**Topic Name:**

The main aim of this lab session is to provide hands-on experience on

* Explore file structure
* File management commands
* Absolute path and Relative path
* Globbing
* Scripting

File Structure

1. Under the root directory there are many files like

/bin , /boot , /dev , /etc , ….

Find out the importance of those files

Example : /etc is for user account details

|  |  |  |
| --- | --- | --- |
| **S.No** | **Directory** | **Usage** |
| 1 | / | Root directory |
| 2 | /bin | Binary files |
| 3 | /boot | Contain files needed to boot system |
| 4 | /dev | Contain device file that contain hardware informatiion |
| 5 | /etc | Contain system configuration files and directories |
| 6 | /home | Contain user home directories |
| 7 | /lib | Contain library file that are needed to run ‘bin’ and ‘sbin’ files |
| 8 | /proc | A virtual filesystem that provides an interface to kernel data structures. |
| 9 | /sbin | Contains system binaries, essential for system administration tasks. |
| 10 | /tmp | Used for temporary file storage. |
| 11 | /var | Contains variable data such as logs, mail spools, and temporary files. |
| 12 |  |  |
| 13 |  |  |

1. In Linux, there are three different files

Regular file

Directory

Special file

Block file

Character file

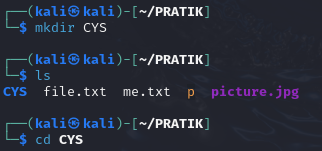
Socket file

Pipe file

Fill the below table:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| File Type | Represented by (Hint ls ) | Role | How to create | How to check | Location | Screen shot |
| Regular file | Minus (-) | To store data | touch filename | NA | Any directory |  |
| * Text file | Minus (-) | File containing plain text. | echo "text" > file.txt | NA | Any directory |  |
| * Compressed file | Minus (-) | File compressed to save space | gzip file | NA | Any directory |  |
| * Image | Minus (-) | File containing image | NA | NA | Any directory |  |
| Directory | ‘d’ | File containing sub directories and files | mkdir directory\_name | NA | Any directory |  |
| Block file | ‘b’ | Represents a block device | NA | NA | /dev |  |
| Character file | ‘c’ | Represents a character | NA | NA | /dev |  |
| Socket file | ‘s’ | Used for inter-process communication | NA | NA | Varies |  |
| pipe file | ‘p’ | Used for inter-process communication | NA | NA | Any directory |  |

1. Globbing
2. Go back to CYS



1. Create multiple subdirectories using single command

LS

Unit1

command

glob

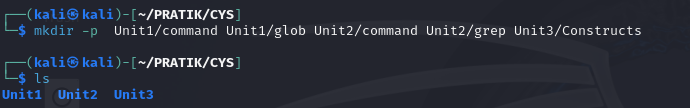
Unit2

command

grep

Unit3

Constructs



1. Navigate to unit1/glob



1. Create the following files :

Commands.txt

Commands1.txt

Commands2.txt

page1.html

page2.html

page3.html

file1

file10

file11

file2

File2

File3

file33

fileAB

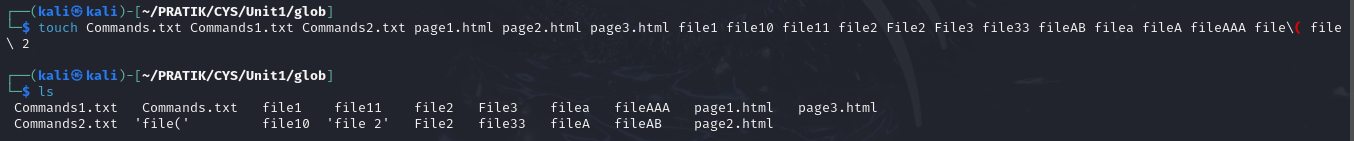
filea

fileA

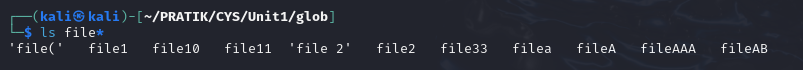
fileAAA

file(

file 2



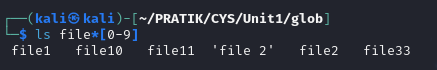
* + 1. List all files starting with file



* + 1. List all files starting with File



* + 1. List all files starting with file and ending in a number.



* + 1. List all files starting with file and ending with a letter



* + 1. List all files starting with File and having a digit as fifth character.



* + 1. List all files starting with File and having a digit as fifth character and nothing else.



* + 1. List (with ls) all files starting with a letter and ending in a number.



* + 1. List (with ls) all files that have exactly five characters.



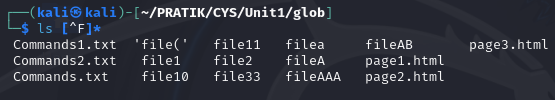
* + 1. List (with ls) all files that start with f or F and end with 3 or A.



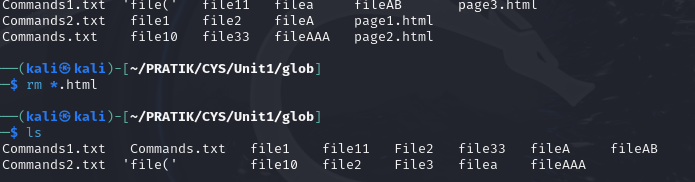
* + 1. List (with ls) all files that start with f have i or R as second character and end in a number.



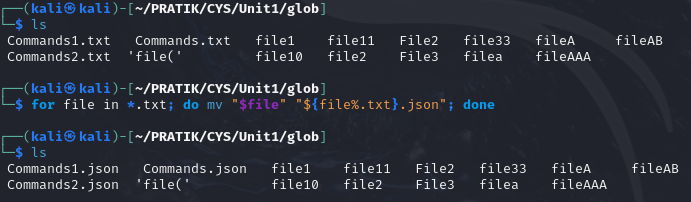
* + 1. List all files that do not start with the letter F.



* + 1. Remove all the \*.html

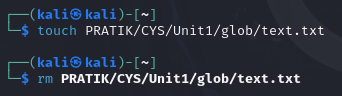
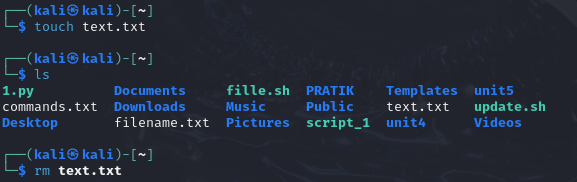


* + 1. Rename \*.txt to \*.json



1. Absolute path and relative path

Use rm, mv, cp, ls with absolute path and relative path as per your choice.

1. Wildcards

|  |  |  |  |
| --- | --- | --- | --- |
| Notation | Use | Example | Screenshot |
| \* | |  | | --- | |  |   Matches any number of characters | ls file\*[0-9] |  |
| ? | Matches exactly one character | ls file? |  |
| [ ] | Matches any one of the enclosed characters | ls file[0-9] |  |
| [! ] | Matches any character that is not a member of the set characters | ls \*[!F] |  |
| { } | Matches a range of characters | ls {0..9}.txt |  |

More on Character class

|  |  |  |  |
| --- | --- | --- | --- |
| Notation | Use | Example | Screenshot |
| [:alnum:] | Matches any alphanumeric character |  |  |
| [:alpha:] | Matches any alhpabetic characyer |  |  |
| [:digit:] | Matches any digit between(0-9) |  |  |
| [:lower:] | Matches any lowercase letter |  |  |
| [:upper:] | Matches any uppercase letter |  |  |

4. change permission

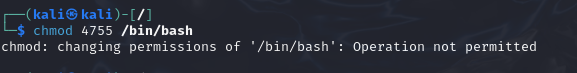
1. Change the permission set of /work/readme.txt so that only the user (owner) can read,write, and execute it. Use absolute mode.



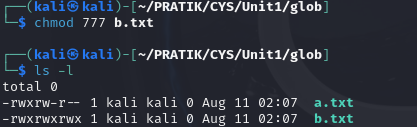
1. Change the permission set of /work/readme.txt so that any user can read it, the group can read/write to it and the user (owner) can read/write/execute it. Use absolute mode.



1. Change the permission set of /bin/bash so that only the user (owner) can read/write/ execute, group, and any user can execute it. However, whenever anyone executes it, it should run with the privileges of the owner user. Use absolute mode.



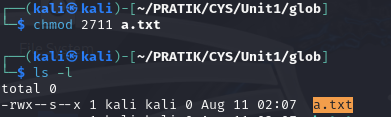
1. Change the permission set of /work/readme.txt so that only the user (owner) can read, write, and execute it. Use relative mode.



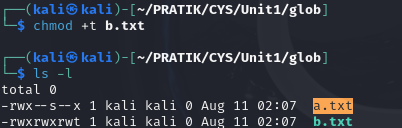
1. Change the permission set of /work/readme.txt so that any user can read it, the group can read/write to it and the user (owner) can read/write/execute it. Use relative mode.



1. Change the permission set of /work/readme.txt so that only the user (owner) can read/write/ execute, group, and any user can execute it. However, whenever anyone executes it, it should run with the privileges of the group. Use absolute mode.



1. Change the permission set of /work/readme.txt so that only the owner can rename or delete this file while maintaining the existing permissions. Use absolute mode.



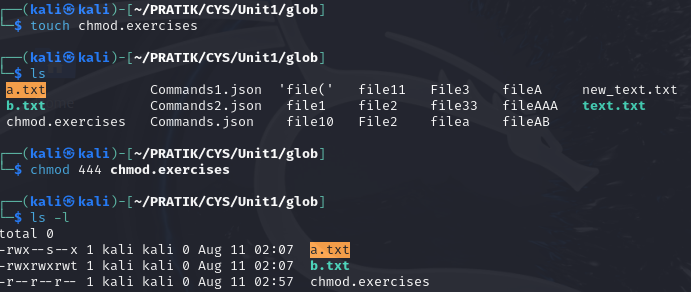
1. What are the default permissions for the new file?

The default permissions for a new file are typically ‘rw-r--r--’(644)

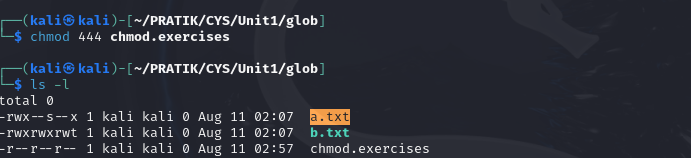
1. What was the command to view the file permissions?



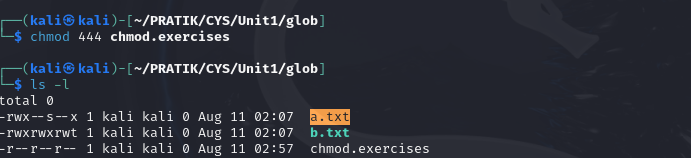
1. Change chmod.exercises permissions to -r--r--r—



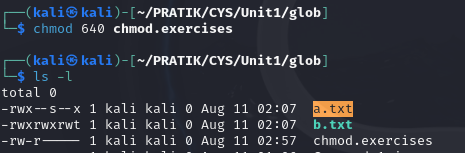
1. Change the file permissions to Read only for the owner, group and all other users.



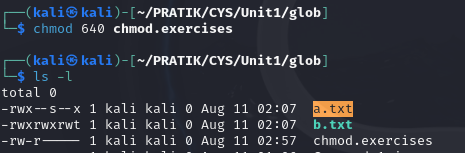
1. What was the command for changing the file permissions to -r--r--r--?



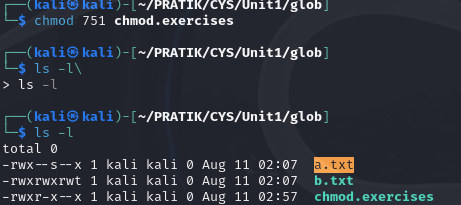
1. Change chmod.exercises permissions to -rw-r-----



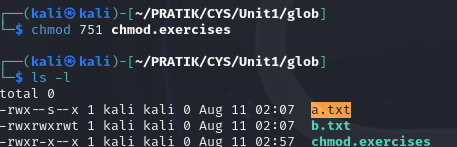
1. Change the file permissions to match the following:
   1. owner: Read and Write
   2. group: Read
   3. other: no permissions (None)



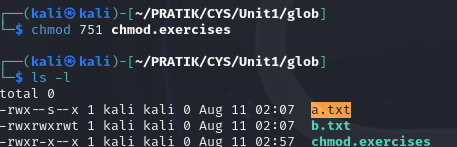
1. What was the command for changing the file permissions to -rw-r-----?



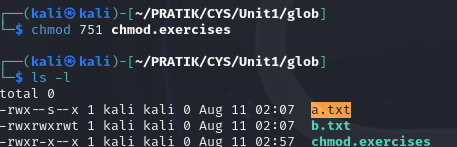
1. Change chmod.exercises permissions to -rwxr-x—x



1. Change the file permissions to match the following:
   1. owner: Read, Write and Execute
   2. group: Read and Execute
   3. other: Execute



1. What was the command for changing the file permissions to -rwxr-x--x?



Evaluation :

Marks : 10 (Deadline : 4 – Originality :3 – Completeness :3 )

Deadline: 06.08.2024

In life there are no shortcuts. All things are connected. For success there is no fast lane. Work hard. Focus your energy, practice, remain honest, Truthful, loyal and committed.

-unknown