



Module Code & Module Title

Level 5 – CT5052, Network Operating Systems

Assessment Type

Logbook 8.

Semester

2023/24 Spring/Autumn

Student Name: Asim Sapkota

London Met ID: 23048768

College ID: NP04CP4A230225

Assignment Due Date: 28th Dec 2024

Assignment Submission Date: 28th Dec 2024

Submitted to: Mr Prasant Adhikari

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Table of Contents

Introduction	3
Objectives	4
Required Tools and Concepts	4
Steps of Replicate:	6
Conclusion.....	16
References	17

Table of Figures

Figure 1: Creating new directory structure W8/8cat-grep	6
Figure 2: Changing to the 8cat-grep directory.....	6
Figure 3: Creating two files using cat command.....	7
Figure 4: Reading the two files	7
Figure 5: Searching for lines containing "ll" in the testa file.	8
Figure 6: Searching for lines not containing "ll" in the testa file.	8
Figure 7: Displaying line numbers for lines containing "ll" in the testa file.....	8
Figure 8: Listing files containing "ll" in their content.....	8
Figure 9: Performing a case-insensitive search for "ll" in all files.	9
Figure 10: Performing a case-insensitive search for "LL" in all files.	9
Figure 11: Counting the number of lines containing "ll" in each file.....	9
Figure 12: Searching for lines starting with "K" in the testa and testb files.	9
Figure 13: Displaying all lines with their line numbers from the testa file.	10
Figure 14: Defining the lsal alias for the ls -al command and checking it is stored using alias	10
Figure 15: Using the alias in our home directory.....	11
Figure 16: Removing the alias and verifying it is not stored.....	12
Figure 17: Defining the alias again and making it permanent	12
Figure 18: Checking the define alias work in another terminal or not	13
Figure 19: Defining the nwho alias and running nwho alias command.....	13
Figure 20: List of commands executed giving the history command	14
Figure 21: Re-executing the last but one command using the redo(r) command and the number of the event	14
Figure 22: Re-executing the command !-3 three commands ago using the negative integer	15
Figure 23: Re-executing the last command which name begins with "l"	15

Introduction

UNIX is a powerful, multi-tasking, and multi-user operating system. From the very beginning, UNIX was designed to be simple, efficient, and flexible. Its underlying philosophy is one of modularity—creating small, focused programs that each do one thing well. This approach allows users to combine these small programs to accomplish more complex tasks. The UNIX programming environment encourages programmers to write simple tools and then use them in combination to solve more complicated problems. These tools are often referred to as UNIX utilities and form the backbone of the operating system's power. (Brian W. Kernighen, 1984)

The power of UNIX utilities lies in their ability to manage files, directories, and system permissions with remarkable efficiency. Unlike graphical user interfaces (GUIs), which prioritize user-friendliness, UNIX commands provide fine-grained control and are highly customizable. They are indispensable for automation, scripting, and managing large-scale systems. For instance, a task that might require multiple steps in a GUI can often be accomplished with a single, well-crafted UNIX command. This efficiency underscores the enduring relevance of UNIX utilities, even in today's GUI-dominated landscape.

This workshop focuses on practical exercises with fundamental UNIX commands such as `mkdir`, `cd`, `cat`, `grep`, `alias`, and `history`. Participants will explore the creation of directory structures, manipulation of files, and management of aliases, all of which are foundational skills for navigating and utilizing the UNIX operating system. By practicing these commands, learners will gain a deeper understanding of the UNIX philosophy of doing one thing well, and they will develop the technical skills necessary to perform tasks efficiently.

In addition to these basic utilities, learners will also explore advanced features such as the use of aliases for command simplification, the `grep` command for text searching, and the `history` command for managing command-line history. These exercises will not only enhance their command-line proficiency but also provide insights into the powerful scripting and automation capabilities of UNIX.

Mastery of UNIX commands and utilities equips learners with the tools to handle complex tasks, from file manipulation to system-level operations. It also lays a strong foundation for understanding advanced topics in operating systems and network management, making it a valuable skill set for IT professionals. (Evi Nemeth, 2017)

Objectives

The Objectives of this workshop is to apply a hands on exercise with some basic UNIX utilities and commands that deal with file manipulation, text searching, managing of aliases, and the use of command history. This workshop aims to teach the way of how to work with files using the cat command as well as to navigate in the directories using the relative and the absolute pathnames and to search for the text in the files using different grep options. The workshop also teaches how to define and manage aliases for commands you frequently run to streamline workflow, and how to do so such that subsequent sessions will dutifully run the aliases this time and not next time. Additionally, students will be introduced to the use of command history to dramatically increase efficiency in re-executing previous commands. The workshop also concludes with practical exercises that gather system information, like users with `getent`. And these skills are very important for working in a UNIX environment (used for system administration, automation, and scripts) since they won't only help you but all other users around you.

Required Tools and Concepts

Software:

A UNIX-based operating system (Linux or macOS) that provides access to the command line interface (CLI). We will use any terminal or shell, such as Bash (Bourne Again Shell), which is commonly available on UNIX systems.

Basic UNIX Commands:

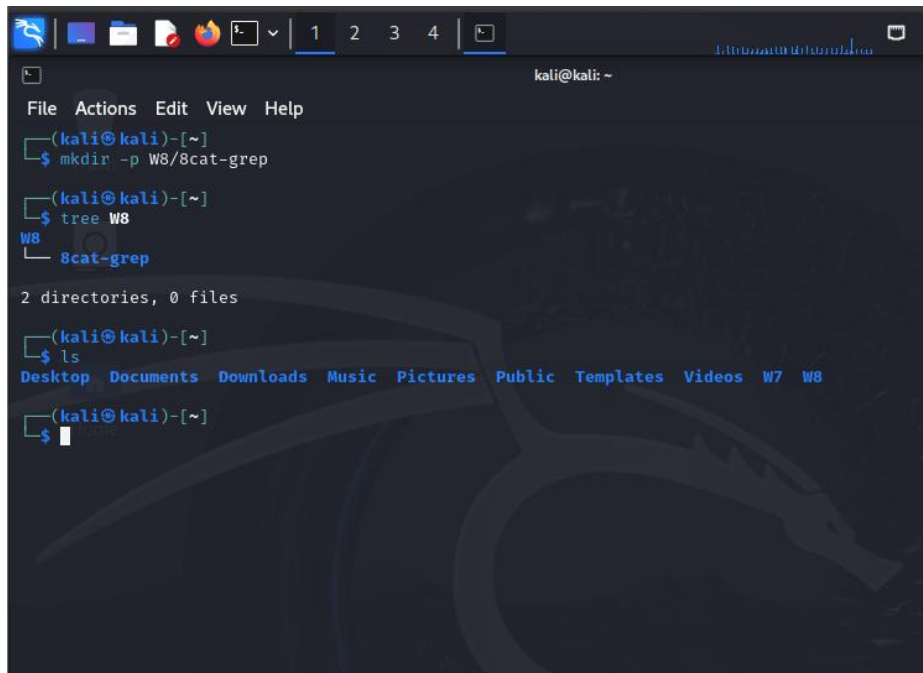
- cat: This utility allows us to create and display the contents of files.
- grep: A powerful command-line tool for searching text patterns within files. It supports various options such as -n for line numbers, -i for case-insensitivity, and -v for inverted matching.
- ls: Used for listing directory contents. We will also work with aliases to customize this command.
- alias: A command that lets us define shortcuts for frequently used commands.
- history: Displays the command history, allowing us to reuse previous commands.
- fc: A command for editing and re-executing commands from the history.
- getent: A command to retrieve system information, such as the number of users, by accessing the system's databases.

Concepts:

- File Manipulation: We will create, display, and edit files in the UNIX environment using cat and other text editors.
- Search Techniques: We will use grep with various options to search for specific patterns in files, including case-sensitive, case-insensitive, and regular expressions.
- Aliases: We will create custom shortcuts for commands to improve efficiency in our tasks. Understanding how to define, use, and remove aliases both temporarily and permanently is crucial.
- Command History: We will leverage the history and fc commands to quickly re-execute past commands and streamline our workflow.
- UNIX File System Navigation: We will use relative and absolute pathnames to navigate and manipulate files and directories effectively.

Steps of Replicate:


1.

A terminal window on a Kali Linux system. The prompt is (kali@kali)-[~]. The user enters 'mkdir -p W8/8cat-grep'. The prompt changes to (kali@kali)-[~] and the user enters 'tree W8'. The output shows a tree structure with 'W8' as the root and '8cat-grep' as a subdirectory. Below this, it says '2 directories, 0 files'. The user then enters 'ls' and the prompt changes to (kali@kali)-[~]. The terminal shows a list of directories: Desktop, Documents, Downloads, Music, Pictures, Public, Templates, Videos, W7, and W8. The cursor is at the end of the prompt (kali@kali)-[~].

```
(kali@kali)-[~]  
$ mkdir -p W8/8cat-grep  
  
(kali@kali)-[~]  
$ tree W8  
W8  
└── 8cat-grep  
  
2 directories, 0 files  
  
(kali@kali)-[~]  
$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos W7 W8  
  
(kali@kali)-[~]  
$
```

Figure 1: Creating new directory structure W8/8cat-grep

2.

A terminal window on a Kali Linux system. The prompt is (kali@kali)-[~]. The user enters 'cd W8/8cat-grep'. The prompt changes to (kali@kali)-[~/W8/8cat-grep]. The cursor is at the end of the prompt (kali@kali)-[~/W8/8cat-grep].

```
(kali@kali)-[~]  
$ cd W8/8cat-grep  
  
(kali@kali)-[~/W8/8cat-grep]  
$
```

Figure 2: Changing to the 8cat-grep directory

3.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ cat > testa
Kkkll
lllmm
oo-oo
mmmdd
dddkk

(kali㉿kali)-[~/W8/8cat-grep]
$ cat > testb
KKKKK
LLLLL
MMMMM
DDDDD

(kali㉿kali)-[~/W8/8cat-grep]
$
```

Figure 3: Creating two files using cat command

```
(kali㉿kali)-[~/W8/8cat-grep]
$ cat testa
Kkkll
lllmm
oo-oo
mmmdd
dddkk

(kali㉿kali)-[~/W8/8cat-grep]
$ cat testb
KKKKK
LLLLL
MMMMM
DDDDD

(kali㉿kali)-[~/W8/8cat-grep]
$
```

Figure 4: Reading the two files

4.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep ll testa
Kkkll
llmm

(kali㉿kali)-[~/W8/8cat-grep]
$
```

Figure 5 Searching for lines containing "ll" in the testa file.:

```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep -v ll testa
oo-oo
mmdd
ddkk

(kali㉿kali)-[~/W8/8cat-grep]
$
```

Figure 6: Searching for lines not containing "ll" in the testa file.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep -n ll testa
1:Kkkll
2:llmm
```

Figure 7: Displaying line numbers for lines containing "ll" in the testa file.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep -l ll *
testa
```

Figure 8: Listing files containing "ll" in their content.


```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep -i ll *
testa:Kkkll
testa:lllmm
testb:LLLLL
```

Figure 9: Performing a case-insensitive search for "ll" in all files.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep -i LL *
testa:Kkkll
testa:lllmm
testb:LLLLL
```

Figure 10: Performing a case-insensitive search for "LL" in all files.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep -c ll *
testa:2
testb:0
```

Figure 11: Counting the number of lines containing "ll" in each file.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep '^K' testa testb
testa:Kkkll
testb:KKKKK

(kali㉿kali)-[~/W8/8cat-grep]
$
```

Figure 12: Searching for lines starting with "K" in the testa and testb files.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ grep -n '^' testa
1:Kkkll
2:lllmm
3:oo-oo
4:mmmdd
5:dddkk

(kali㉿kali)-[~/W8/8cat-grep]
$
```

Figure 13: Displaying all lines with their line numbers from the testa file.

6.

```
(kali㉿kali)-[~/W8/8cat-grep]
$ alias lsal="ls -al"

(kali㉿kali)-[~/W8/8cat-grep]
$ lsal
total 16
drwxrwxr-x 2 kali kali 4096 Dec 26 09:54 .
drwxrwxr-x 3 kali kali 4096 Dec 26 09:50 ..
-rw-rw-r-- 1 kali kali 29 Dec 26 09:54 testa
-rw-rw-r-- 1 kali kali 23 Dec 26 09:54 testb

(kali㉿kali)-[~/W8/8cat-grep]
$
```

Figure 14: Defining the lsal alias for the ls -al command and checking it is stored using alias

```

(kali㉿kali)-[~/W8/scat-grep]
$ cd

(kali㉿kali)-[~]
$ ls -la
total 188
drwx----- 18 kali kali 4096 Dec 26 09:50 .
drwxr-xr-x  3 root root 4096 Aug 18 15:57 ..
-rw-r--r--  1 kali kali  220 Aug 18 15:57 .bash_logout
-rw-r--r--  1 kali kali 5551 Aug 18 15:57 .bashrc
-rw-r--r--  1 kali kali 3526 Aug 18 15:57 .bashrc.original
drwxrwxr-x  9 kali kali 4096 Dec 23 00:55 .cache
drwxr-xr-x 12 kali kali 4096 Dec 21 05:53 .config
drwxr-xr-x  2 kali kali 4096 Dec 21 05:52 Desktop
-rw-r--r--  1 kali kali  35 Dec 21 05:52 .dmrc
drwxr-xr-x  2 kali kali 4096 Dec 21 05:52 Documents
drwxr-xr-x  2 kali kali 4096 Dec 21 05:52 Downloads
-rw-r--r--  1 kali kali 11759 Aug 18 15:57 .face
lrwxrwxrwx  1 kali kali  5 Aug 18 15:57 .face.icon -> .face
drwx----- 3 kali kali 4096 Dec 21 05:52 .gnupg
-rw-----  1 kali kali  0 Dec 21 05:52 .ICEauthority
drwxr-xr-x  3 kali kali 4096 Aug 18 15:57 .java
drwxr-xr-x  4 kali kali 4096 Dec 21 05:52 .local
drwx----- 4 kali kali 4096 Dec 23 00:55 .mozilla
drwxr-xr-x  2 kali kali 4096 Dec 21 05:52 Music
drwxr-xr-x  2 kali kali 4096 Dec 21 06:21 Pictures
-rw-r--r--  1 kali kali  807 Aug 18 15:57 .profile
drwxr-xr-x  2 kali kali 4096 Dec 21 05:52 Public
drwxr-xr-x  2 kali kali 4096 Dec 21 05:52 Templates
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-clipboard-tty7-control.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-clipboard-tty7-service.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-display-svg-x11-tty7-control.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-display-svg-x11-tty7-service.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-draganddrop-tty7-control.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-draganddrop-tty7-service.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-hostversion-tty7-control.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-seamless-tty7-control.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-seamless-tty7-service.pid
-rw-r--r--  1 kali kali  5 Dec 26 09:14 .vboxclient-vmvga-session-tty7-control.pid
drwxr-xr-x  2 kali kali 4096 Dec 21 05:52 Videos
drwxrwxr-x  4 kali kali 4096 Dec 21 05:59 W7
drwxrwxr-x  3 kali kali 4096 Dec 26 09:50 W8
-rw-----  1 kali kali  49 Dec 26 09:14 .Xauthority
-rw-----  1 kali kali 7703 Dec 26 09:16 .xsession-errors
-rw-----  1 kali kali 8583 Dec 23 00:55 .xsession-errors.old
-rw-----  1 kali kali  615 Dec 23 00:55 .zsh_history
-rw-r--r--  1 kali kali 10868 Aug 18 15:57 .zshrc

(kali㉿kali)-[~]
$

```

Figure 15: Using the alias in our home directory

7.

```

(kali㉿kali)-[~]
$ unalias lsal

(kali㉿kali)-[~]
$ lsal
Command 'lsal' not found, did you mean:
  command 'lsar' from debunar
Try: sudo apt install <deb name>

(kali㉿kali)-[~]
$ 

```

Figure 16: Removing the alias and verifying it is not stored

8.

```

zstyle :completion:*:*:kitt:*:processes list-colors-=(#b) #([0-9]#)*=0;31
fi

# some more ls aliases
alias ll='ls -l'
alias la='ls -A'
alias l='ls -CF'

# enable auto-suggestions based on the history
if [ -f /usr/share/zsh-autosuggestions/zsh-autosuggestions.zsh ]; then
  . /usr/share/zsh-autosuggestions/zsh-autosuggestions.zsh
  # change suggestion color
  ZSH_AUTOSUGGEST_HIGHLIGHT_STYLE='fg=#999'
fi

# enable command-not-found if installed
if [ -f /etc/zsh_command_not_found ]; then
  . /etc/zsh_command_not_found
fi

alias lsal="ls -al"

```

[^]G Help [^]O Write Out [^]F Where Is [^]K Cut [^]T Execute [^]C Location ^M-^U Undo
[^]X Exit [^]R Read File [^]\ Replace [^]U Paste [^]J Justify [^]/ Go To Line ^M-^E Redo

Figure 17: Defining the alias again and making it permanent

```

(kali@kali)~$ cat /etc/passwd
total 192
drwxr-xr-x 10 kali kali 4096 Dec 26 11:32 .
drwxr-xr-x 2 root root 4096 Aug 18 15:57 ..
-rw-r--r- 1 kali kali 228 Aug 18 15:57 .bash_logout
-rw-r--r- 1 kali kali 5551 Aug 18 15:57 .bashrc
-rw-r--r- 1 kali kali 324 Aug 18 15:57 .bashrc.original
drwxr-xr-x 9 kali kali 4096 Dec 21 05:53 .cache
-rw-r--r- 1 kali kali 4096 Dec 21 05:53 .config
drwxr-xr-x 2 kali kali 4096 Dec 21 05:52 Desktop
-rw-r--r- 1 kali kali 35 Dec 21 05:52 .dmrc
drwxr-xr-x 2 kali kali 4096 Dec 21 05:52 Documents
drwxr-xr-x 2 kali kali 4096 Dec 21 05:52 Downloads
-rw-r--r- 1 kali kali 11759 Aug 18 15:57 .face
drwxr-xr-x 2 kali kali 4096 Dec 21 05:52 .face.icon → .face
-rw-r--r- 1 kali kali 0 Dec 21 05:52 .ICGauthority
drwxr-xr-x 3 kali kali 4096 Aug 18 15:57 .java
drwxr-xr-x 4 kali kali 4096 Dec 21 05:52 .local
drwxr-xr-x 4 kali kali 4096 Dec 21 04:55 .mozilla
drwxr-xr-x 2 kali kali 4096 Dec 21 05:52 .Music
-rw-r--r- 1 kali kali 807 Aug 18 15:57 .profile
drwxr-xr-x 2 kali kali 4096 Dec 21 05:52 .Public
drwxr-xr-x 2 kali kali 4096 Dec 21 05:52 Templates
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-clipboard-tty7-control.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-clipboard-tty7-service.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-display-svga-x11-tty7-control.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-display-svga-x11-tty7-service.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-draganddrop-tty7-control.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-draganddrop-tty7-service.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-hostversion-tty7-control.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-hostversion-tty7-service.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-seamless-tty7-control.pid
-rw-r--r- 1 kali kali 5 Dec 26 09:14 .vboxclient-seamless-tty7-service.pid
drwxr-xr-x 2 kali kali 4096 Dec 21 05:52 .Videos
drwxr-xr-x 2 kali kali 4096 Dec 21 05:59 #?
drwxr-xr-x 3 kali kali 4096 Dec 26 09:50 #?
-rw-r--r- 1 kali kali 49 Dec 26 09:14 .Xauthority
-rw-r--r- 1 kali kali 9553 Dec 21 05:53 .xsession-errors
-rw-r--r- 1 kali kali 8583 Dec 23 00:55 .xsession-errors.old
-rw-r--r- 1 kali kali 1546 Dec 26 11:32 .zsh_history
-rw-r--r- 1 kali kali 18889 Dec 26 11:32 .zshrc
-rw-r--r- 1 kali kali 0 Dec 26 11:21 .zshrc

```

Figure 18: Checking the define alias work in another terminal or not

9 and 10

```

(kali@kali)~$ alias nwho="getent passwd|wc -l"
(kali@kali)~$ nwho
57
(kali@kali)~$

```

Figure 19: Defining the nwho alias and running nwho alias command

11.

```
(kali㉿kali)-[~]
$ history
1  mkdir -p W7/{W7-1/{1level3,2level3},W7-2/{3level3,4level3}}
2  tree W6
3  tree W7
4  figlet ashim
5  cd W7/W7-1/1level3
6  cat>file
7  cat file
8  cd W7/W7-1
9  cd ../
10 ls -l 1level3
11 chmod -rwx 1level3/file
12 ls -l 1level3
13 ls -l 1level3/file
14 cat>>file
15 ls -l 1level3/file
16 cat file
17 cd
18 rm -r W7
19 tree W7
20 mkdir -p W7/{W7-1/{1level3,2level3},W7-2/{3level3,4level3}}
21 tree W7
22 cd W7/W7-1/1level3
23 cat>file
24 cat file
25 cat>>file
26 ls -l 1level3/file
27 cat file
28 ls -l file
29 ls -l 1level3
30 cd ../
31 ls -l 1level3
32 chmod -rwx 1level3/
33 ls -l
34 cat 1level3/file
35 cat>>1level3/file
36 ls 1level3/
37 chmod u+rwx 1level3/
38 ls -l
39 cat 1level3/file
40 cat>>1level3/file
41 ls -l
42 ls 1level3/
43 cd W8/8cat-grep
44 lsal
45 echo $0
46 nano .zshrc
47 cat .zshrc
```

Figure 20: List of commands executed giving the history command

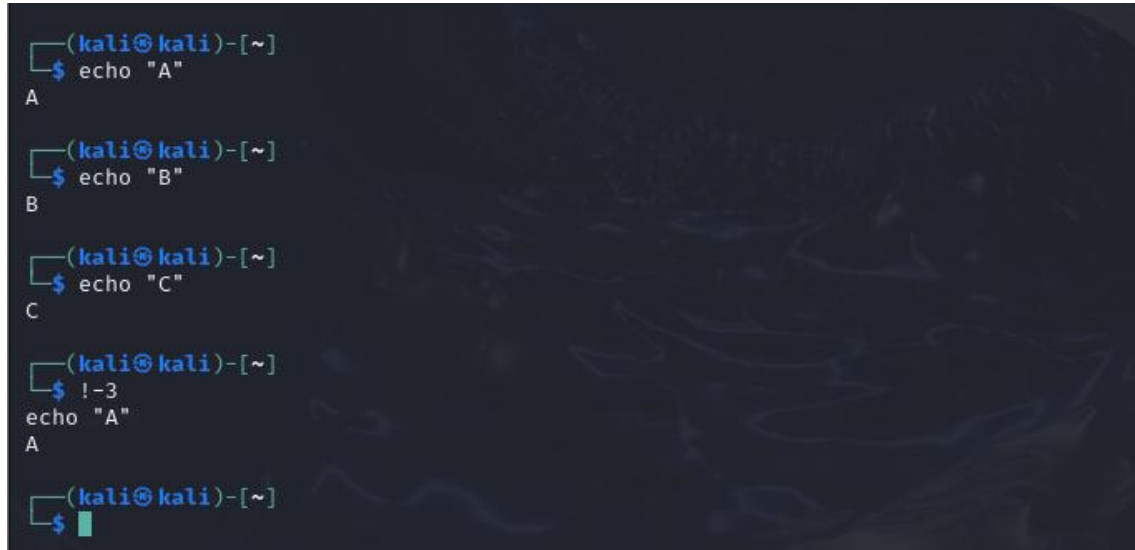
12.

```
(kali㉿kali)-[~]
$ fc -r
help ls
bash: help: no help topics match `ls'. Try `help help' or `man -k ls' or `info ls'.

(kali㉿kali)-[~]
$
```

Figure 21: Re-executing the last but one command using the redo(r) command and the number of the event

13.



```
(kali㉿kali)-[~]  
$ echo "A"  
A  
  
(kali㉿kali)-[~]  
$ echo "B"  
B  
  
(kali㉿kali)-[~]  
$ echo "C"  
C  
  
(kali㉿kali)-[~]  
$ !-3  
echo "A"  
A  
  
(kali㉿kali)-[~]  
$
```

Figure 22: Re-executing the command !-3 three commands ago using the negative integer

14.



```
(kali㉿kali)-[~]  
$ fc -s l  
ls  
Desktop Documents Downloads Music Pictures Public Templates Videos W7 W8 zshrc  
  
(kali㉿kali)-[~]  
$
```

Figure 23: Re-executing the last command which name begins with "l"

Conclusion

In this workshop, we explored a range of essential UNIX utilities that are fundamental for efficient system management and file handling. We gained practical experience in creating and navigating directories using relative and absolute pathnames. Additionally, we learned how to use the `cat` command to create files, which formed the foundation for later tasks involving text manipulation. By practicing with the `grep` command, we explored its various options for searching text patterns within files, such as case-sensitive, case-insensitive, and line-numbered searches. These exercises provided valuable insights into how powerful and versatile the `grep` tool is, especially for data analysis and system administration.

We also learned the importance of managing aliases in UNIX to simplify and optimize our workflow. Through this workshop, we defined custom aliases for commonly used commands, reducing the time spent on repetitive tasks. We also explored how to preserve aliases across sessions, ensuring that our environment remains efficient even after a reboot. This aspect of the workshop helped us understand the flexibility of UNIX in adapting to user needs by providing options for customization, which is crucial for both system administrators and developers working in a UNIX-based environment.

Moreover, the hands-on experience with UNIX's command history feature allowed us to re-execute previous commands quickly and efficiently. This not only saved time but also highlighted how UNIX enables users to work more effectively by offering commands like `fc` and the history list for easy command retrieval. The combination of these tools—file management, text searching, alias creation, and command history—enhanced our overall understanding of the UNIX system. The skills we have developed through this workshop will undoubtedly be valuable in future tasks involving system administration, scripting, and automation, where efficiency and precision are essential.

References

- Brian W. Kernighen, R. P. (1984). Introduction to UNIX and its Philosophy. In R. P. Brian W. Kernighen, *The Unix Programming Environment* (p. 368). Prentice Hall.
- Evi Nemeth, G. S. (2017). Exploring UNIX and Linux Utilities. In G. S. Evi Nemeth, *Unix and Linux System Administration Handbook* (p. 134). Pearson Education.
- Shotts, W. (2019). Exploring UNIX Command Line Tools for Workflow Optimization. In W. Shotts, *The Linux Command Line: A Complete Introduction* (p. 251). No Starch Press.