Linux

Process Management

A process is an instance of a program currently running on a computer system. It involves controlling and monitoring the processes running on a Linux system, including managing process resources, scheduling processes to run on the CPU, and terminating processes when necessary.

There are basically 2 types of processes.

1. Foreground processes:

- This processes are also known as interactive processes.
- This processes is executed or initiated by user, they cannot be initialized by system services.
- Such processes take input from the user and return the output.
- While these processes are running we can not directly initiate a new process from the same terminal.
- A foreground process would be like you are running an Office application on the Linux system.

2. **Background processes:**

- This processes are also known as non-interactive processes.
- These are the processes that are to be executed or initiated by the system itself.
- Processes can be system processes or user processes. System processes are initiated by the kernel, while users initiate User Processes.

Process management can help organizations improve their operational efficiency, reduce costs, increase customer satisfaction, and maintain compliance with regulatory requirements. It involves analyzing the performance of existing processes, identifying bottlenecks, and making changes to optimize the process flow.

States of Process :

- New**:** Newly Created Process (or) being-created process.
- Ready**:** After the creation process moves to the Ready state, i.e. the process is ready for execution.

- Run**:** Currently running process in CPU (only one process at a time can be under execution in a single processor)
- Wait (or Block): When a process requests I/O access.
- Complete (or Terminated): The process completed its execution.

Identifying and Terminating Processes in Linux:

Identification of processes and termination of processes can be done by

1.The ps command is used to find the process ID of the process you want to manage.

2. The kill command in Linux is used to terminate processes by their process IDs.

3. The killall command in Linux is used to terminate processes by their names.

Commands on Process Management:

sleep : This command will be executed in the terminal and we would be able to execute another command after the execution of the above command.

jobs : To get the list of jobs that are either running or stopped.

bg : To run all the pending and force stopped jobs in the background.

`ps -ef | grep sleep` : To get details of a process running in background.

fg : To run all the pending and force stopped jobs in the foreground.

top : To get the list of all the running processes on your Linux machine.

free : Shows the status of your RAM.

nice : Adjusts the priority of a process.

kill : Terminates a process by sending a signal to it.

ps : Displays information about the processes running currently.

pidof : Shows the Process ID of a process.

Disk and File Management in Linux

1. File Management

File management in Linux refers to organizing, manipulating, and controlling files within the Linux operating system. It includes creating, listing, copying, moving, renaming, and deleting files. Understanding the different types of files and their characteristics is crucial for effective file management in Linux.

In Linux, most of the operations are performed on files. And to handle these files Linux has directories which are maintained in a tree-like structure. Linux has 3 types of files:

- **Regular Files:**

Regular files can store text, images, audio, video, executables, and more. For example, a text file named example.txt an image file named image.jpg are considered regular files.

- **Directories:**

These are the files that store the list of file names and the related information. One can create file by using mkdir command. For instance, a directory named documents can contain multiple files such as linux.doc and presentation.ppt, as well as other directories like images and videos.

- **Special Files:**

It Represents a real physical device such as a printer which is used for IO operations.

There are different types of files which are essential for effective file management in Linux and for interacting with the files and directories within the operating system .

File Listing:

To list the files in a directory in Linux, you can use the ls command. The ls command displays a list of files and directories in the current directory.

File Creating:

To create a new file in Linux, you can use the touch command followed by the desired file name.

Displaying File Contents:

The cat command displays the file contents in the terminal.

Coping a File:

The cp command serves this purpose by enabling you to specify the source file and the destination for the copy.

Moving a File:

The mv command is used for this operation, allowing you to specify the current file location and the target directory where you want to move it.

Commands on File Mangement:

cat : This command is used to input standard output into a file.
more : It allow the user to view the contents of a file one screenful at a time.
cp : This command is used to copy the contents of one file to another.
mv : This command allows a user to move or rename a file.
touch : this command to create or update a file.
less : It also includes some extra features such as 'adjustment in the terminal.
head : This command allows a user to output the first 10 lines of any file.
tail : This command allows the user to view the last 10 lines of any file.
du : check the disk space with the help of the du command.
df-h : shows disk space in human-readable format.
Ls –al : lists the entire contents, along with their size, of a particular directory.
fdisk -l : It will list the known partition types.

Networking and networking stack in Linux

A network, whether it is a LAN or WAN, can be thought of as having two parts: computers, and everything that goes between the computers. This focuses on connectivity: firewalls, wireless access points, secure remote administration, remote helpdesk, remote access for users, virtual private networks, authentication, system and network monitoring.

The Network Stack is what allows the applications to be able to access a network through a physical networking device. Networking devices can be modems, cable modems, Ethernet cards, etc.

The Network Stack works in two ways. The first way is when a user makes a request to the network, such as for a file from a network server. The second way is when the request is fulfilled, the file is returned to the user. The Linux network stack contains layers as:

- Application Layer
- System call Interface
- Protocol Agnostic Interface
- Network Protocol
- Device Agnostic Interface
- Device Drivers
- Physical Hardware

There are 7 layers in network stack. The various layers each exist within a set. There are three sets, the User Space, Kernel Space and the Physical Layer. They are grouped as follows:

- The top most layer (Application Layer) is part of the User Space.
- The next five layers are the Kernel Space.
- The final layer is the Physical Layer.

Some Commands on Networking:

host - Performs DNS lookups.
arp - View or add contents of the kernel's ARP table.
iwconfig - Used to configure wireless network interface.
hostname - To identify a network name.
curl or wget - To download a file from internet.
Fconfig - Display and manipulate route and network interfaces.
ip - It is a replacement of ifconfig command.(ip a, ip addr)
Traceroute - Network troubleshooting utility.
tracepath - Similar to traceroute but doesn't require root privileges.
Ping - To check connectivity between two nodes.
netstat - Display connection information.
ss - It is a replacement of netstat.
dig - Query DNS related information.

1**.Application Layer:**

It comes to the TCP and the UDP protocols. On the application layer, applications like different browser or other application which accesses the Internet work in this layer.

2.System Call Interface :

The call is made from the Application Layer to the Kernel. For example of using a word processor to request a file from a network location. In this case, we can send a read call to the Kernel from the word processor.

3**. Protocol Agnostic Interface:**

It is where the socket is created. The socket is a basic communications that is "talking" or "listening". Each socket has an ID which is used specifically for an application.

4. Network Protocol :

It is responsible for how the data is sent or received. The layer controls the information for getting data across the network.

5. Device Agnostic Interface :

It is used to connect the data from/to the Kernel and the network device drivers.

6. Device Drivers:

the device driver of the network device being used. This allows the data to be prepared for transmission over the medium from the network device.

7. Physical Hardware:

The Physical Hardware layer is also known as the hardware or network hardware. It is the actual network hardware. This is where the data packets are transmitted and received from the network medium being used, whether cable or wireless.