**Chapter 2 Lab**

**ACCESSING THE COMMAND LINE**

**PERFORMANCE CHECKLIST**

In this lab, you will use the Bash shell to execute commands.

**OUTCOMES**

• Successfully run simple programs using the Bash shell command line.

• Execute commands used to identify file types and display parts of text files.

• Practice using some Bash command history "shortcuts" to more efficiently repeat commands or parts of commands.

**BEFORE YOU BEGIN**

Log in to workstation as student using student as the password.

On workstation, run the lab cli-review startscript to set up a clean lab environment. The script also copies the zcatfile to student's home directory.

[student@workstation ~]$ lab cli-review start

Note: In this lab you will use the redirect character, >, to record the output of your work. Your instructor should explain this character before this lab is started. Verify your redirected output with the cat /home/student/file.txt command, where file is the name of the file.

**1.** Use the datecommand to display the current time and date.

[student@workstation ~]$ date

Thu Jan 22 10:13:04 PDT 2019

* 1. Record the output in file /home/student/step1.txt.
  2. Verify the file contents with the command cat /home/student/step1.txt.

**2.** Display the current time in 12-hour clock time (for example, 11:42:11 AM). Hint: The format string that displays that output is %r. Use the +%rargument with the datecommand to display the current time in 12-hour clock time.

[student@workstation ~]$ date +%r

10:14:07 AM

2.1 Record the output in file /home/student/step2.txt.

2.1 Verify the file contents with the command cat /home/student/step2.txt.

**3.** What kind of file is **/home/student/zcat**? Is it readable by humans? Use the **file** command to determine its file type.

[student@workstation ~]$ **file zcat**

zcat: POSIX shell script, ASCII text executable

3.1 Record the output in file /home/student/step3.txt.  
  
3.2 Verify the file contents with the command cat /home/student/step3.txt.

3.3 Before proceeding, issue the command **file zcat** again. This will prepare the Bash history shortcut in step 4 to work properly.

**4.** Use the wccommand and Bash shortcuts to display the size of zcat.

The wccommand can be used to display the number of lines, words, and bytes in the zcatscript. Instead of retyping the file name, use the Bash history shortcut Esc+.(the keys Escand .pressed at the same time) to reuse the argument from the previous command.

[student@workstation ~]$ wc Esc+.

[student@workstation ~]$ wc zcat

51 299 1983 zcat

4.1 Record the output in file /home/student/step4.txt.

4.2 Verify the file contents with the command cat /home/student/step4.txt.

**5.** Display the first 10 lines of zcat.

The headcommand displays the beginning of the file.

[student@workstation ~]$ head zcat

#!/bin/sh

# Uncompress files to standard output.

# Copyright (C) 2007, 2010-2018 Free Software Foundation, Inc.

# This program is free software; you can redistribute it and/or modify

# it under the terms of the GNU General Public License as published by

# the Free Software Foundation; either version 3 of the License, or

# (at your option) any later version.

5.1 Record the output in file /home/student/step5.txt.

5.2 Verify the file contents with the command cat /home/student/step5.txt.

**6.** Display the last 10 lines of the zcatfile.

Use the **tail** command to display the last 10 lines of the zcatfile.

[student@workstation ~]$ **tail zcat**

With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>."

case $1 in

--help) printf '%s\n' "$usage" || exit 1;;

--version) printf '%s\n' "$version" || exit 1;;

esac

exec gzip -cd "$@"

6.1 Record the output in file /home/student/step6.txt.

6.2 Verify the file contents with the command cat /home/student/step6.txt.

**7.** Repeat the previous command exactly with three or fewer keystrokes.

Repeat the previous command exactly. Either press the **UpArrow** key once to scroll back through the command history one command and then press **Enter** (uses two keystrokes), or enter the shortcut command **!!** and then press **Enter** (uses three keystrokes) to run the most recent command in the command history . (Try both.)

[student@workstation]$ **!!**

**tail zcat**

With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>."

case $1 in

--help) printf '%s\n' "$usage" || exit 1;;

--version) printf '%s\n' "$version" || exit 1;;

esac

exec gzip -cd "$@"

**8.** Repeat the previous command, but use the **-n 20** option to display the last 20 lines in the file. Use command-line editing to accomplish this with a minimal number of keystrokes. **UpArrow** displays the previous command. **Ctrl**+**A** makes the cursor jump to the beginning of the line. **Ctrl**+**RightArrow** jumps to the next word, then add the **-n 20** option and hit **Enter** to execute the command.

[student@workstation ~]$ **tail -n 20 zcat**

-l, --list list compressed file contents

-q, --quiet suppress all warnings

-r, --recursive operate recursively on directories

-S, --suffix=SUF use suffix SUF on compressed files

--synchronous synchronous output (safer if system crashes, but slower)

-t, --test test compressed file integrity

-v, --verbose verbose mode

--help display this help and exit

--version display version information and exit

With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>."

case $1 in

--help) printf '%s\n' "$usage" || exit 1; exit;;

--version) printf '%s\n' "$version" || exit 1; exit;;

esac

exec gzip -cd "$@"

8.1 Record the output in file /home/student/step8.txt.

8.2 Verify the file contents with the command cat /home/student/step8.txt.

**9.** Use the shell history to run the **date +%r** command again.

Use the **history** command to display the list of previous commands and to identify the

specific **date** command to be executed. Use **!*number*** to run the command, where *number* is

the command number to use from the output of the **history** command.

Note that your shell history may be different from the following example. Determine the

command number to use based on the output of your own **history** command. Note- your output may look different.

[student@workstation ~]$ **history**

1 date

2 date +%r

3 file zcat

4 wc zcat

5 head zcat

6 tail zcat

7 tail -n 20 zcat

8 history

[student@workstation ~]$ **!2**

date +%r

10:49:56 AM

9.1 Record the shell history shortcut number for the first instance of **date +%r** in /home/student/step9.txt.

[student@workstation ~]$ **echo 2 >/home/student/step9.txt.**

9.2 Verify the file contents with the command cat /home/student/step9.txt.

9.3 Type the command history -a to append the commands to the history file:

[student@workstation ~]$ **history -a**

**10.** Evaluation

On workstation, run the **lab cli-review grade** script to confirm success on this exercise.

[student@workstation ~]$ **lab cli-review grade**

**11.** Finish

On workstation, run the **lab cli-review finish** script to complete the lab.

[student@workstation ~]$ **lab cli-review finish**

12. This concludes the lab.