

**Instituto**

**Politécnico**

**Nacional**

**Escuela Superior de Cómputo**

**Practica 1: Comandos UNIX y uso de terminal**

**Materia:**

Sistemas Operativos

**Grupo:**

2CM8

**Alumno:**

Castro Cruces Jorge Eduardo

**Fecha:**

Miércoles, febrero 12, 2020

**OBJETIVO**

* Conocer algunos de los comandos fundamentales de UNIX y aprender el uso del shell.

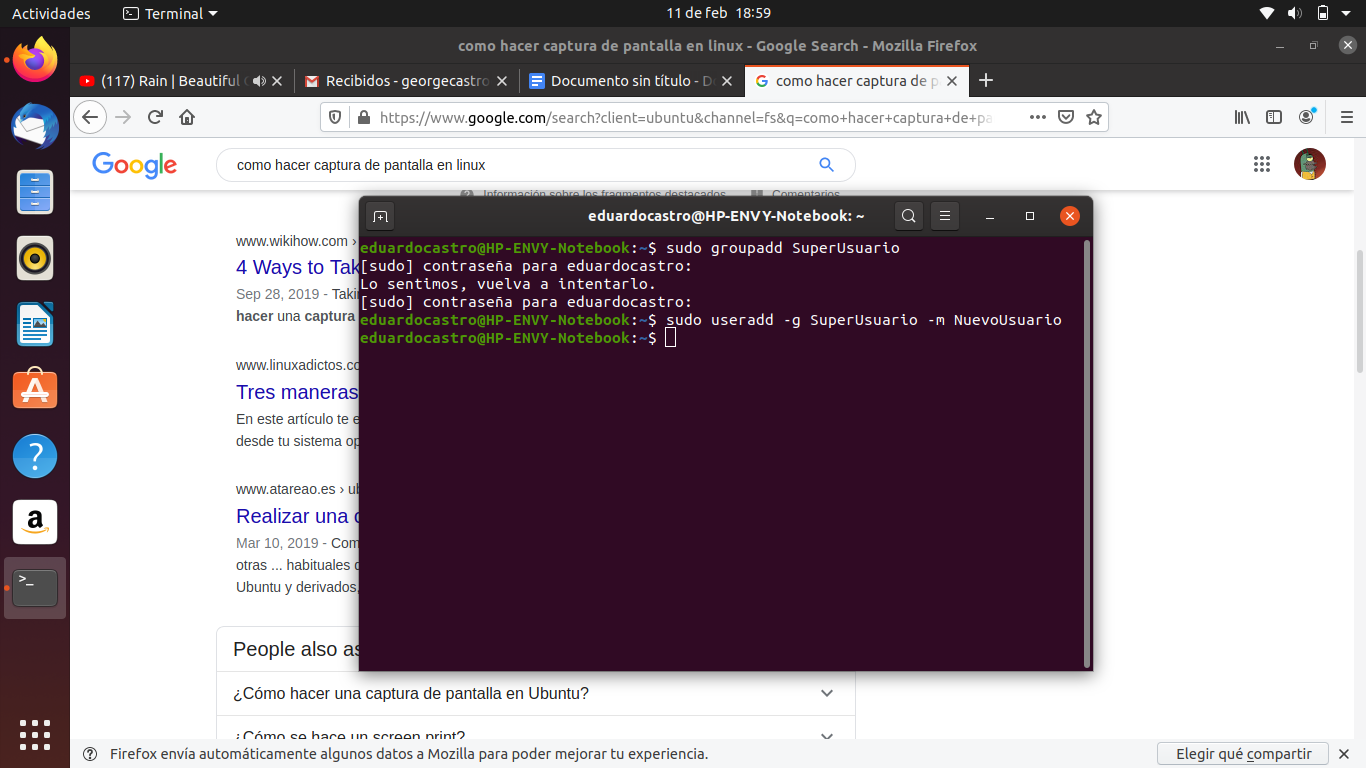
**INTRODUCCIÓN**

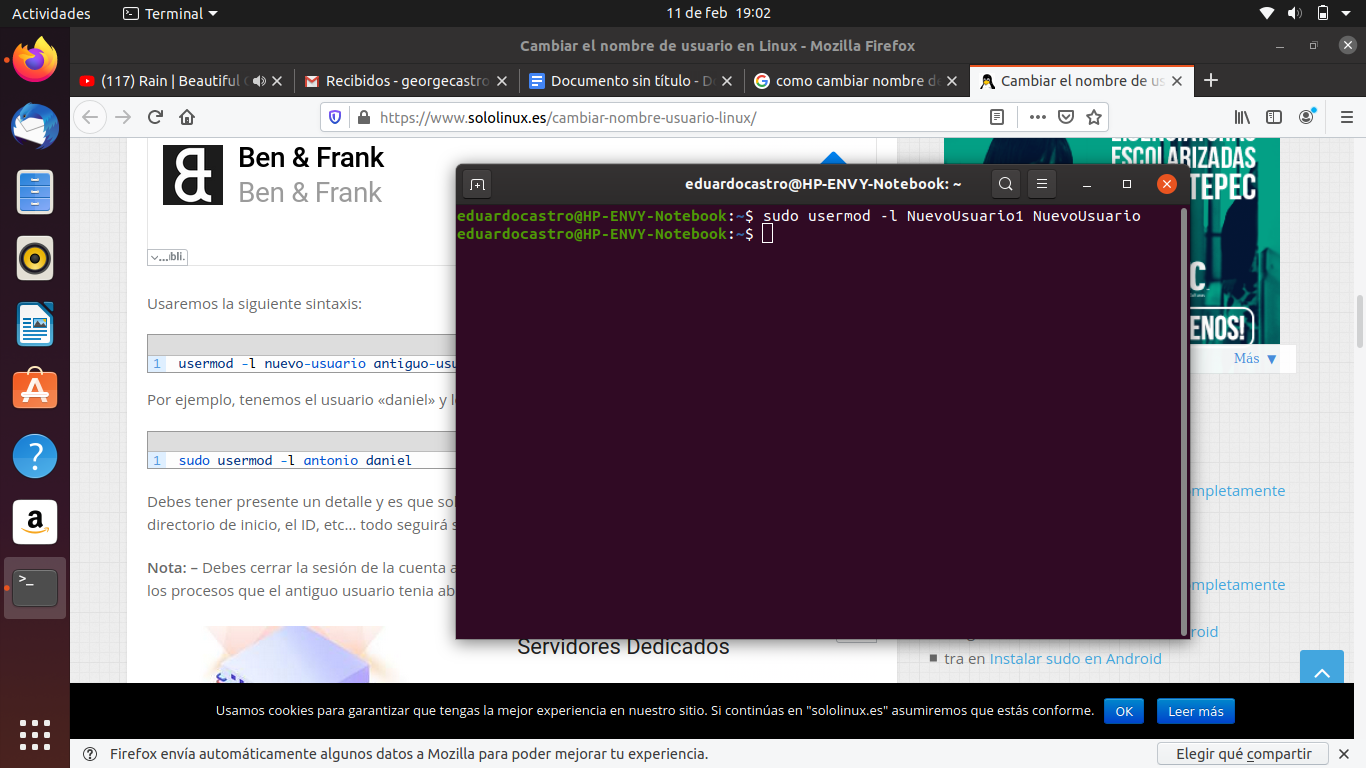
En esta práctica, se pretende conocer algunos de los comandos fundamentales de UNIX y aprender el uso del shell. Mediante la practica y ejecución dentro de la consola, para el manejo de archivos y concatenación de caracteres.

**DESARROLLO**

**Ejercicio 1**. Investiga el uso de los siguientes comandos y caracteres especiales, y muestra su funcionamiento en una terminal.

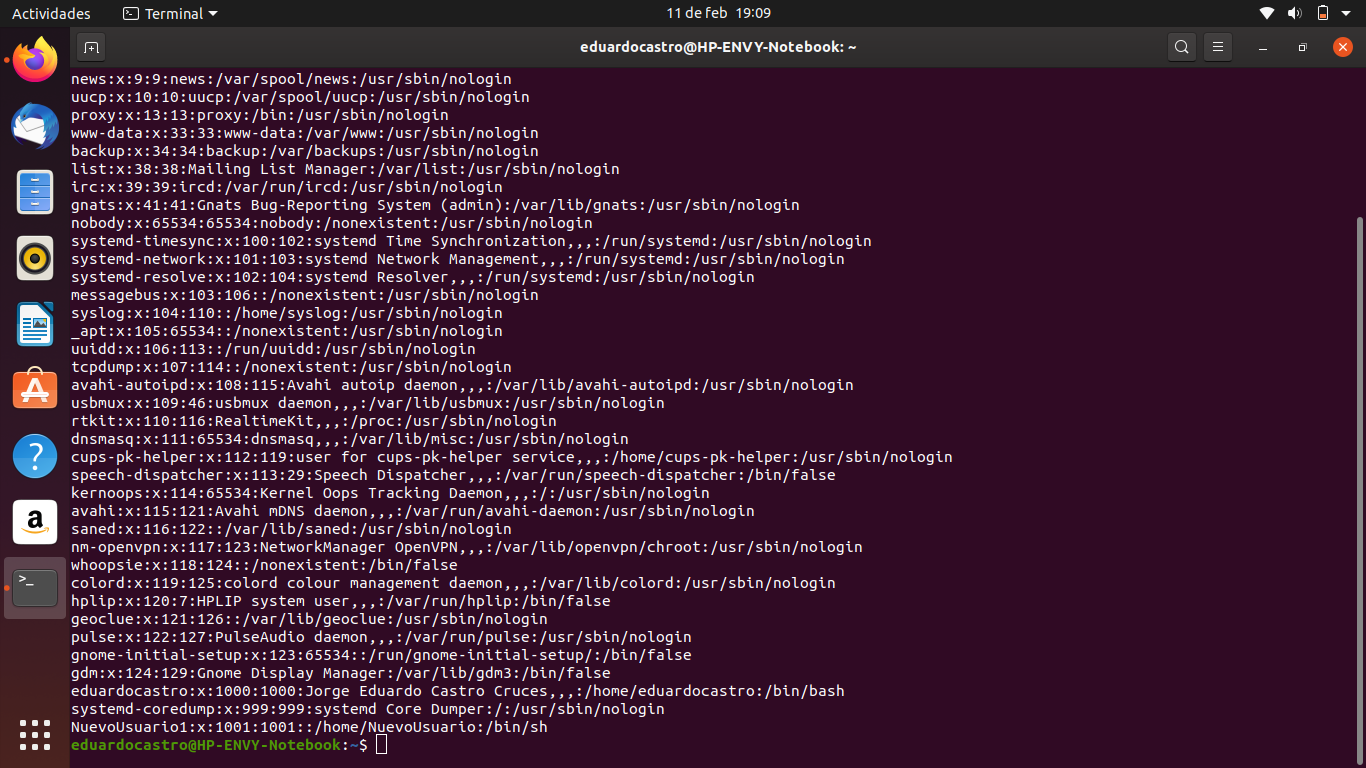
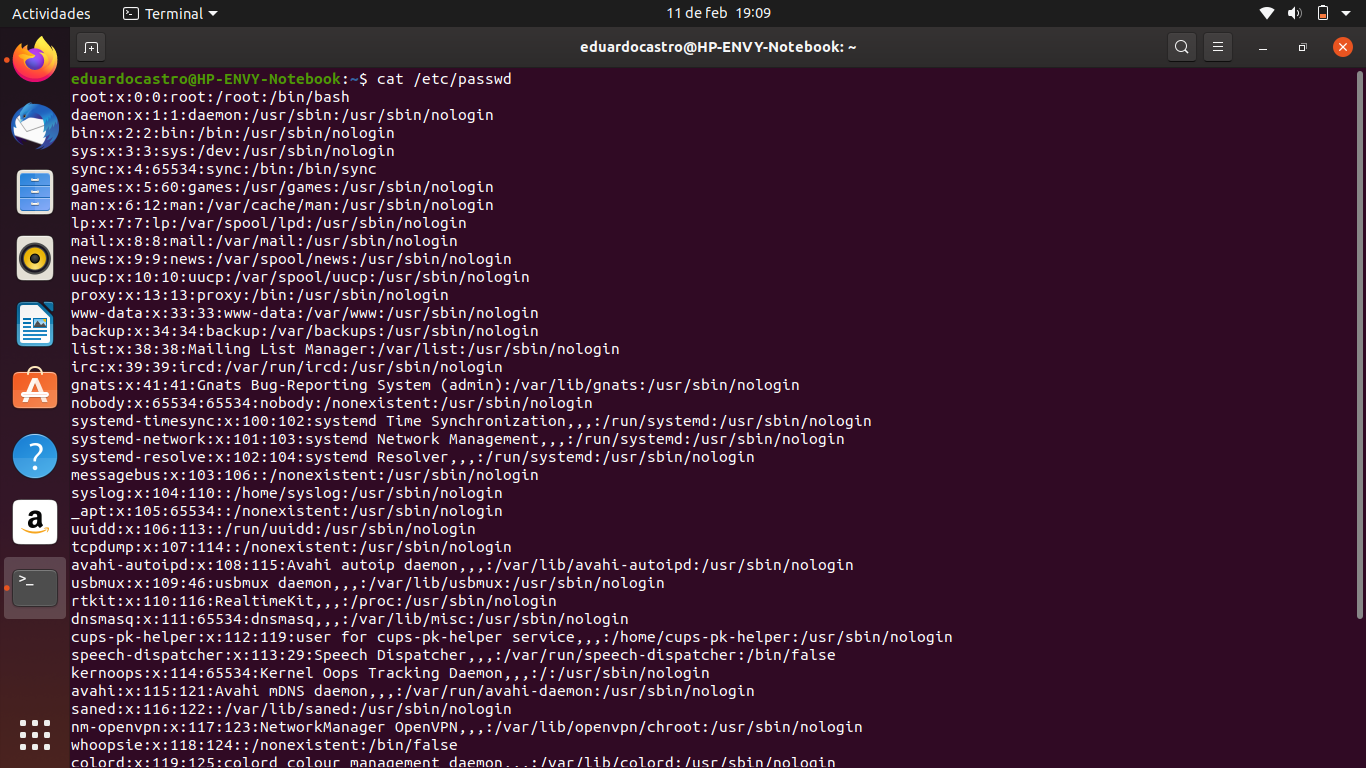
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| cp | | cp arch1 arch2  Copia archivos | cp orla /tmp/orla.bk  (copia el archivo *orla* del directorio actual al archivo *orla.bk* en el directorio /*tmp*) | |
| mv | | mv arch1 arch2  mv arch1 directorio  Traslada o renombra archivos | mv facturas trabajo/  (mueve el archivo facturas al directorio trabajo) | |
| date | | date  Retorna el día y la hora |  | |
| cal | | al [mes] año  Devuelve un calendario del año (y el mes) especificados. | al 09 1993  (devuelve el calendario de septiembre de 1993) | |
| uname | | El comando linux uname sirve para imprimir información del sistema linux. | El uso más práctico es:  uname -a | |
| mkdir | | mkdir directorio  Crea un nuevo directorio | mkdir trabajo  (crea un nuevo subdirectorio denominado *trabajo* en el directorio actual) | |
| man | | man [sección]  Permite consultar el manual | man grep  (obtiene ayuda sobre el comando *grep*) | |
| who | | who [am I]  Indica los usuarios conectados al sistema | who  (informa sobre los usuarios conectados en ese momento) | |
| su | | El comando "su" (Switch User, llamado (sin ser la buena connotación) "Super User" ) permite abrir una sesión con el ID (ID identificante) de un otro usuario, o de iniciar un shell de conexión con el nuevo ID. | ~$ su Password: debian:/home/ccm# suspend [1]+ Stopped su  ~$ fg su debian:/home/ccm# | |
| history | | “History” es un comando muy útil para averiguar los últimos comandos que se han ejecutado en un Servidor | sudo !!  Ejecuta el último comando | |
| wc | | wc [-l,-w,-c] archivo ...  Cuenta las palabras de un archivo (WordCount). | wc -l diario  (cuenta las líneas del archivo *diario*)  l Cuenta líneas  w Cuenta palabras  c Cuenta caracteres | |
| ifconfig | | ‘Ifconfig’, abreviatura de ‘Interface Configuration’, es una utilidad muy importante de los sistemas Linux. Se utiliza para comprobar y configurar las interfaces de red utilizando el terminal o CLI de máquinas Linux | ‘Ifconfig -a’ muestra información de todas las interfaces incluyendo las interfaces que actualmente no están activas también. | |
| pstree | | pstree es un comando poderoso y útil para mostrar procesos en ejecución en Linux. Estos procesos están organizados en un árbol en lugar de en una lista. | Pstree  Este comando por sí solo mostrará una lista de todos los procesos en ejecución. | |
| head | | Por defecto, imprime las primeras 10 líneas de los archivos especificados. Si se proporciona más de un nombre de archivo, los datos de cada archivo están precedidos por su nombre de archivo. | head [OPTION]... [FILE]... | |
| ls | | ls [-{a,c,l,p,r,s}] archivo ...  Lista el contenido de un directorio | ls -la  (lista todos los archivos del directorio actual con información extendida)  a Lista todos los archivos  c Ordena por fecha  l Listado Largo  p Señala cuáles son directorios con /  r Invierte el orden del listado  s Indica el tamaño en bloques | |
| whoami | | who [am I]  Indica los usuarios conectados al sistema | am I Sólo proporciona información sobre el propio usuario. | |
| which | | Indica la ubicación del programa que ejecuta la shell, que estamos empleando cuando escribimos la orden. | Por ejemplo para determinar la ubicación del comando awk ejecutamos:  # which vim  /usr/bin/awk | |
| w | | El comando w muestra información sobre los usuarios que están conectados en ese momento a la máquina y sobre sus procesos | # w -f  06:12:47 up 7:49, 2 users, load average: 0,00, 0,02, 0,07  USER TTY LOGIN@ IDLE JCPU PCPU WHAT  root tty7 22:25 10:49m 6.13s 0.09s gdm-session-worker [pam/gdm3]  root pts/0 22:25 7.00s 1.43s 0.00s w -f | |
| sudo | | El comando Sudo otorga algunos privilegios de administrador a usuarios que no son administradores. | sudo dpkg -i software.deb | |
| ps | ps [-a, -u, -x]  Retorna los procesos activos | | | ps -aux  (visualiza en formato extendido todos los procesos activos y sus usuarios)  a Retorna todos los procesos  u Indica el usuario de cada proceso  x Información extendida | |
| grep | grep [-n,-c,-v] patrón archivo  Busca ocurrencias de un patrón en un archivo | | | grep valencia cities  (busca todas las ocurrencias de la palabra *valencia* en el archivo *cities*)  a Poner el número de cada línea donde aparece el patrón  c Sólo aparecen las líneas que contienen el patrón  v Aparecen las líneas que no contienen el patrón | |
| find | find ruta criterio acción...  Busca archivos que cumplen el criterio y realiza una acción (o varias) sobre ellos. | | | find /tmp -name fact\* -print  (busca e imprime el nombre de los archivos en el directorio /*tmp* que comiencen por *fact*)  Los criterios para los archivos pueden ser:  -name archivo Con ese nombre  -size [+|-]n De tamaño n bloques  -links [+|-]n Con n links  -ntime [+|-]n Accedido hace n días  -mtime [+|-]n Modificado hace n días  -newer archivo Modificado después de *archivo*  Donde aparece n se asume el valor exacto. +n y -n significan mayor o menor que, reespectivamente.  Las acciones pueden ser:  -print Imprime la ruta de los archivos encontrados  -exe comando Ejecuta comando  -ok comando Ejecuta comando pidiendo confirmación.  En estos dos últimos casos el archivo encontrado se repesenta con {} | |
| echo | Imprime *cadena* | | | echo [cadena] | |
| clear | Limpia la pantalla | | | clear | |
| df | El comando linux df nos informa acerca del espacio total, ocupado y libre en nuestro sistema. | | | Para que nos de la misma información en Gb, Mb y Kb usamos el modificador -h (legible por humanos):  df -h | |
| exit | Exit es un comando básico en Linux que cierra las ventanas, las conexiones remotas o la pantalla del terminal | | | exit | |
| cat | cat archivo ...  Concatena e Imprime | | | cat arch2  (envía el archivo *arch2* a la pantalla) | |
| rm | rm [-l,-r] archivo ...  Borra archivos | | | rm -r trabajo  (elimina el directorio trabajo y todos los archivos y subdirectorios que cuelgan de él)  l Protege archivos existentes  r Borra recursivamente todos los subdirectorios y archivos que cuelgan del directorio a borrar | |
| sort | sort [-b,-d,-f,-n,-o archsal, -r] archent  Ordena las líneas de un archivos según orden lexicográfico. | | | sort -f -o salida entrada  (ordena el archivo *entrada* en el archivo *salida* ignorando mayúsculas y minúsculas)  b Ignorar los espacios del principio de línea  d Ordena según diccionario  f Ignora diferencias entre mayúsculas y minúsculas  n Ordena los números por el valor y no por orden lexicográfico  o archivo El resultado se escribe en archivo  r Revierte el orden | |
| pwd | pwd  Indica el directorio actual | | |  | |
| | | Permiten redireccionar la entrada o salida de un comando a otro comando. | | | ls -l | lp  (realiza un listado del directorio actual por impresora) | |
| > | Permiten redireccionar la entrada o salida de un comando a un archivo.  cat arch1 arch2 >> arch3  (añade *arch1* y *arch2* al final de *arch3*) | | | Redireccionan la entrada y la salida (respectivamente) borrando previamente el archivo de destino | |
| < |
| >> | Redireccionan la entrada y la salida a continuación del archivo de destino. | |
| << |

**Ejercicio 2**. Desde la terminal, crea un usuario con permisos de root.

**Ejercicio 3**. Desde la terminal, cambia el nombre del usuario anterior.

**Ejercicio 4.** Muestra en la terminal solo la información de tu usuario, consultando el

archivo referente a los usuarios del sistema (/etc/passwd).



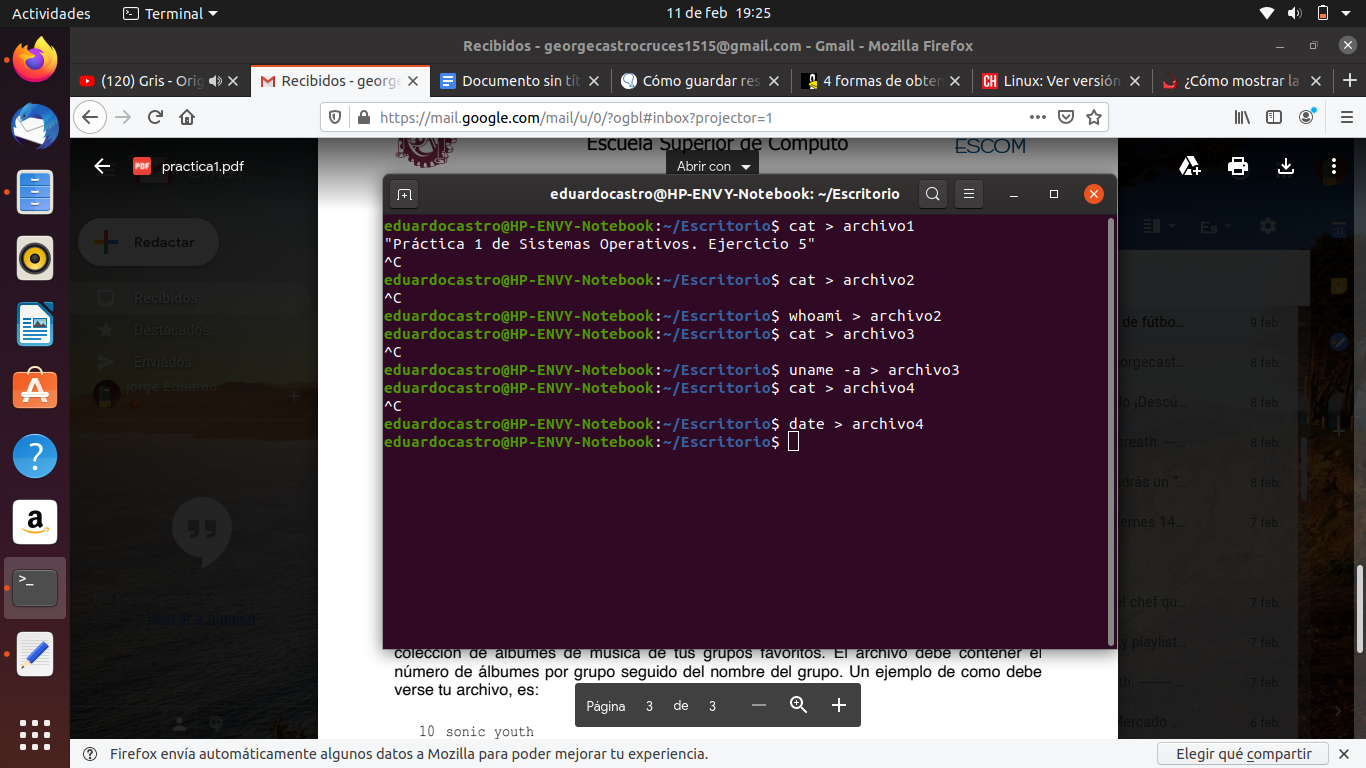
**Ejercicio 5**. Con el direccionamiento de salida (>) crea 4 archivos que contengan la siguiente información:

*Nombre del archivo Contenido*

*archivo1 “Práctica 1 de Sistemas Operativos. Ejercicio 5”*

*archivo2 El nombre del usuario actual*

*archivo3 nombre del kernel del sistema*

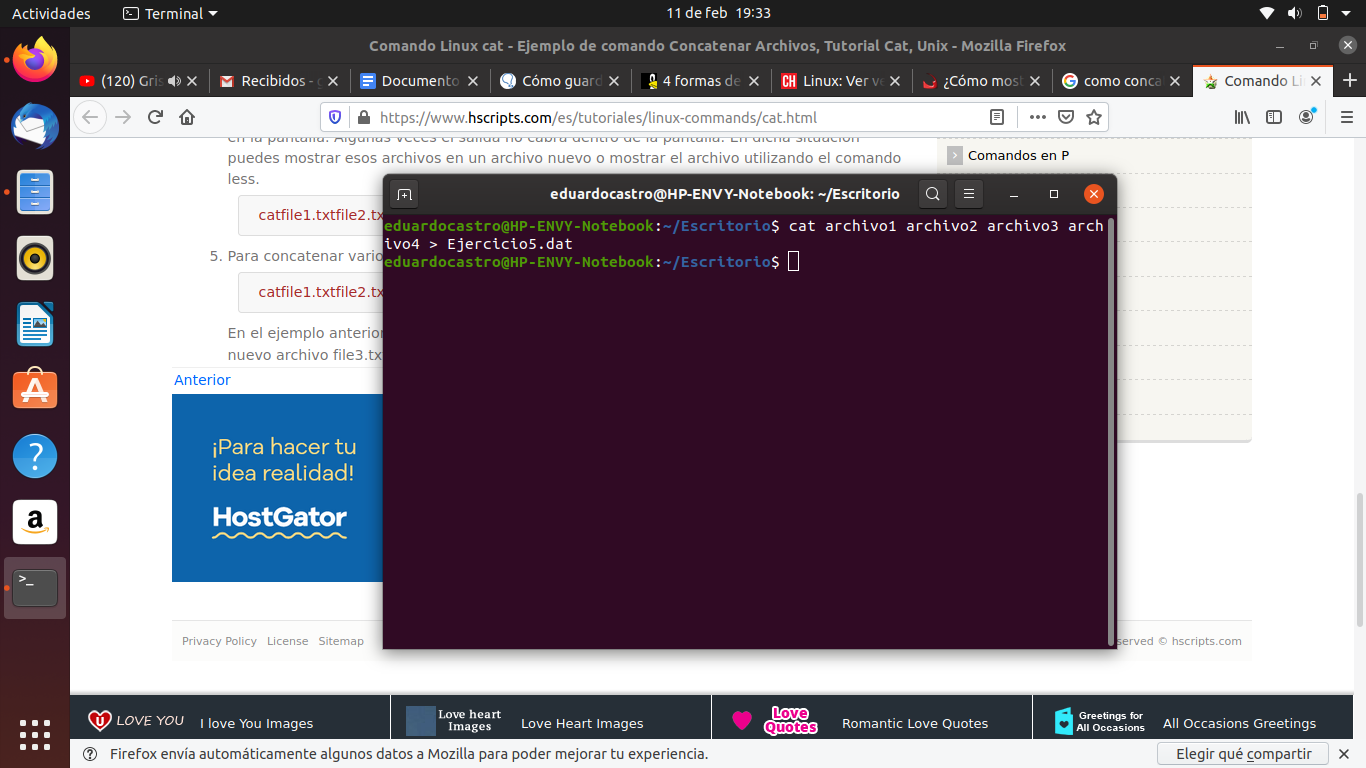
*archivo4 Fecha y hora del sistema actual*

Después concatena la información de todos los archivos en uno solo con el nombre Ejercicio5.dat, de tal forma que se vea de la siguiente manera:

*“Práctica 1 de Sistemas Operativos. Ejercicio 3”*

*Fecha y hora del sistema actual*

*El nombre del usuario actual*

*nombre del kernel del sistema*

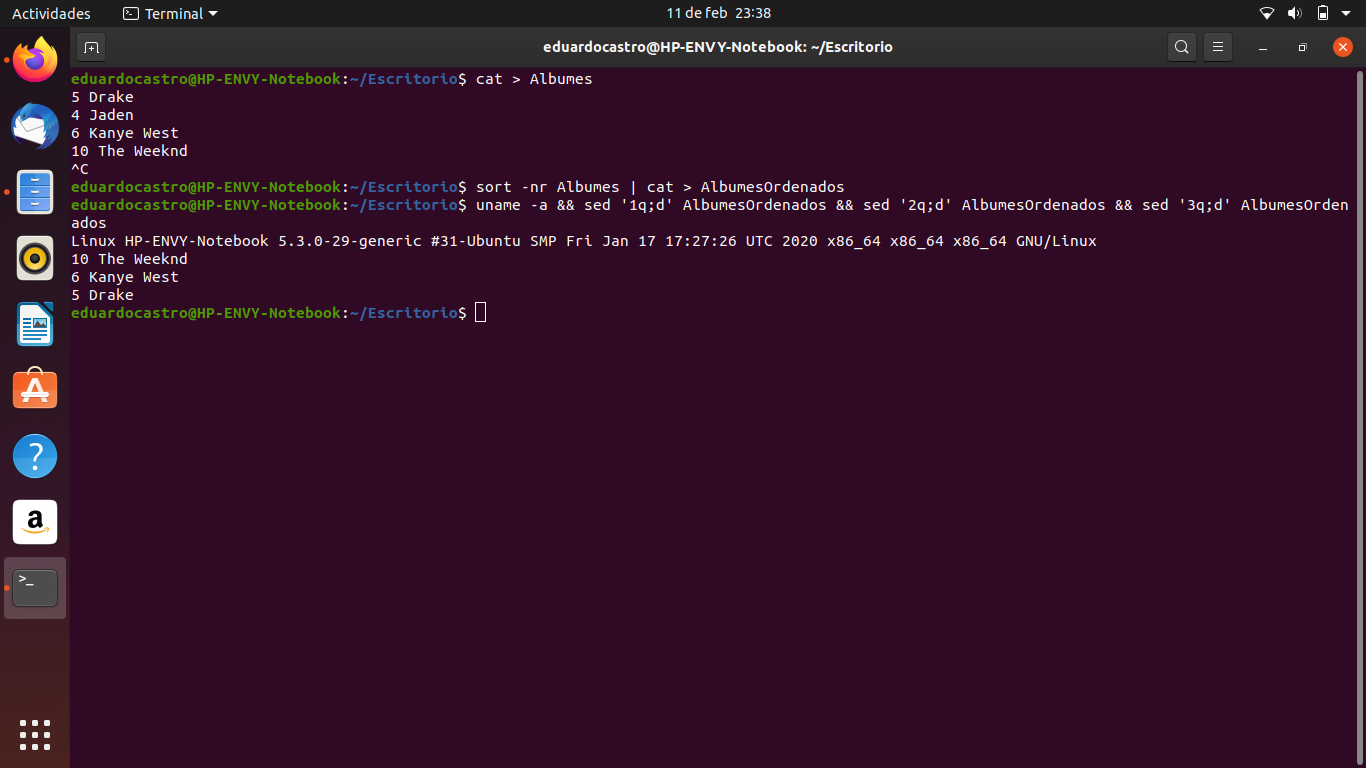
**Ejercicio 6**. Elige el editor de tu preferencia y crea un archivo donde coloques tu colección de álbumes de música de tus grupos favoritos. El archivo debe contener el número de álbumes por grupo seguido del nombre del grupo. Un ejemplo de cómo debe verse tu archivo, es:

*10 sonic youth*

*3 Cat Power*

*4 miami horror*

*8 Beatles*

En la terminal posiciónate en el directorio de trabajo donde se encuentra tu archivo y coloca el conjunto de comandos que imprima las 3 líneas mayores, es decir los 3 grupos con los que tienes más álbumes. El resultado con el ejemplo anterior se vería de la siguiente forma:

**CONCLUSIONES**

En esta práctica, se logro Conocer algunos de los comandos fundamentales de UNIX y aprender el uso del shell. Asi mismo, se logro la mejor comprension y conocimiento de los principales comandos de la consola y su variado manejo y funcionalidad. Por otro lado, se realizo una busqueda de comandos auxiliares en el manejo de archivos y su escritura.

**REFERENCIAS**

[**https://www.por-correo.com/index.php/articulos-de-interes/51-ayuda-linux-listado-de-comandos-importantes-para-linux-unix.html**](https://www.por-correo.com/index.php/articulos-de-interes/51-ayuda-linux-listado-de-comandos-importantes-para-linux-unix.html)