

## Laboratorio 2

### Introducción

La presente práctica de laboratorio se tenía como objetivo conocer y aprender acerca de el sistema operativo Linux. En la primera parte del laboratorio se inició la computadora en Linux y se probaron algunos de los comandos básicos en la terminal. En la segunda parte del laboratorio se instaló Visual Studio Code desde la terminal y se compilaron y ejecutaron programas para observar que efecto tenían en los procesos del sistema.

### Incisos

Prime Parte:

Inciso 4:

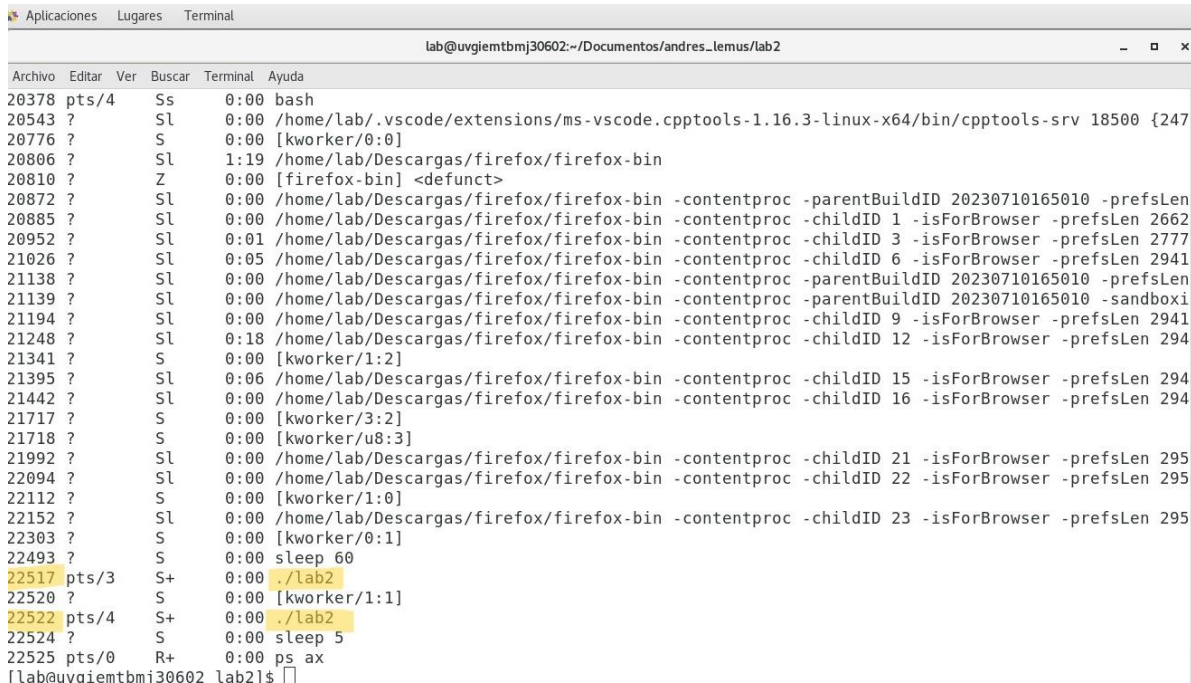
- **man:** Este comando se usa para acceder a las páginas del manual del sistema, donde "man" significa "manual" y permite a los usuarios ver documentación e información detallada sobre varios comandos, llamadas al sistema, funciones de biblioteca, archivos de configuración, entre otros. Proporciona una vista detallada del comando que incluye su nombre, sinopsis, descripción, opciones, estado, valores a devolver, errores, archivos, versiones, ejemplos, autores, etc. `man ls` despliega el manual para el comando `ls`, que significa `list`.
- **ls:** se utiliza para listar archivos y directorios en un directorio específico. Proporciona información sobre archivos, directorios y sus atributos. `ls -l` proporciona información extra acerca de los archivos y directorios, y `ls -la` lista incluso los archivos y directorios escondidos.
- **pwd:** Muestra la ruta del directorio de trabajo actual, comenzando desde la raíz (root). `PWD` significa *Print Working Directory*.
- **cd:** Se utiliza para cambiar el directorio de trabajo actual a diferentes directorios en el sistema. Para usarlo, se escribe "cd" seguido del nombre del directorio o ruta al que desea cambiar.
- **mkdir:** Se utiliza para crear nuevos directorios (carpetas) en el sistema. Significa "make directory". Dependiendo de donde se quiera crear la carpeta se necesitan los permisos apropiados.
- **cp:** Se utiliza para copiar archivos o grupos de archivos o directorios de un lugar a otro. Significa "copy". El comando `cp` requiere al menos dos nombres de archivo en sus argumentos.

- **rm:** Se usa para eliminar objetos como archivos, directorios, enlaces simbólicos, etc. del sistema. Por sí mismo no puede eliminar directorios. Se debe tener mucho cuidado al ejecutar este comando ya que una vez se eliminan los archivos, no podrá recuperar el contenido de los archivos y directorios. Significa "remove".
- **mv:** Se utiliza para mover o renombrar directorios y archivos de una ubicación a otra dentro de un sistema. Significa "move".
- **clear:** Se utiliza para limpiar la pantalla del terminal, eliminando todos los comandos anteriores y la salida de la ventana del terminal. Este comando ignorará cualquier parámetro de línea de comandos que pueda estar presente.
- **ps:** Se utiliza para enumerar y mostrar información acerca de los procesos que se están ejecutando actualmente y sus PID. Significa "process status". Algunas de las opciones que se usan junto a este comando están ps ax que muestra los procesos suyos y de otros usuarios.
- **top:** Es un comando de monitoreo que proporciona una vista en tiempo real del sistema en ejecución. Este comando muestra la información resumida del sistema y la lista de procesos o subprocesos que actualmente administra el Kernel de Linux, además contendrá las estadísticas de procesos y uso de recursos.
- **who:** Se usa para mostrar información sobre los usuarios que actualmente están conectados al sistema, despliega información como el nombre de usuario, terminal o sesión de inicio de sesión, la fecha y hora en que iniciaron sesión y la dirección IP desde la que están conectados.
- **date:** Se utiliza para mostrar la fecha y la hora del sistema además de poder proporcionar información extra sobre la fecha y la hora actuales en función del reloj del sistema. Se puede utilizar también para configurar la fecha y la hora del sistema. De forma predeterminada, el comando de fecha muestra la fecha en la zona horaria en la que está configurado el sistema operativo. Debe ser el super usuario para poder cambiar la fecha y la hora.
- **ifconfig:** Este comando se usa para configurar las interfaces de red que se encuentran en el kernel. Se utiliza en el momento del arranque para configurar las interfaces según sea necesario. El comando se usa para asignar la dirección IP y la máscara de red a una interfaz o para habilitar o deshabilitar una interfaz determinada. Las versiones más recientes de algunas distribuciones de Linux no tienen el comando ifconfig preinstalado, en su lugar se puede utilizar el comando ip.

Segunda parte:

## Inciso 14

En la práctica se realizó un programa que contara de 1 a 20 con un segundo entre cada conteo y que se mostrará en la terminal. El mismo programa se corrió en dos terminales distintas, y luego en una tercera terminal escribir el comando `ps ax` para poder los procesos en todos los usuarios.



```
lab@uvgiemtmbmj30602:~/Documentos/andres_lemus/lab2
Archivo Editar Ver Buscar Terminal Ayuda
20378 pts/4 Ss 0:00 bash
20543 ? Sl 0:00 /home/lab/.vscode/extensions/ms-vscode.cpptools-1.16.3-linux-x64/bin/cpptools-srv 18500 {247
20776 ? S 0:00 [kworker/0:0]
20806 ? Sl 1:19 /home/lab/Descargas/firefox/firefox-bin
20810 ? Z 0:00 [firefox-bin] <defunct>
20872 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -parentBuildID 20230710165010 -prefsLen
20885 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 1 -isForBrowser -prefsLen 2662
20952 ? Sl 0:01 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 3 -isForBrowser -prefsLen 2777
21026 ? Sl 0:05 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 6 -isForBrowser -prefsLen 2941
21138 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -parentBuildID 20230710165010 -prefsLen
21139 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -parentBuildID 20230710165010 -sandboxi
21194 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 9 -isForBrowser -prefsLen 2941
21248 ? Sl 0:18 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 12 -isForBrowser -prefsLen 294
21341 ? S 0:00 [kworker/1:2]
21395 ? Sl 0:06 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 15 -isForBrowser -prefsLen 294
21442 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 16 -isForBrowser -prefsLen 294
21717 ? S 0:00 [kworker/3:2]
21718 ? S 0:00 [kworker/u8:3]
21992 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 21 -isForBrowser -prefsLen 295
22094 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 22 -isForBrowser -prefsLen 295
22112 ? S 0:00 [kworker/1:0]
22152 ? Sl 0:00 /home/lab/Descargas/firefox/firefox-bin -contentproc -childID 23 -isForBrowser -prefsLen 295
22303 ? S 0:00 [kworker/0:1]
22493 ? S 0:00 sleep 60
22517 pts/3 S+ 0:00 ./lab2
22520 ? S 0:00 [kworker/1:1]
22522 pts/4 S+ 0:00 ./lab2
22524 ? S 0:00 sleep 5
22525 pts/0 R+ 0:00 ps ax
lab@uvgiemtmbmj30602 lab21$
```

Figura 1. Procesos corriendo en el sistema.

Al observar la Figura 1 se puede notar que hay dos procesos que están corriendo el mismo ejecutable el cual es `lab2`. Hasta la izquierda se pueden ver números, estos números son distintos para ambos procesos a pesar de que están corriendo el mismo ejecutable. Ese número es el PID (Process ID) que es único para cada proceso. En conclusión, se puede decir que a pesar de que están ejecutando el mismo código y programa, al ser dos terminales distintas son dos procesos distintos con su propio identificador.

## Referencias

Información de los comandos obtenidos de:

Team GeeksforGeeks. (2023). Linux Commands. <https://www.geeksforgeeks.org/linux-commands/>