

Linux+ Support Skills

GNU and Unix Commands

- **Introduction**
 - **Work on the Command Line**
 - **Create, Monitor and Kill Processes**
 - **Use Streams Pipes and Redirects**
 - **Modify Process Execution Priorities**
 - **Summary**
-

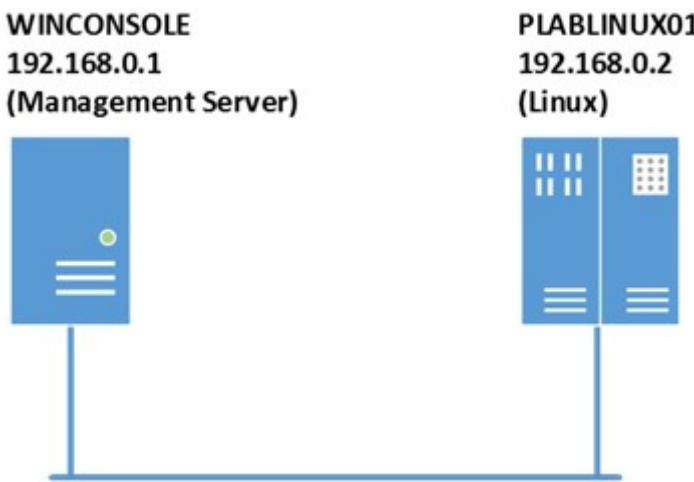
Introduction

The **GNU and Unix Commands** module provides you with the instruction and Server hardware to develop your hands on skills in the defined topics. This module includes the following exercises:

- 1) Work on the Command Line
- 2) Create, Monitor and Kill Processes
- 3) Use Streams Pipes and Redirects
- 4) Modify Process Execution Priorities

Lab Diagram

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



Connecting to your lab

In this module you will be working on the following equipment to carry out the steps defined in each exercise.

- WINCONSOLE (Management Server)
- PLABLINUX01 (Linux Server)

Each exercise will detail which console you are required to work on to carry out the steps.

To start simply click on the named Server from the device list (located on the left hand side of the screen) and click the **Power on** from the in tools bar. In some cases the devices may power on automatically.

During the boot up process an activity indicator will be displayed in the name tab:

- Black - Powered Off
- Blue - Working on your request
- Green - Ready to access

If the remote console is not displayed automatically in the main window (or popup) click the **Connect** icon located in the tools bar to start your session.

If the remote console does not appear please try the following option:

- Switch between the HTML 5 and Java client versions in the tools bar.

In the event this does not resolve your connectivity problems please visit our Help / Support pages for additional resolution options.

Copyright Notice

This document and its content is copyright of Practice-IT - © Practice-IT 2013. All rights reserved. Any redistribution or reproduction of part or all of the contents in any form is prohibited other than the following:

- 1) You may print or download to a local hard disk extracts for your personal and non-commercial use only.
- 2) You may copy the content to individual third parties for their personal use, but only if you acknowledge the website as the source of the material. You may not, except with our express written permission, distribute or commercially exploit the content. Nor may you transmit it or store it in any other website or other form of electronic retrieval system.

Exercise 1 - Work on the Command Line

In this exercise, you will understand how to get the command history on the command line. Please refer to your course material or use your favourite search engine to research **command-line history in linux**.

In this exercise, you will configure:

- Use single shell commands and one line command sequences to perform basic tasks on the command
- Use and modify the shell environment including defining, referencing and exporting
- Use and edit command history
- Invoke commands inside and outside the defined path

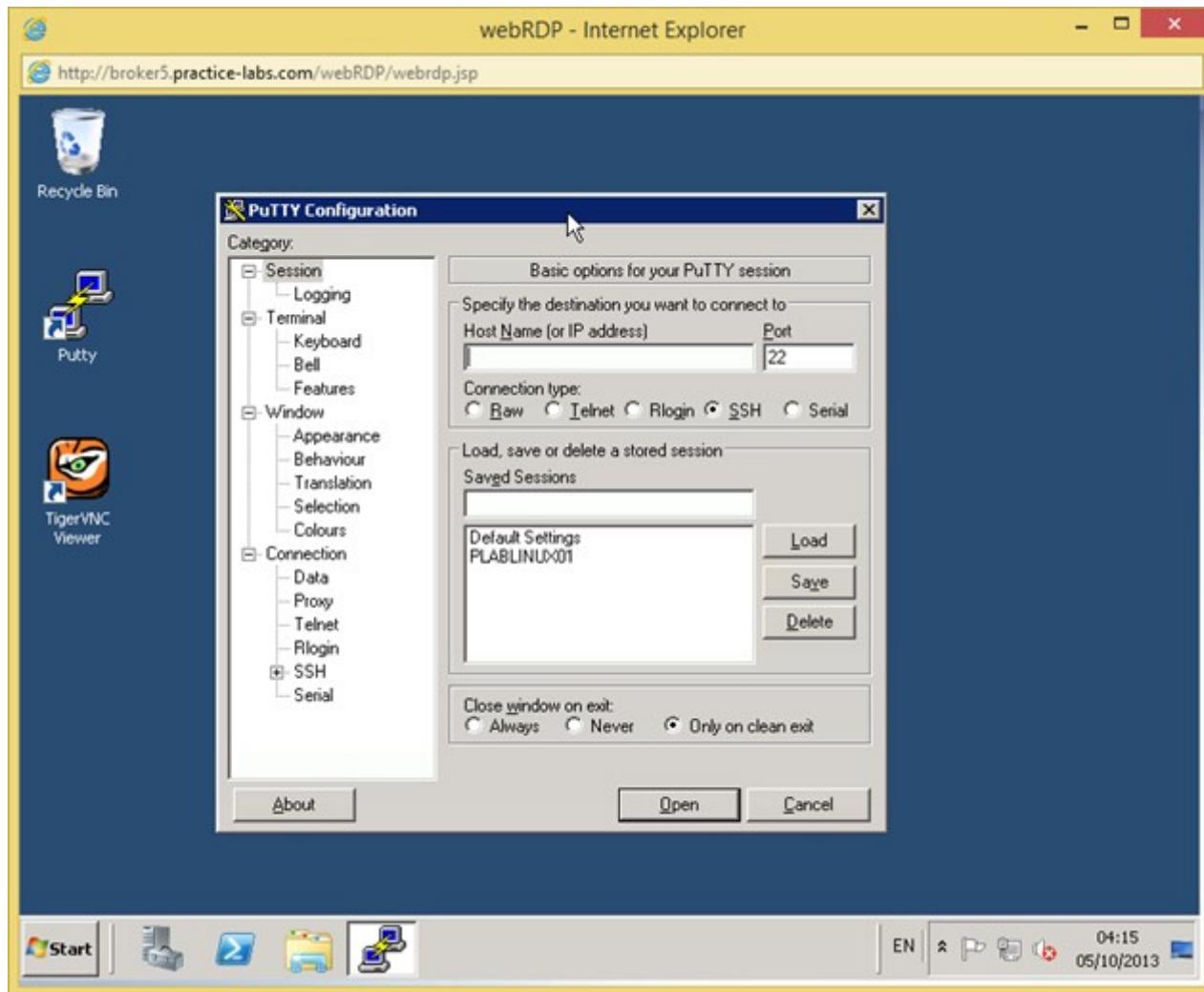
Use Single Shell Commands and One Line Command Sequences to Perform Basic Tasks on the Command Line

Step 1

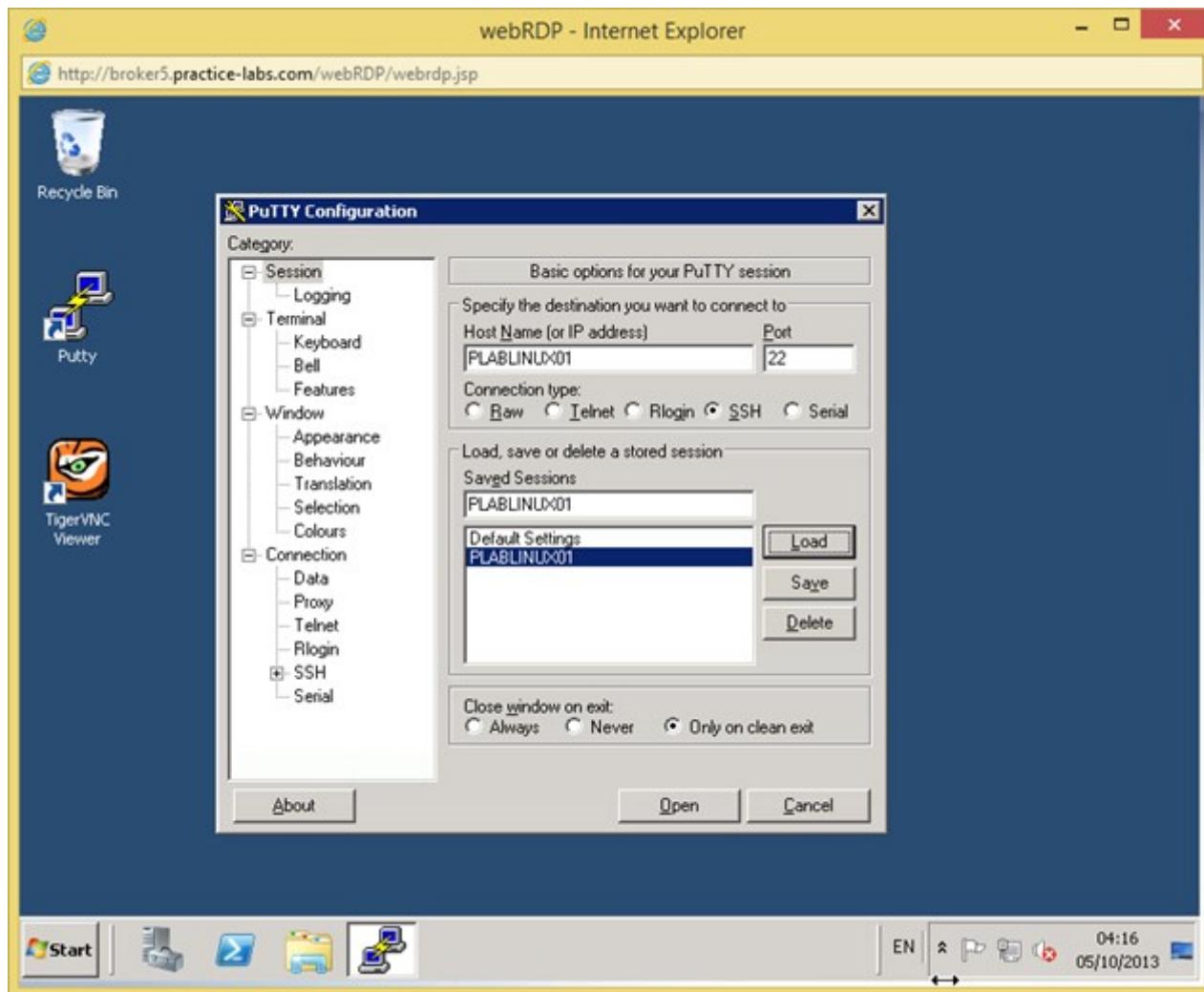
Ensure you have powered on the required devices and click **WINCONSOLE**.

You will be logged into the **Management Server** and should now see the desktop. On the **Desktop**, double-click **PuTTY** to start it.

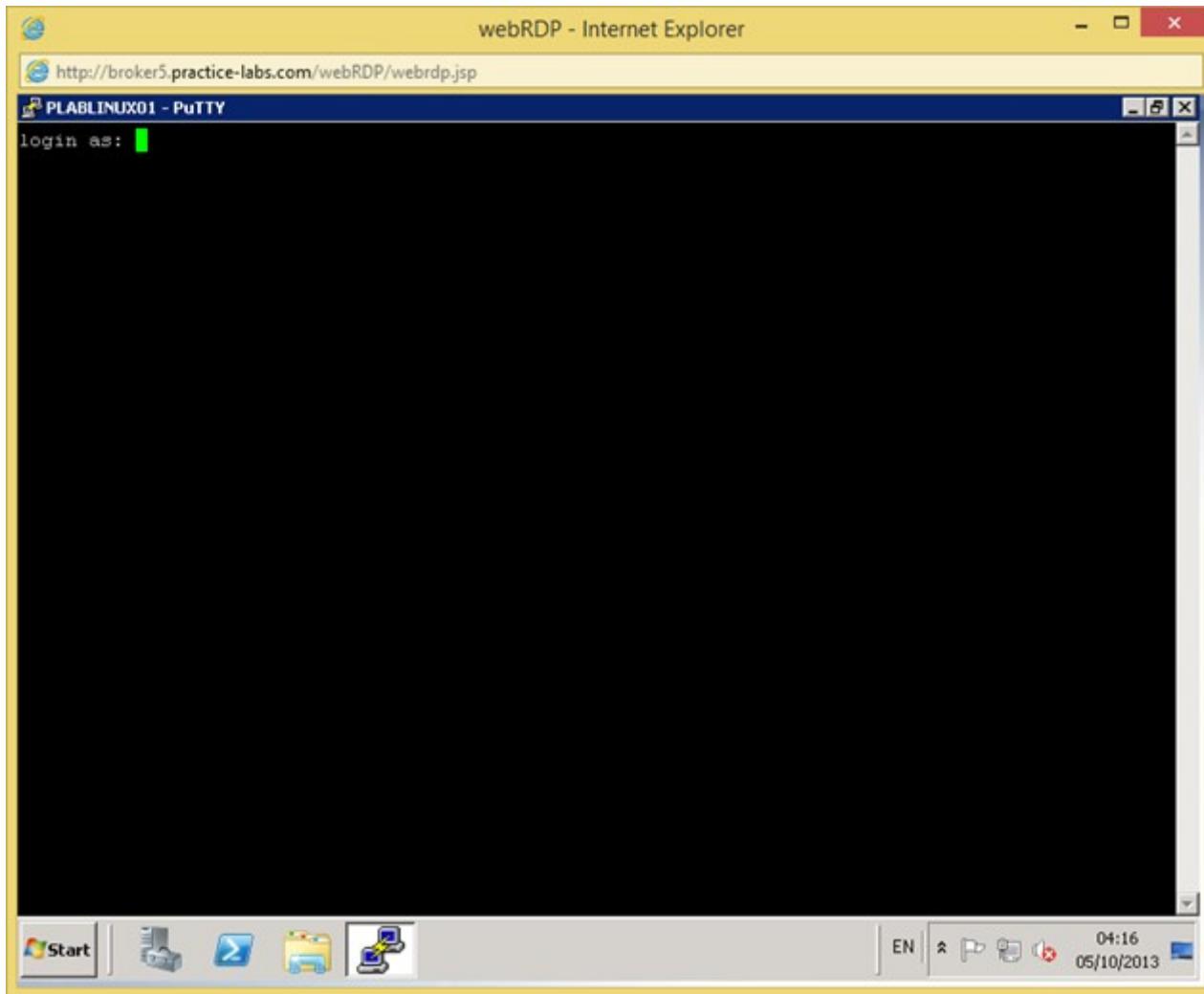
The **PuTTY Configuration** dialog box appears.



In the **PuTTY Configuration** dialog box, select **PLABLINUX01** in the **Load, save or delete a stored session** and click **Load**.

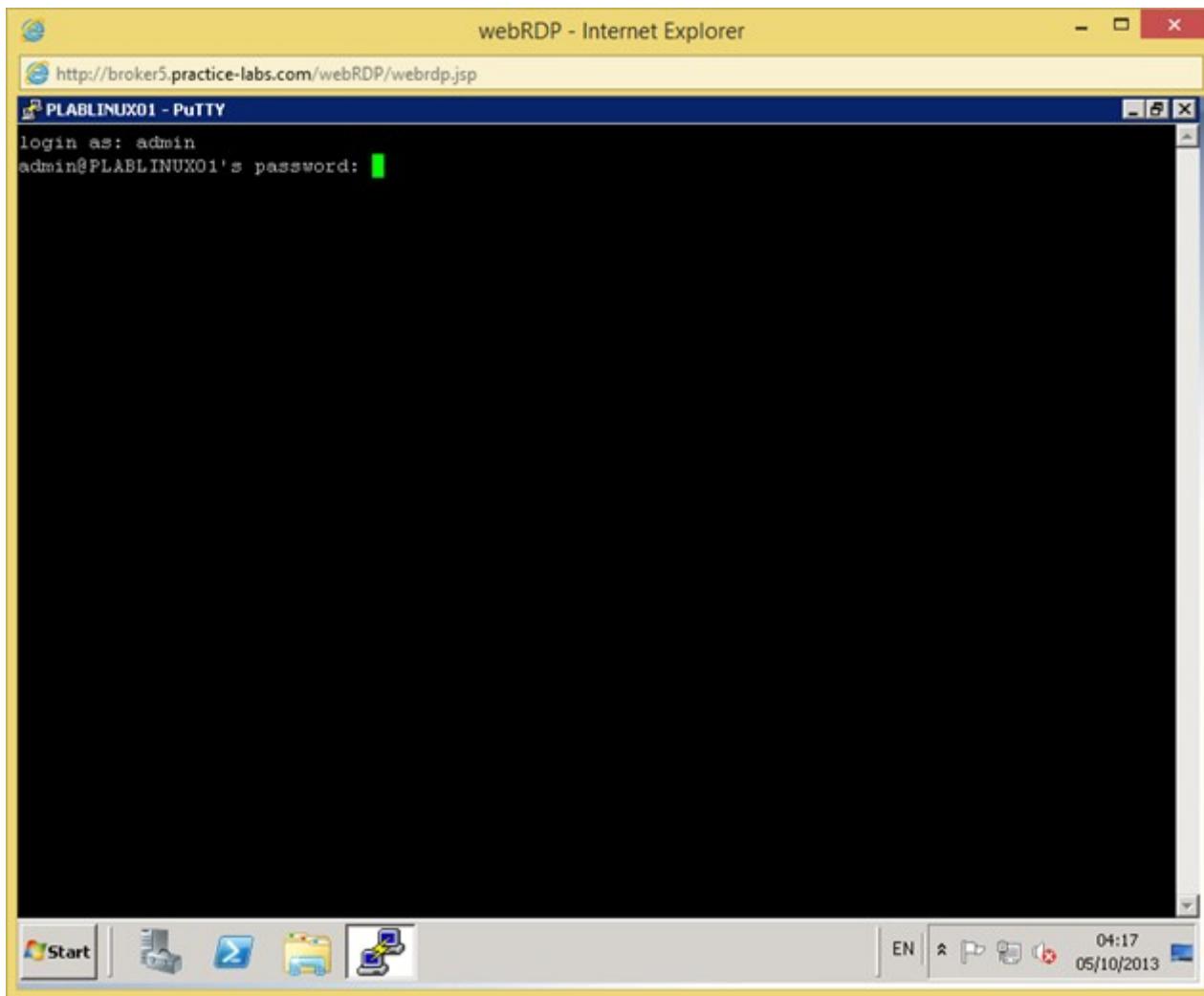


Click **Open**. The **PLABLINUX01 – PuTTY command line** will appear.



Step 2

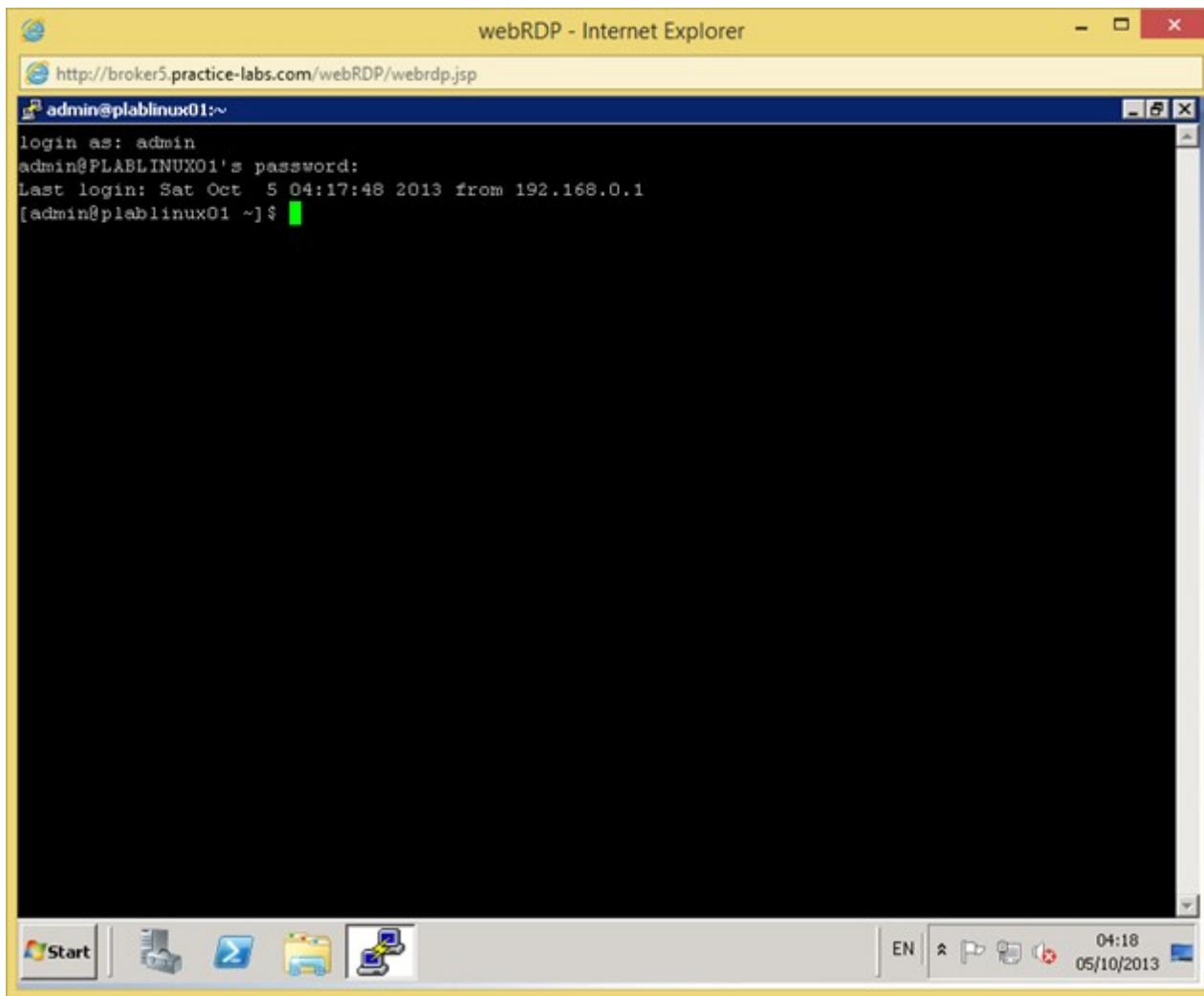
On the **login** screen, enter **admin**



At the **admin@plablinux01**'s password prompt, type password as **Password** and press **Enter**.

Note: Unlike Windows the username is case-sensitive on Linux systems. Also the password field does not show any characters, so input the password above and press enter.

You have successfully logged into the **PLABLINUX01** system.

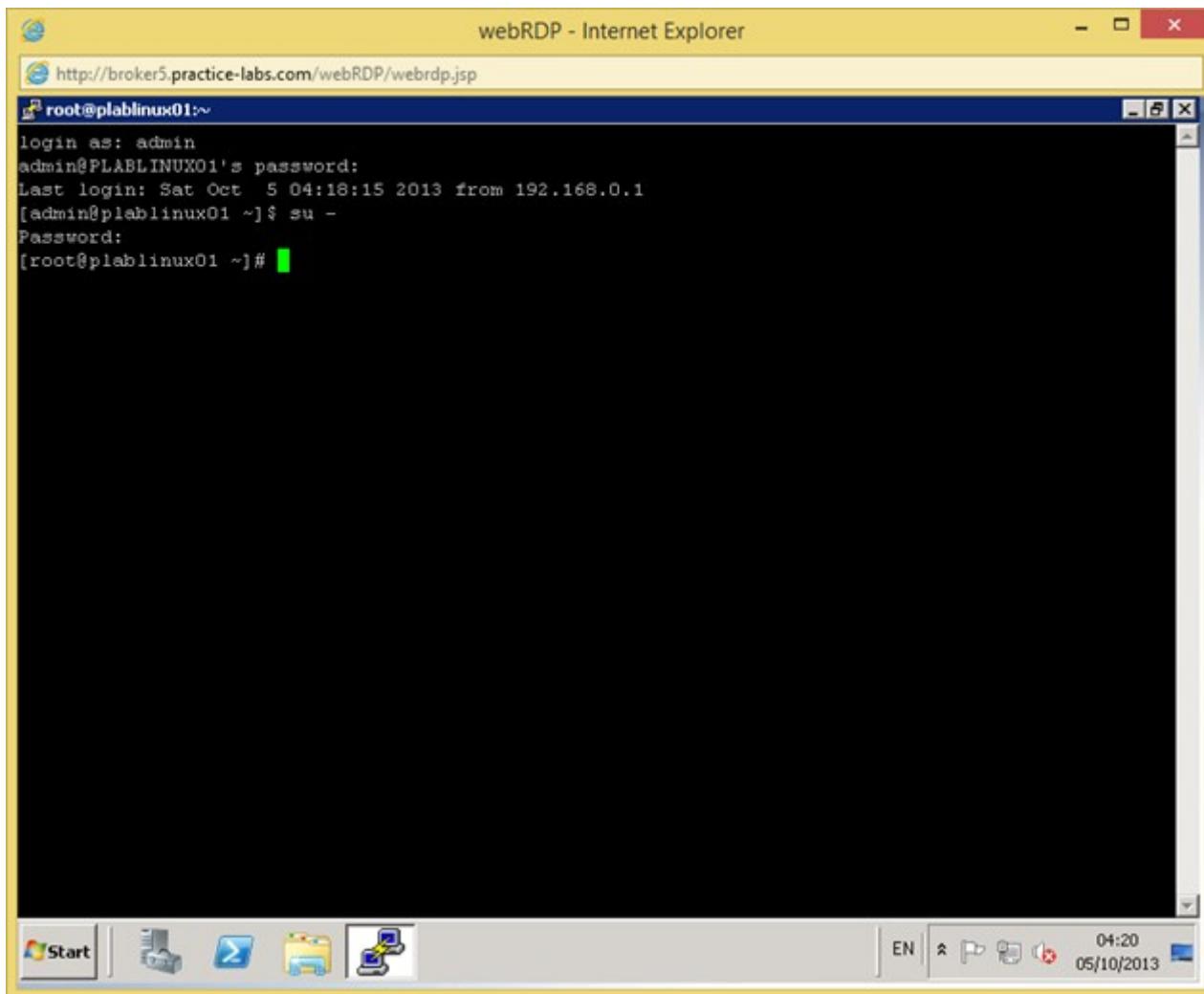


At the command prompt, gain the root privileges by typing the following command:

```
SU -
```

Enter the **Password** at the Password prompt.

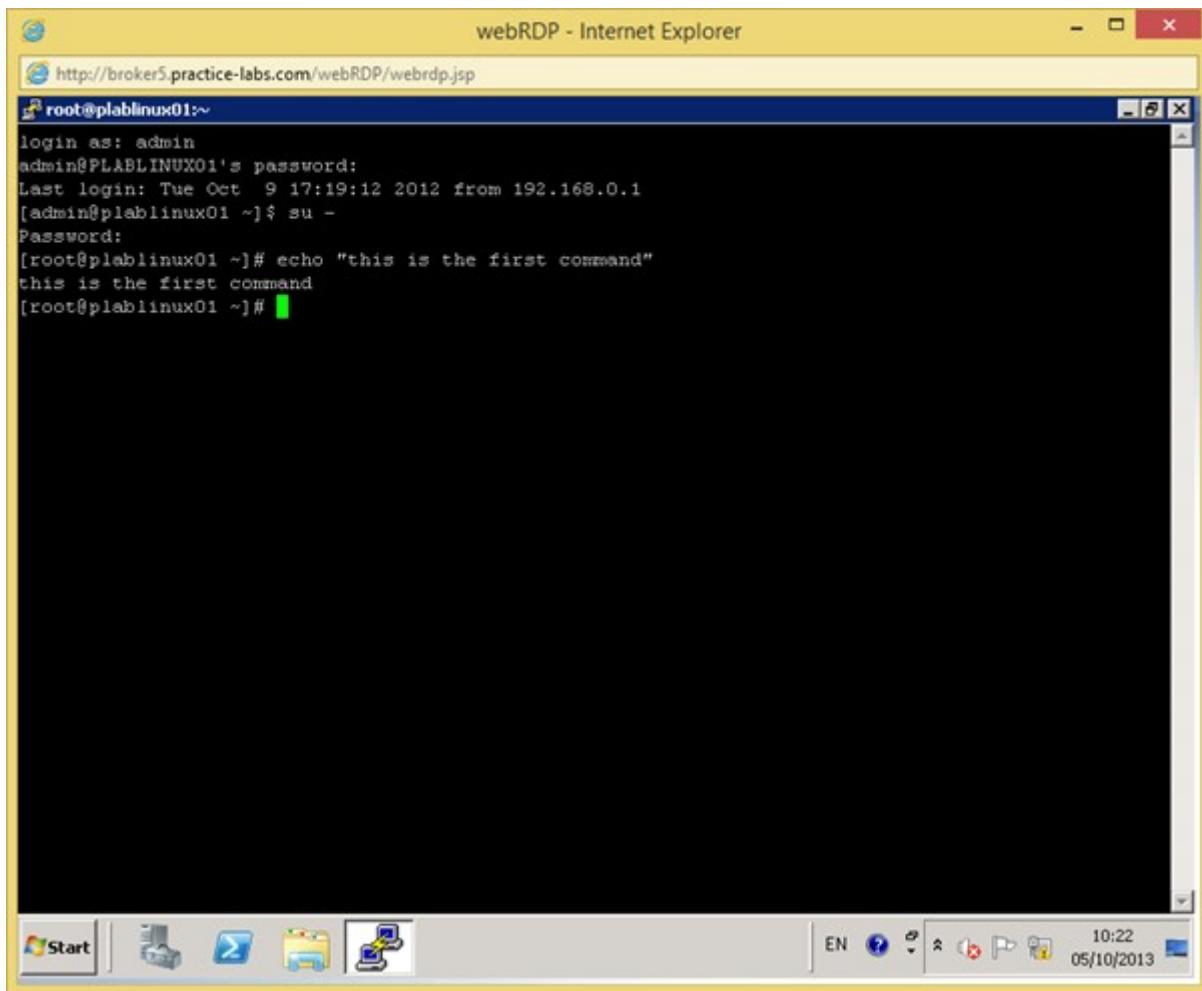
The root command prompt is displayed.



Step 3

Use the echo command to display a message on the screen. To display the echo command, enter the following command:

```
echo "this is the first command"
```



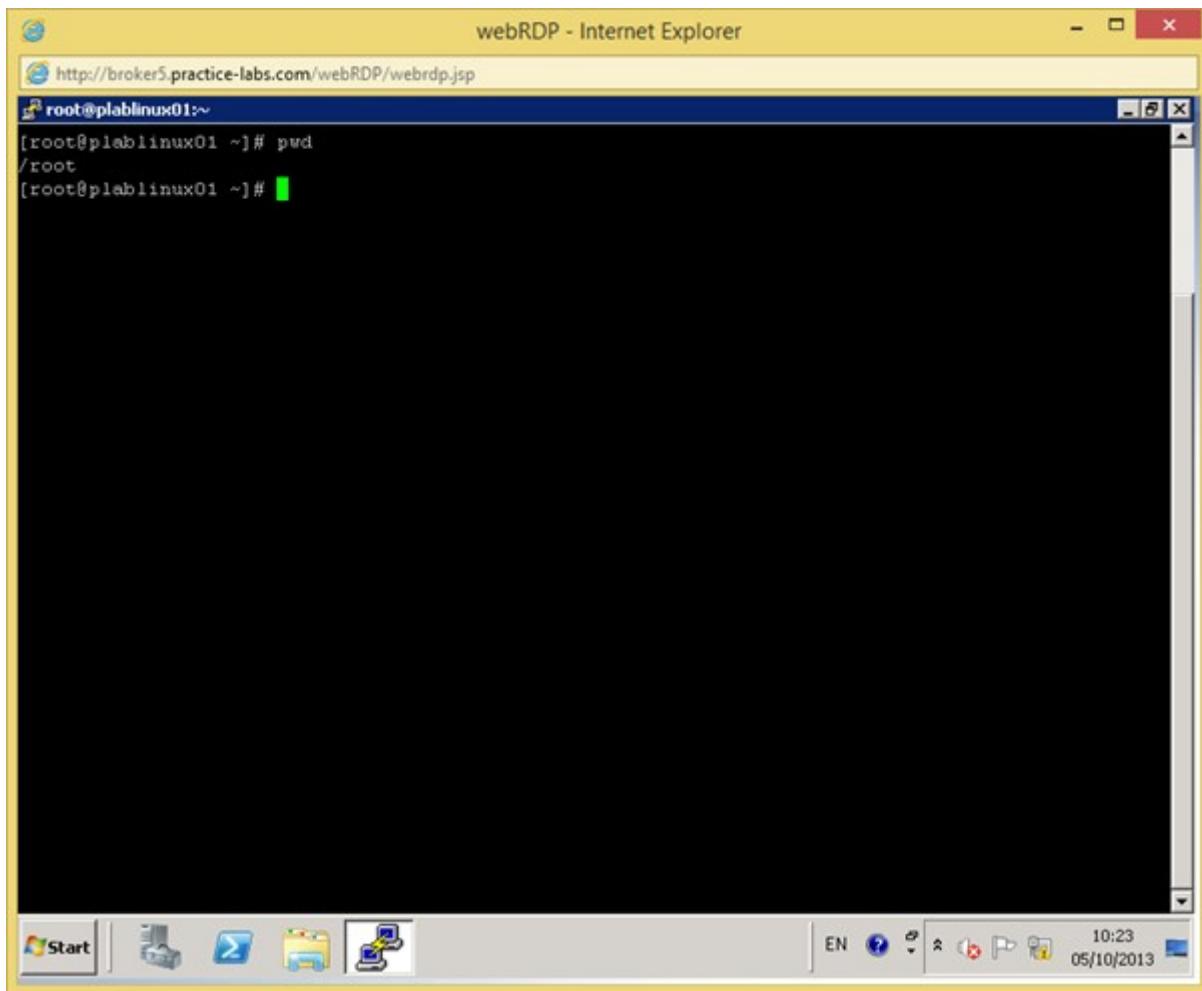
Step 4

Clear the screen using the following command:

```
clear
```

To display the current directory, enter the following command:

```
pwd
```



Step 5

The **exec** command is part of the shell that is used to start other commands. When you use the exec command to start a program, the exec command reoccupies the process that used to start it.

The current shell is replaced, to use exec command, enter the following set of commands:

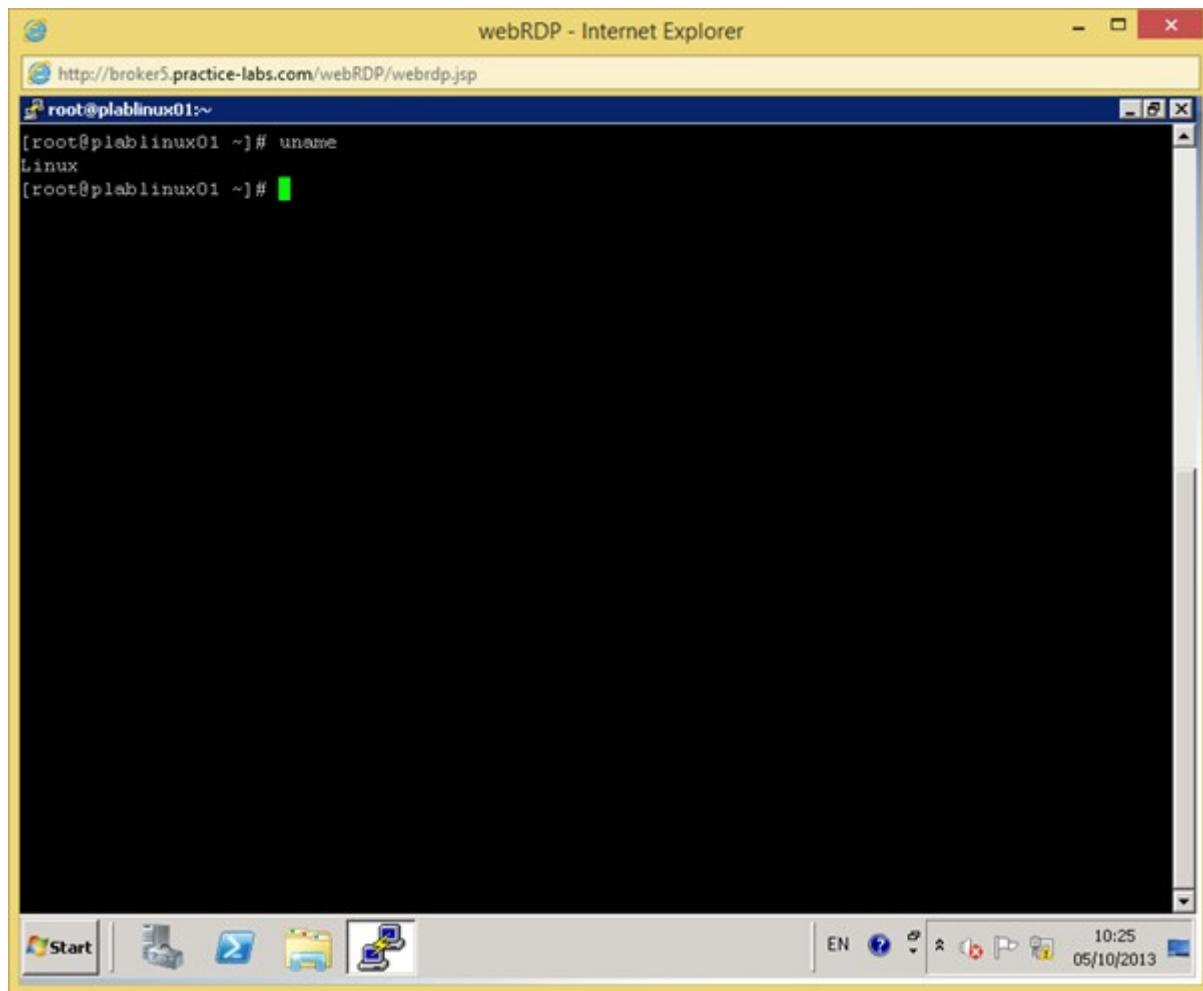
```
echo $$  
bash  
echo $$  
echo hello  
echo $$  
exec echo hello  
echo $$
```

```
[root@plablinux01 ~]# echo $$
1462
[root@plablinux01 ~]# bash
[root@plablinux01 ~]# echo $$
1526
[root@plablinux01 ~]# echo hello
hello
[root@plablinux01 ~]# echo $$
1526
[root@plablinux01 ~]# exec echo hello
hello
[root@plablinux01 ~]# echo $$
1462
[root@plablinux01 ~]#
```

Step 6

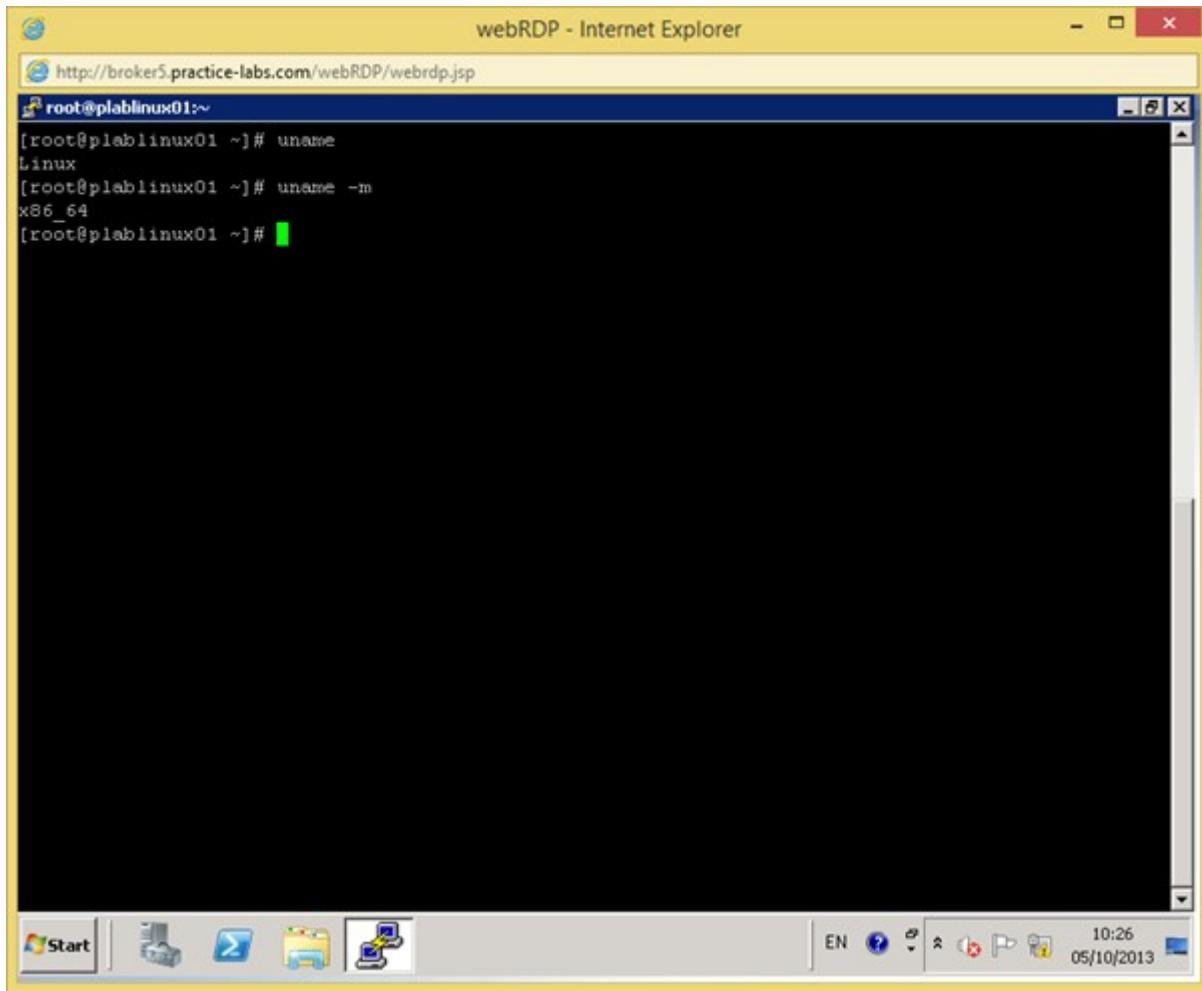
You can use the **uname** command to display different information of your system. To use uname command, enter the following set of commands:

```
uname
```



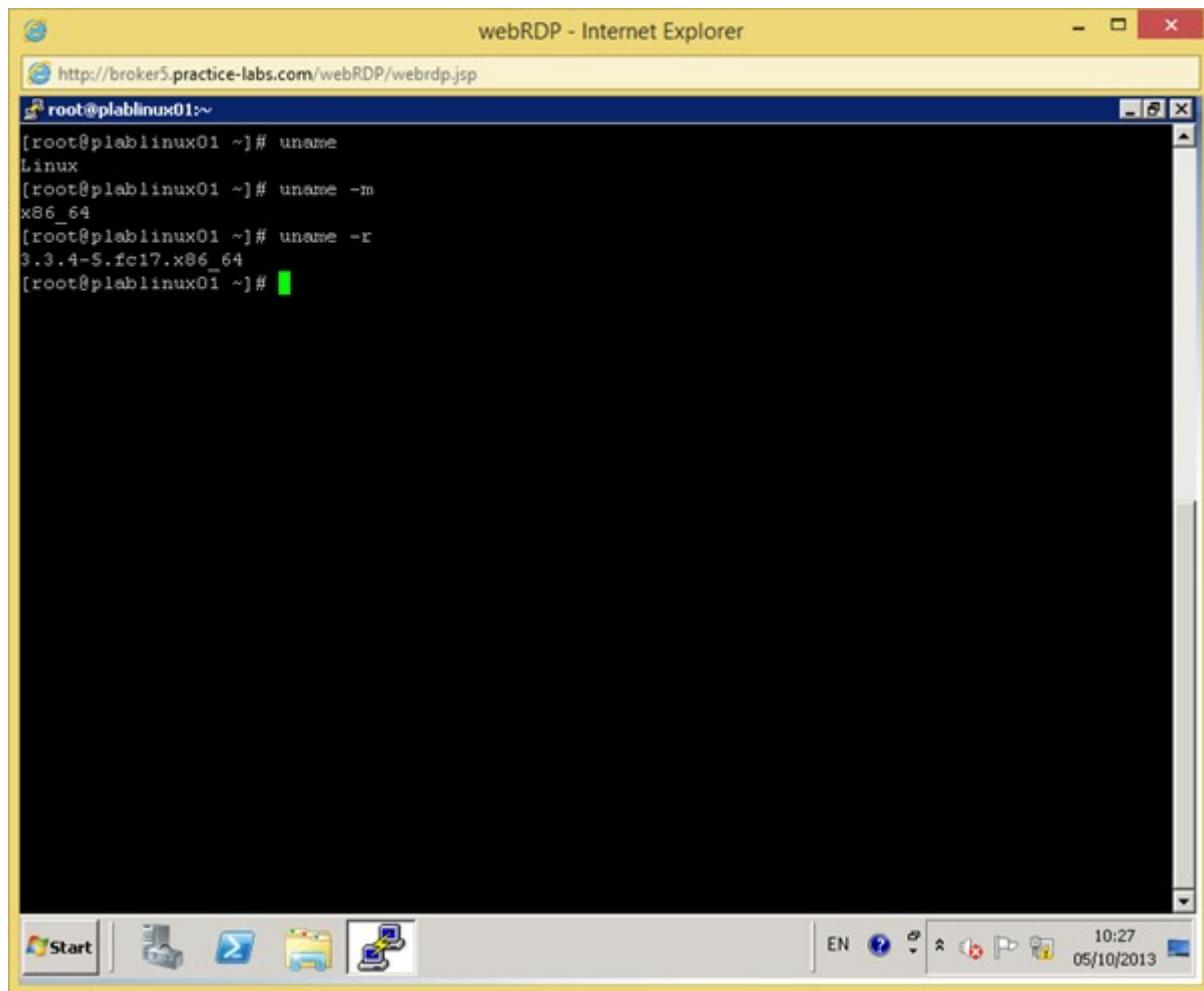
To display the hardware architecture, enter the following command:

```
uname -m
```



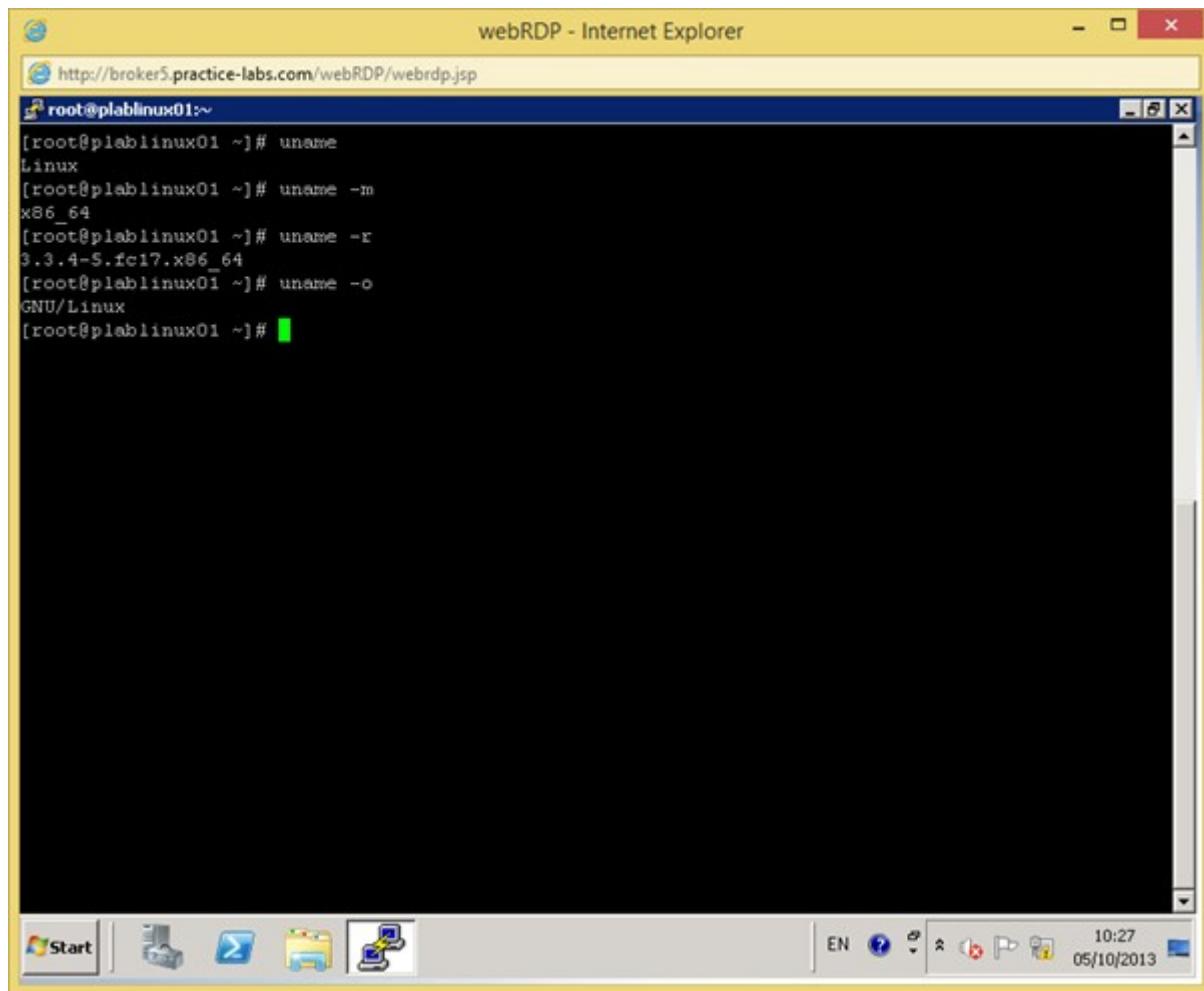
To display the kernel version, enter the following command:

```
uname -r
```



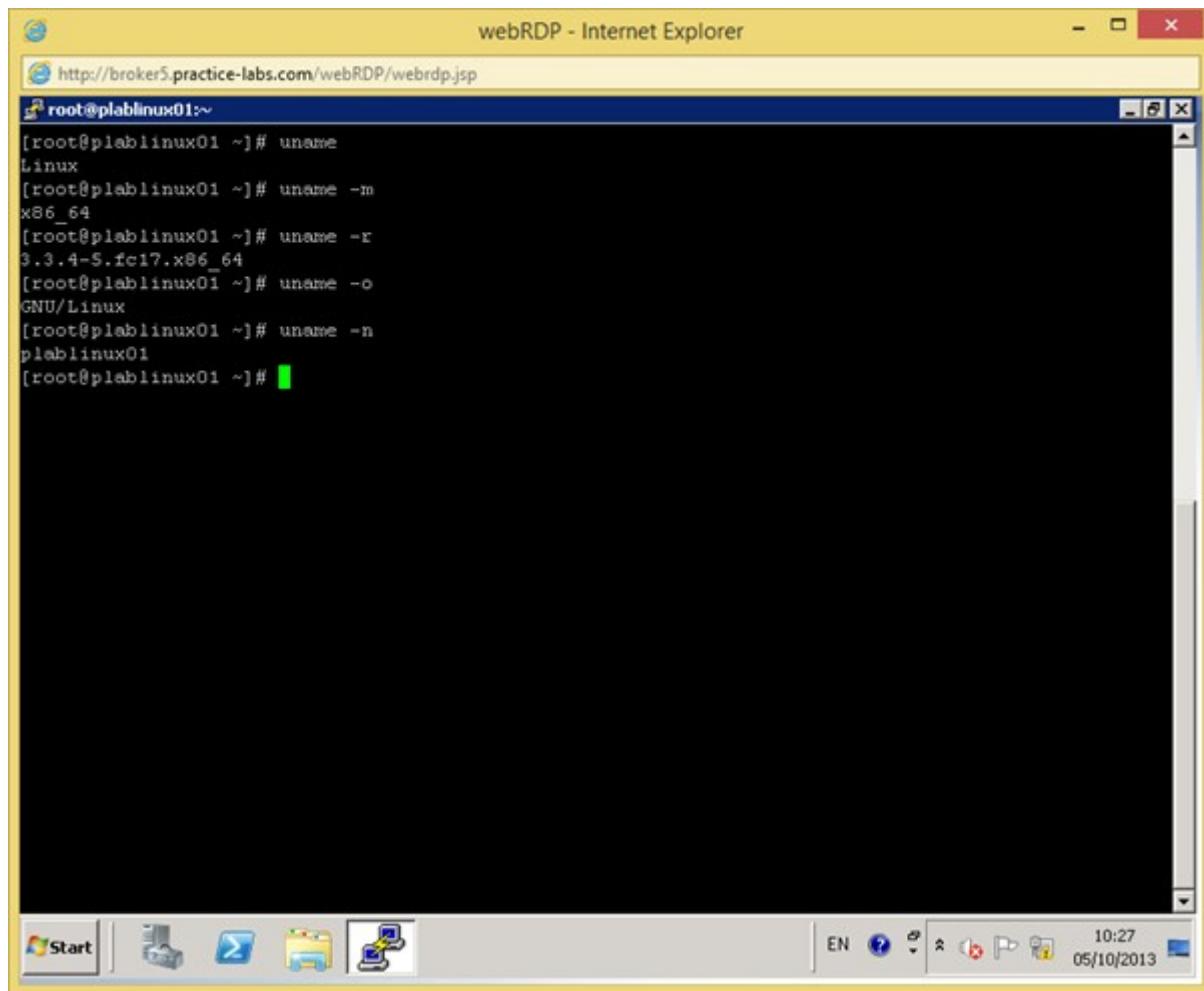
To display the operating system, enter the following command:

```
uname -o
```



To get the name of the system, enter the following command:

```
uname -n
```



```
[root@piablinux01 ~]# uname
Linux
[root@piablinux01 ~]# uname -m
x86_64
[root@piablinux01 ~]# uname -r
3.3.4-5.fc17.x86_64
[root@piablinux01 ~]# uname -o
GNU/Linux
[root@piablinux01 ~]# uname -n
piablinux01
[root@piablinux01 ~]#
```

Use and Modify the Shell Environment Including Defining, Referencing and Exporting

To obtain information on packages, perform the following steps:

Step 1

Ensure httpd package is installed to complete this task.

```
yum install httpd
```

```
--> Running transaction check
--> Package httpd.x86_64 0:2.2.23-1.fc17 will be installed
--> Processing Dependency: httpd-tools = 2.2.23-1.fc17 for package: httpd-2.2.23-1.fc17.x86_64
--> Processing Dependency: apr-util-ldap for package: httpd-2.2.23-1.fc17.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.2.23-1.fc17.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.2.23-1.fc17.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.4.6-1.fc17 will be installed
--> Package apr-util.x86_64 0:1.4.1-2.fc17 will be installed
--> Package apr-util-ldap.x86_64 0:1.4.1-2.fc17 will be installed
--> Package httpd-tools.x86_64 0:2.2.23-1.fc17 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package          Arch      Version       Repository      Size
=====
Installing:
httpd           x86_64   2.2.23-1.fc17    updates        831 k
Installing for dependencies:
apr              x86_64   1.4.6-1.fc17     fedora        99 k
apr-util         x86_64   1.4.1-2.fc17     fedora        78 k
apr-util-ldap   x86_64   1.4.1-2.fc17     fedora        17 k
httpd-tools     x86_64   2.2.23-1.fc17    updates        71 k

Transaction Summary
=====
Install 1 Package (+4 Dependent packages)

Total download size: 1.1 M
Installed size: 3.5 M
Is this ok [y/N]: 
```

Note: First the dependencies for the httpd package are resolved. Then, detailed description of the httpd package are given.

When prompted to confirm the installation, enter **y**. Package downloading starts. The installation and verification of packages continues.

The screenshot shows a terminal window titled "root@plablinux01:~" running on a Windows operating system. The terminal displays the output of a "yum install httpd" command. The output includes dependency resolution, package installation details, transaction summary, and download progress. The terminal window has a yellow header bar with the title "webRDP - Internet Explorer". The taskbar at the bottom shows icons for Start, Computer, File Explorer, and Control Panel, along with the date and time (06:49, 05/10/2013).

```
Dependencies Resolved

=====
Package          Arch      Version       Repository      Size
=====
httpd           x86_64    2.2.23-1.fc17   updates        831 k
Installing for dependencies:
apr              x86_64    1.4.6-1.fc17    fedora        99 k
apr-util         x86_64    1.4.1-2.fc17    fedora        78 k
apr-util-ldap   x86_64    1.4.1-2.fc17    fedora        17 k
httpd-tools     x86_64    2.2.23-1.fc17   updates        71 k

Transaction Summary
=====
Install 1 Package (+4 Dependent packages)

Total download size: 1.1 M
Installed size: 3.5 M
Is this ok [y/N]: y
Downloading Packages:
(1/5): apr-1.4.6-1.fc17.x86_64.rpm | 99 kB     00:00
(2/5): apr-util-1.4.1-2.fc17.x86_64.rpm | 78 kB     00:00
(3/5): apr-util-ldap-1.4.1-2.fc17.x86_64.rpm | 17 kB     00:00
(4/5): httpd-2.2.23-1.fc17.x86_64.rpm | 831 kB    00:00
(5/5): httpd-tools-2.2.23-1.fc17.x86_64.rpm | 71 kB     00:00

Total                                         1.0 MB/s | 1.1 MB  00:01
Running Transaction Check
Running Transaction Test
Transaction Test Succeeded
Running Transaction
[green progress bar]
```

After all packages are installed and verified, you are prompted with the **Complete!** message.

webRDP - Internet Explorer
http://broker5.practice-labs.com/webRDP/webrdp.jsp

```
[root@plablinux01:~]
Is this ok [y/N]: y
Downloading Packages:
(1/5): apr-1.4.6-1.fc17.x86_64.rpm | 99 kB     00:00
(2/5): apr-util-1.4.1-2.fc17.x86_64.rpm | 78 kB     00:00
(3/5): apr-util-ldap-1.4.1-2.fc17.x86_64.rpm | 17 kB     00:00
(4/5): httpd-2.2.23-1.fc17.x86_64.rpm | 831 kB    00:00
(5/5): httpd-tools-2.2.23-1.fc17.x86_64.rpm | 71 kB     00:00
-----
Total                                         1.0 MB/s | 1.1 MB   00:01

Running Transaction Check
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : apr-1.4.6-1.fc17.x86_64                               1/5
  Installing : apr-util-1.4.1-2.fc17.x86_64                         2/5
  Installing : apr-util-ldap-1.4.1-2.fc17.x86_64                     3/5
  Installing : httpd-tools-2.2.23-1.fc17.x86_64                      4/5
  Installing : httpd-2.2.23-1.fc17.x86_64                           5/5
  Verifying   : apr-util-ldap-1.4.1-2.fc17.x86_64                     1/5
  Verifying   : httpd-tools-2.2.23-1.fc17.x86_64                      2/5
  Verifying   : apr-util-1.4.1-2.fc17.x86_64                         3/5
  Verifying   : httpd-2.2.23-1.fc17.x86_64                           4/5
  Verifying   : apr-1.4.6-1.fc17.x86_64                             5/5

Installed:
  httpd.x86_64 0:2.2.23-1.fc17

Dependency Installed:
  apr.x86_64 0:1.4.6-1.fc17                                apr-util.x86_64 0:1.4.1-2.fc17
  apr-util-ldap.x86_64 0:1.4.1-2.fc17                         httpd-tools.x86_64 0:2.2.23-1.fc17

Complete!
[root@plablinux01 ~]#
```

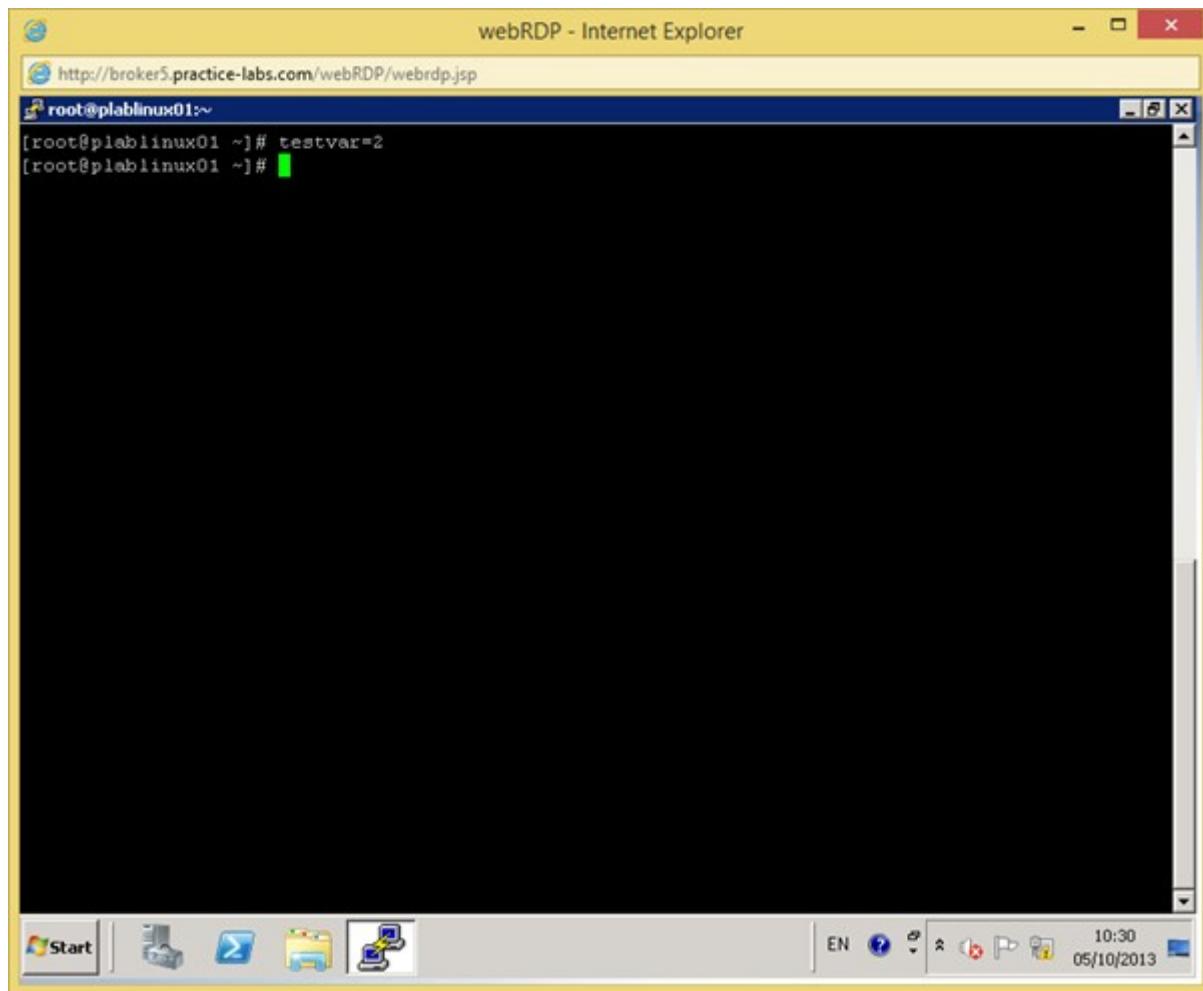
Clear the screen by entering the following command:

```
clear
```

Step 2

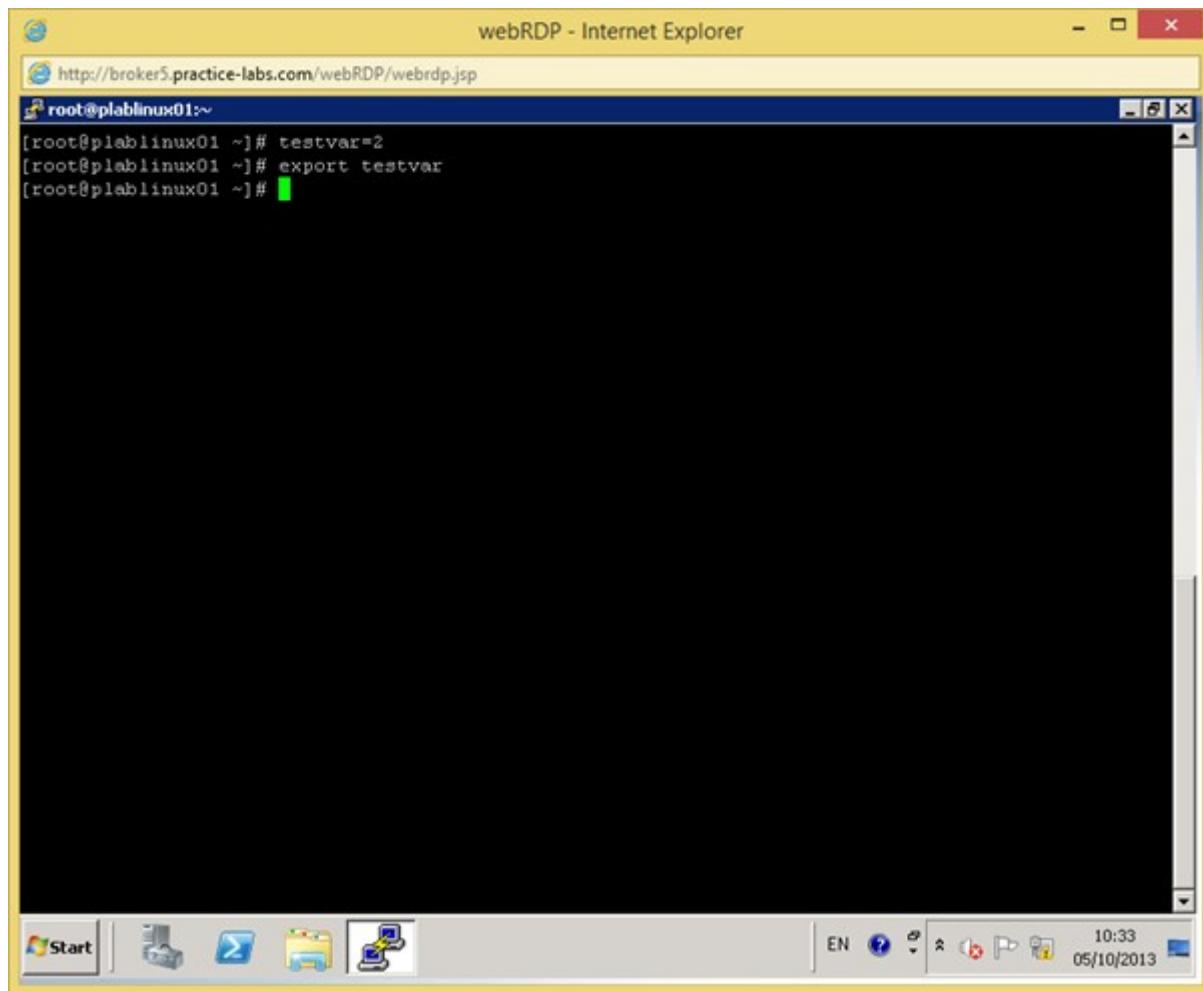
First declare a variable. To declear a variable, enter the following command:

```
testvar=2
```



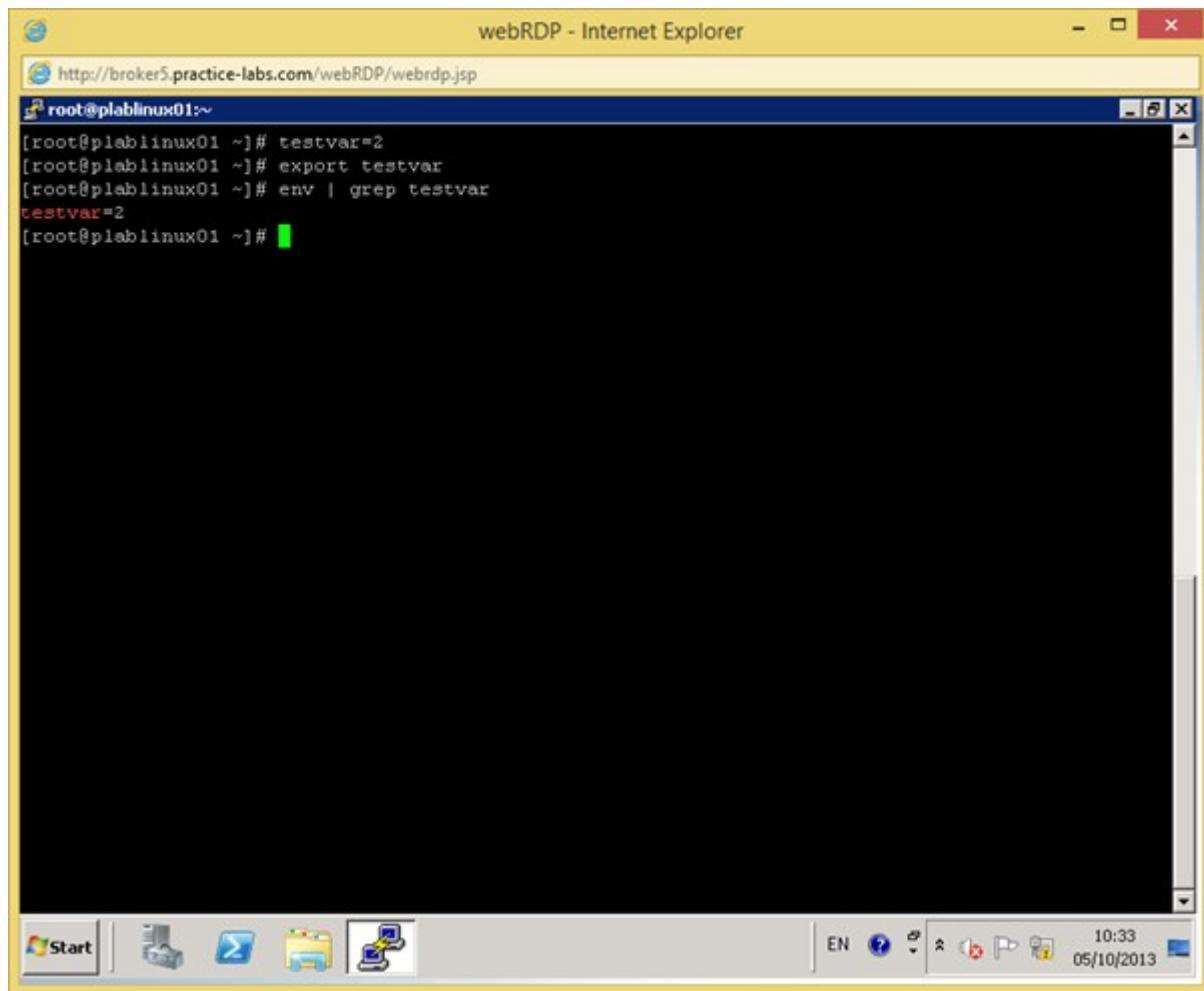
Now, export the variable. To export the variable, enter the following command:

```
export testvar
```



To verify that the variable has been exported, enter the following command:

```
env | grep testvar
```



The variable value is returned as the output.

To unset the variable, enter the following command:

```
unset testvar
```

```
[root@plablinux01 ~]# testvar=2
[root@plablinux01 ~]# export testvar
[root@plablinux01 ~]# env | grep testvar
testvar=2
[root@plablinux01 ~]# unset testvar
[root@plablinux01 ~]#
```

Step 7

You can use the **man** command to display complete information of a specific command. To use the **man** command, enter the following command to view information about the **pwd** command:

```
man pwd
```

Use and Edit Command History

To use and edit command history, perform the following steps:

Step 1

To view the command history, enter the following command:

```
history
```

```
root@plablinux01:~#
87 systemctl start vncserver@:1.service
88 reboot
89 ping 192.168.0.1
90 cl -a
91 shutdown -h now
92 echo "this is the first command"
93 clear
94 pwd
95 clear
96 echo $$
97 bash
98 echo $$
99 clear
100 uname
101 unam -m
102 clear
103 uname
104 uname -m
105 uname -r
106 uname -o
107 uname -n
108 clear
109 testvar=2
110 export testvar
111 clear
112 testvar=2
113 export testvar
114 env | grep testvar
115 unset testvar
116
117 clear
118 history
[root@plablinux01 ~]#
```

At the command prompt, use the following keys to navigate through the commands that you have entered so far:

Keys for the Command History

Ctrl+P	Previous line (equivalent to Up-arrow)
Ctrl+n	Next line (equivalent to Down-arrow)
Ctrl+b	Go back one character on the line (equivalent to Left-Arrow)

Ctrl+f	Go forward one character on the line (equivalent to Right-Arrow)
Ctrl+a	Go to the beginning of the line (equivalent to <Home>)
Ctrl+e	Go to the end of the line (equivalent to <End>)

To enter a specific command from the history, first view the number and then run the command with the number. You will need to use ! as a prefix to the command number. Enter the following command:

```
!22
```

```
90 cl -a
91 shutdown -h now
92 echo "this is the first command"
93 clear
94 pwd
95 clear
96 echo $$
97 bash
98 echo $$
99 clear
100 uname
101 unam -m
102 clear
103 uname
104 uname -m
105 uname -r
106 uname -o
107 uname -n
108 clear
109 testvar=2
110 export testvar
111 clear
112 testvar=2
113 export testvar
114 env | grep testvar
115 unset testvar
116
117 clear
118 history
[root@plablinux01 ~]# !22
ls
anaconda-ks.cfg Desktop Documents Downloads Music Pictures Public Templates Videos
[root@plablinux01 ~]#
```

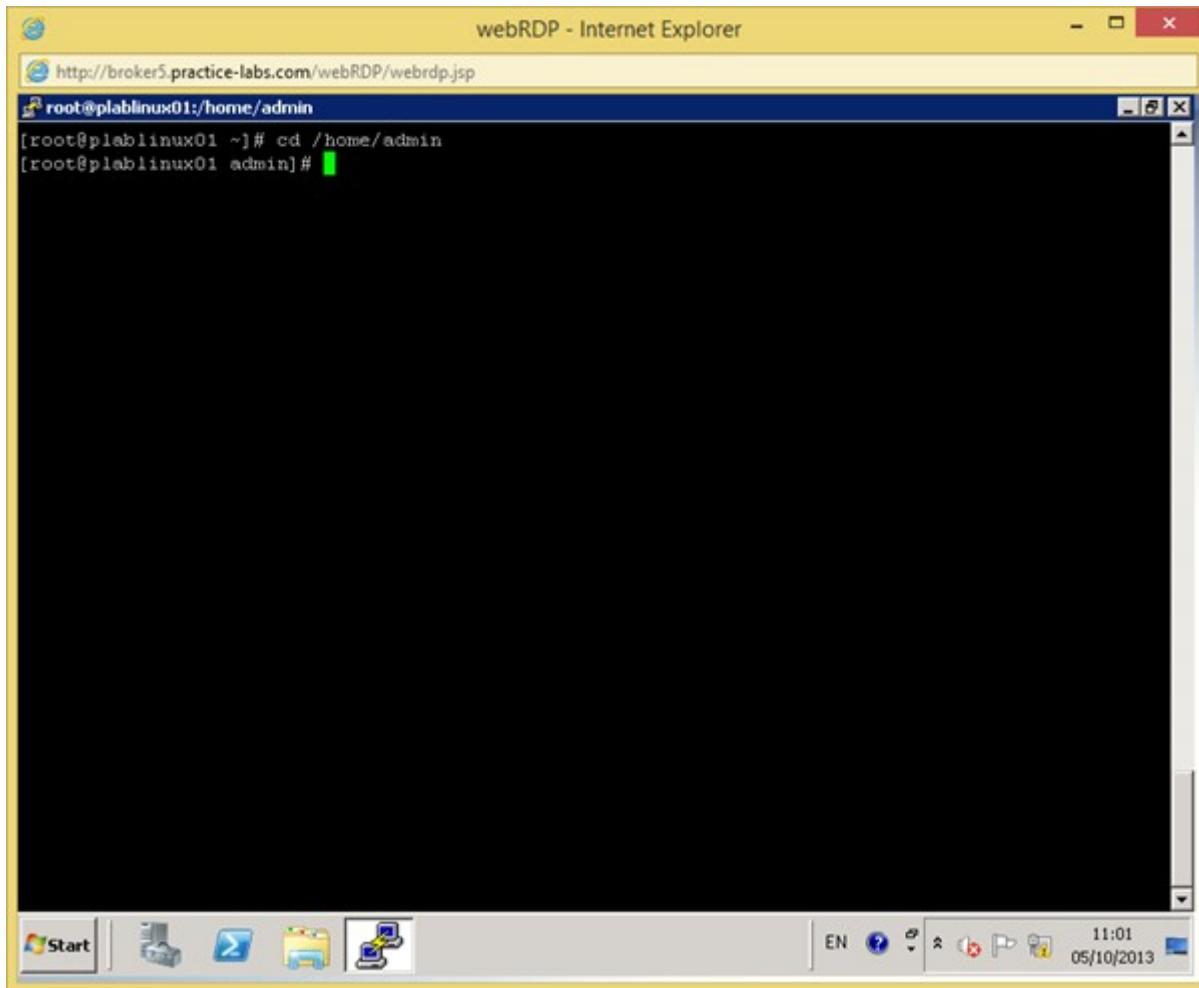
Invoke Commands Inside and Outside the Defined Path

To understand the terms and utilities, perform the following steps:

Step 1

To navigate to a directory using absolute path, enter the following command:

```
cd /home/admin
```

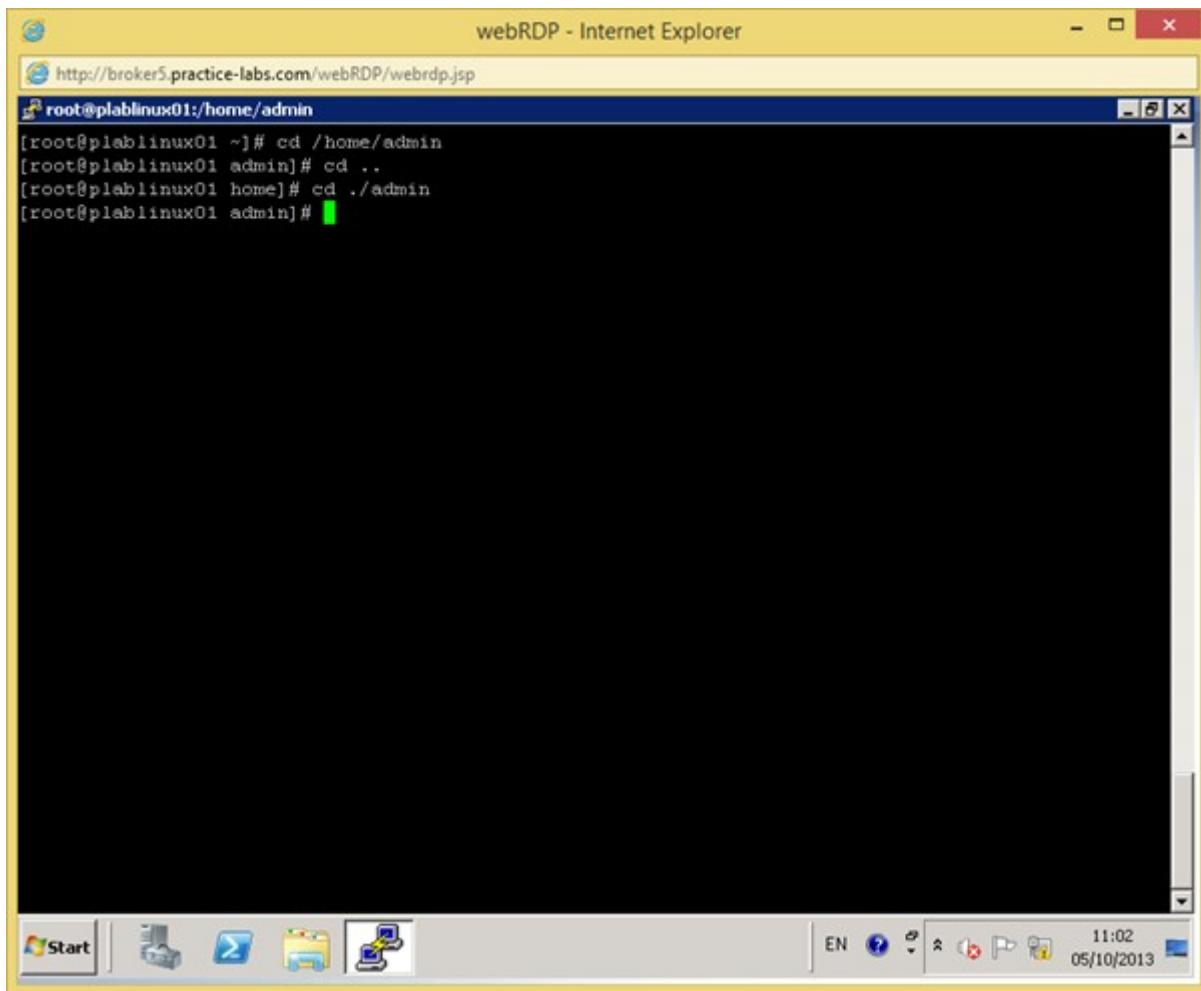


To go back to the home directory, enter the following command:

```
cd ..  
/
```

To use the relative path to the admin directory, enter the following command:

```
cd ./admin
```



Leave the devices you have powered on in their current state and proceed to the next exercise.

Exercise 2 - Create, Monitor and Kill Processes

In this exercise you will understand how manage processes. Please refer to your course material or use your favourite search engine to research **Create, Monitor and Kill Processes in Linux**.

In this exercise, you will configure:

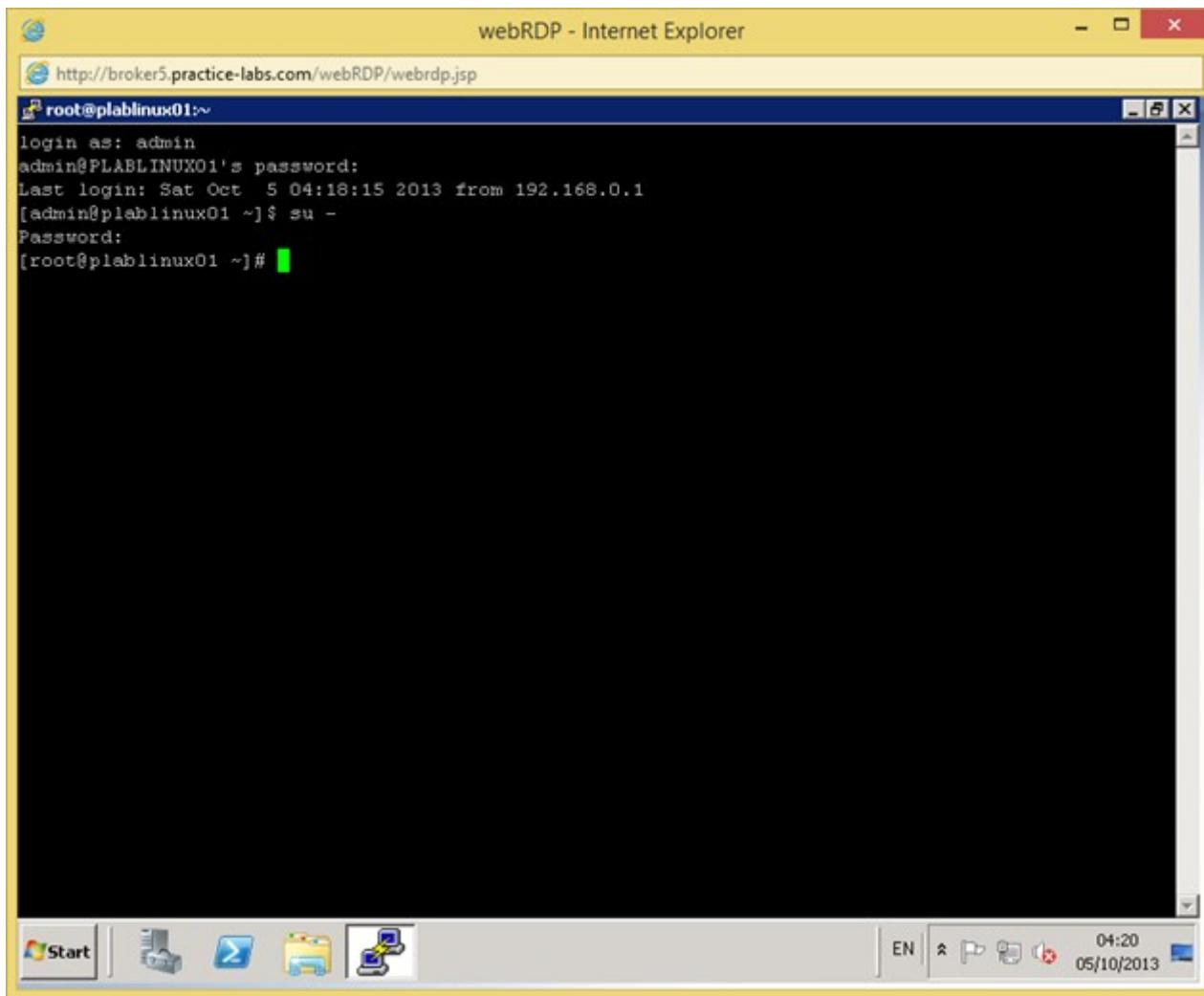
- Run jobs in the foreground and background
- Signal a program to continue running after logout
- Monitor active processes
- Select and sort processes for display
- Send signals to processes
- Terms and Utilities

Run Jobs in the Foreground and Background

To run jobs in the foreground and background, perform the following steps:

Step 1

From the command console using the root privileges prompt, defined by the # symbol.



Step 2

Let's first list all the processes that are running on the system. To list the processes, enter the following command:

```
ps -e
```

The screenshot shows a webRDP session in Internet Explorer. A terminal window is open with the command 'ps aux' running. The output lists numerous processes, including system daemons like gconfd, dconf-service, and notification-daemon, along with user applications like gnome-panel, gnome-screensaver, and evolution-alarm. The terminal prompt '[root@plablinux01 admin]#' is visible at the bottom.

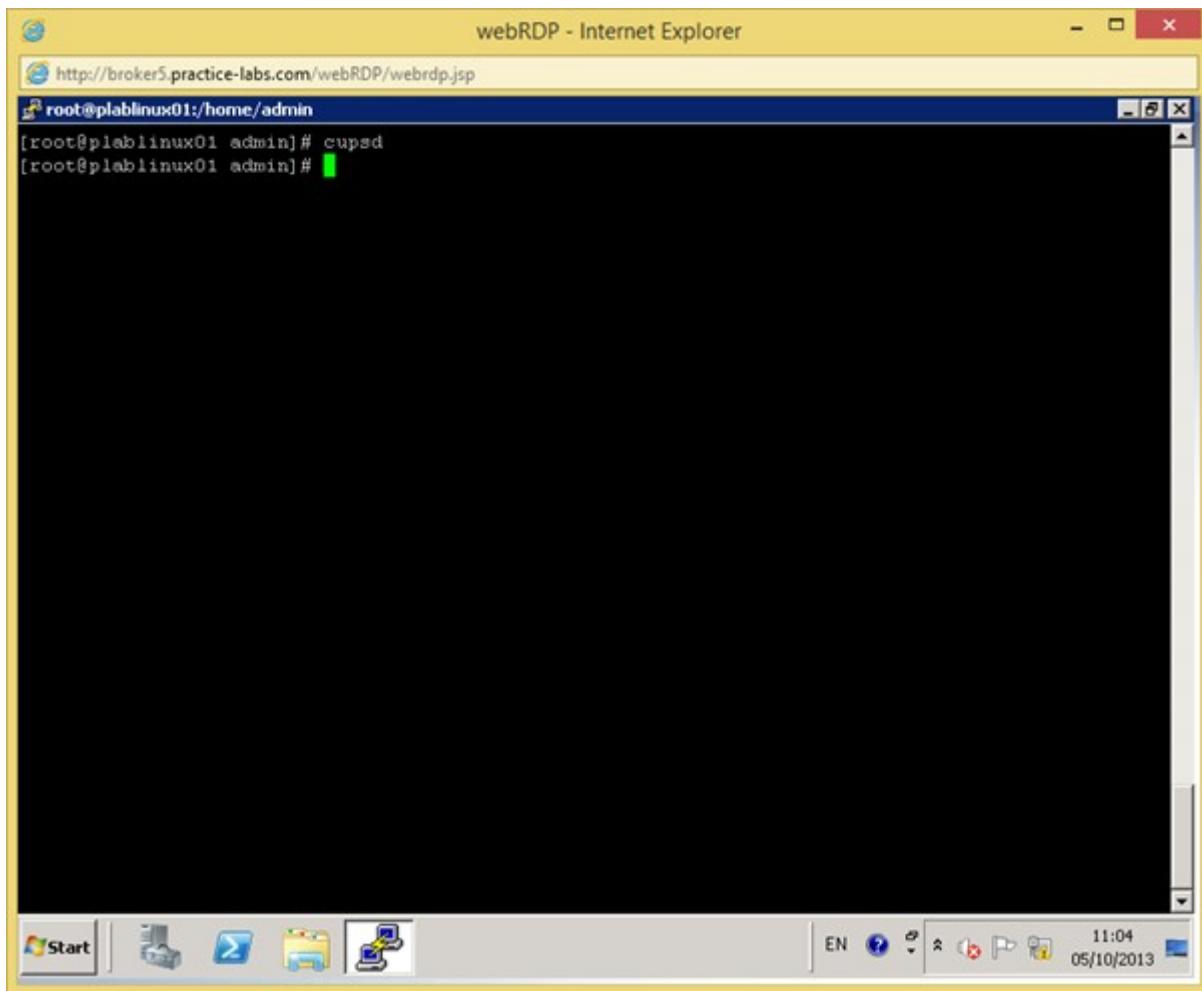
```
1202 ? 00:00:00 gnome-panel
1208 ? 00:00:00 gconfd-2
1210 ? 00:00:00 dconf-service
1215 ? 00:00:00 gnome-screensav
1217 ? 00:00:00 tracker-miner-f
1219 ? 00:00:00 nm-applet
1220 ? 00:00:00 seapplet
1222 ? 00:00:00 deja-dup-monito
1228 ? 00:00:00 notification-da
1231 ? 00:00:00 gnome-sound-app
1234 ? 00:00:00 evolution-alarm
1237 ? 00:00:00 abrt-applet
1246 ? 00:00:00 bluetooth-apple
1250 ? 00:00:00 tracker-store
1256 ? 00:00:05 vmtoolsd
1259 ? 00:00:00 gnome-fallback-
1313 ? 00:00:00 wnck-applet
1315 ? 00:00:00 clock-applet
1317 ? 00:00:00 notification-ar
1356 ? 00:00:00 sendmail
1371 ? 00:00:00 sendmail
1397 ? 00:00:00 sshd
1401 ? 00:00:00 sshd
1402 pts/0 00:00:00 bash
1458 pts/0 00:00:00 su
1462 pts/0 00:00:00 bash
1523 ? 00:00:01 kworker/0:0
1604 ? 00:00:00 kworker/0:1
1607 ? 00:00:00 flush-253:1
1621 ? 00:00:00 anacron
1630 ? 00:00:00 kworker/0:2
1632 pts/0 00:00:00 ps
[root@plablinux01 admin]#
```

You can clear the screen using the **clear** command.

The complete list of processes is displayed. You can now choose a process that you want to run in the background. For demonstration purposes of this lab, you will run cupsd in the background. Enter the following command:

```
cupsd
```

Note that no result will be displayed.



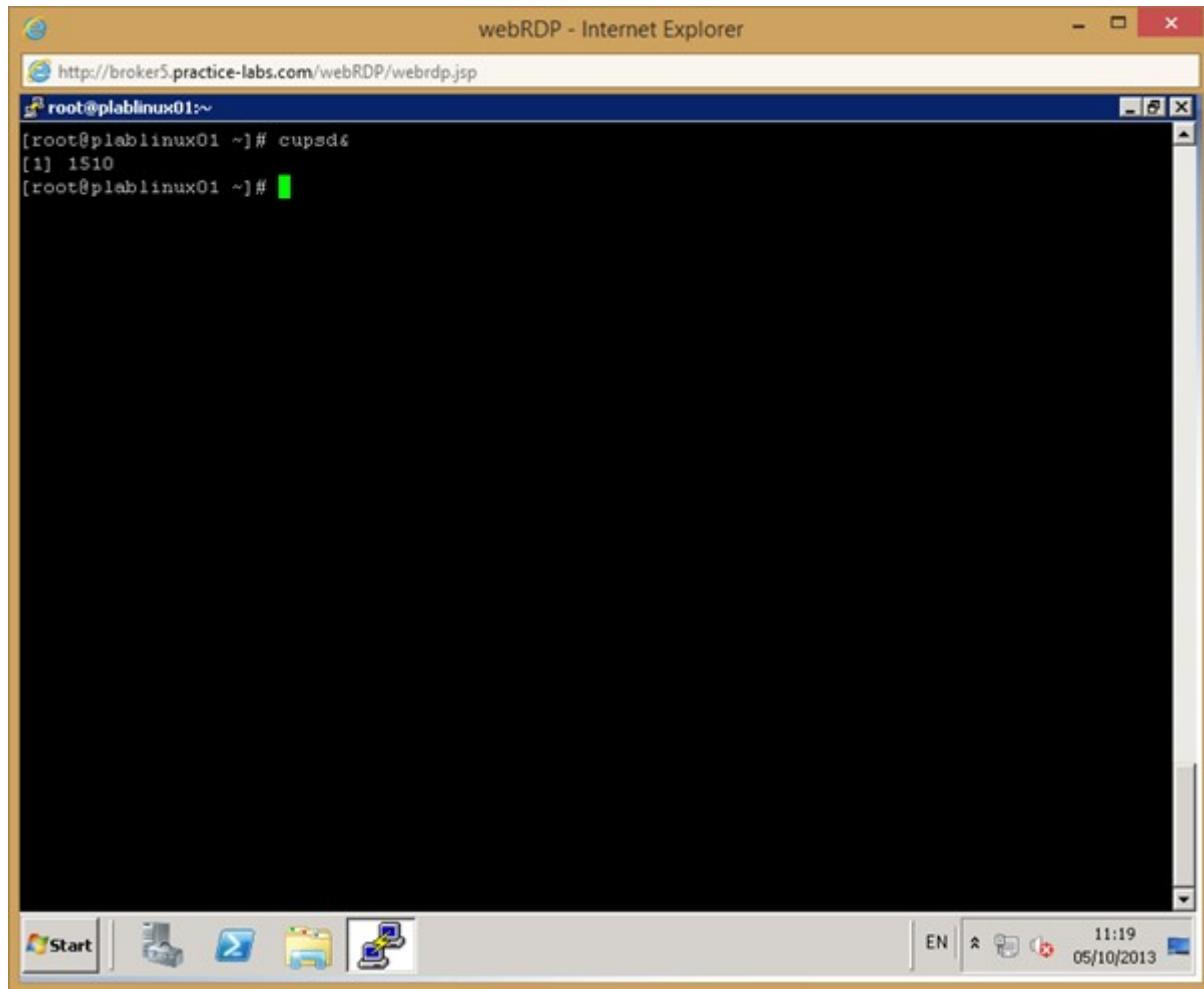
Now enter the following command:

```
bg
```

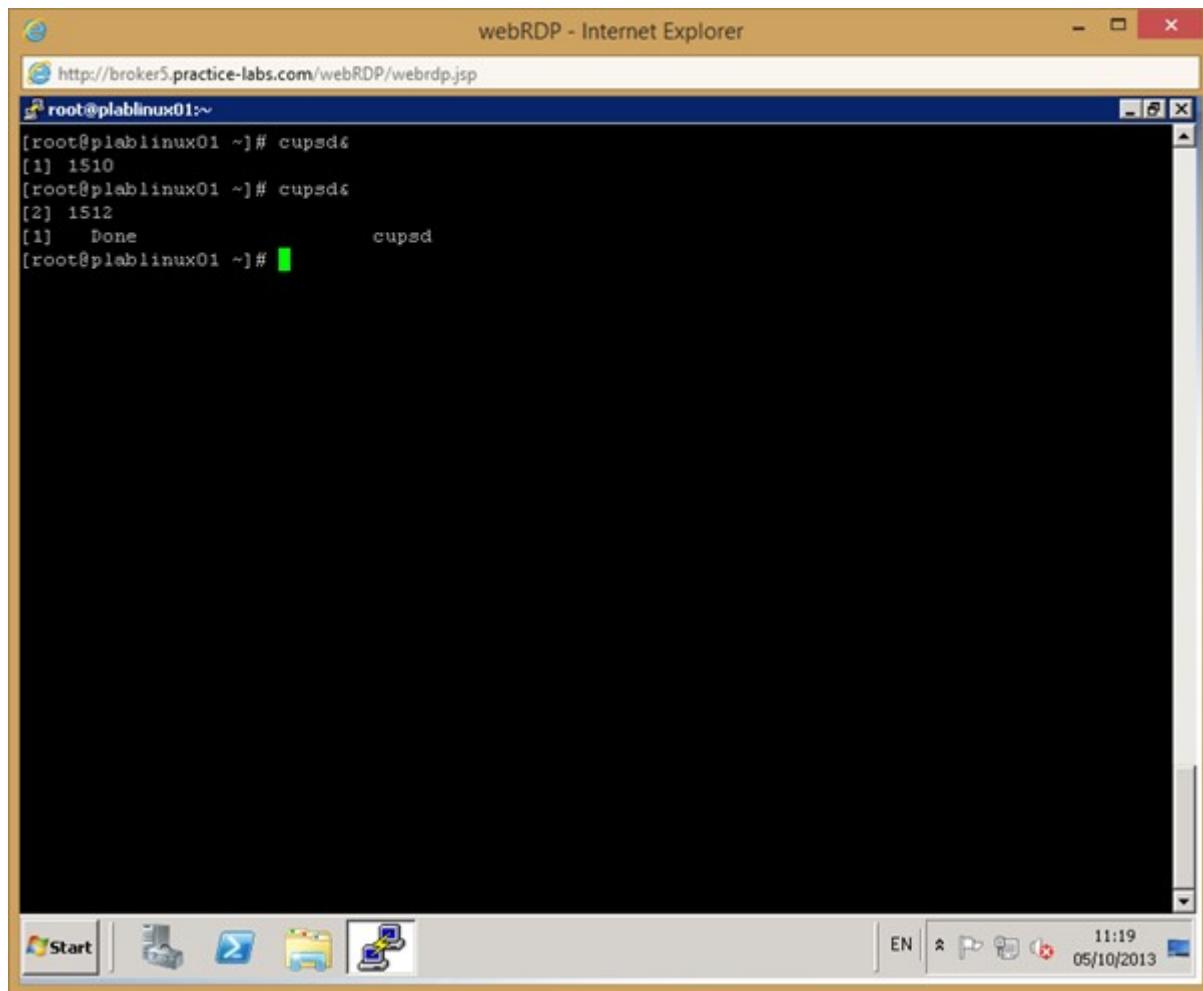
Note: You might get an error when you execute this command.

You can also put the process in the background by adding & as suffix to the process name. If the process was running, it will be stopped. However, if you execute the same command, it will start running in the background. Enter the following command:

```
cupsd&
```

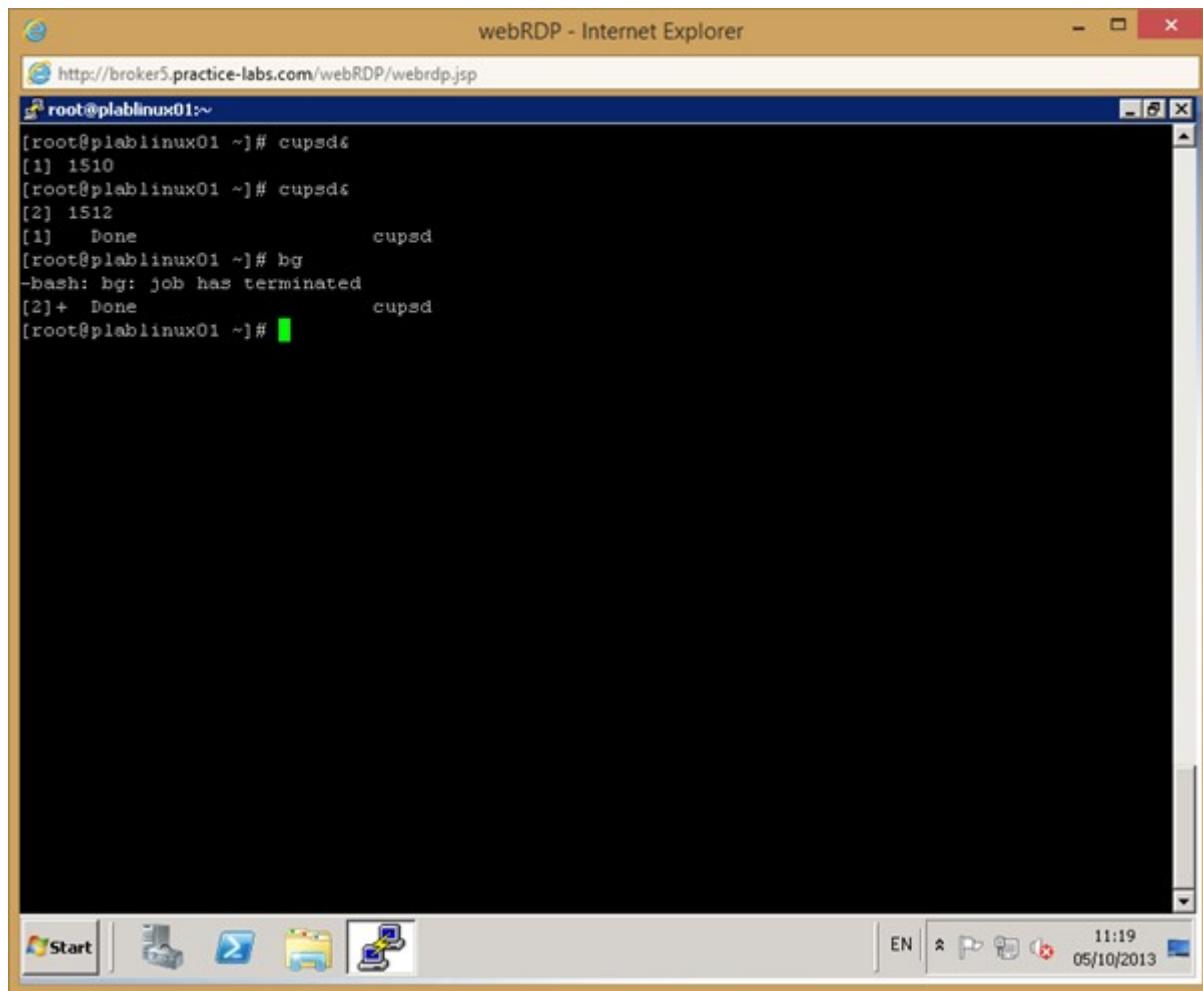


Enter same command once again.



Now enter the following command:

```
bg
```



Signal a Program to Continue Running After Logout

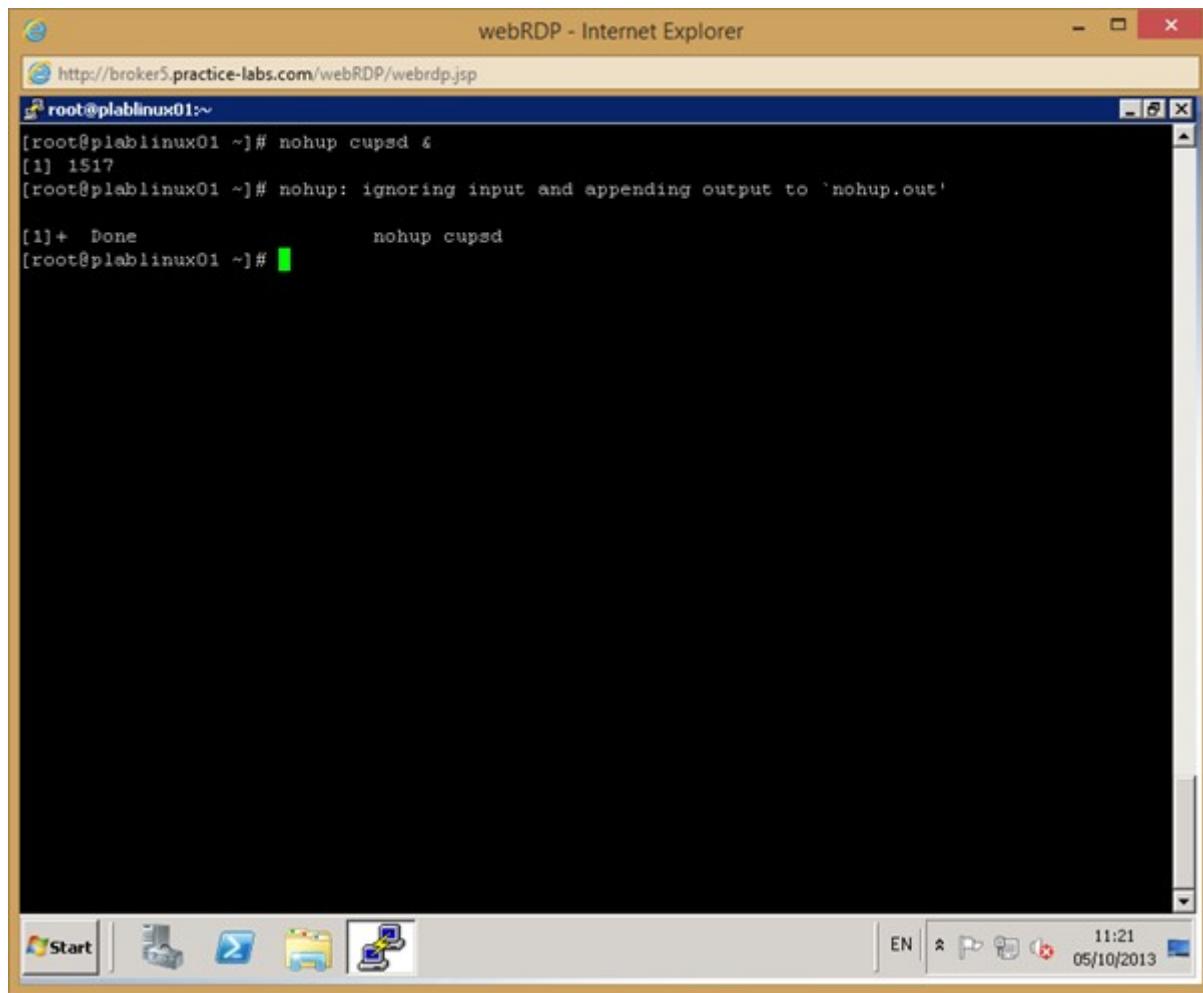
The **nohup** utility continues to run a program even after the user logs out. The **SIGHUP** signals are ignored for the program that is specified to run with the nohup utility.

To signal a program to continue running after logout, perform the following steps:

Step 1

From the command shell list all processes by entering the following commands:

```
nohup cupsd &
```



Similarly, you can create a batch script and run with the nohup program. The batch script will continue to run even if you as a user log out.

Monitor Active Processes

A system will have different types of processes running at any given point of time. You can monitor all, specific processes, or processes for a specific user.

To monitor active processes, perform the following steps:

Step 1

List all processes by entering the following commands:

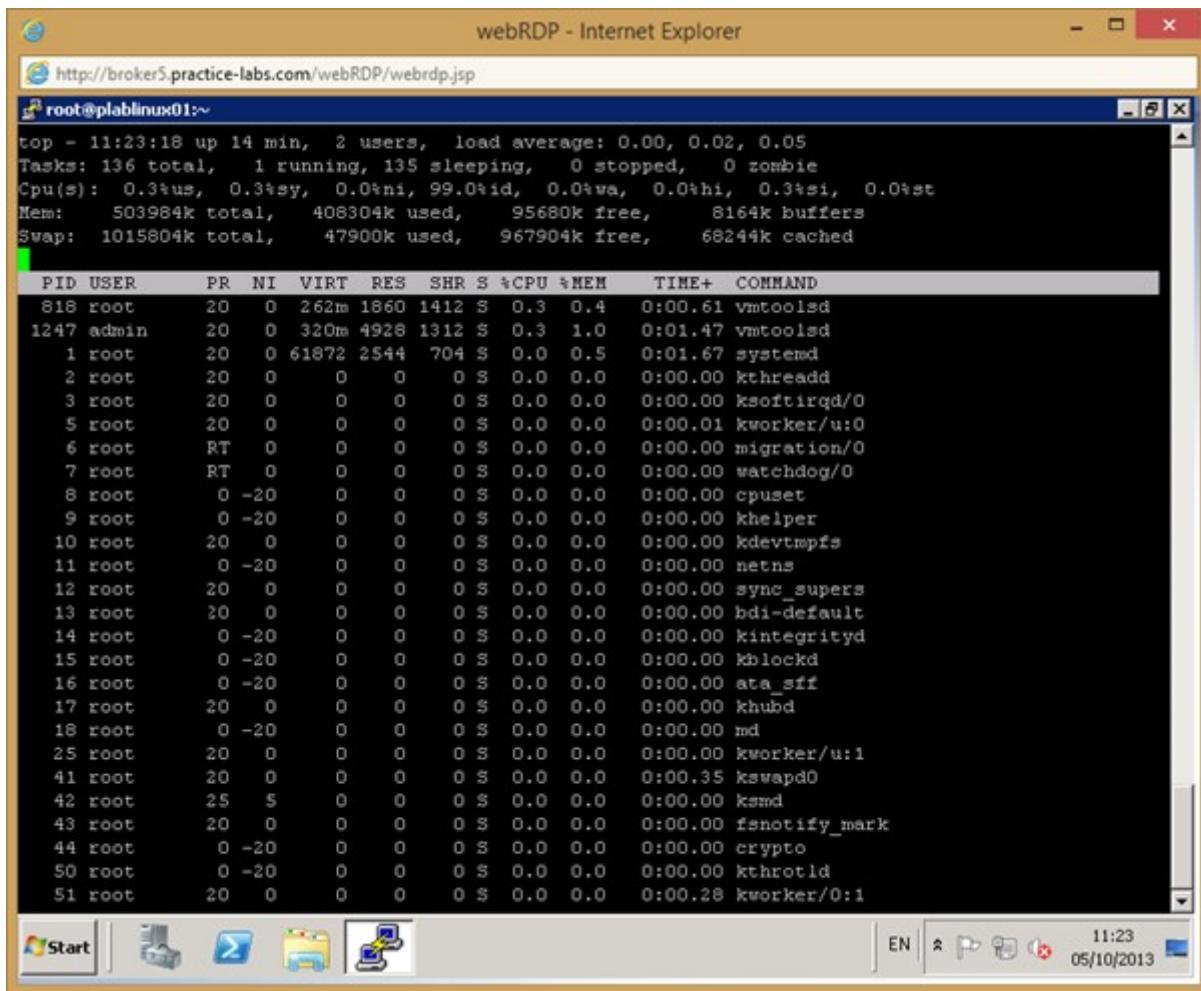
```
ps -e
```

The screenshot shows a webRDP session in Internet Explorer. The terminal window displays the output of the 'ps' command, listing numerous processes running on the system. The processes include system daemons like 'gnome-screensav', 'tracker-miner-f', 'nm-applet', 'seapplet', 'deja-dup-monito', 'notification-da', 'gnome-sound-app', 'evolution-alarm', 'abrt-applet', 'bluetooth-apple', 'tracker-store', 'vmtoolsd', 'gnome-fallback-', 'wnck-applet', 'clock-applet', 'notification-ar', 'sendmail', 'sendmail', 'kworker/0:0', 'sshd', 'sshd', 'bash', 'su', 'bash', 'cupsd', 'cupsd', 'cupsd', 'cupsd', 'cupsd', 'cupsd', 'cupsd', 'ps'. The command '[root@plablinux01 ~]#' is visible at the bottom of the terminal window.

```
1207 ? 00:00:00 gnome-screensav
1209 ? 00:00:00 tracker-miner-f
1210 ? 00:00:00 nm-applet
1212 ? 00:00:00 seapplet
1214 ? 00:00:00 deja-dup-monito
1218 ? 00:00:00 notification-da
1221 ? 00:00:00 gnome-sound-app
1224 ? 00:00:00 evolution-alarm
1225 ? 00:00:00 abrt-applet
1228 ? 00:00:00 bluetooth-apple
1237 ? 00:00:00 tracker-store
1247 ? 00:00:01 vmtoolsd
1248 ? 00:00:00 gnome-fallback-
1306 ? 00:00:00 wnck-applet
1307 ? 00:00:00 clock-applet
1310 ? 00:00:00 notification-ar
1348 ? 00:00:00 sendmail
1350 ? 00:00:00 sendmail
1370 ? 00:00:00 kworker/0:0
1379 ? 00:00:00 sshd
1383 ? 00:00:00 sshd
1384 pts/0 00:00:00 bash
1441 pts/0 00:00:00 su
1445 pts/0 00:00:00 bash
1499 ? 00:00:00 cupsd
1502 ? 00:00:00 cupsd
1504 ? 00:00:00 cupsd
1505 ? 00:00:00 kworker/0:2
1511 ? 00:00:00 cupsd
1513 ? 00:00:00 cupsd
1518 ? 00:00:00 cupsd
1520 pts/0 00:00:00 ps
[root@plablinux01 ~]#
```

Now, you will list all active processes that are running on the system. Enter the following command:

```
top
```



Select and sort processes for display

You can select and sort processes for display. To select and sort processes for display, perform the following steps:

Step 1

List all the processes and locate cupsd with its process id by entering the following commands:

```
ps ux
```

```

root      668  0.0  0.5 222464  2712 ?          S1   11:08  0:00 /usr/libexec/polkit-1/polkitd --n
root      672  0.0  0.2 71152   1016 ?          S    11:08  0:00 /usr/sbin/modem-manager
root      673  0.0  0.0 23168   476 ?          Ss   11:08  0:00 /usr/sbin/bluetoothd -n
root      691  0.0  0.1 77600   616 ?          Ss   11:08  0:00 /usr/sbin/sshd -D
root      818  0.0  0.3 268784  1860 ?          S1   11:08  0:00 /usr/sbin/vmtoolsd
root      840  0.0  0.2 211808  1028 ?          Ssl  11:08  0:00 /usr/sbin/gdm-binary -nodaemon
root      845  0.0  0.1 249612  936 ?          S1   11:08  0:00 /usr/libexec/gdm-simple-slave --d
root      887  0.0  1.9 138844  9924 ttys1        Ss+  11:08  0:00 /usr/bin/Xorg :0 -background none
root      890  0.0  0.0     0   0 ?          S<   11:08  0:00 [ttm_swap]
root      902  0.0  0.2 346228  1396 ?          S1   11:08  0:00 gdm-session-worker [pam/gdm-welco
root      905  0.0  0.2 319728  1348 ?          Ssl  11:08  0:00 /usr/libexec/accounts-daemon
root      912  0.0  0.0     0   0 ?          S    11:08  0:00 [flush-253:1]
root      926  0.0  0.3 230148  1980 ?          Ssl  11:08  0:00 /usr/libexec/upowerd
root      1132 0.0  0.3 237136  1832 ?          S1   11:08  0:00 gdm-session-worker [pam/gdm-passw
root      1161 0.0  0.3 204412  1512 ?          Ss   11:08  0:00 /usr/sbin/cupsd -f
root      1170 0.0  0.5 373872  2552 ?          Ssl  11:08  0:00 /usr/lib/udisks2/udisksd --no-deb
root      1350 0.0  0.2 99728   1140 ?          Ss   11:09  0:00 sendmail: accepting connections
root      1370 0.0  0.0     0   0 ?          S    11:13  0:00 [kworker/0:0]
root      1379 0.0  0.8 132300  4496 ?          Ss   11:16  0:00 sshd: admin [priv]
root      1441 0.0  0.4 168192  2108 pts/0        S    11:16  0:00 su -
root      1445 0.0  0.5 115236  2852 pts/0        S    11:16  0:00 -bash
root      1499 0.0  0.6 204408  3464 ?          Ss   11:16  0:00 cupsd -C /etc/cups/cupsd.conf
root      1502 0.0  0.6 204408  3460 ?          Ss   11:17  0:00 cupsd -C /etc/cups/cuped.conf
root      1504 0.0  0.6 204408  3464 ?          Ss   11:17  0:00 cuped -C /etc/cups/cupsd.conf
root      1505 0.0  0.0     0   0 ?          S    11:18  0:00 [kworker/0:2]
root      1511 0.0  0.6 204408  3460 ?          Ss   11:19  0:00 cupsd -C /etc/cups/cupsd.conf
root      1513 0.0  0.6 204408  3464 ?          Ss   11:19  0:00 cuped -C /etc/cups/cupsd.conf
root      1518 0.0  0.6 204408  3460 ?          Ss   11:21  0:00 cupsd -C /etc/cups/cupsd.conf
root      1521 0.2  0.2 15256   1204 pts/0        T    11:23  0:00 top
root      1525 0.5  1.0 401916  5444 ?          S1   11:23  0:00 /usr/libexec/packagekitd
root      1531 0.0  0.0     0   0 ?          S    11:23  0:00 [kworker/0:1]
root      1533 0.0  0.2 115736  1200 pts/0        R+  11:23  0:00 ps ux

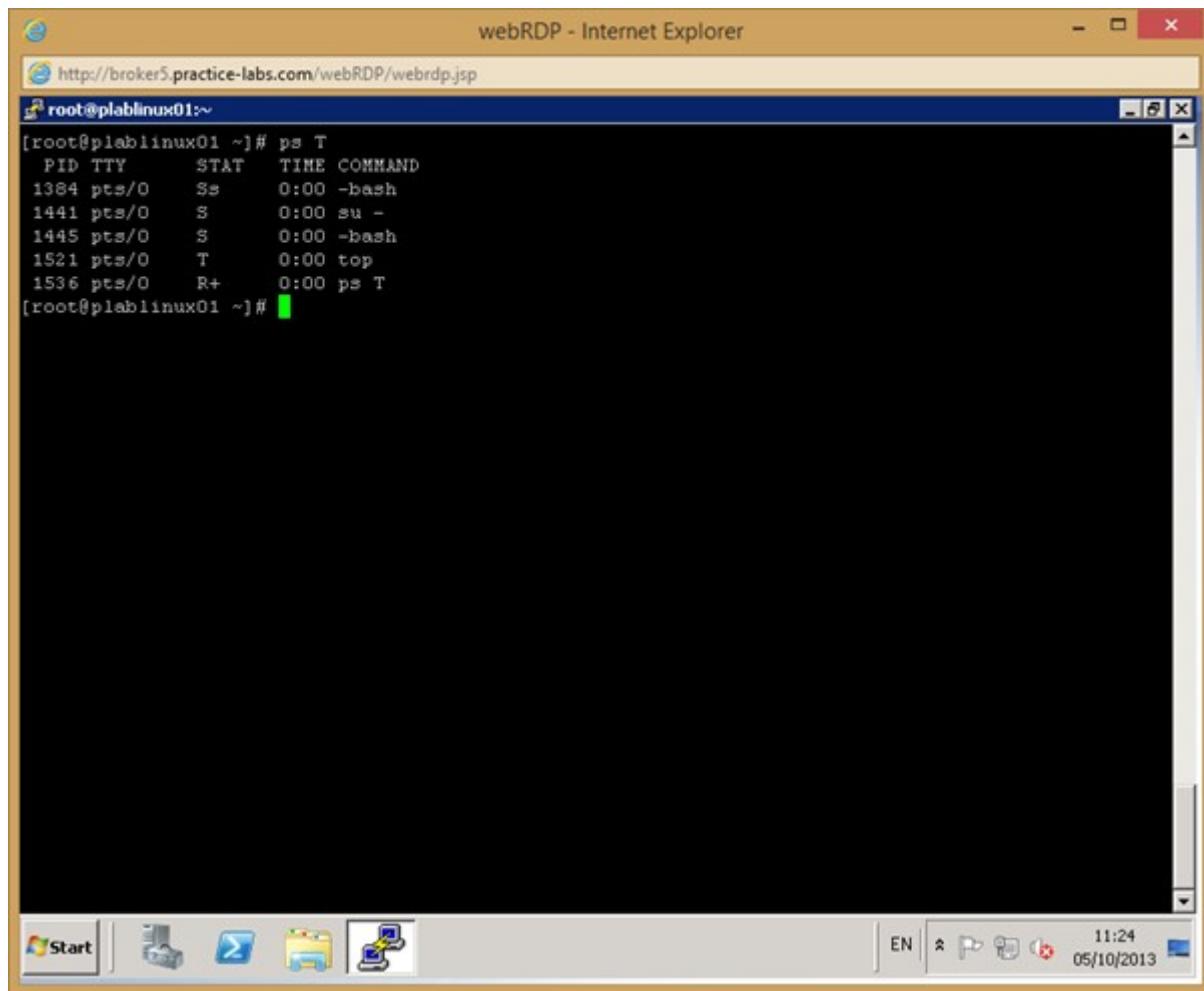
```

Step 2

Clear the screen using the **clear** command.

To list processes run under a terminal session by a user, enter the following commands:

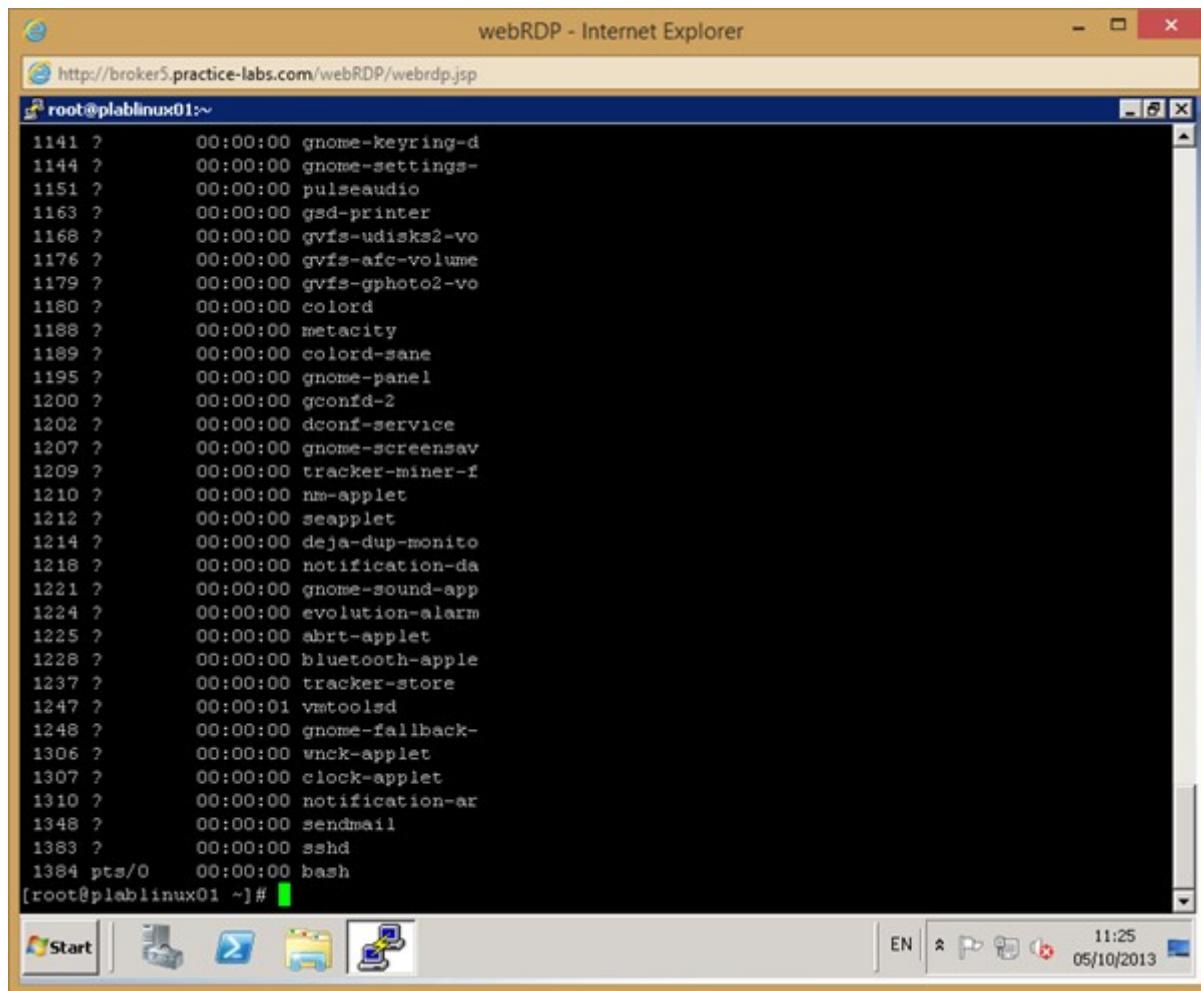
```
ps T
```



Step 3

To list every process except those running as root, enter the following commands:

```
ps -U root -u root -N
```



To list the processes by a specific user, enter the following command:

```
ps -u admin
```

The screenshot shows a Windows desktop environment. In the center is a terminal window titled "webRDP - Internet Explorer" with the URL "http://broker5.practice-labs.com/webRDP/webrdp.jsp". The terminal window displays a list of processes from the command "ps -e". The processes listed include gvfsd, gvfs-fuse-daemo, ssh-agent, gnome-keyring-d, gnome-settings-, pulseaudio, gsd-printer, gvfs-udisks2-vo, gvfs-afc-volume, gvfs-ghoto2-vo, metacity, gnome-panel, gconfd-2, dconf-service, gnome-screensav, tracker-miner-f, nm-applet, seapplet, deja-dup-monito, notification-da, gnome-sound-app, evolution-alarm, abrt-applet, bluetooth-apple, tracker-store, vmtoolsd, gnome-fallback-, wnck-applet, clock-applet, notification-ar, sshd, and bash. The terminal prompt "[root@plablinux01 ~]#" is visible at the bottom. The desktop taskbar at the bottom shows icons for Start, File Explorer, Task View, and Control Panel. The system tray shows the date and time as 11:26 05/10/2013.

```
1077 ? 00:00:00 gvfsd
1079 ? 00:00:00 gvfs-fuse-daemo
1123 ? 00:00:00 ssh-agent
1141 ? 00:00:00 gnome-keyring-d
1144 ? 00:00:00 gnome-settings-
1151 ? 00:00:00 pulseaudio
1163 ? 00:00:00 gsd-printer
1168 ? 00:00:00 gvfs-udisks2-vo
1176 ? 00:00:00 gvfs-afc-volume
1179 ? 00:00:00 gvfs-ghoto2-vo
1188 ? 00:00:00 metacity
1195 ? 00:00:00 gnome-panel
1200 ? 00:00:00 gconfd-2
1202 ? 00:00:00 dconf-service
1207 ? 00:00:00 gnome-screensav
1209 ? 00:00:00 tracker-miner-f
1210 ? 00:00:00 nm-applet
1212 ? 00:00:00 seapplet
1214 ? 00:00:00 deja-dup-monito
1218 ? 00:00:00 notification-da
1221 ? 00:00:00 gnome-sound-app
1224 ? 00:00:00 evolution-alarm
1225 ? 00:00:00 abrt-applet
1228 ? 00:00:00 bluetooth-apple
1237 ? 00:00:00 tracker-store
1247 ? 00:00:01 vmtoolsd
1248 ? 00:00:00 gnome-fallback-
1306 ? 00:00:00 wnck-applet
1307 ? 00:00:00 clock-applet
1310 ? 00:00:00 notification-ar
1383 ? 00:00:00 sshd
1384 pts/0 00:00:00 bash
[root@plablinux01 ~]#
```

Send Signals to Processes

You can send termination signals to the running processes. To signals to processes, perform the following steps:

Step 1

List all the processes and locate cupsd with its process id by entering the following commands:

```
ps -e
```

```
root@plablinux01:~# ps -aux
 1210 ? 00:00:00 nm-applet
 1212 ? 00:00:00 seapplet
 1214 ? 00:00:00 deja-dup-monito
 1218 ? 00:00:00 notification-da
 1221 ? 00:00:00 gnome-sound-app
 1224 ? 00:00:00 evolution-alarm
 1225 ? 00:00:00 abrt-applet
 1228 ? 00:00:00 bluetooth-apple
 1237 ? 00:00:00 tracker-store
 1247 ? 00:00:02 vmtoolsd
 1248 ? 00:00:00 gnome-fallback-
 1306 ? 00:00:00 unck-applet
 1307 ? 00:00:00 clock-applet
 1310 ? 00:00:00 notification-ar
 1348 ? 00:00:00 sendmail
 1350 ? 00:00:00 sendmail
 1370 ? 00:00:00 kworker/0:0
 1379 ? 00:00:00 sshd
 1383 ? 00:00:00 sshd
 1384 pts/0 00:00:00 bash
 1441 pts/0 00:00:00 su
 1445 pts/0 00:00:00 bash
 1499 ? 00:00:00 cupsd
 1502 ? 00:00:00 cupsd
 1504 ? 00:00:00 cupsd
 1511 ? 00:00:00 cupsd
 1513 ? 00:00:00 cupsd
 1518 ? 00:00:00 cupsd
 1521 pts/0 00:00:00 top
 1531 ? 00:00:00 kworker/0:1
 1540 ? 00:00:00 kworker/0:2
 1544 pts/0 00:00:00 ps
[root@plablinux01 ~]# kill 1 1513
```

To kill the cupsd process with the process id 1513, enter the following command:

Note: In your system, the process id for cupsd may be different. Replace 1513 with the relevant process id.

```
kill 1 1513
```

The screenshot shows a Windows desktop environment with a terminal window titled "webRDP - Internet Explorer". The terminal window displays a list of processes from the command line. The user runs the command "ps -e" to list all processes, then uses "kill 1 1513" to terminate the process with ID 1513. The desktop taskbar at the bottom shows icons for Start, File Explorer, Task View, and Control Panel.

```
root@plablinux01:~# ps -e
1214 ? 00:00:00 deja-dup-monito
1218 ? 00:00:00 notification-da
1221 ? 00:00:00 gnome-sound-app
1224 ? 00:00:00 evolution-alarm
1225 ? 00:00:00 abrt-applet
1228 ? 00:00:00 bluetooth-apple
1237 ? 00:00:00 tracker-store
1247 ? 00:00:03 vmtoolsd
1248 ? 00:00:00 gnome-fallback-
1306 ? 00:00:00 wnck-applet
1307 ? 00:00:00 clock-applet
1310 ? 00:00:00 notification-ar
1348 ? 00:00:00 sendmail
1350 ? 00:00:00 sendmail
1370 ? 00:00:01 kworker/0:0
1379 ? 00:00:00 sshd
1383 ? 00:00:00 sshd
1384 pts/0 00:00:00 bash
1441 pts/0 00:00:00 su
1445 pts/0 00:00:00 bash
1499 ? 00:00:00 cupsd
1502 ? 00:00:00 cupsd
1504 ? 00:00:00 cupsd
1511 ? 00:00:00 cupsd
1513 ? 00:00:00 cupsd
1518 ? 00:00:00 cupsd
1521 pts/0 00:00:00 top
1545 ? 00:00:00 kworker/0:1
1547 ? 00:00:00 kworker/0:2
1551 ? 00:00:00 flush-253:1
1553 pts/0 00:00:00 ps
[root@plablinux01 ~]# kill 1 1513
[root@plablinux01 ~]#
```

To verify that the process cupsd is terminated, list the current running processes with the ps –e command. The process cupsd with the process id 1513 should not be present in the list.

```

root@plablinux01:~#
1210 ? 00:00:00 nm-applet
1212 ? 00:00:00 seapplet
1214 ? 00:00:00 deja-dup-monito
1218 ? 00:00:00 notification-da
1221 ? 00:00:00 gnome-sound-app
1224 ? 00:00:00 evolution-alarm
1225 ? 00:00:00 abrt-applet
1228 ? 00:00:00 bluetooth-apple
1237 ? 00:00:00 tracker-store
1247 ? 00:00:03 vmtoolsd
1248 ? 00:00:00 gnome-fallback-
1306 ? 00:00:00 unck-applet
1307 ? 00:00:00 clock-applet
1310 ? 00:00:00 notification-ar
1348 ? 00:00:00 sendmail
1350 ? 00:00:00 sendmail
1370 ? 00:00:01 kworker/0:0
1379 ? 00:00:00 sshd
1383 ? 00:00:00 sshd
1384 pts/0 00:00:00 bash
1441 pts/0 00:00:00 su
1445 pts/0 00:00:00 bash
1499 ? 00:00:00 cupsd
1502 ? 00:00:00 cupsd
1504 ? 00:00:00 cupsd
1511 ? 00:00:00 cupsd
1518 ? 00:00:00 cupsd
1521 pts/0 00:00:00 top
1545 ? 00:00:00 kworker/0:1
1547 ? 00:00:00 kworker/0:2
1551 ? 00:00:00 flush-253:1
1563 pts/0 00:00:00 ps
[root@plablinux01 ~]#

```

Note that you can terminate a process in different ways:

- 1 or SIGHUP hangup or disconnect the process
- 2 or SIGINT same as Ctrl+C interrupt
- 3 or SIGQUIT quit
- 9 or SIGKILL kill the process through a kernel call
- 15 or SIGTERM terminate a process 'nicely'. This is the DEFAULT signal.

Terms and Utilities

To understand various terms and utilities on this concept, perform the following steps:

Note: If you continuing from the previous task and have not logged out, then you can skip **Step 1**.

Step 1

To verify the hierarchy of the processes in the system, enter the following command:

pstree

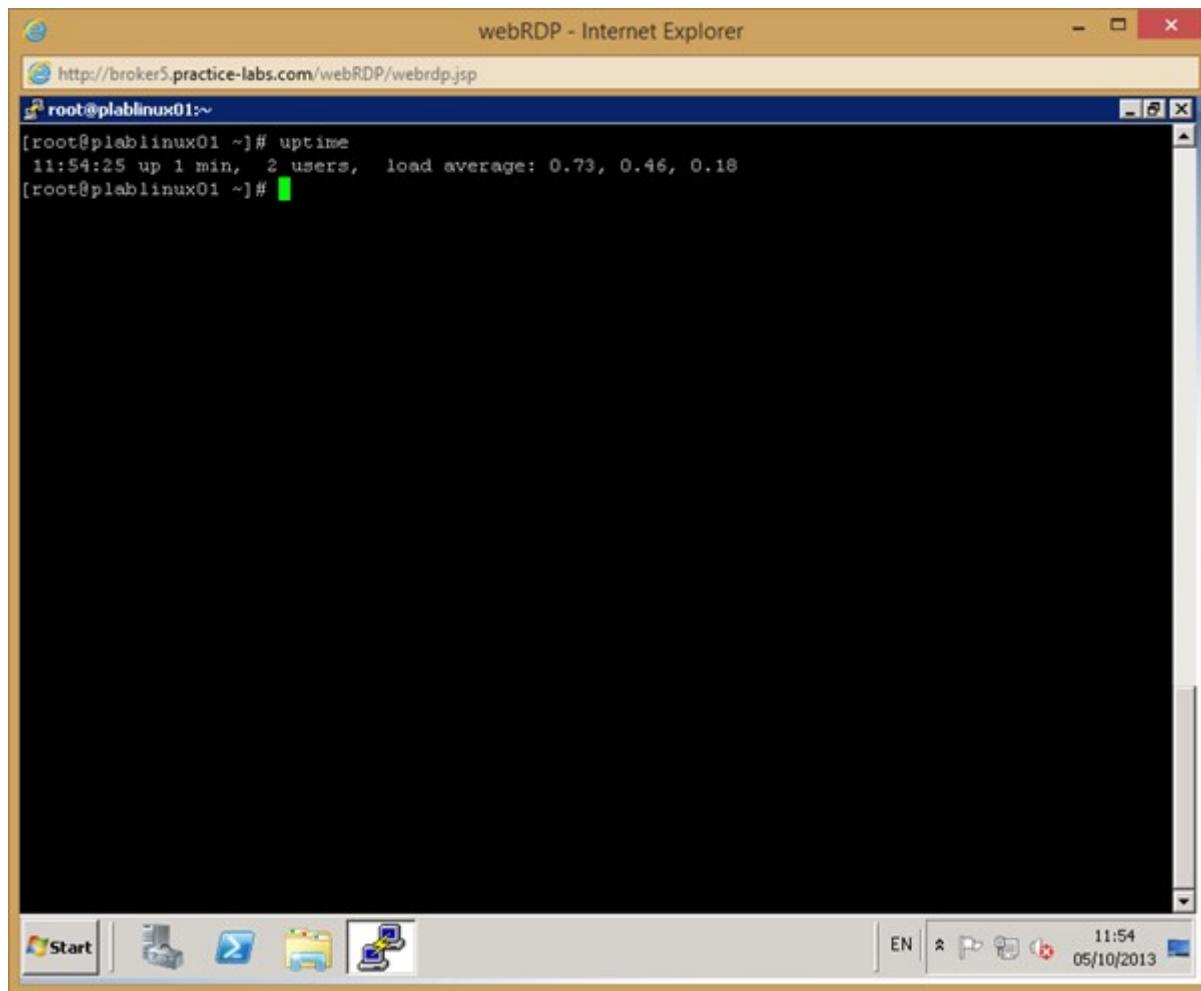
The screenshot shows a Windows desktop environment with a terminal window titled 'webRDP - Internet Explorer'. The window displays the output of the 'pstree' command, which shows a tree of processes on a Linux system. The root process is 'root@plablinux01:~'. Other visible processes include 'gnome-session', 'gvfsd-printer', 'gvfs-afc-volume', 'gvfs-fuse-daemon', 'gvfs-gphoto2-vo', 'gvfs-udisks2-vo', 'gvfsd', 'imsettings-daemon', 'celog', 'modem-manager', 'dispatcher', 'notification-ar', 'packagekitd', 'polkitd', 'pulseaudio', 'rsyslogd', 'rtkit-daemon', 'sendmail', 'sshd', 'system-setup-ke', 'systemd-journal', 'systemd-logind', 'udevd', 'udisksd', 'upowerd', 'vmtoolsd', and 'wnck-applet'. The terminal window has a blue title bar and a black background for the code. The desktop taskbar at the bottom shows icons for Start, Print, Mail, and File Explorer, along with system status indicators like battery level and date/time.

```
root@plablinux01:~  
âmeapplet  
âssh-agent  
âtracker-miner-fââ2*((tracker-miner-f)  
âtracker-storeââ6*((tracker-store)  
âvnconfig  
â3*((gnome-session)  
âgsd-printerââ1(gsd-printer)  
âgvfs-afc-volumeââ(gvfs-afc-volume)  
âgvfs-fuse-daemoââ3*((gvfs-fuse-daemon)  
âgvfs-gphoto2-vo  
âgvfs-udisks2-voââ(gvfs-udisks2-vo)  
â2*[gvfsd]  
âimsettings-daemââ2*((imsettings-daem)  
âcelog  
âmodem-manager  
âm-dispatcher.âââ10-sendmailâââsystemctl  
ânotification-arâââ(notification-ar)  
âpackagekitdââ2*((packagekitd)  
âpolkitdâââ(polkitd)  
â2*[pulseaudioâââ(pulseaudio)  
ârsyslogdââ3*((rsyslogd)  
ârtkit-daemonââ2*((rtkit-daemon)  
â2*[sendmail]  
âsshdâââsshdâââsshdâââbashâââsuâââbashâââpstree  
âsystem-setup-ke  
âsystemd-journal  
âsystemd-logind  
âudevdâââ2*(udevd)  
âudisksdâââ3*((udisksd)  
âupowerdâââ2*((upowerd)  
â2*[vmtoolsd]  
âwnck-appletâââ2*((wnck-applet)  
[root@plablinux01 ~]#
```

Step 2

To verify the system uptime, enter the following command:

uptime



Step 3

To check the memory usage and available memory, enter the following command:

```
free
```

The screenshot shows a Windows desktop environment. In the center is a terminal window titled "webRDP - Internet Explorer" running on "root@plablinux01". The window displays the output of several commands:

```
[root@plablinux01 ~]# uptime
11:54:52 up 2 min, 2 users, load average: 0.48, 0.42, 0.17
[root@plablinux01 ~]# free
total used free shared buffers cached
Mem: 503984 496892 7092 0 16988 132572
/+ buffers/cache: 347332 156652
Swap: 1015804 9332 1006472
[root@plablinux01 ~]#
```

The desktop taskbar at the bottom includes icons for Start, File Explorer, Task View, and Control Panel. The system tray shows the date (05/10/2013) and time (11:54).

Leave the devices you have powered on in their current state and proceed to the next exercise.

Exercise 3 - Use Streams Pipes and Redirects

In this exercise you will understand how use streams, pipes, and redirects. Please refer to your course material or use your favourite search engine to research **Use Streams, Pipes And Redirects in Linux.**

In this exercise, you will configure:

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

Redirecting Standard Input, Standard Output and Standard Error

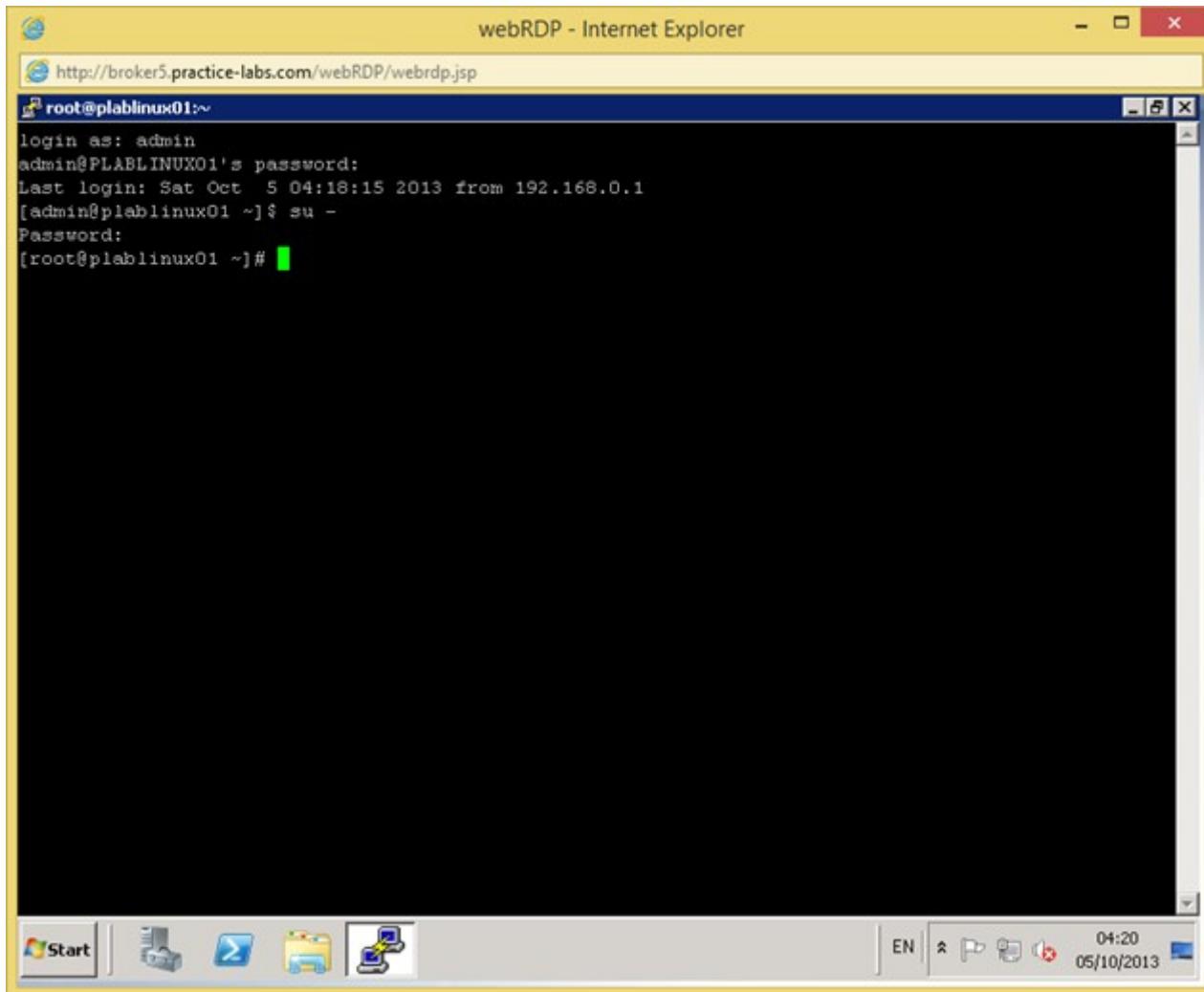
For each command that you execute in a terminal window, different streams are used by Linux to get input, output, or to generate an error. Each type of stream is defined with a value.

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

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

Step 1

From the command console using the root privileges prompt, defined by the # symbol.



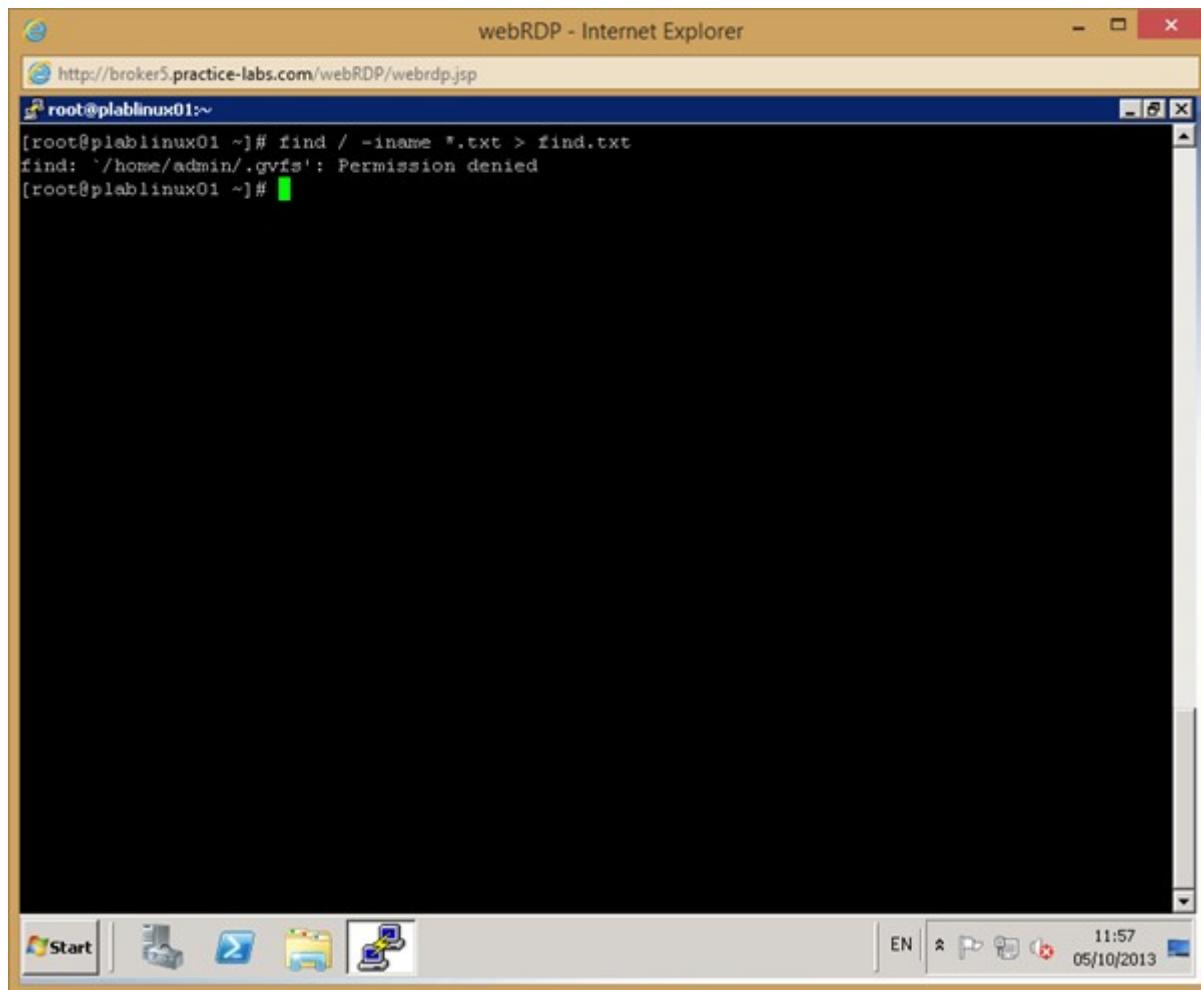
Clear the screen by entering the following command:

```
clear
```

Step 2

To redirect the output to a file rather than the screen, enter the following command:

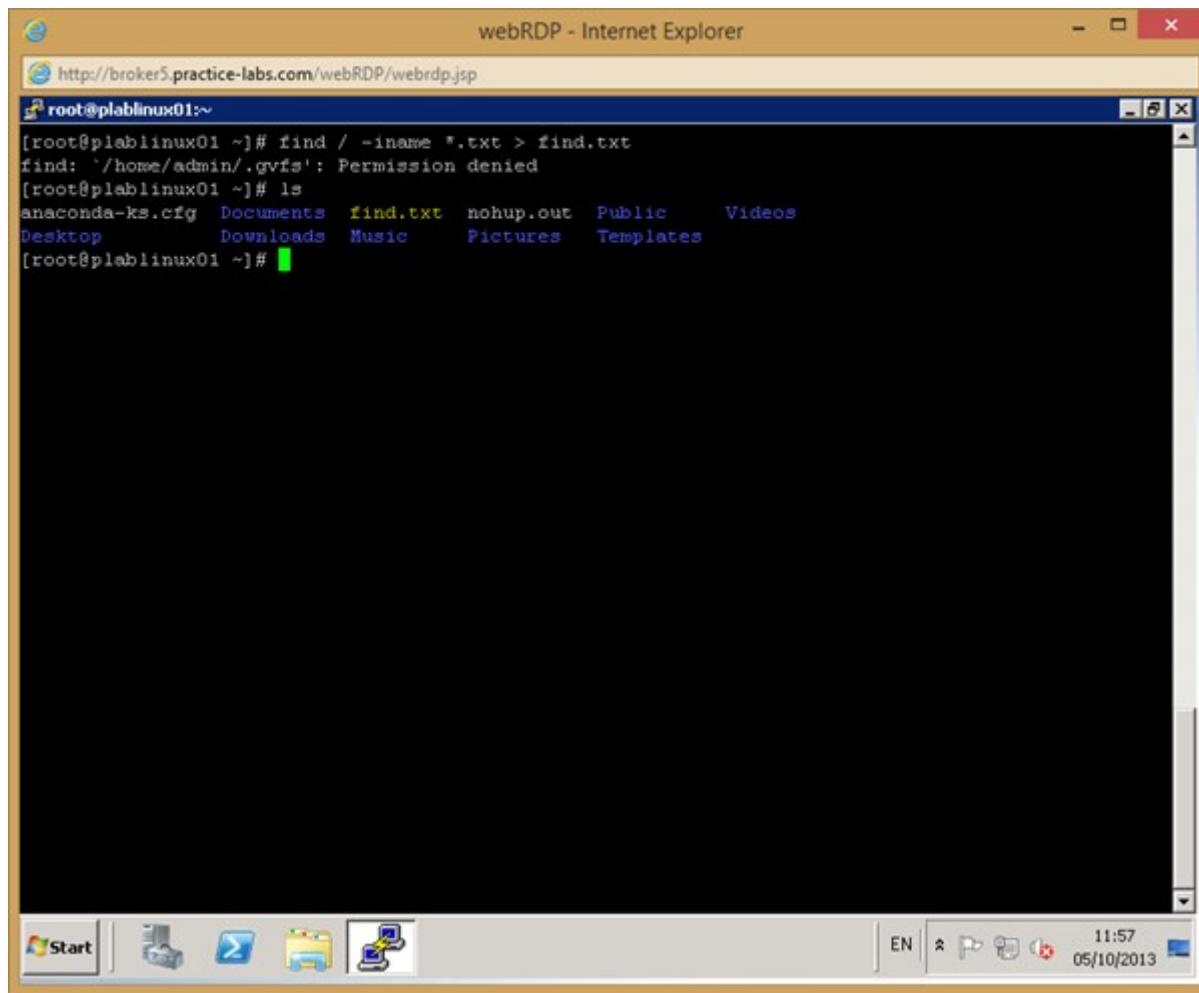
```
find / -iname *.txt > find.txt
```



Note: Due to permissions, you might be flagged with an error but the file will be generated.

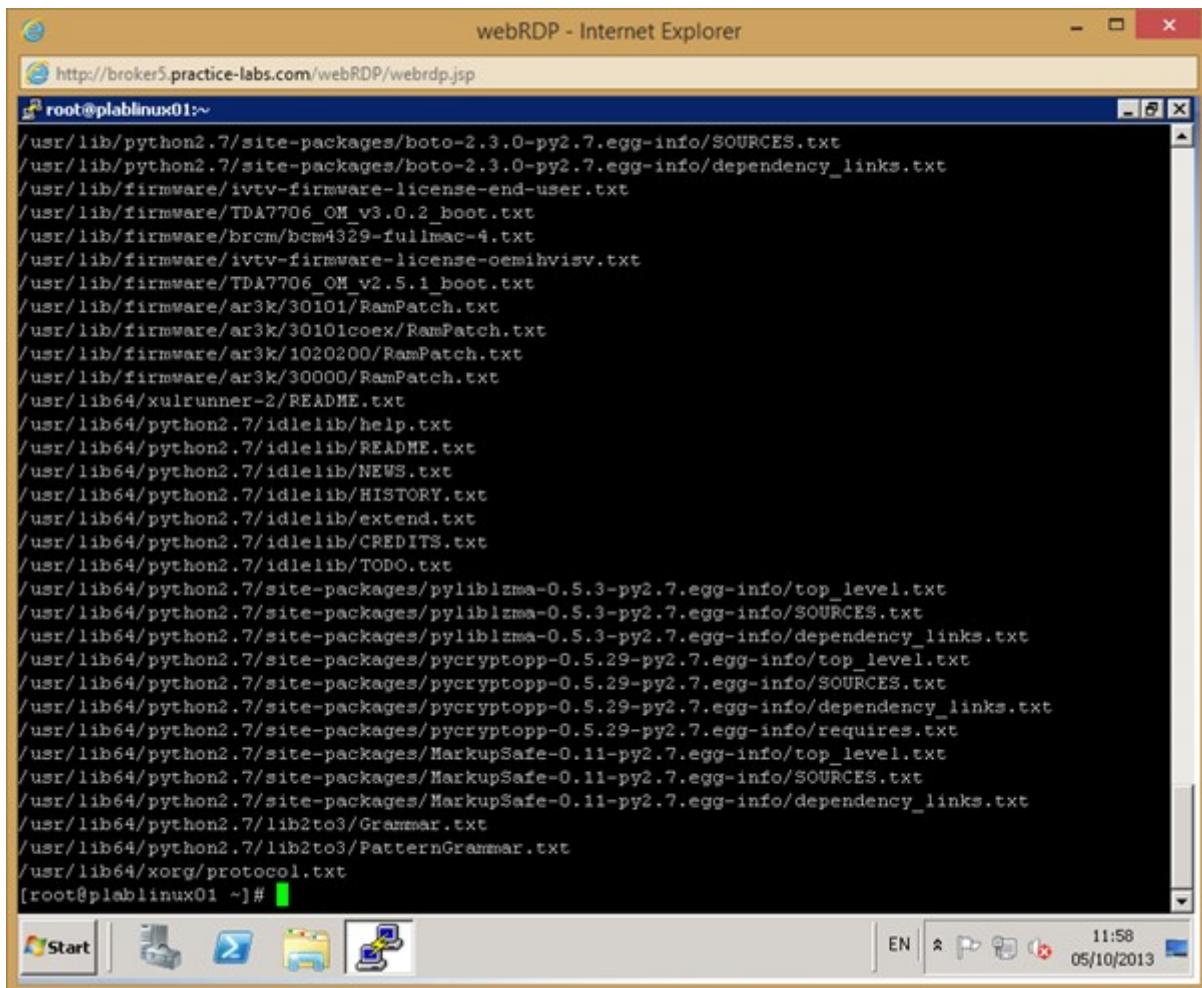
Let's verify if the file exists. Enter the following command:

```
ls
```



Note that the file is generated. 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
```



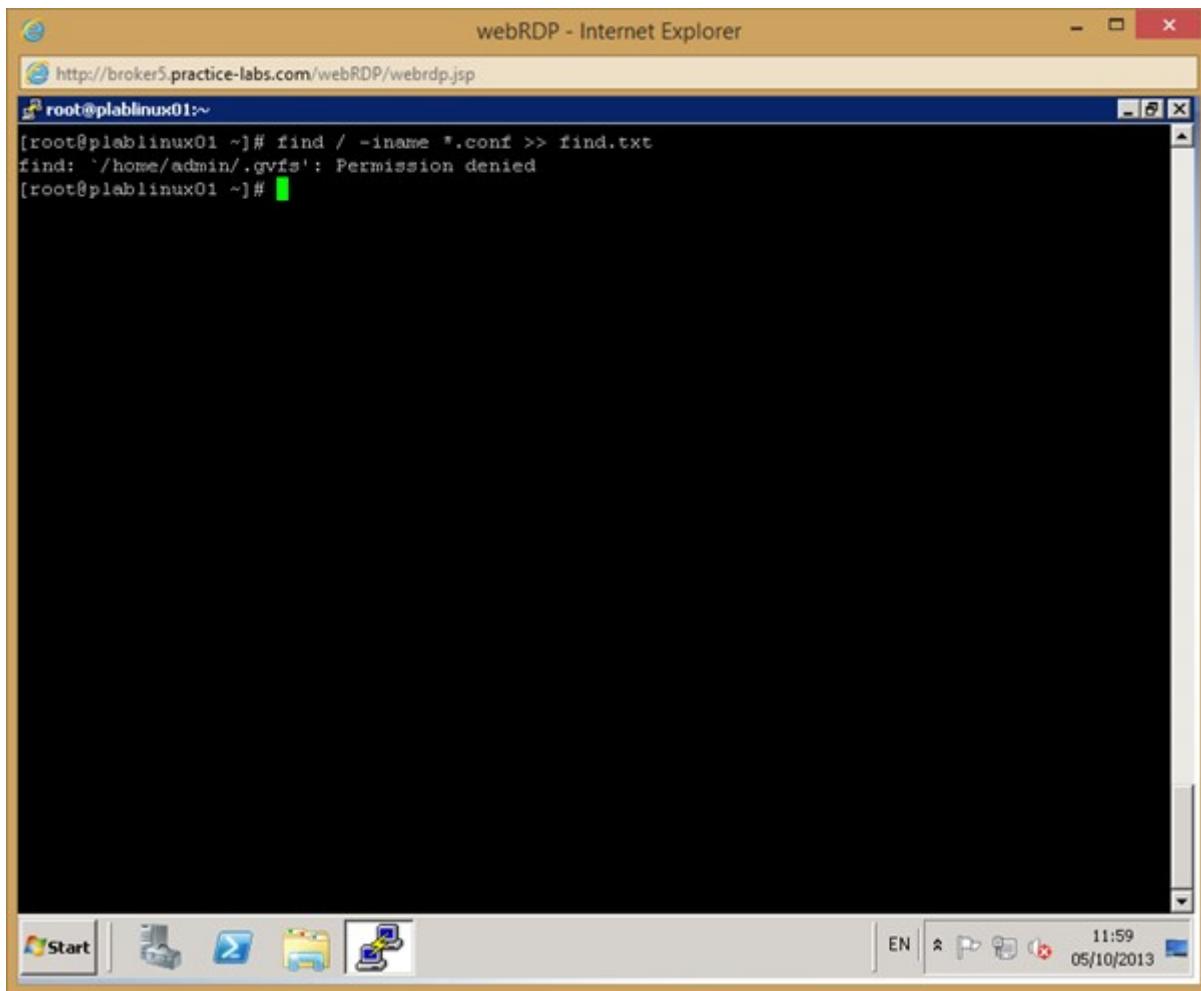
The screenshot shows a Windows desktop environment with a taskbar at the bottom. On the taskbar, there is a Start button, a printer icon, a file explorer icon, and a desktop icon. The system tray shows the date as 05/10/2013 and the time as 11:58. Above the taskbar, a window titled "webRDP - Internet Explorer" is open, displaying a command-line session. The session shows the user is root on a plablinux01 machine. The command entered is "ls /". The output lists numerous files and directories starting with "/usr/lib/python2.7/site-packages/boto-2.3.0-py2.7.egg-info/" and ending with "/root@plablinux01 ~]#".

```
root@plablinux01:~ ls /  
/usr/lib/python2.7/site-packages/boto-2.3.0-py2.7.egg-info/SOURCES.txt  
/usr/lib/python2.7/site-packages/boto-2.3.0-py2.7.egg-info/dependency_links.txt  
/usr/lib/firmware/ivtv-firmware-license-end-user.txt  
/usr/lib/firmware/TDA7706_OM_v3.0.2_boot.txt  
/usr/lib/firmware/brcm/bcm4329-fujimac-4.txt  
/usr/lib/firmware/ivtv-firmware-license-oemihvisv.txt  
/usr/lib/firmware/TDA7706_OM_v2.5.1_boot.txt  
/usr/lib/firmware/ar3k/30101/RamPatch.txt  
/usr/lib/firmware/ar3k/30101coex/RamPatch.txt  
/usr/lib/firmware/ar3k/1020200/RamPatch.txt  
/usr/lib/firmware/ar3k/30000/RamPatch.txt  
/usr/lib64/xulrunner-2/README.txt  
/usr/lib64/python2.7/idlelib/help.txt  
/usr/lib64/python2.7/idlelib/README.txt  
/usr/lib64/python2.7/idlelib/NEWS.txt  
/usr/lib64/python2.7/idlelib/HISTORY.txt  
/usr/lib64/python2.7/idlelib/extend.txt  
/usr/lib64/python2.7/idlelib/CREDITS.txt  
/usr/lib64/python2.7/idlelib/TODO.txt  
/usr/lib64/python2.7/site-packages/pyliblzma-0.5.3-py2.7.egg-info/top_level.txt  
/usr/lib64/python2.7/site-packages/pyliblzma-0.5.3-py2.7.egg-info/SOURCES.txt  
/usr/lib64/python2.7/site-packages/pyliblzma-0.5.3-py2.7.egg-info/dependency_links.txt  
/usr/lib64/python2.7/site-packages/pycryptopp-0.5.29-py2.7.egg-info/top_level.txt  
/usr/lib64/python2.7/site-packages/pycryptopp-0.5.29-py2.7.egg-info/SOURCES.txt  
/usr/lib64/python2.7/site-packages/pycryptopp-0.5.29-py2.7.egg-info/dependency_links.txt  
/usr/lib64/python2.7/site-packages/pycryptopp-0.5.29-py2.7.egg-info/requirements.txt  
/usr/lib64/python2.7/site-packages/MarkupSafe-0.11-py2.7.egg-info/top_level.txt  
/usr/lib64/python2.7/site-packages/MarkupSafe-0.11-py2.7.egg-info/SOURCES.txt  
/usr/lib64/python2.7/site-packages/MarkupSafe-0.11-py2.7.egg-info/dependency_links.txt  
/usr/lib64/python2.7/lib2to3/Grammar.txt  
/usr/lib64/python2.7/lib2to3/PatternGrammar.txt  
/usr/lib64/xorg/protocol.txt  
[root@plablinux01 ~]#
```

Step 3

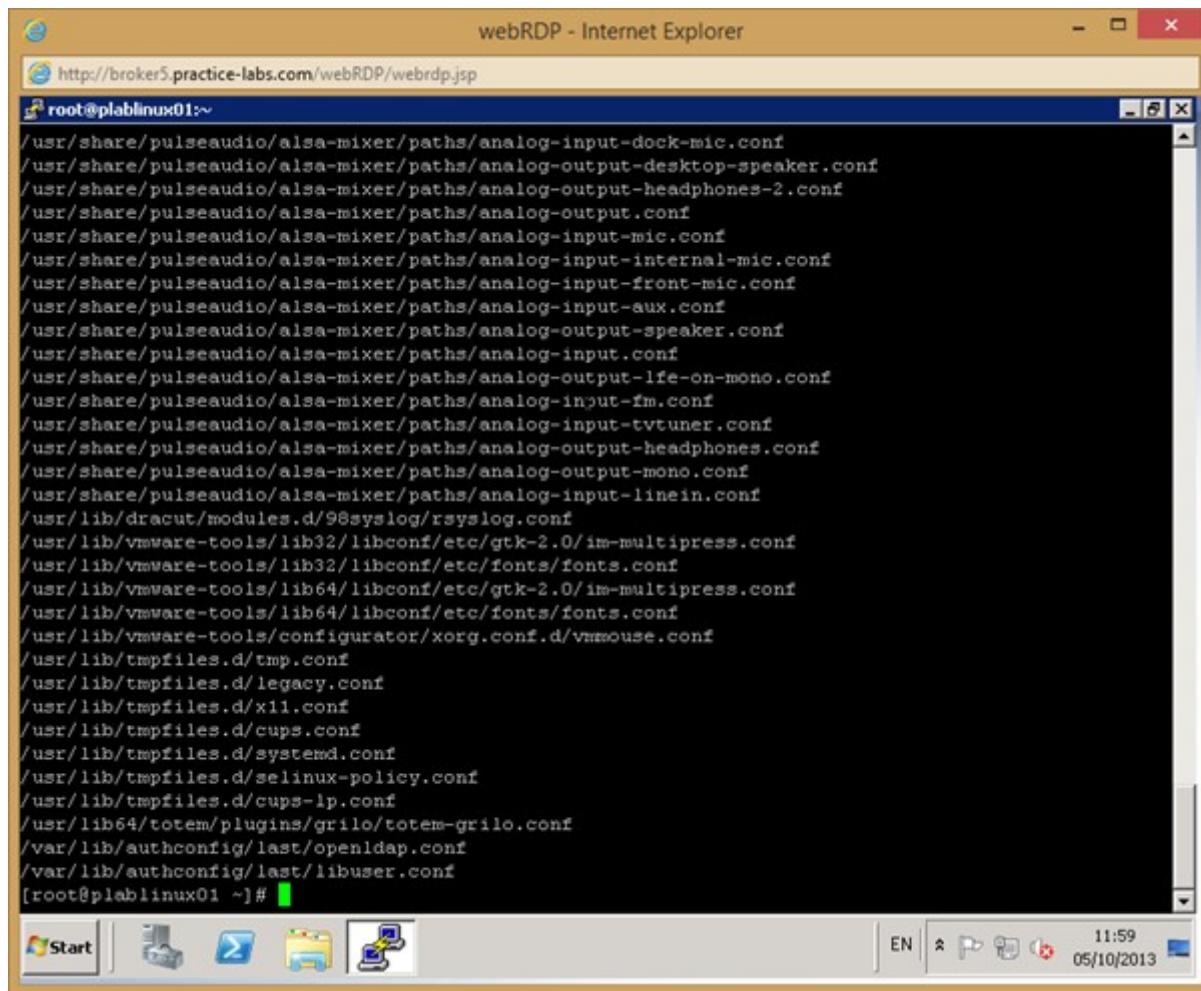
You can also append more information into the same file if you do not want to create a new file. To do this, enter the following command:

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



View the contents of the file by using the cat command.

```
cat find.txt
```

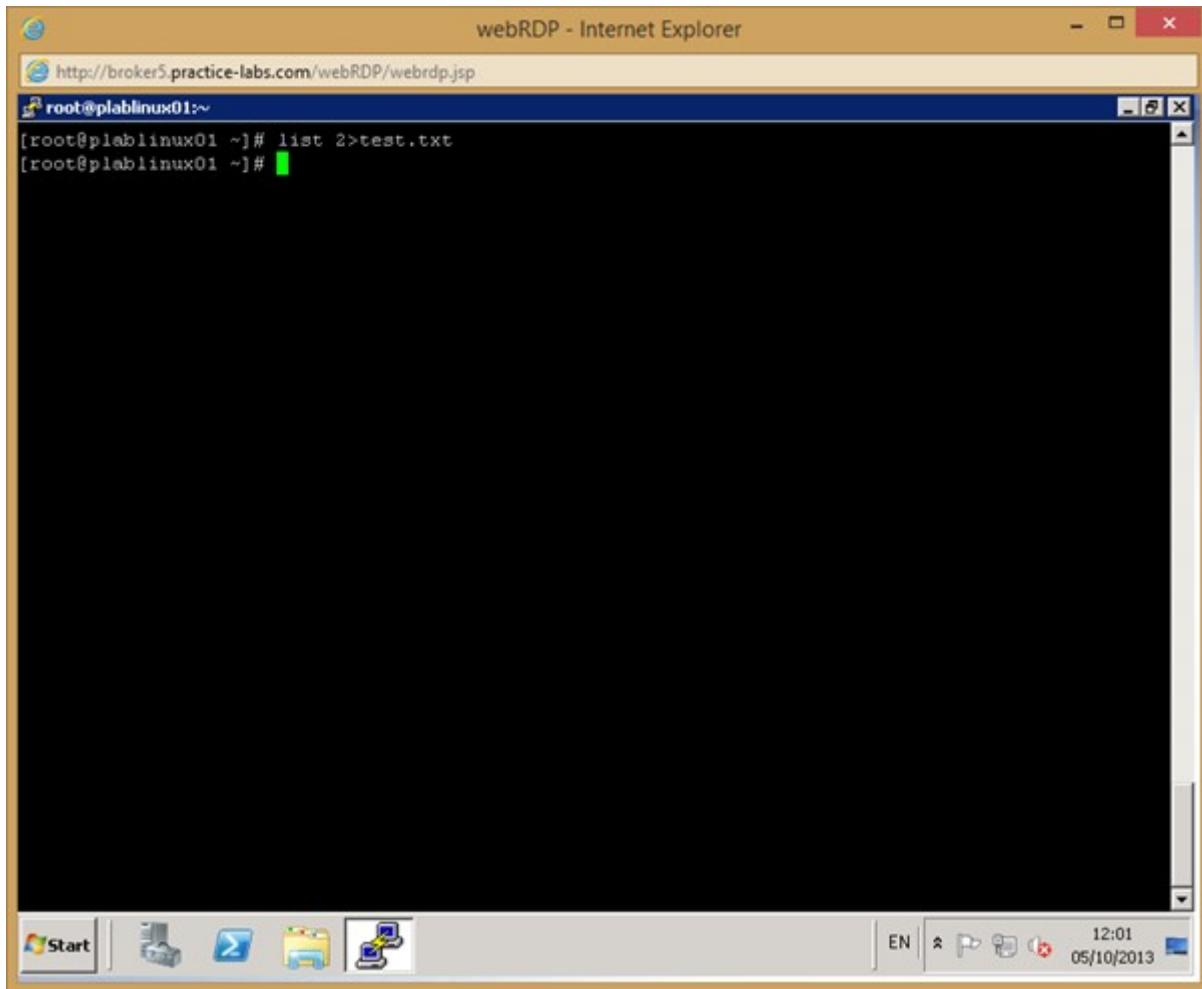


The screenshot shows a terminal window titled "webRDP - Internet Explorer" running on a Linux system. The command entered is "ls /usr/share/pulseaudio/alsa-mixer/paths/*". The output lists numerous configuration files related to audio paths, including "analog-input-dock-mic.conf", "analog-output-desktop-speaker.conf", "analog-output-headphones-2.conf", "analog-output.conf", "analog-input-mic.conf", "analog-input-internal-mic.conf", "analog-input-front-mic.conf", "analog-input-aux.conf", "analog-output-speaker.conf", "analog-input.conf", "analog-output-lfe-on-mono.conf", "analog-input-fm.conf", "analog-input-tvtuner.conf", "analog-output-headphones.conf", "analog-output-mono.conf", "analog-input-linein.conf", "98syslog.rsyslog.conf", "libconf/etc/gtk-2.0/im-multipress.conf", "libconf/etc/fonts/fonts.conf", "libconf/etc/gtk-2.0/im-multipress.conf", "libconf/etc/fonts/fonts.conf", "configurator/xorg.conf.d/vmmouse.conf", "tmpfiles.d/tmp.conf", "tmpfiles.d/legacy.conf", "tmpfiles.d/x11.conf", "tmpfiles.d/cups.conf", "tmpfiles.d/systemd.conf", "selinux-policy.conf", "cups-lp.conf", "totem/plugins/grilo/totem-grilo.conf", "authconfig/last/openldap.conf", and "authconfig/last/libuser.conf". The terminal prompt is "[root@plablinux01 ~]#".

Step 4

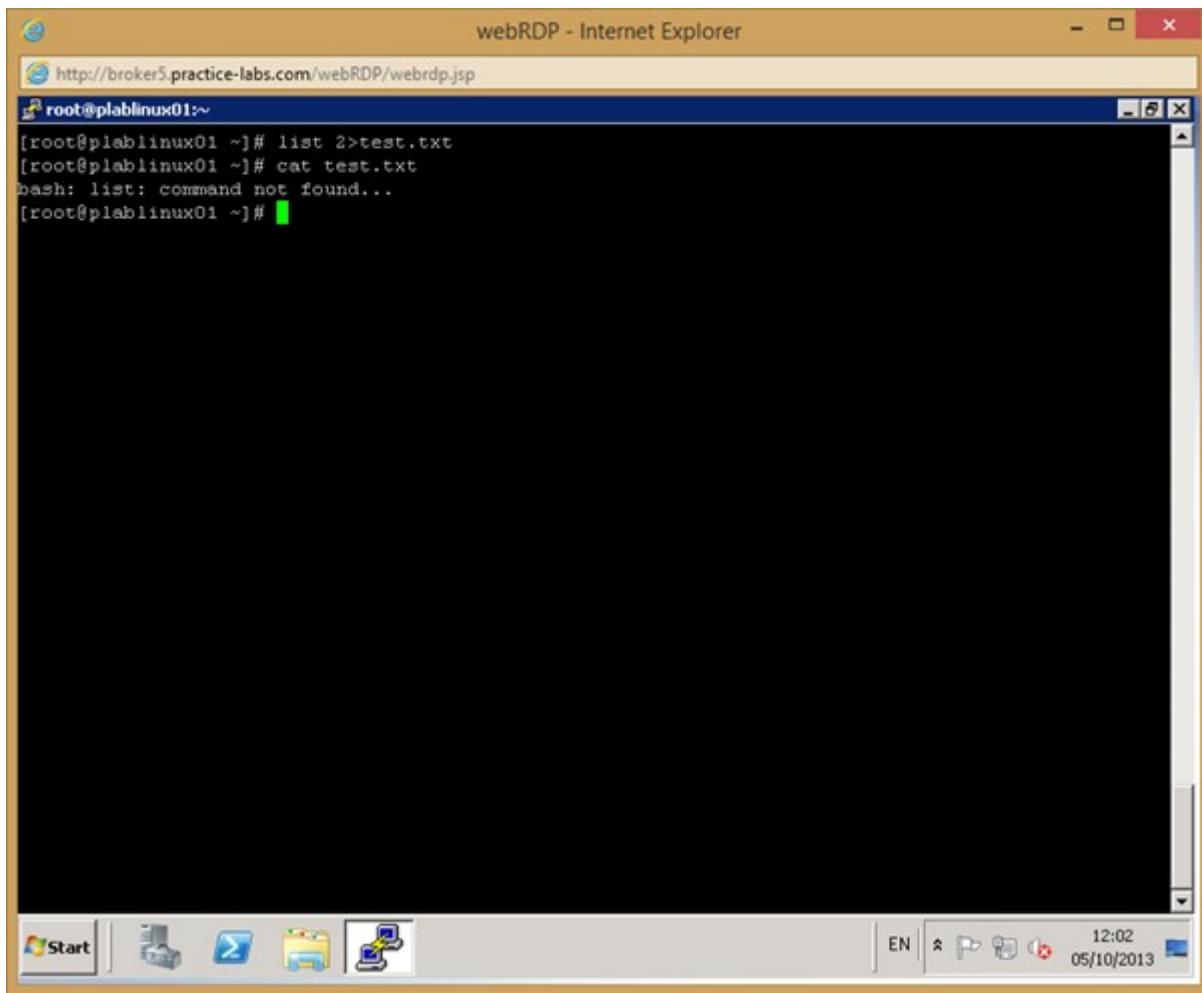
You can redirect the errors to a file just like you redirect the output. In this task, you will use a command that does not exist so that the error can be redirected to a file. To do this, enter the following command:

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



View the text.txt file.

```
cat test.txt
```



Note that this file contains the error.

Step 5

You can also combine the output and error or simply two outputs using '&'. This example demonstrates showing one output on the screen and another to the file To do this, enter the following command:

```
ls & ls -l >test.txt
```

The screenshot shows a Windows desktop environment. At the top, there is a taskbar with several icons: Start, Control Panel, File Explorer, Task View, and others. The system tray shows the date and time as 05/10/2013 at 12:03. Above the taskbar, a window titled "webRDP - Internet Explorer" is open. The address bar shows the URL "http://broker5.practice-labs.com/webRDP/webrdp.jsp". The main content area of the browser displays a terminal session. The terminal output is as follows:

```
[root@plablinux01 ~]# list 2>test.txt
[root@plablinux01 ~]# cat test.txt
bash: list: command not found...
[root@plablinux01 ~]# ls & ls -l >test.txt
[1] 1537
anaconda-ks.cfg  Documents  find.txt  nohup.out  Public      test.txt
Desktop          Downloads  Music     Pictures   Templates  Videos
[1]+  Done                    ls --color=auto
[root@plablinux01 ~]#
```

When you view the **test.txt** file, you will find the following output as shown in the graphic below.

```
[root@piablinux01 ~]# list 2>test.txt
[root@piablinux01 ~]# cat test.txt
bash: list: command not found...
[root@piablinux01 ~]# ls & ls -l >test.txt
[1] 1537
anaconda-ks.cfg  Documents  find.txt  nohup.out  Public    test.txt
Desktop        Downloads   Music     Pictures   Templates  Videos
[1]+  Done                  ls --color=auto
[root@piablinux01 ~]#
[root@piablinux01 ~]# cat test.txt
total 9872
-rw-----. 1 root root    1042 Oct  9  2012 anaconda-ks.cfg
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Desktop
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Documents
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Downloads
-rw-r--r--. 1 root root 10063901 Oct  5 11:59 find.txt
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Music
-rw-----. 1 root root      0 Oct  5 11:21 nohup.out
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Pictures
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Public
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Templates
-rw-r--r--. 1 root root      0 Oct  5 12:03 test.txt
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Videos
[root@piablinux01 ~]#
```

Pipe the Output of One Command to The Input of Another Command

Pipes provide a method to apply multiple commands concurrently. To pipe the output of one command to the input of another command, perform the following steps:

Note: If you continuing from the previous task and have not logged out, then you can skip **Step 1**.

Step 1

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

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

The screenshot shows a Windows desktop environment. In the center is a terminal window titled "webRDP - Internet Explorer" running under "root@plablinux01:~". The window displays the output of the "ls -l" command, listing various system files and devices. The output is as follows:

```
total 0
crw-rw----. 1 root video    10, 175 Oct  5 11:52 eppgارت
crw-----. 1 root root     10, 235 Oct  5 11:52 autoofs
drwxr-xr-x. 2 root root      360 Oct  5 11:52 block
drwxr-xr-x. 2 root root      120 Oct  5 11:52 bsg
c-----. 1 root root     10, 234 Oct  5 11:52 btrfs-control
lrwxrwxrwx. 1 root root       3 Oct  5 11:52 cdrom -> sr0
drwxr-xr-x. 2 root root     2680 Oct  5 11:52 char
crw-----. 1 root root       5,  1 Oct  5 11:52 console
lrwxrwxrwx. 1 root root      11 Oct  5 11:52 core -> /proc/kcore
drwxr-xr-x. 3 root root      80 Oct  5 11:52 cpu
crw-----. 1 root root     10,  62 Oct  5 11:52 cpu_dma_latency
drwxr-xr-x. 6 root root     120 Oct  5 11:52 disk
brw-rw----. 1 root disk    253,   0 Oct  5 11:52 dm-0
brw-rw----. 1 root disk    253,   1 Oct  5 11:52 dm-1
drwxr-xr-x. 2 root root      80 Oct  5 11:52 dri
lrwxrwxrwx. 1 root root       3 Oct  5 11:52 fb -> fb0
crw-rw----. 1 root video    29,   0 Oct  5 11:52 fb0
lrwxrwxrwx. 1 root root      13 Oct  5 11:52 fd -> /proc/self/fd
crw-rw-rw-. 1 root root       1,   7 Oct  5 11:52 full
crw-rw-rw-. 1 root root     10, 229 Oct  5 11:52 fuse
crw-----. 1 root root     10, 228 Oct  5 11:52 hpet
drwxr-xr-x. 2 root root       0 Oct  5 11:52 hugepages
prv-----. 1 root root       0 Oct  5 11:52 initctl
drwxr-xr-x. 3 root root     160 Oct  5 11:52 input
crw-----. 1 root root       1,  11 Oct  5 11:52 kmsg
srw-rw-rw-. 1 root root       0 Oct  5 11:52 log
brw-rw----. 1 root disk     7,   0 Oct  5 11:52 loop0
brw-rw----. 1 root disk     7,   1 Oct  5 11:52 loop1
brw-rw----. 1 root disk     7,   2 Oct  5 11:52 loop2
brw-rw----. 1 root disk     7,   3 Oct  5 11:52 loop3
brw-rw----. 1 root disk     7,   4 Oct  5 11:52 loop4
--More--
```

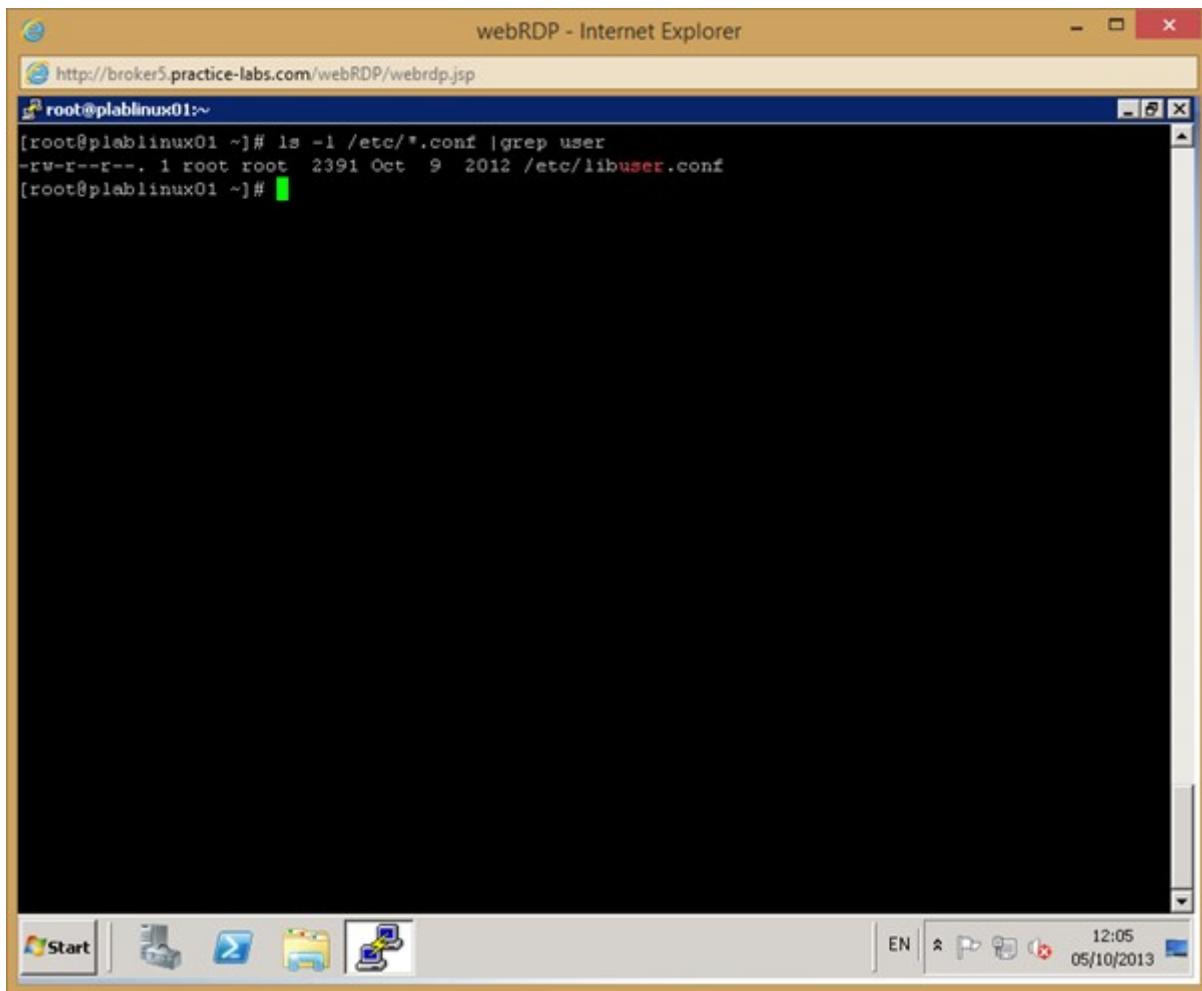
You can continue to press Enter to view the complete output or terminate ‘**CTRL + c**’ to terminate the output.

Use the Output of One Command as Arguments to Another Command

Step 1

Enter the following command:

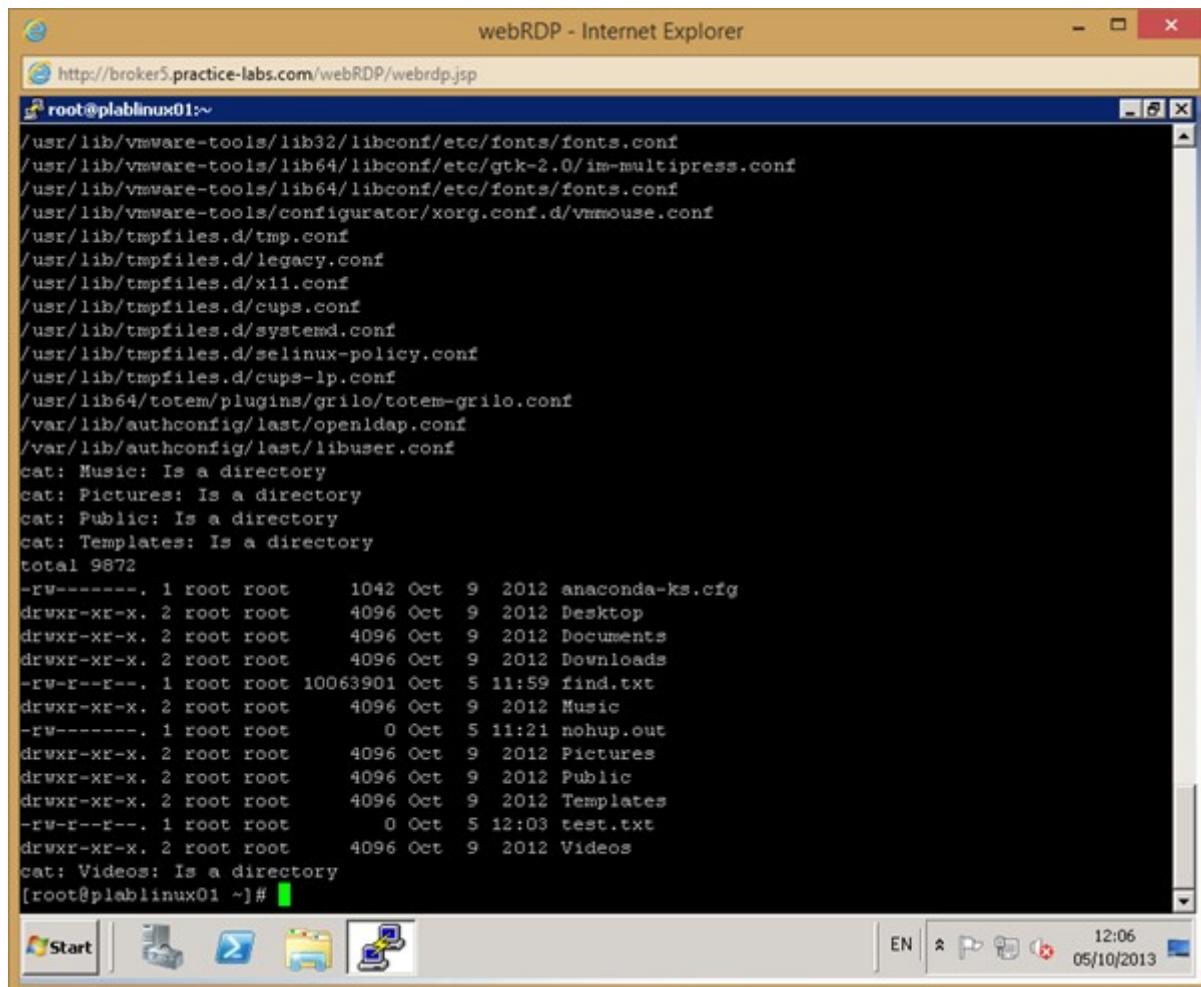
```
ls -l /etc/*.conf | grep user
```



Step 2

The xargs utility generates an argument list using standard input for a command. The **xargs** command is typically used with a **pipe** (|). To use xargs, enter the following command:

```
ls | xargs cat
```



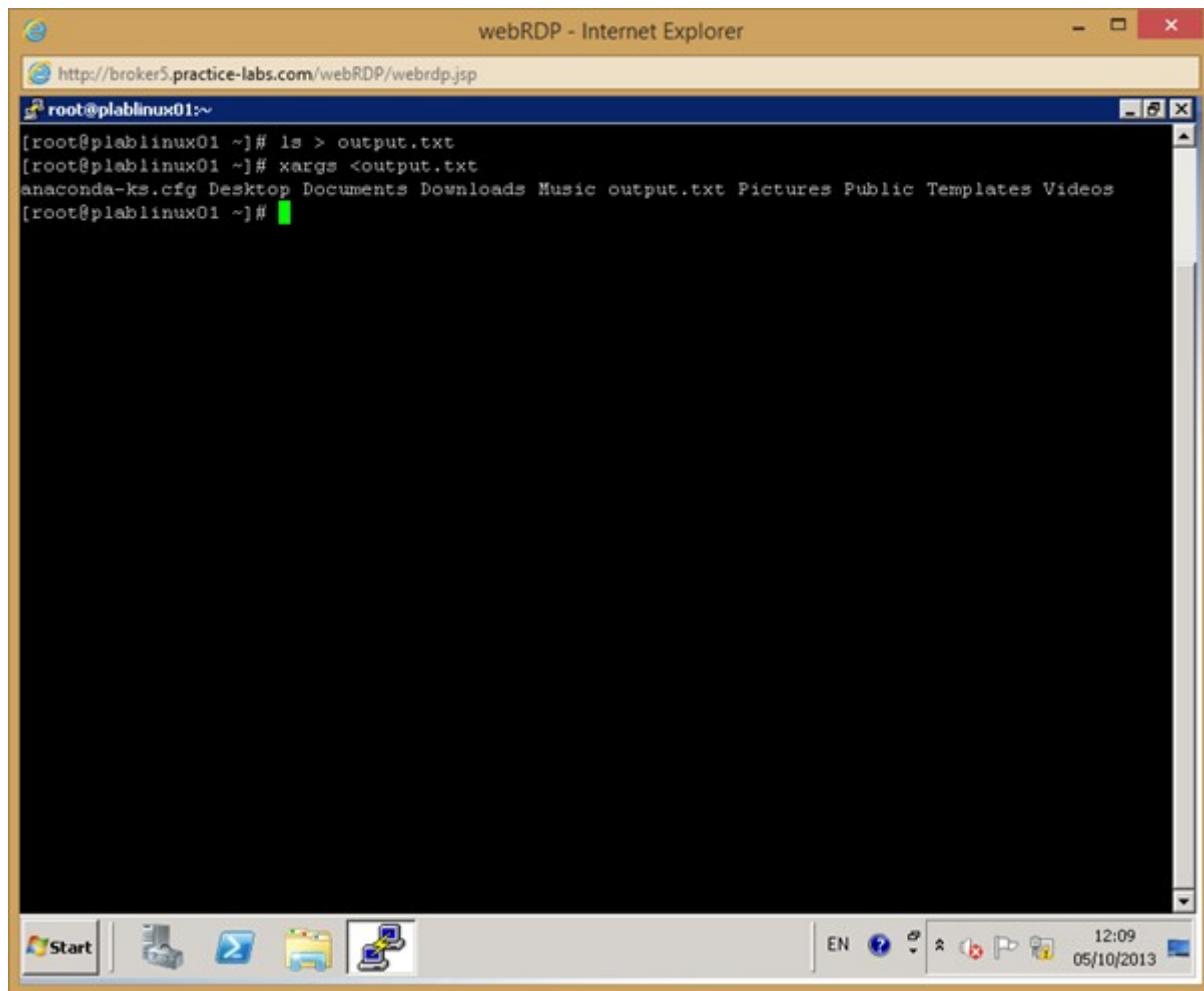
```
root@plablinux01:~#
/usr/lib/vmware-tools/lib32/libconf/etc/fonts/fonts.conf
/usr/lib/vmware-tools/lib64/libconf/etc/gtk-2.0/im-multipress.conf
/usr/lib/vmware-tools/lib64/libconf/etc/fonts/fonts.conf
/usr/lib/vmware-tools/configurator/xorg.conf.d/vmmouse.conf
/usr/lib/tmpfiles.d/tmp.conf
/usr/lib/tmpfiles.d/legacy.conf
/usr/lib/tmpfiles.d/x11.conf
/usr/lib/tmpfiles.d/cups.conf
/usr/lib/tmpfiles.d/systemd.conf
/usr/lib/tmpfiles.d/selinux-policy.conf
/usr/lib/tmpfiles.d/cups-lp.conf
/usr/lib64/totem/plugins/grilo/totem-grilo.conf
/var/lib/authconfig/last/openldap.conf
/var/lib/authconfig/last/libuser.conf
cat: Music: Is a directory
cat: Pictures: Is a directory
cat: Public: Is a directory
cat: Templates: Is a directory
total 9872
-rw-----. 1 root root    1042 Oct  9  2012 anaconda-ks.cfg
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Desktop
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Documents
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Downloads
-rw-r--r--. 1 root root 10063901 Oct  5 11:59 find.txt
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Music
-rw-----. 1 root root      0 Oct  5 11:21 nohup.out
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Pictures
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Public
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Templates
-rw-r--r--. 1 root root      0 Oct  5 12:03 test.txt
drwxr-xr-x. 2 root root    4096 Oct  9  2012 Videos
cat: Videos: Is a directory
[root@plablinux01 ~]#
```

Create a file with the following command:

```
ls > output.txt
```

Use the xargs command to view the output:

```
xargs <output.txt
```



Send Output to Both Stdout and a File

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

Step 1

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
```

```

root@plablinux01:~# ls
crw--w----. 1 root tty      4,   60 Oct  5 12:06 tty60
crw--w----. 1 root tty      4,   61 Oct  5 12:06 tty61
crw--w----. 1 root tty      4,   62 Oct  5 12:06 tty62
crw--w----. 1 root tty      4,   63 Oct  5 12:06 tty63
crw--w----. 1 root tty      4,   7 Oct   5 12:06 tty7
crw--w----. 1 root tty      4,   8 Oct   5 12:06 tty8
crw--w----. 1 root tty      4,   9 Oct   5 12:06 tty9
crw-rw----. 1 root dialout  4,   64 Oct  5 12:06 ttyS0
crw-rw----. 1 root dialout  4,   65 Oct  5 12:06 ttyS1
crw-rw----. 1 root dialout  4,   66 Oct  5 12:06 ttyS2
crw-rw----. 1 root dialout  4,   67 Oct  5 12:06 ttyS3
crw-r----. 1 root root     10,  223 Oct  5 12:06 uinput
crw-rw-rw-. 1 root root    1,   9 Oct   5 12:06 urandom
crw-----. 1 root root    252,  0 Oct   5 12:06 usbmon0
crw-r----. 1 root tty      7,   0 Oct   5 12:06 vcs
crw-rw----. 1 root tty      7,   1 Oct   5 12:06 vcs1
crw-rw----. 1 root tty      7,   2 Oct   5 12:06 vcs2
crw-rw----. 1 root tty      7,   3 Oct   5 12:06 vcs3
crw-rw----. 1 root tty      7,   4 Oct   5 12:06 vcs4
crw-rw----. 1 root tty      7,   5 Oct   5 12:06 vcs5
crw-rw----. 1 root tty      7,   6 Oct   5 12:06 vcs6
crw-rw----. 1 root tty      7,  128 Oct  5 12:06 vcsa
crw-rw----. 1 root tty      7,  129 Oct  5 12:06 vcsa1
crw-rw----. 1 root tty      7,  130 Oct  5 12:06 vcsa2
crw-rw----. 1 root tty      7,  131 Oct  5 12:06 vcsa3
crw-rw----. 1 root tty      7,  132 Oct  5 12:06 vcsa4
crw-rw----. 1 root tty      7,  133 Oct  5 12:06 vcsa5
crw-rw----. 1 root tty      7,  134 Oct  5 12:06 vcsa6
crw-----. 1 root root    10,  63 Oct  5 12:06 vga_arbiter
drwxr-xr-x. 2 root root     80 Oct  5 12:06 vg_plablinux01
c-----. 1 root root    10, 238 Oct  5 12:06 vhost-net
crw-rw-rw-. 1 root root    1,   5 Oct   5 12:06 zero
[root@plablinux01 ~]#

```

```

root@plablinux01:~# ls
anaconda-ks.cfg  Documents  Music      Pictures  Templates
Desktop          Downloads  output.txt  Public    Videos
[root@plablinux01 ~]#

```

View the contents of the output.txt file using the ‘**cat**’ command.

```
root@plablinux01:~# ls -l
crw--w----. 1 root tty      4,   60 Oct  5 12:06 tty60
crw--w----. 1 root tty      4,   61 Oct  5 12:06 tty61
crw--w----. 1 root tty      4,   62 Oct  5 12:06 tty62
crw--w----. 1 root tty      4,   63 Oct  5 12:06 tty63
crw--w----. 1 root tty      4,   7 Oct   5 12:06 tty7
crw--w----. 1 root tty      4,   8 Oct   5 12:06 tty8
crw--w----. 1 root tty      4,   9 Oct   5 12:06 tty9
crw-rw----. 1 root dialout 4,   64 Oct  5 12:06 ttyS0
crw-rw----. 1 root dialout 4,   65 Oct  5 12:06 ttyS1
crw-rw----. 1 root dialout 4,   66 Oct  5 12:06 ttyS2
crw-rw----. 1 root dialout 4,   67 Oct  5 12:06 ttyS3
crw-r----. 1 root root     10,  223 Oct  5 12:06 uinput
crw-rw-rw-. 1 root root    1,   9 Oct   5 12:06 urandom
crw-----. 1 root root    252,  0 Oct   5 12:06 usbmon0
crw-rw----. 1 root tty      7,   0 Oct   5 12:06 vcs
crw-rw----. 1 root tty      7,   1 Oct   5 12:06 vcs1
crw-rw----. 1 root tty      7,   2 Oct   5 12:06 vcs2
crw-rw----. 1 root tty      7,   3 Oct   5 12:06 vcs3
crw-rw----. 1 root tty      7,   4 Oct   5 12:06 vcs4
crw-rw----. 1 root tty      7,   5 Oct   5 12:06 vcs5
crw-rw----. 1 root tty      7,   6 Oct   5 12:06 vcs6
crw-rw----. 1 root tty      7,  128 Oct  5 12:06 vcsa
crw-rw----. 1 root tty      7,  129 Oct  5 12:06 vcsa1
crw-rw----. 1 root tty      7,  130 Oct  5 12:06 vcsa2
crw-rw----. 1 root tty      7,  131 Oct  5 12:06 vcsa3
crw-rw----. 1 root tty      7,  132 Oct  5 12:06 vcsa4
crw-rw----. 1 root tty      7,  133 Oct  5 12:06 vcsa5
crw-rw----. 1 root tty      7,  134 Oct  5 12:06 vcsa6
crw-----. 1 root root    10,  63 Oct  5 12:06 vga_arbiter
drwxr-xr-x. 2 root root     80 Oct  5 12:06 vg_plablinux01
c-----. 1 root root    10, 238 Oct  5 12:06 vhost-net
crw-rw-rw-. 1 root root    1,   5 Oct   5 12:06 zero
[root@plablinux01 ~]#
```

Leave the devices you have powered on in their current state and proceed to the next exercise.

Exercise 4 - Modify Process Execution Priorities

In this exercise you will understand how to modify process execution priorities. Please refer to your course material or use your favourite search engine to research **Modify Process Execution Priorities in Linux**.

In this exercise, you will configure:

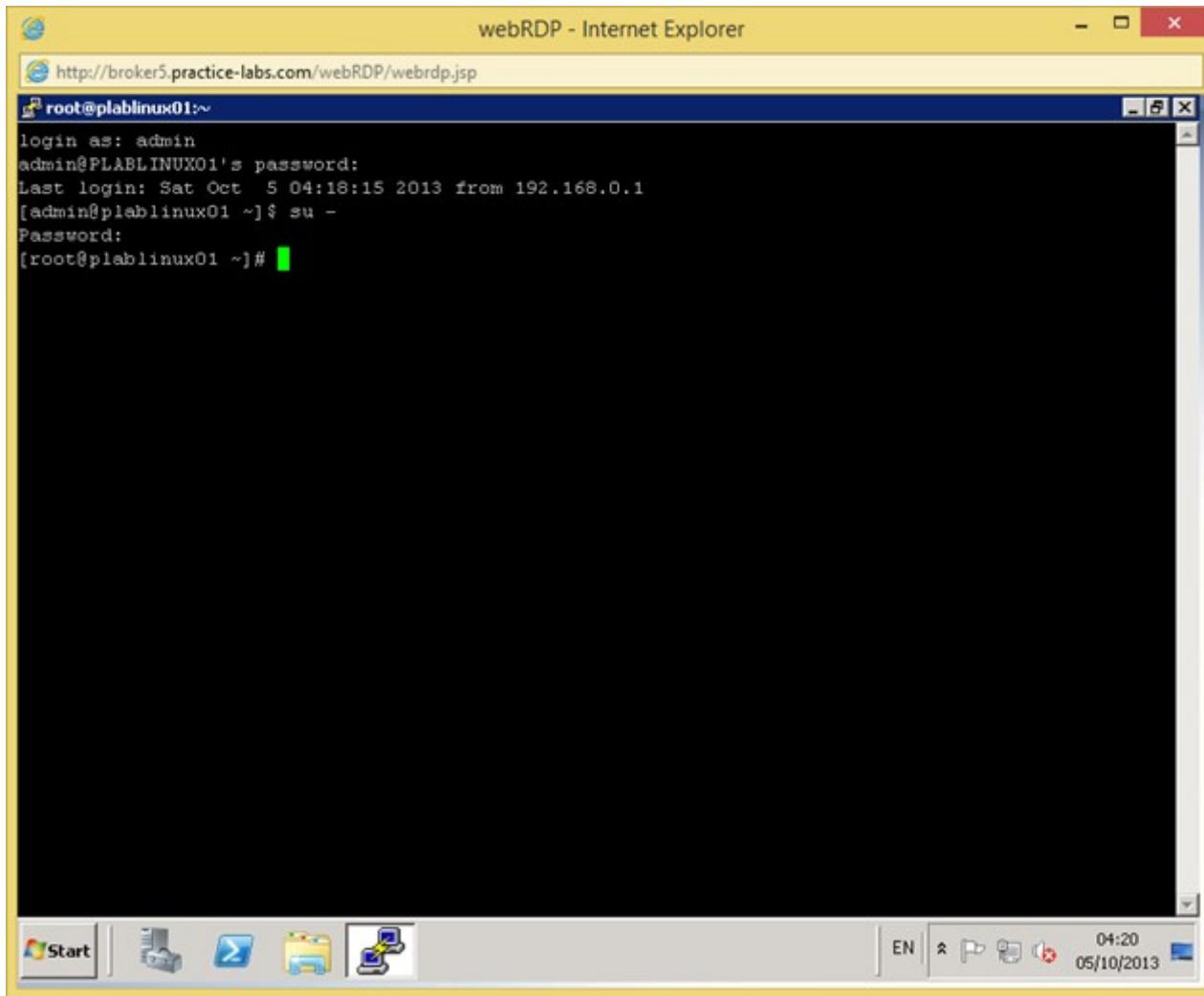
- Know the default priority of a job that is created
- Run a program with higher or lower priority than the default

- Change the priority of a running process

Know the Default Priority of a Job that is Created

Step 1

From the command console using the root privileges prompt, defined by the # symbol.



Let's first list all active processes that are running on the system. Enter the following command:

```
top
```

The screenshot shows a Windows desktop environment with a terminal window titled "webRDP - Internet Explorer". The window displays system monitoring information from the "top" command and a detailed process list from the "ps" command.

```

top - 12:11:43 up 5 min, 2 users, load average: 0.01, 0.06, 0.05
Tasks: 133 total, 1 running, 132 sleeping, 0 stopped, 0 zombie
Cpu(s): 2.9%us, 2.2%sy, 0.2%ni, 94.3%id, 0.3%wa, 0.1%hi, 0.1%si, 0.0%st
Mem: 503984k total, 492360k used, 11624k free, 16852k buffers
Swap: 1015804k total, 8664k used, 1007140k free, 130524k cached

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
 1 root 20 0 61872 16m 2028 S 0.0 3.3 0:01.74 systemd
 2 root 20 0 0 0 S 0.0 0.0 0:00.00 kthreadd
 3 root 20 0 0 0 S 0.0 0.0 0:00.00 ksoftirqd/0
 5 root 20 0 0 0 S 0.0 0.0 0:00.01 kworker/u:0
 5 root 20 0 0 0 S 0.0 0.0 0:00.01 kworker/u:0
 6 root RT 0 0 0 S 0.0 0.0 0:00.00 migration/0
 7 root RT 0 0 0 S 0.0 0.0 0:00.00 watchdog/0
 8 root 0 -20 0 0 S 0.0 0.0 0:00.00 cpuset
 9 root 0 -20 0 0 S 0.0 0.0 0:00.00 khelper
10 root 20 0 0 0 S 0.0 0.0 0:00.00 kdevtmpfs
11 root 0 -20 0 0 S 0.0 0.0 0:00.00 netns
12 root 20 0 0 0 S 0.0 0.0 0:00.00 sync_supers
13 root 20 0 0 0 S 0.0 0.0 0:00.00 bdi-default
14 root 0 -20 0 0 S 0.0 0.0 0:00.00 kintegrityd
15 root 0 -20 0 0 S 0.0 0.0 0:00.00 kblockd
16 root 0 -20 0 0 S 0.0 0.0 0:00.00 ata_sff
17 root 20 0 0 0 S 0.0 0.0 0:00.00 khubd
18 root 0 -20 0 0 S 0.0 0.0 0:00.00 md
25 root 20 0 0 0 S 0.0 0.0 0:00.00 kworker/u:1
41 root 20 0 0 0 S 0.0 0.0 0:00.06 kswapd0
42 root 25 5 0 0 S 0.0 0.0 0:00.00 ksmd
43 root 20 0 0 0 S 0.0 0.0 0:00.00 fsnotify_mark
44 root 0 -20 0 0 S 0.0 0.0 0:00.00 crypto
50 root 0 -20 0 0 S 0.0 0.0 0:00.00 kthrotld
51 root 20 0 0 0 S 0.0 0.0 0:00.02 kworker/0:1
52 root 20 0 0 0 S 0.0 0.0 0:00.01 scsi_eh_0

```

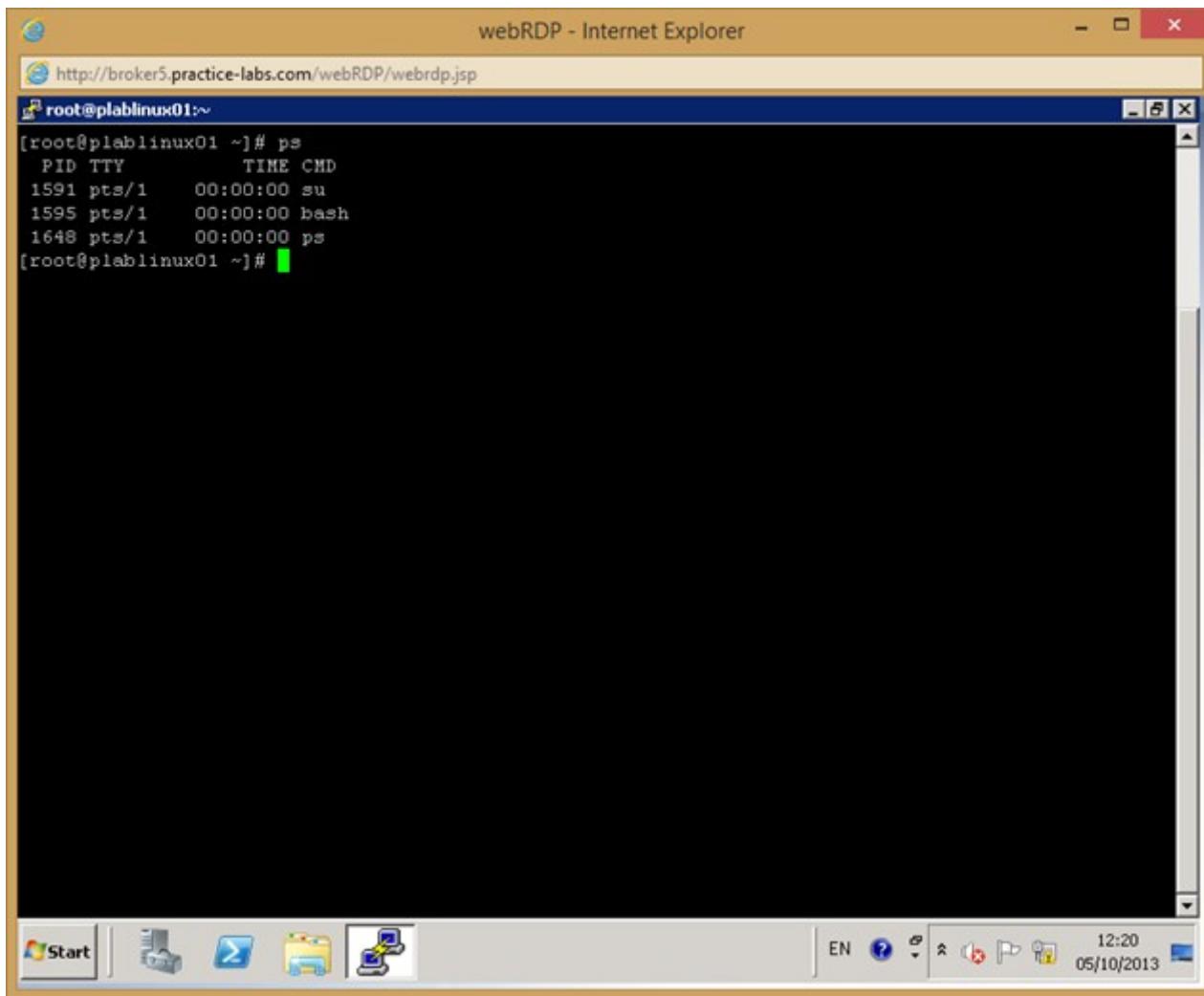
The terminal window also shows a Windows taskbar at the bottom with icons for Start, File Explorer, Task View, and others. The system tray shows the date and time as 12:11 05/10/2013.

Note that **NI** field displays the priority of a process that is running.

Step 2

Along with the **top** command, you can also use the **ps** command to display the process information. By default, the ps command displays limited information. Enter the following command:

```
ps
```



Note that the information is displayed for three specific processes: **su**, **bash**, and **ps**.

To display more information, like priority of a process, enter the following command:

```
ps -l
```

```
[root@plablinux01 ~]# ps
 PID TTY      TIME CMD
 1591 pts/1    00:00:00 su
 1595 pts/1    00:00:00 bash
 1648 pts/1    00:00:00 ps
[root@plablinux01 ~]# ps -l
F S  UID   PID  PPID C PRI  NI ADDR SZ WCHAN  TTY      TIME CMD
4 S     0 1591  1535 0  80    0 - 42048 wait    pts/1    00:00:00 su
4 S     0 1595  1591 0  80    0 - 28780 wait    pts/1    00:00:00 bash
0 R     0 1649  1595 0  80    0 - 28409 -      pts/1    00:00:00 ps
[root@plablinux01 ~]#
```

You can also displays the priorities for a specific user. To display priorities for a specific user, enter the following command:

```
ps -l -u admin
```

Note that the UID, which is the user ID, is 1000 for the admin user.

```

root@plablinux01:~#
1 S 1000 1137 1006 0 80 0 - 17171 poll_s ?
0 S 1000 1150 1 0 80 0 - 102578 poll_s ?
0 S 1000 1157 1006 0 80 0 - 202961 poll_s ?
1 S 1000 1166 1 0 80 0 - 98806 poll_s ?
0 S 1000 1173 1 0 80 0 - 116966 poll_s ?
0 S 1000 1178 1 0 80 0 - 64357 poll_s ?
0 S 1000 1186 1 0 80 0 - 61496 poll_s ?
0 S 1000 1189 1 0 80 0 - 42438 poll_s ?
0 S 1000 1199 1006 0 80 0 - 107214 poll_s ?
0 S 1000 1205 1006 0 80 0 - 147451 poll_s ?
0 S 1000 1210 1 0 80 0 - 34608 poll_s ?
0 S 1000 1212 1 0 80 0 - 64440 poll_s ?
0 S 1000 1217 1006 0 80 0 - 114390 poll_s ?
0 S 1000 1219 1006 0 99 19 - 133385 poll_s ?
0 S 1000 1221 1006 0 80 0 - 121695 poll_s ?
0 S 1000 1222 1006 0 80 0 - 62314 poll_s ?
0 S 1000 1225 1006 0 80 0 - 93023 poll_s ?
0 S 1000 1236 1006 0 80 0 - 106605 poll_s ?
0 S 1000 1239 1006 0 80 0 - 113625 poll_s ?
0 S 1000 1246 1006 0 80 0 - 138480 poll_s ?
0 S 1000 1250 1006 0 80 0 - 61847 poll_s ?
0 S 1000 1254 1006 0 80 0 - 107759 poll_s ?
0 S 1000 1256 1006 0 80 0 - 107355 poll_s ?
0 S 1000 1262 1 0 80 0 - 82171 poll_s ?
0 S 1000 1265 1006 0 80 0 - 122708 poll_s ?
0 S 1000 1313 1 0 80 0 - 129003 poll_s ?
0 S 1000 1314 1 0 80 0 - 129876 poll_s ?
0 S 1000 1321 1 0 80 0 - 107201 poll_s ?
5 S 1000 1384 1379 0 80 0 - 33075 poll_s ?
0 S 1000 1385 1384 0 80 0 - 28781 wait pts/0
5 S 1000 1534 1530 0 80 0 - 33075 poll_s ?
0 S 1000 1535 1534 0 80 0 - 28781 wait pts/1

```

[root@plablinux01 ~]#

Run a Program with Higher or Lower Priority than the Default

To run a program with higher or lower priority than the default, perform the following steps:

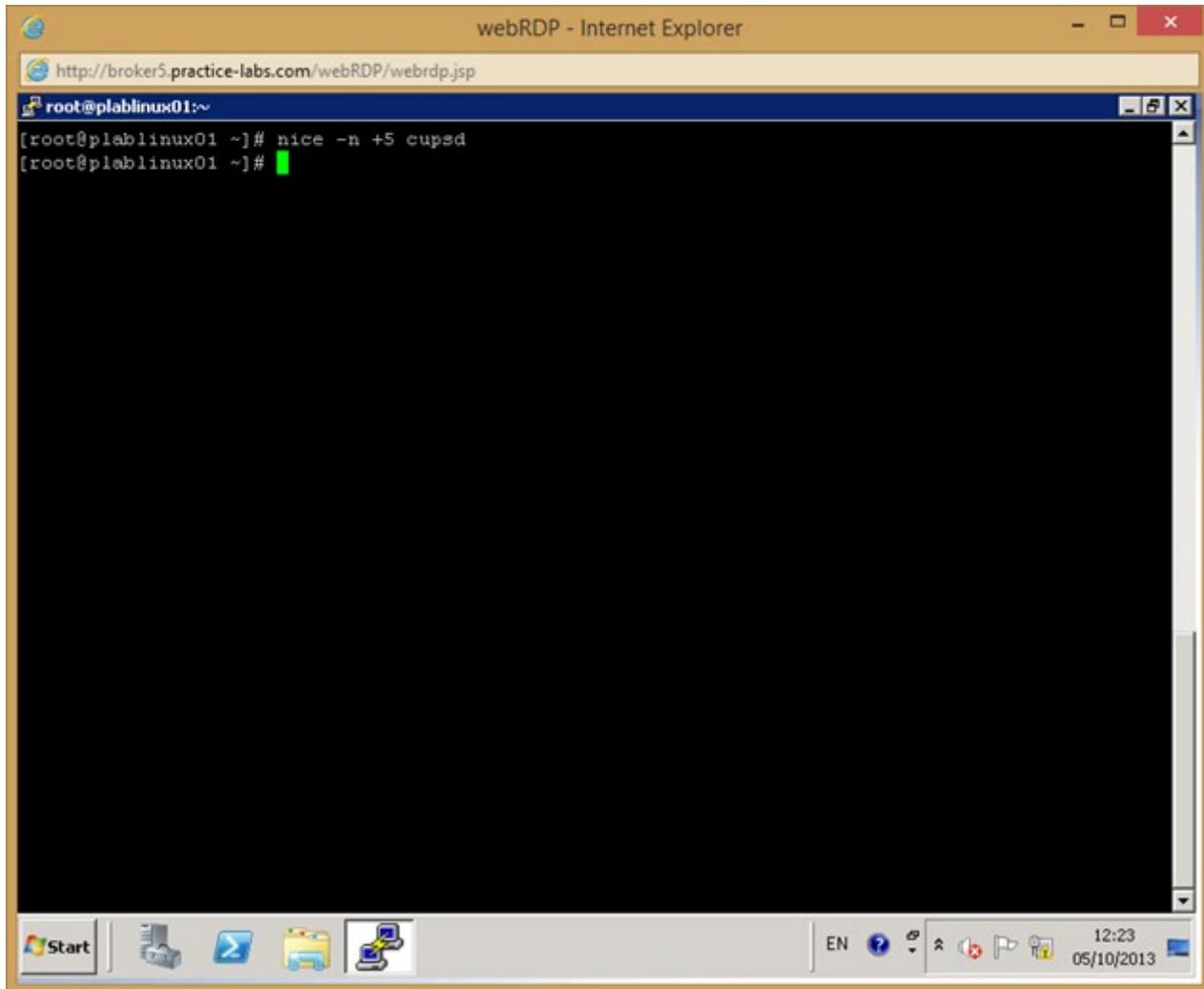
Step 1

You can run a program with higher or lower priority. Even though the default priority is 0, you can change the priority to either of the following:

- Up to -1 to -19 where -19 is the highest priority
- From 1 to +20 where +20 is the lowest priority

Enter the following command:

```
nice -n +5 cupsd
```



Step 2

You can also verify whether the program is running with the priority that you have set. To do this, enter the following command:

```
ps -l -u root
```

```

root@plablinux01:~#
 4 S    0  648   1  0  80   0 - 29598 hrtime ?      00:00:00 crond
 4 S    0  653   1  0  80   0 -  5297 pause ?      00:00:00 atd
 4 S    0  655   1  0  80   0 - 63398 poll_s ?     00:00:00 rsyslogd
 4 S    0  661  620  0  76 -4 - 5355 unix_s ?     00:00:00 sedispatch
 4 S    0  678   1  0  80   0 - 55616 poll_s ?     00:00:00 polkitd
 4 S    0  682   1  0  80   0 - 17788 poll_s ?     00:00:00 modem-manager
 4 S    0  683   1  0  80   0 -  5792 poll_s ?     00:00:00 bluetoothd
 4 S    0  701   1  0  80   0 - 19400 poll_s ?     00:00:00 sshd
 0 S    0  828   1  0  80   0 - 67177 poll_s ?     00:00:00 vmtoolsd
 4 S    0  850   1  0  80   0 - 52952 poll_s ?     00:00:00 gdm-binary
 4 S    0  859  850  0  80   0 - 62403 poll_s ?     00:00:00 gdm-simple-slav
 4 S    0  897  859  0  80   0 - 34711 poll_s tty1  00:00:00 Xorg
 1 S    0  900   2  0  60 -20 - 0 rescue ?      00:00:00 ttm_swap
 4 S    0  912  859  0  80   0 - 86557 poll_s ?     00:00:00 gdm-session-wor
 4 S    0  915   1  0  80   0 - 79932 poll_s ?     00:00:00 accounts-daemon
 1 S    0  920   2  0  80   0 - 0 bdi_ur ?      00:00:00 flush-253:1
 0 S    0  936   1  0  80   0 - 57537 poll_s ?     00:00:00 upowerd
 0 S    0 1099  859  0  80   0 - 59284 poll_s ?     00:00:00 gdm-session-wor
 4 S    0 1171   1  0  80   0 - 51102 epoll_?     00:00:00 cupsd
 0 S    0 1180   1  0  80   0 - 93468 poll_s ?     00:00:00 udisksd
 5 S    0 1358   1  0  80   0 - 24933 poll_s ?     00:00:00 sendmail
 4 S    0 1379  701  0  80   0 - 33075 poll_s ?     00:00:00 sshd
 4 S    0 1441 1385  0  80   0 - 42048 wait pts/0   00:00:00 su
 4 S    0 1445 1441  0  80   0 - 28780 n_tty pts/0  00:00:00 bash
 1 S    0 1526   2  0  80   0 - 0 worker ?      00:00:00 kworker/0:1
 4 S    0 1530  701  0  80   0 - 33075 poll_s ?     00:00:00 sshd
 4 S    0 1591 1535  0  80   0 - 42048 wait pts/1   00:00:00 su
 4 S    0 1595 1591  0  80   0 - 28780 wait pts/1   00:00:00 bash
 1 S    0 1652   2  0  80   0 - 0 worker ?      00:00:00 kworker/0:0
 4 S    0 1654   1  0  80   0 - 100479 poll_s ?     00:00:00 packagekitd
 4 S    0 1663   1  0  85   5 - 51102 epoll_-?    00:00:00 cupsd
 0 R    0 1664 1595  0  80   0 - 28934 -      pts/1   00:00:00 ps
[root@plablinux01 ~]#

```

Note that the process number 1663 is now set with the priority of 5.

Change the Priority of a Running Process

In the previous task you saw that you can start a program with a different priority than the default, which is 0. You can also change the priority when a process is running. You can change the priority using the renice command.

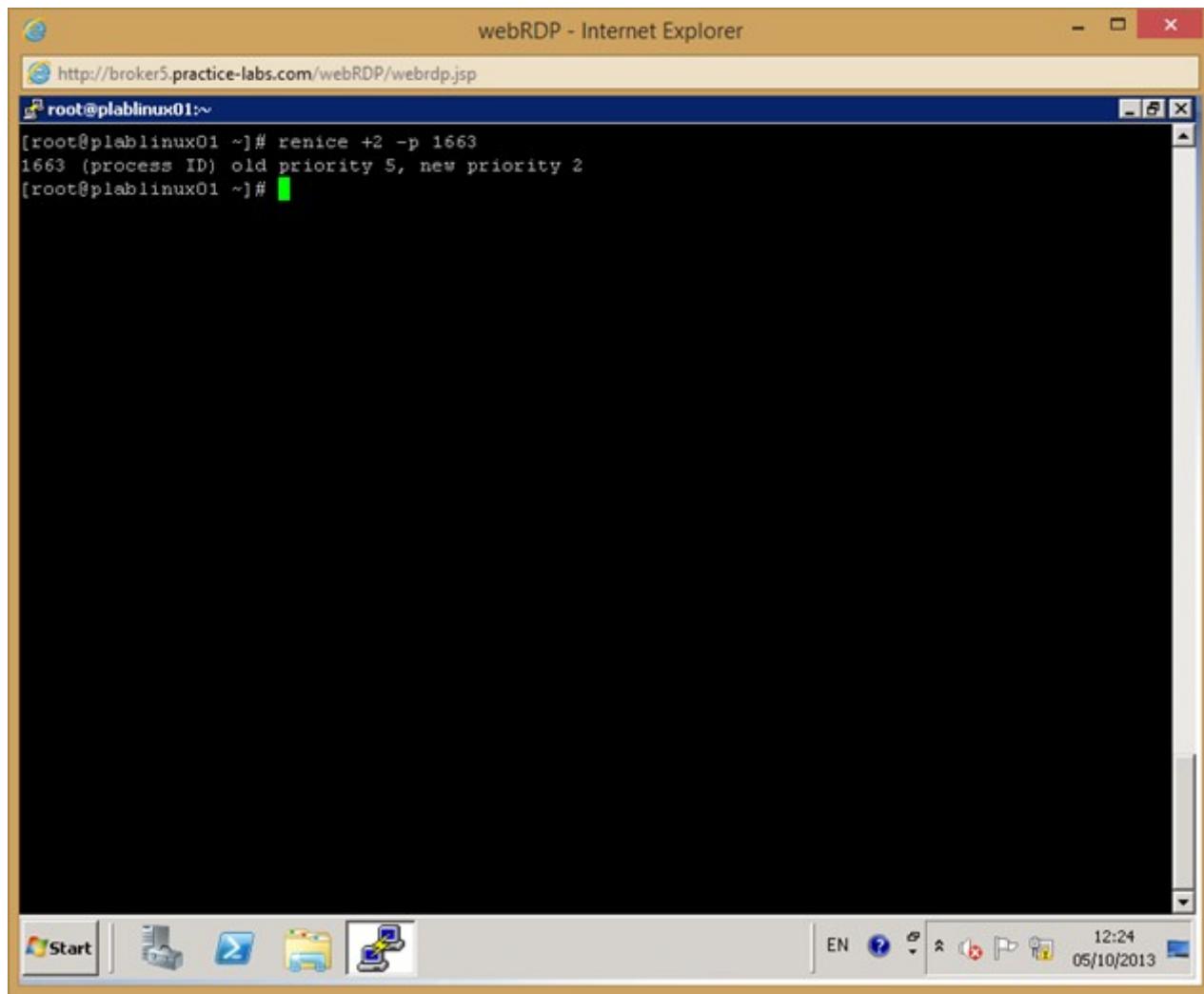
To change the priority of an active process, perform the following steps:

Step 1

To change the priority of an active process, enter the following command:

```
renice +2 -p 1663
```

Note that for the lab demonstration purpose, process id 1663 is selected. You can choose any other process from the active process list.



You are immediately notified that the process priority is now changed from 5, which you had set when you started the process, to 2. You can verify the changed process id by entering the following command:

```
ps -l -u root
```

```

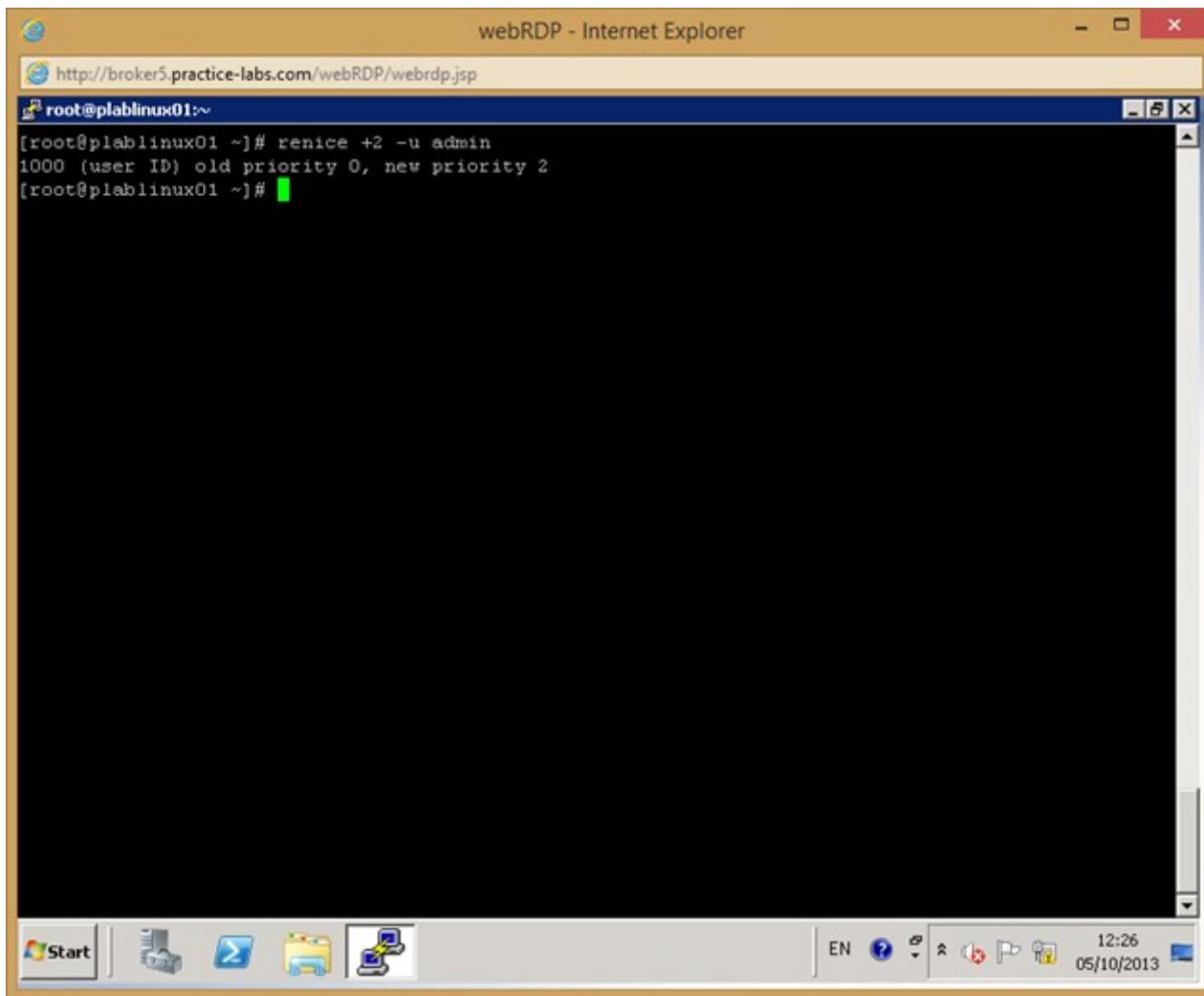
root@plablinux01:~#
4 S 0 648 1 0 80 0 - 29598 hrtime ? 00:00:00 crond
4 S 0 653 1 0 80 0 - 5297 pause ? 00:00:00 atd
4 S 0 655 1 0 80 0 - 63398 poll_s ? 00:00:00 rsyslogd
4 S 0 661 620 0 76 -4 - 5355 unix_s ? 00:00:00 sedispatch
4 S 0 678 1 0 80 0 - 55616 poll_s ? 00:00:00 polkitd
4 S 0 682 1 0 80 0 - 17788 poll_s ? 00:00:00 modem-manager
4 S 0 683 1 0 80 0 - 5792 poll_s ? 00:00:00 bluetoothd
4 S 0 701 1 0 80 0 - 19400 poll_s ? 00:00:00 sshd
0 S 0 828 1 0 80 0 - 67177 poll_s ? 00:00:00 vmtoolsd
4 S 0 850 1 0 80 0 - 52952 poll_s ? 00:00:00 gdms-binary
4 S 0 859 850 0 80 0 - 62403 poll_s ? 00:00:00 gdms-simple-slav
4 S 0 897 859 0 80 0 - 34711 poll_s ttys 00:00:00 Xorg
1 S 0 900 2 0 60 -20 - 0 rescue ? 00:00:00 ttm_swap
4 S 0 912 859 0 80 0 - 86557 poll_s ? 00:00:00 gdm-session-wor
4 S 0 915 1 0 80 0 - 79932 poll_s ? 00:00:00 accounts-daemon
1 S 0 920 2 0 80 0 - 0 bdi_vr ? 00:00:00 flush-253:1
0 S 0 936 1 0 80 0 - 57537 poll_s ? 00:00:00 upowerd
0 S 0 1099 859 0 80 0 - 59284 poll_s ? 00:00:00 gdm-session-wor
4 S 0 1171 1 0 80 0 - 51102 epoll_ ? 00:00:00 cupsd
0 S 0 1180 1 0 80 0 - 93468 poll_s ? 00:00:00 udisksd
5 S 0 1358 1 0 80 0 - 24933 poll_s ? 00:00:00 sendmail
4 S 0 1379 701 0 80 0 - 33075 poll_s ? 00:00:00 sshd
4 S 0 1441 1385 0 80 0 - 42048 wait pts/0 00:00:00 su
4 S 0 1445 1441 0 80 0 - 28780 n_tty_ pts/0 00:00:00 bash
1 S 0 1526 2 0 80 0 - 0 worker ? 00:00:00 kworker/0:1
4 S 0 1530 701 0 80 0 - 33075 poll_s ? 00:00:00 sshd
4 S 0 1591 1535 0 80 0 - 42048 wait pts/1 00:00:00 su
4 S 0 1595 1591 0 80 0 - 28780 wait pts/1 00:00:00 bash
1 S 0 1652 2 0 80 0 - 0 worker ? 00:00:00 kworker/0:0
4 S 0 1654 1 0 80 0 - 100479 poll_s ? 00:00:00 packagekitd
4 S 0 1663 1 0 82 2 - 51102 epoll_- ? 00:00:00 cupsd
0 R 0 1667 1595 0 80 0 - 28934 - pts/1 00:00:00 ps
[root@plablinux01 ~]#

```

Step 2

There may be situations where you want to change the priority of all processes for a user. Let's assume that for the admin user, you want to change the priority of all processes. To change the priority of all processes of admin user, enter the following command:

```
renice +2 -u admin
```



You are immediately notified that the process priority for all processes of admin user is now changed to 2. You can verify the changed process id by entering the following command:

```
ps -l -u admin
```

```

root@plablinux01:~#
1 S 1000 1137 1006 0 82 2 - 17171 poll_s ? 00:00:00 ssh-agent
S S 1000 1150 1 0 82 2 - 102578 poll_s ? 00:00:00 gnome-keyring-d
0 S 1000 1157 1006 0 82 2 - 202961 poll_s ? 00:00:00 gnome-settings-
1 S 1000 1166 1 0 82 2 - 98806 poll_s ? 00:00:00 pulseaudio
0 S 1000 1173 1 0 82 2 - 116966 poll_s ? 00:00:00 gsd-printer
0 S 1000 1178 1 0 82 2 - 64357 poll_s ? 00:00:00 gvfs-udisks2-vo
0 S 1000 1186 1 0 82 2 - 61496 poll_s ? 00:00:00 gvfs-afc-volume
0 S 1000 1189 1 0 82 2 - 42438 poll_s ? 00:00:00 gvfs-gphoto2-vo
0 S 1000 1199 1006 0 82 2 - 107214 poll_s ? 00:00:00 metacity
0 S 1000 1205 1006 0 82 2 - 147451 poll_s ? 00:00:00 gnome-panel
0 S 1000 1210 1 0 82 2 - 34608 poll_s ? 00:00:00 gconfd-2
0 S 1000 1212 1 0 82 2 - 64440 poll_s ? 00:00:00 dconf-service
0 S 1000 1217 1006 0 82 2 - 114390 poll_s ? 00:00:00 gnome-screensav
0 S 1000 1219 1006 0 82 2 - 133385 poll_s ? 00:00:00 tracker-miner-f
0 S 1000 1221 1006 0 82 2 - 121695 poll_s ? 00:00:00 nm-applet
0 S 1000 1222 1006 0 82 2 - 62314 poll_s ? 00:00:00 seapplet
0 S 1000 1225 1006 0 82 2 - 93023 poll_s ? 00:00:00 deja-dup-monito
0 S 1000 1236 1006 0 82 2 - 106605 poll_s ? 00:00:00 notification-da
0 S 1000 1239 1006 0 82 2 - 113625 poll_s ? 00:00:00 gnome-sound-app
0 S 1000 1246 1006 0 82 2 - 138480 poll_s ? 00:00:00 evolution-alarm
0 S 1000 1250 1006 0 82 2 - 61847 poll_s ? 00:00:00 abrt-applet
0 S 1000 1254 1006 0 82 2 - 107759 poll_s ? 00:00:00 bluetooth-apple
0 S 1000 1256 1006 0 82 2 - 107355 poll_s ? 00:00:00 tracker-store
0 S 1000 1262 1 0 82 2 - 82171 poll_s ? 00:00:01 vmtoolsd
0 S 1000 1265 1006 0 82 2 - 122708 poll_s ? 00:00:00 gnome-fallback-
0 S 1000 1313 1 0 82 2 - 129003 poll_s ? 00:00:00 wnck-applet
0 S 1000 1314 1 0 82 2 - 129876 poll_s ? 00:00:00 clock-applet
0 S 1000 1321 1 0 82 2 - 107201 poll_s ? 00:00:00 notification-ar
S S 1000 1384 1379 0 82 2 - 33075 poll_s ? 00:00:00 sshd
0 S 1000 1385 1384 0 82 2 - 28781 wait pts/0 00:00:00 bash
S S 1000 1534 1530 0 82 2 - 33075 poll_s ? 00:00:00 sshd
0 S 1000 1535 1534 0 82 2 - 28781 wait pts/1 00:00:00 bash
[root@plablinux01 ~]#

```

Shutdown all virtual machines used in this lab, by using the power functions located in the Tools bar before proceeding to the next module. Alternatively you can log out of the lab platform.

Summary

In this module, you completed the following tasks in Linux:

- Use single shell commands and one line command sequences to perform basic tasks on the command
- Use and modify the shell environment including defining, referencing and exporting
- Use and edit command history
- Invoke commands inside and outside the defined path

- Run jobs in the foreground and background
- Signal a program to continue running after logout
- Monitor active processes
- Select and sort processes for display
- Send signals to processes
- Terms and Utilities
- Redirecting standard input, standard output and standard error
- Pipe the output of one command to the input of another command
- Use the output of one command as arguments to another command
- Send output to both stdout and a file
- Know the default priority of a job that is created
- Run a program with higher or lower priority than the default
- Change the priority of a running process

Also Try

Using your lab infrastructure you can attempt the following topics at your own pace.

- Use the man command to find information on various command like cd, exec,
- Use !2 as the command to see what happens on the system. Note the result.
- Navigate to various subdirectories of /etc directory using absolute and relative paths.
- Define a variable and export its output to a file. View the file to note the result.
Create a user and monitor its active processes
- Monitor active processes. Observe them for a few minutes to review the changes
- Verify the uptime
- Use the pstree command and notice the difference in output when running ps command
- Use the tee command to run the following command: cmd1 | tee file_cmd1 | cmd2 | cmd3 > file1. Replace cmd1 etc with appropriate commands
- Use the cmd > file\$ cmd 1> file command and view the output. Replace cmd1 etc with appropriate commands
- Use the cmd 2>> file command to view the output. Replace cmd1 etc with appropriate commands
- Run the top command and press h.
- Change a process priority using the top command
- Change the process priority from 0 to -20 for root user for all processes

