BASH SCRIPTING

1. INTRODUCTION……………………………………………………………………………………………………….2

2. SCRIPTING………3

3. **FILE SYSTEM AND PERMISSIONS**…………………………………………………………………...4

    a. FILE SYSTEM……………………………………………………………………………………………4

    b. PERMISSIONS…………………………………………………………………………………………..4

    c. OWNERSHIP……………………………………………………………………………………………5

    d. CHANGING PERMISSIONS……………………………………………………………………..6

INTRODUCTION

- Bash is an acronym for the 'Bourne Again SHell'.
- Bash is a command language interpreter which is a default interpreter on many GNU/Linux Systems.
- It is an open source version of the Bourne Shell and was first released in 1989.
- Other shell interpreters are Korn Shell, C shell etc.
- Bash scripts are used by Systems Administrators, Programmers, Network Engineers, Scientists and just about anyone else who uses a Linux/ Unix system regularly.

SCRIPTING

- A script for a computer tells the computer what it should do or say.
- Bash scripts we are telling the Bash shell what it should do.
- A Bash script is a plain text file which contains a series of commands.
- Anything you can run normally on the command line can be put into a script and it will do exactly the same thing and vice versa.
- Instead of executing commands one by one, scripting allows us to execute commands automatically.
- we can write the commands line by line in a text editor and save it with an extension '.sh'.
- We can run bash script as follows:
    - Open a text editor and type commands in it. Then save with ‘.sh’ extension.
    - Then type chmod +x <filename.sh>
    - Type ./ <filename.sh>
- While scripting we have to specify the interpreter explicitly (since many shell interpreters are there).
- To get the full path, use the command ‘which’ and prefix the obtained path with a shebang #!.
- This must be the first line of the script.

FILE SYSTEM AND PERMISSIONS

File System

- A File system is a structure organized with a collection of files and folders.
- Linux file system is a tress like structure consists of lot of directories.
- Directories are just files consisting of list of other files.
- In Linux there is no difference between directories and files, there the files are known as directories.
- In Linux the files are categorized as follows:
    - Ordinary Files : Contains data, text, program instructions and images,
    - Special Files : These files give access to hardware devices.
    - Directories : Contains both ordinary and special files.

Permissions

- Linux-based Operating System requires file permissions to secure its filesystem, as there are file permission-based issues that occur when a user assigns improper permissions to the files and directories.

- These issues may cause malicious or accidental tampering to the filesystem.
- There are three types of permissions associated with the files as follows:
    - Read (r) : The permission by which you can view the contents of the file.
    - Write (w) : The permission by which you can modify the content.
    - Execute (x) : The permission by which one can run the programming file or script.

Ownership

- Three types of Linux users are there :
    - Owner : Owner is the superuser who creates the file. He can access all the permissions associated with a file that includes reading, modifying, and running the file.
    - Group : Group is known as a set of users or multi-users. The superuser creates it. Every member in a group has the same access permissions associated with a file.
    - Other Users : The third-party users can be anybody else who doesn't belong to the Superuser/Group members. They use the permissions associated with any file or directory which are created or owned by the Superuser/Group members.

Changing Permissions

- We can alter the file permissions of each class (owner/group/others) using **chmod** command.
- The way to do it is :
  - chmod [**class**][operator][permission] file_name
  - **class** is represented by the indicators - **u, g, o,** and **a** such that u for the user, g for the group, o for the other, and **a** for all the classes.
  - **operator** (+ or -) is used to add or remove the permission.
  - **permission** is represented by the indicators **r, w, x** to allow access for reading, modifying, or running the script respectively.