	Carátula para entrega de prácticas	
Facultad de Ingeniería	Laboratorio de docencia	

Laboratorios de computación salas A y B

Profesor: Adrián Ulises Martínez Mercado

Asignatura: Fundamentos de Programación

Grupo: 7

No de Práctica(s): #2

Integrante(s): HERNÁNDEZ GONZÁLEZ ANDRÉS SEBASTIÁN

*No. de Equipo de
cómputo empleado:* #26

No. de Lista o Brigada: Brigada #6

Semestre: 1

Fecha de entrega: 29 de Septiembre de 2021

Observaciones:

CALIFICACIÓN: _____

Índice:

Carátula.....	1
Objetivo	3
Introducción.....	3
Desarrollo.....	3-11
Conclusión.....	11
Referencias.....	11

PRÁCTICA #2 GNU LINUX

Objetivo:

- El alumno identificará al sistema operativo como una parte esencial de un sistema de cómputo. Explorará un sistema operativo GNU/Linux con el fin de conocer y utilizar sus comandos básicos.

Introducción:

Los Sistemas operativos son una parte importante entre el o los usuarios y el kernel que a su vez se comunica de la manera más básica con el hardware, de otra manera esta interacción sería demasiado técnica y muy pocos, como lo era hace un tiempo, podrían utilizar una computadora ya no solo para programar sino también con actividades de carácter más administrativas, recreativas, económicas, etc.

A lo largo de la historia los Sistemas operativos han tenido un desarrollo constante siendo cada vez más útiles para los tipos de usuarios a quienes están destinados. De estos podemos destacar a los basados en el kernel Linux y proyecto GNU porque su principal ventaja, con respecto a otros, es que son de código abierto, implicando que cualquiera pueda modificarlo a lo que necesite, ya sea para administrar un servidor, montar una supercomputadora, controlar redes, ciber-seguridad, entornos de programación, creación de nuevos sistemas operativos, desarrollo de aplicaciones y un sin fin de utilidades por lo cual resulta importante conocer un poco más sobre dichos sistemas hechos para el tipo de usuario que podrá ser un programador o desarrollador, en pocas palabras, estamos ante un compendio de herramientas.

Desarrollo:

A lo largo de la práctica se trabajó con una terminal o consola de en la cual se pueden ejecutar una gran variedad de comandos, para ello, se instaló previamente el programa de "SSH Secure Shell Client". Para acceder a la consola, primero es necesario iniciar el programa y hacer clic en el pequeño ícono de la computadora que indica "Connect". (Imagen 1)

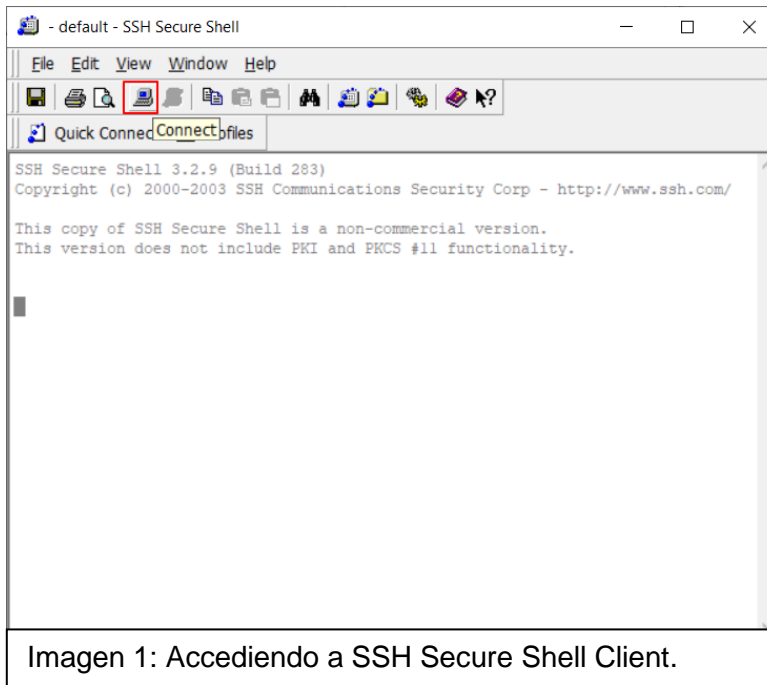


Imagen 1: Accediendo a SSH Secure Shell Client.

En caso de tener Linux ya instalado, no es necesario instalar el programa “SSH Secure Shell Client” puesto que Linux tiene su propia consola de comandos.

Retomando la práctica, luego de hacer clic en dicho ícono, saldrá la ventana siguiente:

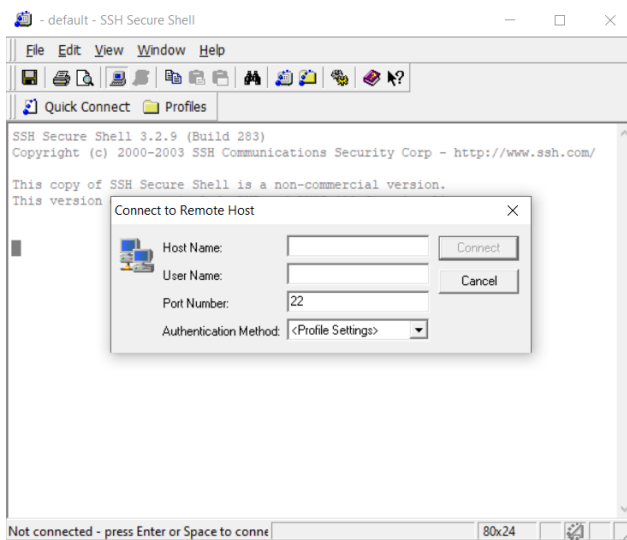
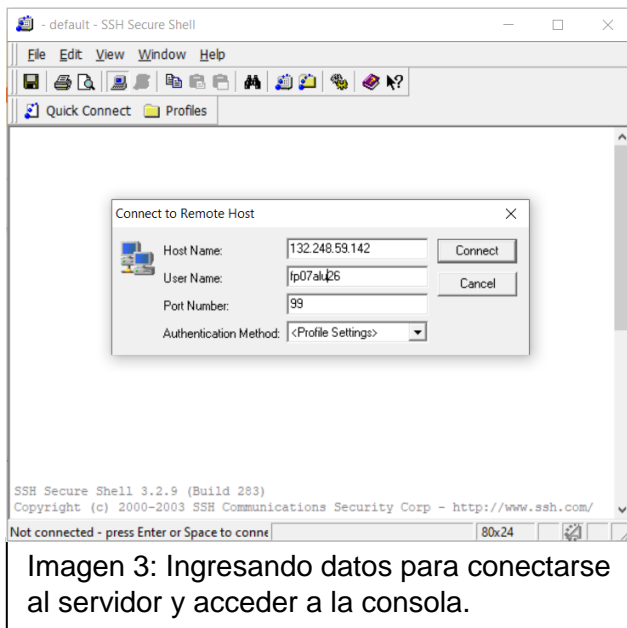


Imagen 2: Ventana emergente para conectarse al servidor.

De la cual se llenan el apartado de “Host name” con el número 132.248.59.142, el “User Name” que es nombre de usuario fp07alu## (## es número de lista asignado a cada alumno que entregó el profesor previo a la práctica) y “Port Number” 99. Por último, se selecciona el recuadro “Connect” para así, estar conectado al servidor de la facultad.



Una vez se accedió al servidor, debe aparecer la siguiente ventana, la cual confirma que se conectó al servidor correctamente y se procede a hacer clic en el botón “Yes”:



Se ingresa la contraseña, que es el número de cuenta del alumno y ahora ya nos encontramos en la interfaz de la consola, en la cual ya se puede escribir los diversos comandos.


```
[fp07alu26@samba ~]$ ls -l
total 194
lrwxrwxrwx. 1 root root 7 jul 31 2015 bin -> usr/bin
dr-xr-xr-x. 5 root root 1024 jul 31 2015 boot
drwx----- 6 web_dba web_dba 4096 ago 8 2015 database
drwxr-xr-x 17 root root 3480 sep 22 12:52 dev
drwxr-xr-x. 147 root root 12288 sep 22 12:52 etc
drwxr-xr-x. 6 root root 4096 feb 26 2016 home
lrwxrwxrwx. 1 root root 7 jul 31 2015 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 jul 31 2015 lib64 -> usr/lib64
drwx----- 2 root root 16384 jul 31 2015 lost+found
drwxr-xr-x 2 root root 40 sep 22 12:52 media
drwxr-xr-x. 2 root root 4096 feb 3 2012 mnt
drwxr-xr-x. 3 root root 4096 feb 3 2012 opt
dr-xr-xr-x 294 root root 0 sep 22 12:52 proc
dr-xr-x--- 28 administrador administrador 4096 sep 22 12:53 root
drwxr-xr-x 38 root root 1380 sep 22 19:26 run
lrwxrwxrwx. 1 root root 8 jul 31 2015/sbin -> usr/sbin
drwxr-xr-x 3 root root 4096 feb 18 2019 sitio
drwxr-xr-x. 4 root root 4096 abr 3 2020 software
drwxr-xr-x. 2 root root 4096 feb 3 2012 srv
drwxr-xr-x 13 root root 0 sep 22 12:52 sys
drwxrwxrwt. 316 root root 114688 sep 22 16:42 tmp
drwxr-xr-x. 59 root root 4096 sep 19 03:56 users
drwxr-xr-x. 13 root root 4096 jul 31 2015 usr
drwxr-xr-x. 23 root root 4096 sep 17 2015 var
drwx----- 16 webserver webserver 4096 mar 16 2021 webserver
```

Imagen 7: Comando ls -l que muestra lista larga del directorio actual y cada una de sus propiedades.

Por otra parte, el comando ls / ,lista el contenido de los elementos del sistema operativo o que se encuentren en la raíz, de los cuales parten otras carpetas, programas, aplicaciones, etc.

```
[fp07alu26@samba ~]$ ls /
bin boot database dev etc home lib lib64 lost+found media mnt opt proc root run/sbin sitio software srv sys tmp users usr var webserver
```

Imagen 8: Comando ls / ,muestra elementos del sistema operativo.

Para ver los usuarios del equipo local, se emplea el comando ls -l /home:

```
[fp07alu26@samba ~]$ ls -l /home
total 16
drwx----- 10 administrador administrador 4096 sep 22 22:44 administrador
drwx----- 4 mena mena 4096 sep 9 2015 mena
drwx----- 4 MNR MNR 4096 oct 5 2015 MNR
drwx----- 4 squid squid 4096 ago 4 2015 squid
```

Imagen 9: Comando ls -l /home , despliega los usuarios del equipo local.

Por otro lado, hay un comando muy útil, el comando: man ls ,que permite visualizar la lista de todos los comandos que se pueden emplear.

```
[fp07alu26@samba /]$ man ls
LS(1) User Commands LS(1)

NAME
    ls - list directory contents

SYNOPSIS
    ls [OPTION]... [FILE]...

DESCRIPTION
    List information about the FILES (the current directory by default).
    Sort entries alphabetically if none of -cftuvSUX nor --sort is speciâ[m]
    fied.

    Mandatory arguments to long options are mandatory for short options
    too.

    -a, --all
        do not ignore entries starting with .

    -A, --almost-all
        do not list implied . and ..

    --author
        with -l, print the author of each file

Manual page ls(1) line 1 (press h for help or q to quit)
LS(1) User Commands LS(1)

NAME
    ls - list directory contents

SYNOPSIS
    ls [OPTION]... [FILE]...

DESCRIPTION
    List information about the FILES (the current directory by default).
    Sort entries alphabetically if none of -cftuvSUX nor --sort is speciâ[m]
    fied.

    Mandatory arguments to long options are mandatory for short options
    too.

    -a, --all
        do not ignore entries starting with .

    -A, --almost-all
        do not list implied . and ..

    --author
        with -l, print the author of each file

    -b, --escape
        print C-style escapes for nongraphic characters

    --block-size=SIZE
        scale sizes by SIZE before printing them. E.g.,
        '--block-size=M' prints sizes in units of 1,048,576 bytes. See
        SIZE format below.

    -B, --ignore-backups
        do not list implied entries ending with ~

    -c
        with -lt: sort by, and show, ctime (time of last modification of
        file status information) with -l: show ctime and sort by name
        otherwise: sort by ctime, newest first

    -C
        list entries by columns

    --color[=WHEN]
        colorize the output. WHEN defaults to 'always' or can be
        'never' or 'auto'. More info below
```

Imagen 10: Lista de comandos para usar, desplegados con el comando man ls

Otro comando visto es el ls .. ó ls ../ que dirige a la carpeta padre:

```
[fp07alu26@samba /]$ ls ..
bin      dev      lib      media    proc     sbin     srv      users    webserver
boot     etc      lib64    mnt      root     sitio    sys      usr
database home     lost+found opt      run      software [img] var
[fp07alu26@samba /]$ ls ../
bin      dev      lib      media    proc     sbin     srv      users    webserver
boot     etc      lib64    mnt      root     sitio    sys      usr
database home     lost+found opt      run      software [img] var
```

Imagen 11: Dirigiéndose a carpeta padre por medio del comando ls.. o ls ../

O localizar archivos por medio de una ruta alternativa y desde la carpeta padre con el comando `ls ../../usr`

```
[fp07alu26@samba /]$ ls ../../usr  
bin  etc  games  include  lib  lib64  libexec  local  sbin  share  src  tmp
```

Imagen 12: Accediendo a carpetas o archivos a través de la carpeta padre.

Posteriormente se pasó a la fase de crear archivos y carpetas. Se empezó accediendo a una carpeta de cada usuario con el comando `cd /users/fp07/fp07alu##` (## numero de lista), se creó un directorio con el comando `mkdir`, se usó el algoritmo `cd` para ubicarse en una carpeta, además se empleó un nuevo comando llamado `touch`, útil para crear archivos de cualquier tipo, por ejemplo texto (.txt), pdf (.pdf), programa c (.c), etc.

```
[fp07alu26@samba ~]$ cd practica02  
[fp07alu26@samba practica02]$ touch practica02.pdf  
[fp07alu26@samba practica02]$ mkdir uno  
[fp07alu26@samba practica02]$ mkdir dos
```

Imagen 13: Ubicarse en una carpeta con comando `cd`, creando archivo con el comando `touch` un archivo en la misma carpeta `practica02` y creando directorio con comando `mkdir`.

Después se pasó al directorio con nombre `uno` con el comando `cd`, en el cual se creó un archivo en programa en lenguaje C, se copió el contenido del archivo `programa2.c` al `programa3.c` con el uso del comando `cp` y se mostró el contenido de ese directorio.

```
[fp07alu26@samba practica02]$ cd uno  
[fp07alu26@samba uno]$ touch programa2.c  
[fp07alu26@samba uno]$ cp programa2.c programa3.c  
[fp07alu26@samba uno]$ ls  
programa2.c  programa3.c
```

Imagen 14: Posicionándose en directorio `uno`, creando archivo con extensión `.c`, copiando contenido de un archivo a otro y mostrando contenido del directorio.

Se prosiguió con más ejemplos, en este caso, se copió el contenido del `programa2.c` al directorio con nombre `dos` y con el nombre `programa4.c`

```
[fp07alu26@samba uno]$ cp programa2.c ../dos/programa4.c  
[fp07alu26@samba uno]$ ls  
programa2.c  programa3.c  
[fp07alu26@samba uno]$ cd ../uno
```

Imagen 15: Copiando contenido de programa a cierto directorio y con un nombre en específico.

Asimismo es posible cambiar el nombre de cierto directorio o archivo en caso de escribirlo mal, haciendo uso del comando mv, escribiendo el nombre original del directorio y seguidamente escribiendo el nombre nuevo deseado. Del mismo modo, este comando es útil para mover cierto archivo a cierta carpeta con mv ubicación_de_origen/ archivo ubicación_de_destino

```
[fp07alu26@samba ~]$ mkdir practicaa02  
[fp07alu26@samba ~]$ mv practicaa02 practica02
```

Imagen 16: Cambiando nombre de directorio con comando mv

Así como se han creado archivos, también es posible removerlos de las carpetas, con el comando rm nombre_de_archivo, de la misma forma es posible eliminar los directorios o carpetas con el mismo comando, pero con algunas instrucciones más: rm -R nombre_de_carpeta_o_directorio_a_eliminar. El comando clear sirve para despejar la pantalla de la consola y no tener todas los comandos que se han ejecutado.

```
[fp07alu26@samba ~]$ rm practica2.txt  
[fp07alu26@samba ~]$ rm practica.pdf  
[fp07alu26@samba ~]$ rm -R uno  
[fp07alu26@samba ~]$ clear
```

Imagen 17: Remover archivos y carpetas con el comando rm y limpiando la pantalla de consola con comando clear.

O si se cuenta con muchos archivos que inicien con las mismas palabras y se desean eliminar dichos archivos se emplea el mismo comando rm -R palabras_con las_que_inician_archivos_a_eliminar*

```
[fp07alu26@samba ~]$ rm -R prog*
```

Imagen 18: Eliminando múltiples archivos que inicien con mismas palabras

Finalmente se usó el comando find . -name nombre_de_archivo_a_buscar ,para poder saber la ubicación de ciertos archivos.

```
[fp07alu26@samba ~]$ find . -name programa2.c
./practica02/dos/programa2.c
./practica02/uno/programa2.c
```

Imagen 19: Encontrar la ubicación de archivos con comando find

Conclusiones:

Conclusión de brigada - El sistema operativo es de suma importancia para un equipo ya que sin él, el equipo no podría funcionar. Existe una variedad muy extensa de sistemas operativos pero los más conocidos son Windows, Unix, Linux y MacOS y a pesar de que a simple vista parecen ser muy diferentes, todos cumplen con un mismo objetivo: ayudar a que el usuario tenga una mejor relación con el equipo. Como brigada podemos concluir que se cumplieron los objetivos de esta práctica ya que identificamos y exploramos de una manera no tan superficial un sistema operativo y realizamos cambios en este, nos dimos cuenta de la importancia que tiene en un sistema de cómputo ya que se encarga de regular el funcionamiento de nuestra computadora y actúa como el intermediario entre la computadora y el usuario, a su vez, aprendimos a utilizar comandos de navegación, creación y edición del servidor al cual accedimos, los cuales son de gran ayuda y necesarios siempre que se realiza una actividad como esta.

Hernández González Andrés Sebastián - A través de la práctica se pudieron identificar una gran variedad de conceptos, entre ellos el del sistema operativo así como los elementos que lo constituyen, sistemas GNU/Linux, software libre, licencia GNU y el kernel; sin embargo, para el desarrollo de la práctica se hizo especial énfasis en el Shell del sistema GNU/Linux, en el cual se pudieron ejecutar una amplia gama de comandos con ayuda del programa SSH Secure Shell Client, el cual es una herramienta de gran ayuda para poder ejecutar cada uno de los comandos vistos a en el sistema operativo y conectándose a un servidor de vía remota. Una práctica bastante completa en cuestión de contenido e interesante en el desarrollo, puesto que se pudieron ver muchos comandos básicos relativamente “sencillos” pero con demasiada utilidad ya sea para explorar archivos, crearlos, copiarlos, removerlos, re direccionarlos, etc. De modo que afortunadamente, el objetivo principal de la práctica se pudo cumplir satisfactoriamente.

Referencias:

- Linux y GNU - Proyecto GNU - Free Software Foundation. (s.f.). n/a. Gnu.org. Recuperado de <https://www.gnu.org/gnu/linux-and-gnu.es.html>
- Soto, J. (2020). ¿Qué es el kernel y para qué sirve? GEEKNETIC. <https://www.geeknetic.es/Kernel/que-es-y-para-que-sirve>
- SISTEMAS OPERATIVOS. (s.f.). n/a. Areatecnologia.com. <https://www.areatecnologia.com/sistemas-operativos.htm>