# Week 10 Linux Projects - Working with the BASH Shell

Activities

[Week 10 Linux Projects - Working with the BASH Shell 1](#_Toc193534815)

[Update Kali Linux 1](#_Toc193534816)

[Project 7-1: Redirect stdout and stderr to a File 2](#_Toc193534817)

[Project 7-2: Redirect stdout and stdin using pipe metacharacters 4](#_Toc193534818)

[Project 7-3: Create and Use a Shell Script 6](#_Toc193534819)

[Project 7-4: Shell Scripts with Decisions and Loops 8](#_Toc193534820)

[Project 7.5: Backup Script with Input 9](#_Toc193534821)

[Decisions 10](#_Toc193534822)

[Positional Parameters 10](#_Toc193534823)

[Project 7.6: Backup Script 11](#_Toc193534824)

[Assignment Submission 12](#_Toc193534825)

**How to Create Screenshots:** Please use the Windows Snip and Sketch Tool or the Snipping Tool. Paste a screenshot of just the program you are working on. If you are snipping a virtual machine, make sure your focus is outside the virtual machine before you snip.

1. Press and hold down the **Windows key** & **Shift**, then Type **S.** This brings up the on-screen snipping tool.
2. Click and Drag your mouse around whatever you want to snip.
3. Release the mouse button. This places the snip into the Windows Clipboard.

Go into Word or wherever you want to paste the snip. Hold down **CTRL**, then Type **V** to paste the snip.

## Update Kali Linux

In Kali Linux in the terminal.

|  |
| --- |
| sudo apt update  sudo apt dist-upgrade -y |

# Project 7-1: Redirect stdout and stderr to a File

Time required: 30 minutes

In this hands-on project, you use the shell to redirect the stdout and stderr to a file and take stderr from a file.

1. Log in to the terminal as user.
2. Type **touch sample1 sample2** and press **Enter** to create two new files named sample1 and sample2 in your home directory.
3. Verify their creation by typing **ls** Press **Enter**.
4. Type **ls -l sample1 sample2 sample3** and press **Enter**.
5. Is there any stdout displayed on the terminal screen? Is there any stderr displayed on the terminal screen? Why?

Click or tap here to enter text.

1. Type **ls -l sample1 sample2 sample3 > file** and press **Enter**.
2. Is there any stdout displayed on the terminal screen? Is there any stderr displayed on the terminal screen? Why?

Click or tap here to enter text.

1. Type **cat file** and press **Enter**.
2. What are the contents of the file and why?

Click or tap here to enter text.

1. Type **ls -l sample1 sample2 sample3 > file** and press **Enter**.
2. Is there any Standard Output displayed on the terminal screen? Is there any Standard Error displayed on the terminal screen? Why?

Click or tap here to enter text.

1. Type **cat file** and press **Enter**.
2. What are the contents of the file and why? Were the previous contents retained? Why?

Click or tap here to enter text.

1. Type **ls -l sample1 sample2 sample3 > file 2> file2** and press **Enter**.
2. Is there any stdout displayed on the terminal screen? Is there any stderr displayed on the terminal screen? Why?

Click or tap here to enter text.

1. Type **cat file** and press **Enter**.
2. What are the contents of the file and why?

Click or tap here to enter text.

1. Insert a screenshot.

Click or tap here to enter text.

1. Type **cat file2** and press **Enter**.
2. What are the contents of file2 and why?

Click or tap here to enter text.

1. Insert a screenshot.

Click or tap here to enter text.

1. Type **ls -l sample1 sample2 sample3 > file 2>&1** and press **Enter**.
2. Is there any stdout displayed on the terminal screen? Is there any stderr displayed on the terminal screen? Why?

Click or tap here to enter text.

1. Type **cat file** and press **Enter**.
2. What are the contents of the file and why?

Click or tap here to enter text.

1. Type **ls -l sample1 sample2 sample3 >&2 2> file2** and press **Enter**.
2. Is there any stdout displayed on the terminal screen? Is there any stderr displayed on the terminal screen? Why?

Click or tap here to enter text.

1. Type **cat file2** and press **Enter**.
2. What are the contents of file2 and why?

Click or tap here to enter text.

1. Insert a screenshot.

Click or tap here to enter text.

1. Type **date >> file** and press **Enter**.
2. Type **cat file** and press **Enter**.
3. What are the contents of the file and why?

Click or tap here to enter text.

1. Type **date >> file** and press **Enter**.
2. Type **cat file** and press **Enter**.
3. What are the contents of the file and why? Can you tell when each date command was run?

Click or tap here to enter text.

1. Type **exit** and press **Enter** to log out of your shell.

# Project 7-2: Redirect stdout and stdin using pipe metacharacters

Time required: 20 minutes

In this hands-on project, you redirect stdout and stdin using pipe metacharacters.

1. Log in to the terminal as user.
2. Type **cat /etc/services** and press **Enter** to view the /etc/*services* file.
3. Type **cat /etc/services | less** and press **Enter** to perform the same task page-by-page.
4. Explain what the | (pipe) metacharacter does in the previous command. How is this different from the **less /etc/services** command?

Click or tap here to enter text.

1. Type **cat /etc/services | grep -i NFS** and press **Enter**.
2. How many lines are displayed? Why did you not need to specify a filename with the grep command?

Click or tap here to enter text.

1. Type **cat /etc/services | grep -i NFS | tr n N** and press **Enter**.
2. Explain the output on the terminal screen.

Click or tap here to enter text.

1. Type **cat /etc/services | grep NFS | tr n N | sort -r** and press **Enter**.
2. Explain the output on the terminal screen.

Click or tap here to enter text.

1. Type **cat /etc/services | grep ntp | tr n N | sort -r | tee file** and press **Enter**.
2. Explain the output on the terminal screen.

Click or tap here to enter text.

1. Type **cat file** and press **Enter**.
2. What are the contents? Why? What does the tee command do in the pipe above?

Click or tap here to enter text.

1. Type **cat /etc/services | grep NFS | tr n N | sort -r | tee file | wc -l**Press **Enter**.
2. Explain the output on the terminal screen.

Click or tap here to enter text.

1. Type **cat file** and press **Enter**.
2. What are the contents and why?

Click or tap here to enter text.

1. Type **cat /etc/services | grep NFS | tr F f | sort -r | sed /udp/d | sed /tcp/s/mount/MOUNT/g** and press **Enter**.
2. Explain the output on the terminal screen. Can this output be obtained with the grep and tr commands instead of sed?

Click or tap here to enter text.

1. Type **cat /etc/hosts**
2. Type **cat /etc/hosts | awk ' /localhost/ {print $1, $3}’** and press **Enter**.
3. Explain the output on the terminal screen.

Click or tap here to enter text.

1. Type **exit** and press **Enter** to log out of your shell.

# Project 7-3: Create and Use a Shell Script

Time required: 15 minutes

In this hands-on project, you create a basic shell script and execute it on the system.

1. Log in to the terminal as **user**

Of course, the first script is going to be Hello World!

1. Use your favorite Linux text editorto open a new file for editing called **hello\_world.sh** in your home directory.

Text, letter

Description automatically generated

1. Type bash **hello\_world.sh** 🡪 press **Enter**.
2. Insert a screenshot of your script results.

Click or tap here to enter text.

1. Use your favorite Linux text editorto open a new file for editing called **myscript.sh** in your home directory.
2. Enter the following text into the **myscript.sh** file.

Text

Description automatically generated

1. Type **ls -l myscript.sh** 🡪 press **Enter**.
2. What permissions does the **myscript.sh** file have?

Click or tap here to enter text.

1. Type **bash myscript.sh** and press **Enter**.
2. Insert a screenshot of your script results.

Click or tap here to enter text.

1. Type **./myscript.sh** and press **Enter**.
2. What error message did you receive and why?

Click or tap here to enter text.

1. Type **chmod 744 myscript.sh** 🡪 press **Enter**.
2. Type **./myscript.sh** and press **Enter**.
3. Did the script execute? Why?

Click or tap here to enter text.

1. Type **ls -l myscript.sh** 🡪 press **Enter**.
2. Look at the permissions for the **myscript.sh**. What changed?

Click or tap here to enter text.

1. Type **exit** and press **Enter** to log out of your shell.

|  |  |
| --- | --- |
| **Escape sequence** | **Description** |
| \\ | backslash |
| \a | ASCII beep |
| \b | backspace |
| \c | Prevents new line |
| \n\ | Starts a new line |
| \t | Horizontal tab |

# Project 7-4: Shell Scripts with Decisions and Loops

Time required: 45 minutes

In this hands-on project, you create a shell script that uses decision and loop constructs to analyze user input.

1. Log in to the terminal as user.
2. Open a new text file for editing called **input.sh** in your home directory.
3. Why is it good form to use a **.sh** extension for shell scripts?

Click or tap here to enter text.

1. Enter the following script including comments.

Text, letter

Description automatically generated

1. Type **chmod 744 input.sh** 🡪 press **Enter**.
2. Execute the script.
3. Insert a screenshot.

Click or tap here to enter text.

# Project 7.5: Backup Script with Input

1. Change directories to Documents.
2. Use your favorite Linux text editor to create a few files.
3. Change back to your home directory.
4. Create the following script.

Text

Description automatically generated

1. Type **chmod 755 backup\_input.sh** 🡪 press **Enter.**
2. Execute the script.

You can use a relative path or absolute path. This example will use a relative path which assumes you are in your home directory.

1. Enter **./Code** for the directory to backup.
2. Insert a screenshot.

Click or tap here to enter text.

## Decisions

Decision constructs are the most common type of construct used in shell scripts. They alter the flow of a program based on whether a command in the program completed successfully or based on a decision that the user makes in response to a question posed by the program.

The most common type of decision construct, the if construct, has the following syntax:

|  |
| --- |
| **if** <this is true>  **then** <do these commands>  **elif** <this is true>  **then** <do these commands>  **else** <do these commands>  **fi** |

Some common rules govern if constructs:

1. **elif** (else if) and **else** statements are optional.
2. You can have an unlimited number of **elif** statements.
3. The <do these commands> section can consist of multiple commands, one per line.
4. The <do these commands> section is typically indented from the left side of the text file for readability but does not need to be.
5. The end of the statement must be a backward **if** (fi).
6. The <this is true> part of the if syntax shown earlier can be a command or a test statement:
   1. Commands return true if they perform their function properly.
   2. Test statements are enclosed within square brackets [ ] or prefixed by the word “test” and used to test certain conditions on the system.

## Positional Parameters

A shell script can take arguments when it is executed on the command line; these arguments are called positional parameters and may be referenced using special variables within the shell script itself. The special variable **$1** refers to the contents of the first argument, **$2** refers to the contents of the second argument, **$3** refers to the contents of the third argument, and so on. If there are more than nine arguments, then you can use curly braces to indicate the appropriate positional parameter; for example, ${10} refers to the contents of the tenth argument. The special variable $0 refers to the command itself.

# Project 7.6: Backup Script

The following script uses a command line argument to choose the directory to backup.

1. Create the following script.

Text

Description automatically generated

1. Type **chmod 755 backup\_arg.sh** 🡪 press **Enter.**
2. What permissions does **755** give?

Click or tap here to enter text.

1. Type **./backup\_arg.sh** 🡪 press **Enter**. Note the error that you receive because you did not specify a positional parameter.
2. Type **./backup\_arg.sh Documents** 🡪 press **Enter.** This will create a backup of the Documents directory within the backupfile.tar.gz file in your home directory.
3. Type **ls -f** and press Enter.
4. Insert a screenshot.

Click or tap here to enter text.

1. Type **exit** and press **Enter** to log out of your shell.

## Assignment Submission

Attach this completed document to the assignment in BlackBoard.