the CPU performance for a specific process. Type the following command:

```
perf stat -p 2194
```

Press **Enter**.

Notice that the perf command does not display any information instantly. You will need to break the command with Ctrl + C to display the results.

**Note:** In your lab environment, the process ID is likely to differ. In order to find your PID, run the “top” command from Task 2, Step 2 of this lab, and write down the PID for the administrator account.

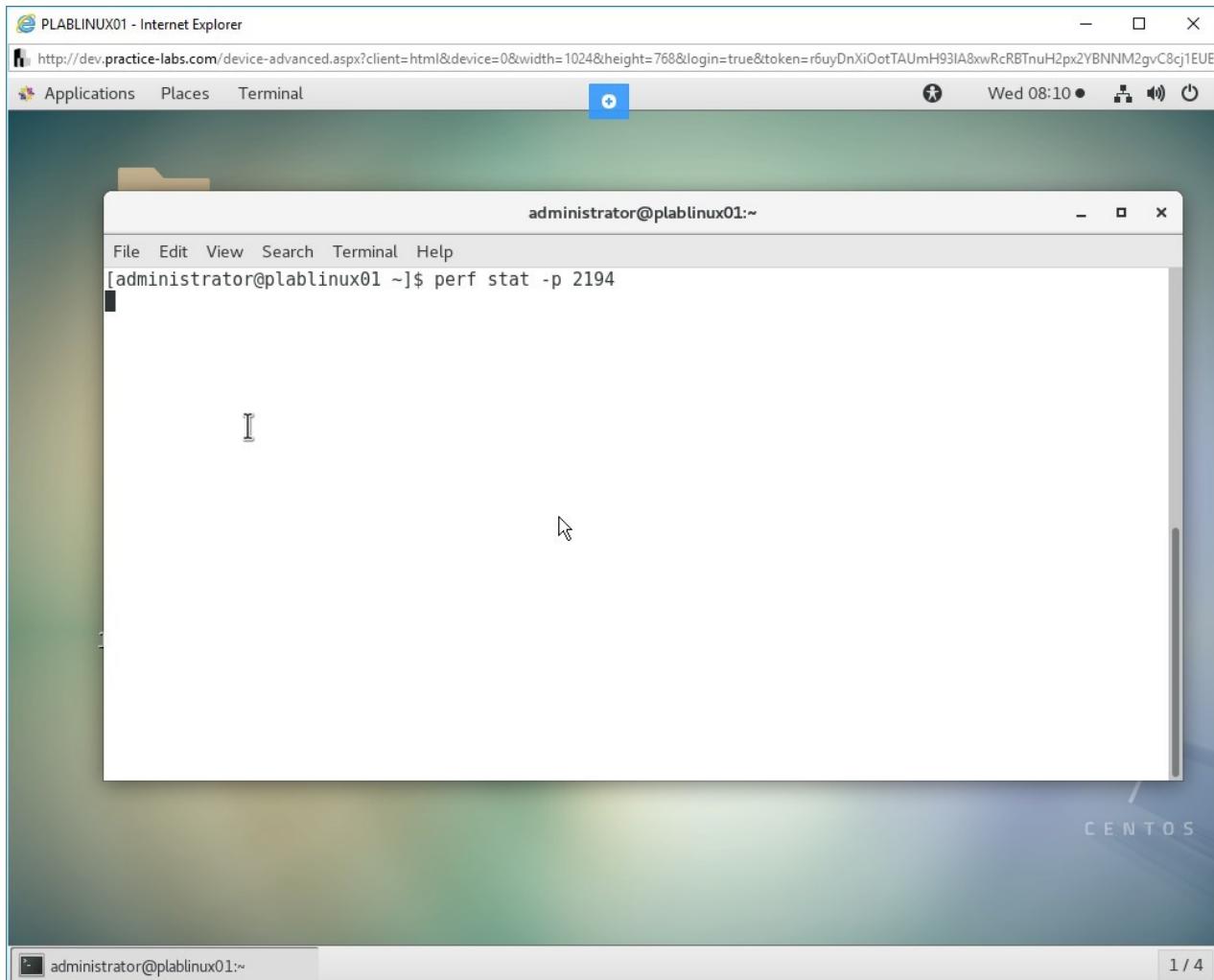


Figure 1.24 Screenshot of PLABLINUX01: Measuring the CPU performance for a specific process using the perf command.

## Step 8

After breaking the command, the profiling information is displayed.

The screenshot shows a terminal window titled "administrator@plablinux01:~". The window displays the output of the command "perf stat -p 2194". The output includes performance counter statistics for process ID 2194, showing utilization percentages for various metrics like task-clock, context-switches, and page-faults. It also shows the total time elapsed as 189.509 seconds.

```
File Edit View Search Terminal Help
[administrator@plablinux01 ~]$ perf stat -p 2194
^C
Performance counter stats for process id '2194':
      587.451557      task-clock:u (msec)      #  0.003 CPUs utilized
          0      context-switches:u           #  0.000 K/sec          (61.27%)
          0      cpu-migrations:u           #  0.000 K/sec          (61.29%)
         10      page-faults:u            #  0.017 K/sec          (61.31%)
<not supported>    cycles:u
<not supported>   instructions:u
<not supported>    branches:u
<not supported> branch-misses:u

 189.509767400 seconds time elapsed

[administrator@plablinux01 ~]$
```

Figure 1.25 Screenshot of PLABLINUX01: Displaying the CPU profiling information.

## Step 9

Clear the screen by entering the following command:

```
clear
```

Using the **perf** command, you can also measure the CPU performance for a specific time period and record it. Type the following command:

```
sudo perf record -e cycles:u -a -- sleep 10
```

Press **Enter**. Notice that sudo command is used. For collecting system-wide profiling information, you will need to use administrative privileges.

When prompted, type the following password:

**Passw0rd**

Press **Enter**. Notice that the perf command does not display any information instantly.

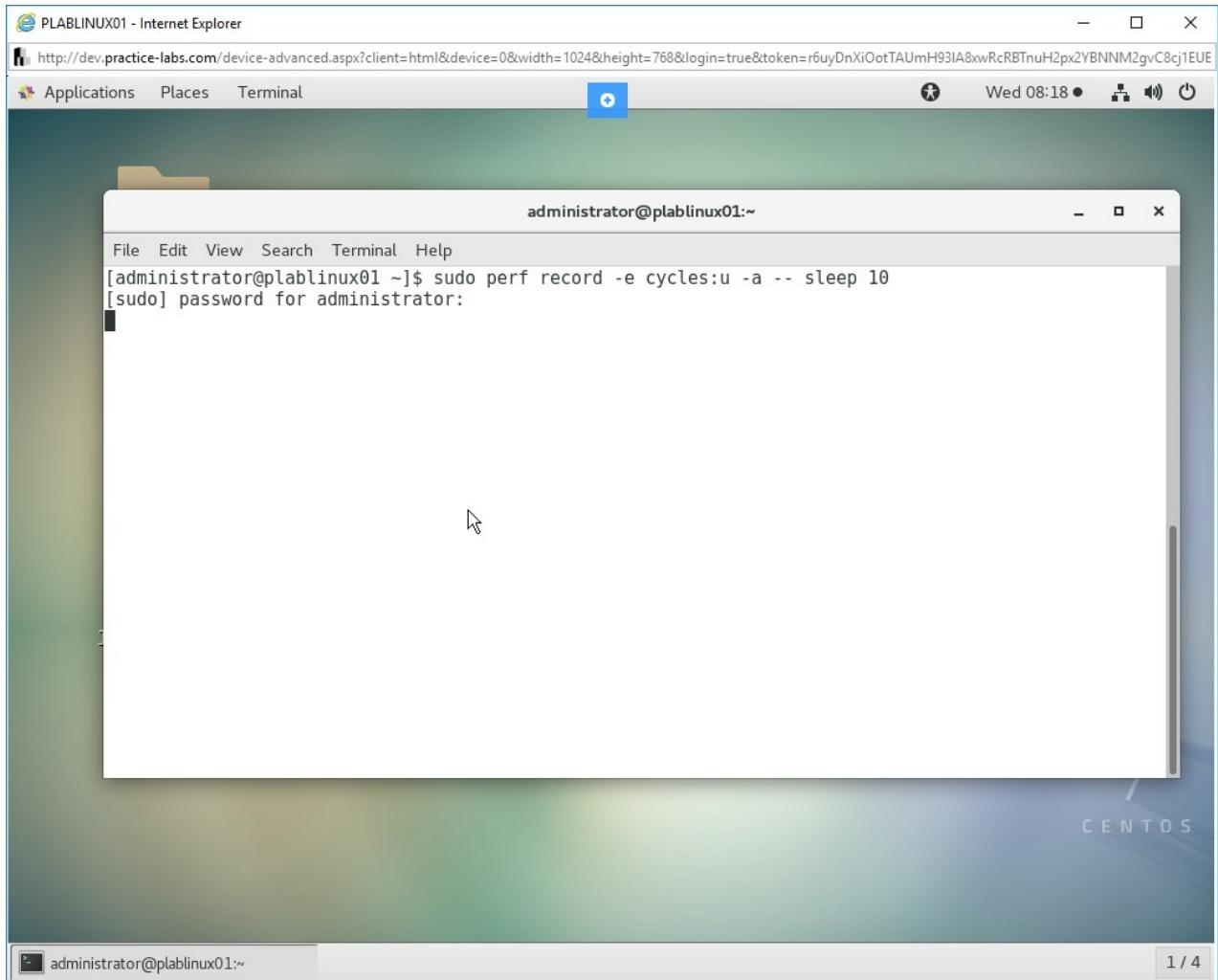


Figure 1.26 Screenshot of PLABLINUX01: Measuring and recording the CPU performance.

## Step 10

After **10** seconds, the profiling information is recorded.

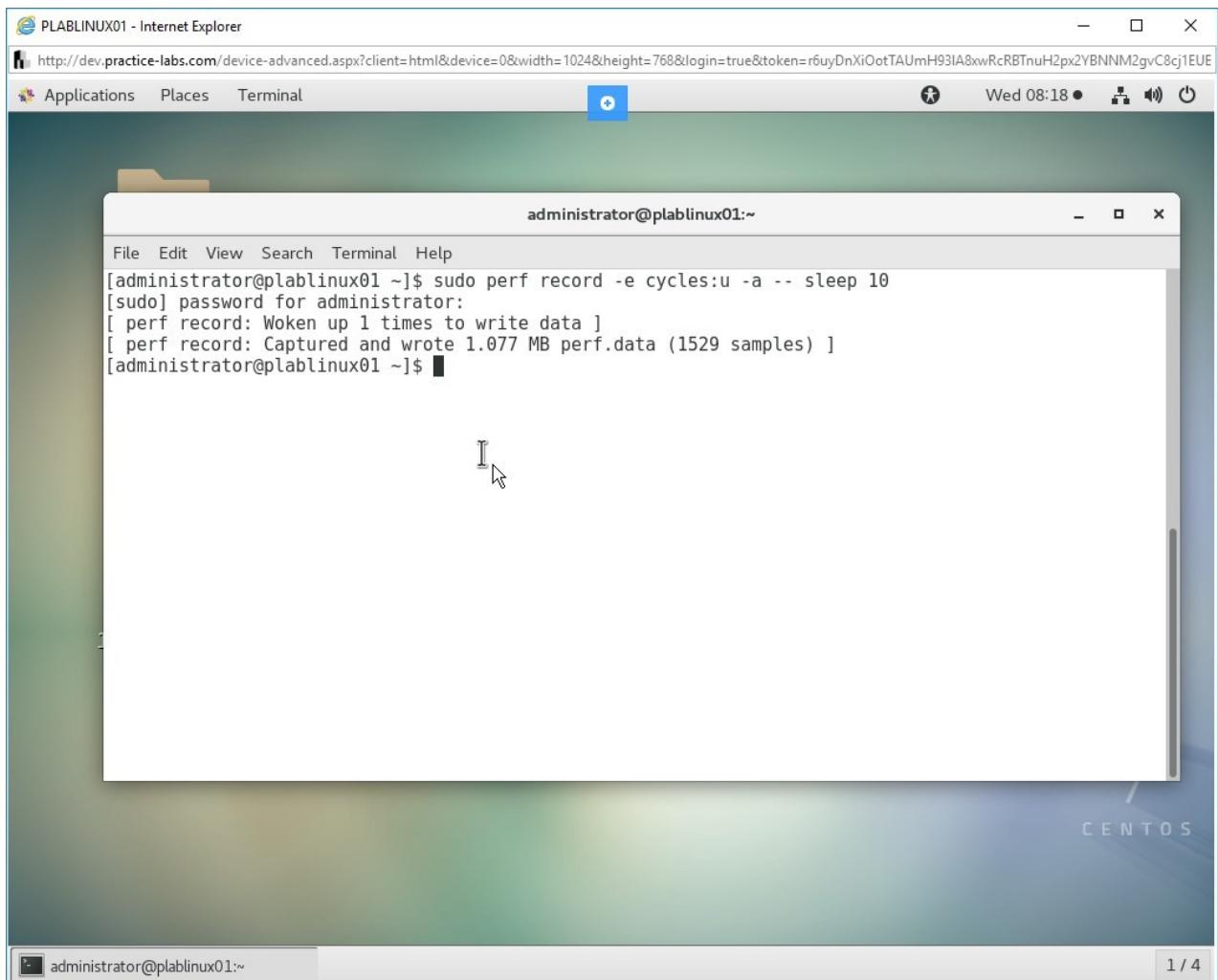


Figure 1.27 Screenshot of PLABLINUX01: Recording the profiling information.

## Step 11

You will now use the perf report command to view the recorded profiling information. Type the following command:

```
sudo perf report
```

Press **Enter**. Notice that sudo command is used.

Notice that the recorded information is displayed.

**Note:** To break the command, press *Ctrl + C*.

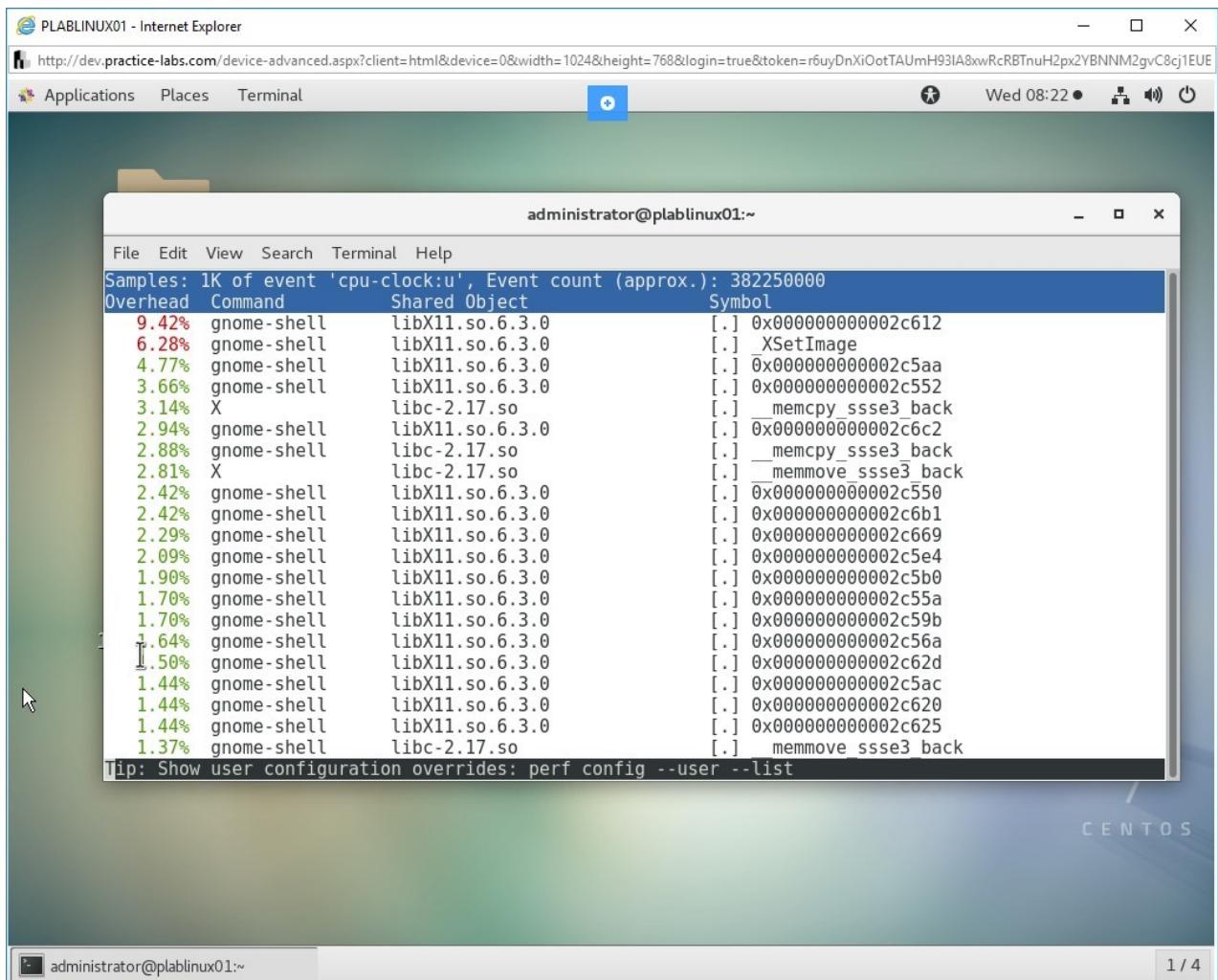


Figure 1.28 Screenshot of PLABLINUX01: Reading through recorded information

## Step 12

Clear the screen by entering the following command:

```
clear
```

You will now also use the **perf top** command to display all cpu-clock related events. Type the following command:

```
sudo perf top -e cpu-clock
```

Press **Enter**. Notice that sudo command is used.

Notice that the recorded information is displayed.

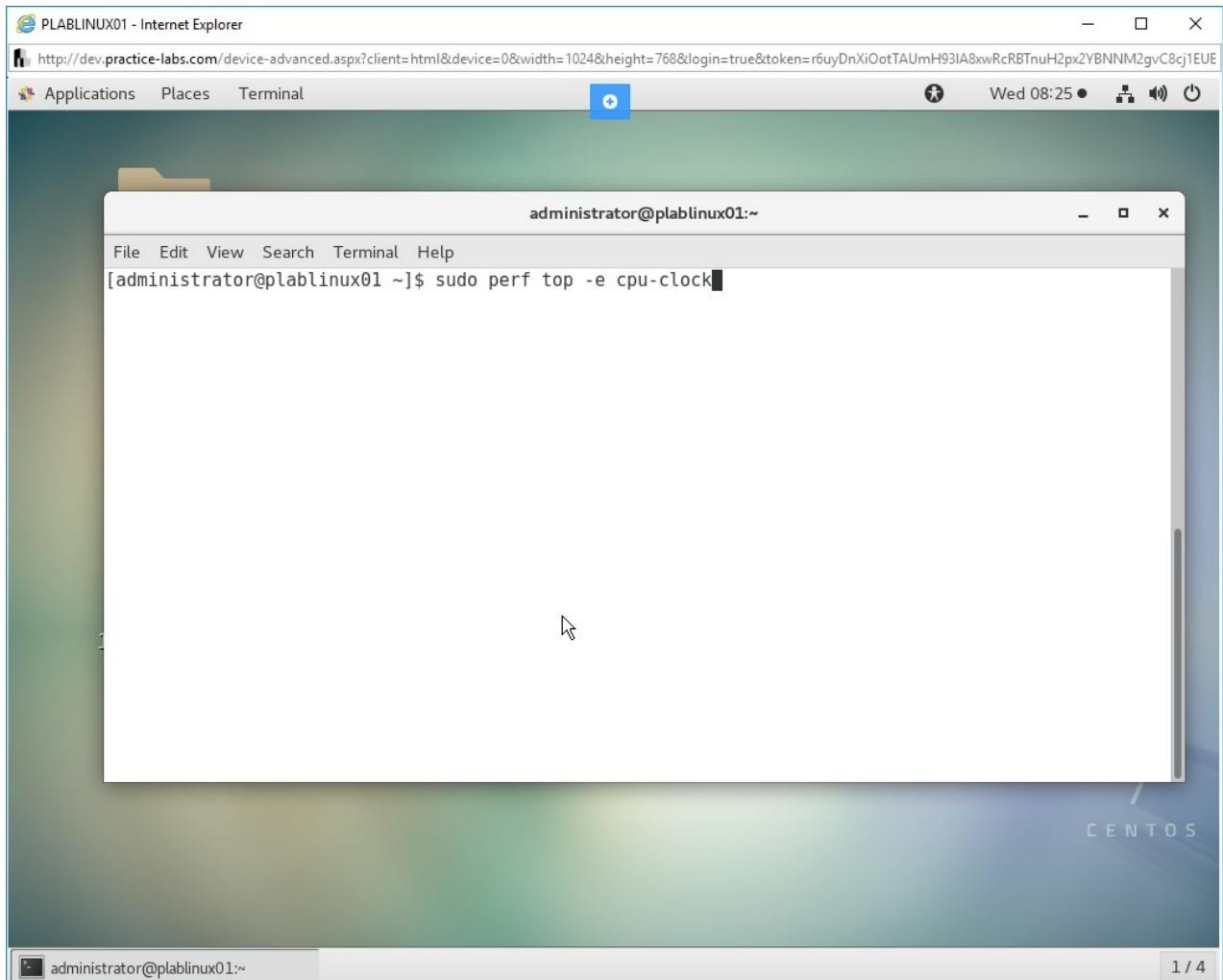


Figure 1.29 Screenshot of PLABLINUX01: Executing the perf top command.

## Step 13

Notice that the output is displayed. This is real-time information that is being displayed.

The screenshot shows a terminal window titled "administrator@plablinux01:~". The window displays the output of the perf top command. The title bar includes the URL "http://dev.practice-labs.com/device-advanced.aspx?client=html&device=0&width=1024&height=768&login=true&token=r6uyDnXiOotTAUmH93IA8xwRcRBtNuH2px2YBNNM2gvC8cj1EUE". The terminal window has a blue header bar with the text "Samples: 2K of event 'cpu-clock', Event count (approx.): 441591720". Below this, a table lists CPU usage by shared object. The table has columns for Overhead, Shared Object, and Symbol. The data shows that libX11.so.6.3.0 is the most active shared object, followed by libc-2.17.so and [kernel]. The terminal also shows a message at the bottom: "Failed to open /tmp/perf-2194.map, continuing without symbols".

Figure 1.30 Screenshot of PLABLINUX01: Displaying the perf top command information.

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

## Review

Well done, you have completed the **Perform CPU Monitoring and Configuration** Practice Lab.

## Summary

You completed the following exercise:

- Exercise 1 - Perform CPU Monitoring

You should now be able to:

- Use various commands to monitor a CPU
- Install and use perf

## Feedback

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