**VYSHAK S BEKAL**

**241059045**

**LINUX OS & SCRIPTING LAB**

**M.E – CYBER SECURITY**

**Topic Name:** 2\_lab\_2024\_Glob\_chmod\_pathname

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 | Used to store the files needed to boot the operating system |
| 4 | /dev | Hardware interface files and kernel interfaces |
| 5 | /etc | Configuration files |
| 6 | /home | User specific files |
| 7 | /lib | Essential shared libraries and kernel modules |
| 8 | /proc | Process filesystem |
| 9 | /sbin | System binaries |
| 10 | /tmp | Temporary files |
| 11 | /var | Variable data |
| 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 | Screen shot |
| Regular file | - | Used to store regular content | Touch “filename” |  |
| Text file | - | Used to store regular contents in text format | Touch “filename” |  |
| Compressed file | - | Used to compres file size | gzip “filename” |  |
| Image |  |  | NA |  |
| Directory | d | Used in purpose of organizing and sharing files |  |  |
| Block file |  |  | NA |  |
| Character file |  |  | NA |  |
| Socket file |  |  | NA |  |
| pipe file |  |  | NA |  |

1. Globbing
2. Go back to CYS
3. Create multiple subdirectories using single command

LS

Unit1

Command

glob

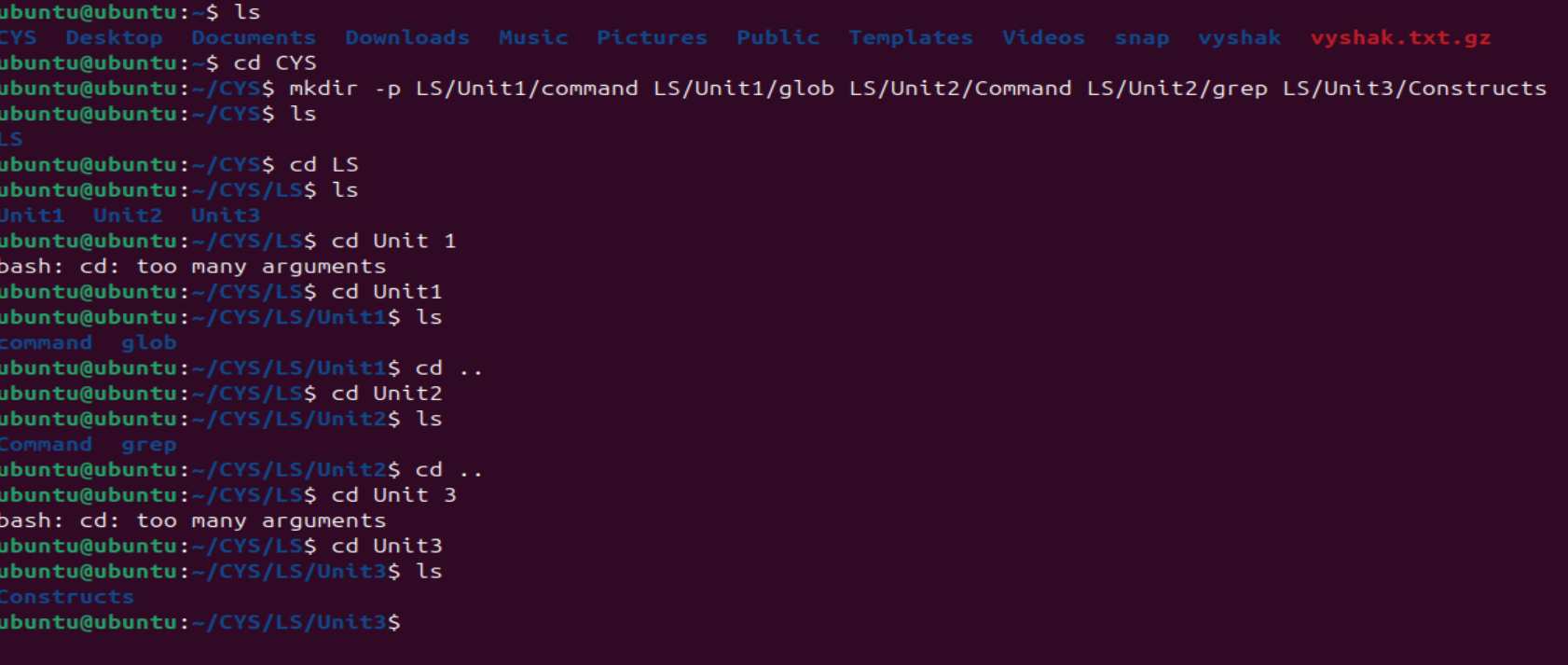
Unit2

command

grep

Unit3

Constructs



1. Navigate to unit1/glob
2. Create the following files :

Commands.txt

Commands1.txt

Commands2.txt

page1.html

page2.html

page3.html

file1

file10

file11

file2

File2

File3

file33

fileAB

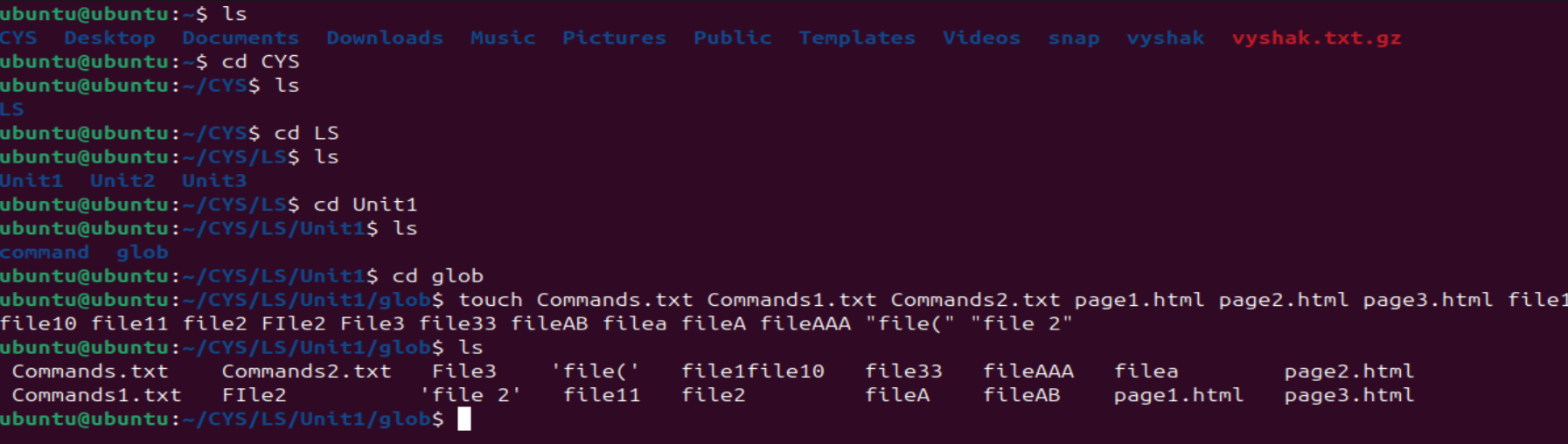
filea

fileA

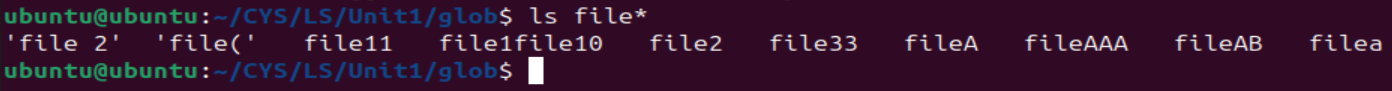
fileAAA

file(

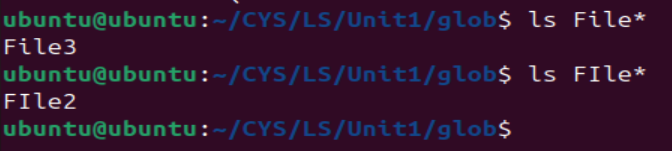
file 2



* 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.



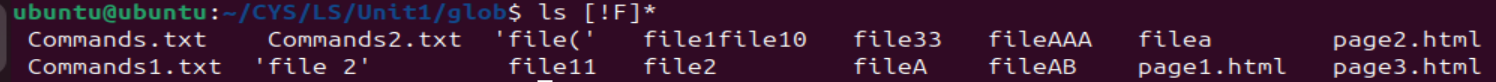
* + 1. List (with ls) all files that start with f or F and end with 3 or A.



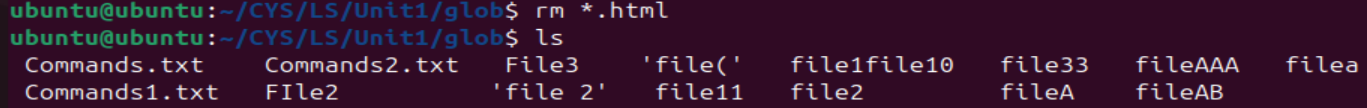
* + 1. List (with ls) all files that start with f have i or R as second character and end in a number.



* + 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 \* |  |
| ? | Matches a single character. | ls ?file\* |  |
| [ ] | Matches any single characterwithinbrackets. | ls file[0-9]\* |  |
| [! ] | Matches any character that is not a member of the set characters | ls [!F] |  |
| { } | Matches a comma-separated list of values | ls {file3, filea} |  |

More on Character class

|  |  |
| --- | --- |
| Notation | Use |
| [:alnum:] | Matches any alphanumeric character. |
| [:alpha:] | Matches any alphabetic character. |
| [:digit:] | Matches any digit. |
| [:lower:] | Matches any lowercase letter. |
| [:upper:] | Matches any uppercase letter. |

4. change permission

1. work/readme.txt - User can read/write/execute (absolute mode):

chmod 700 /work/readme.txt

1. /work/readme.txt - Any user can read, group can read/write, user can read/write/execute (absolute mode):

chmod 764 /work/readme.txt

1. /bin/ - User can read/write/execute, group and others can execute, runs with owner privileges (absolute mode):

chmod 4751 /bin/

1. /work/readme.txt - User can read/write/execute (relative mode):

chmod u+rwx,go-rwx /work/readme.txt

1. /work/readme.txt - Any user can read, group can read/write, user can read/write/execute (relative mode):

chmod u+rwx,g+rw,o+r /work/readme.txt

1. /work/readme.txt - User can read/write/execute, group and others can execute, runs with group privileges (absolute mode):

chmod 2755 /work/readme.txt

1. /work/readme.txt - Only the owner can rename or delete, maintaining existing permissions (absolute mode):

chmod 700 /work/readme.txt

1. Default permissions for a new file:

Typically -rw-r--r-- (umask 022).

1. Command to view file permissions:

ls -l

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

chmod 444 chmod.exercises

1. Change file permissions to Read-only for owner, group, and others:

chmod 444 filename

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

chmod 640 chmod.exercises

1. Command for changing file permissions to -rw-r-----:

chmod 640 filename

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

chmod 751 chmod.exercises

1. Command for changing file permissions to -rwxr-x--x:

chmod 751 filename