IFA. Práctica de laboratorio 02

Hugo Fonseca Díaz
email uo258318@uniovi.es

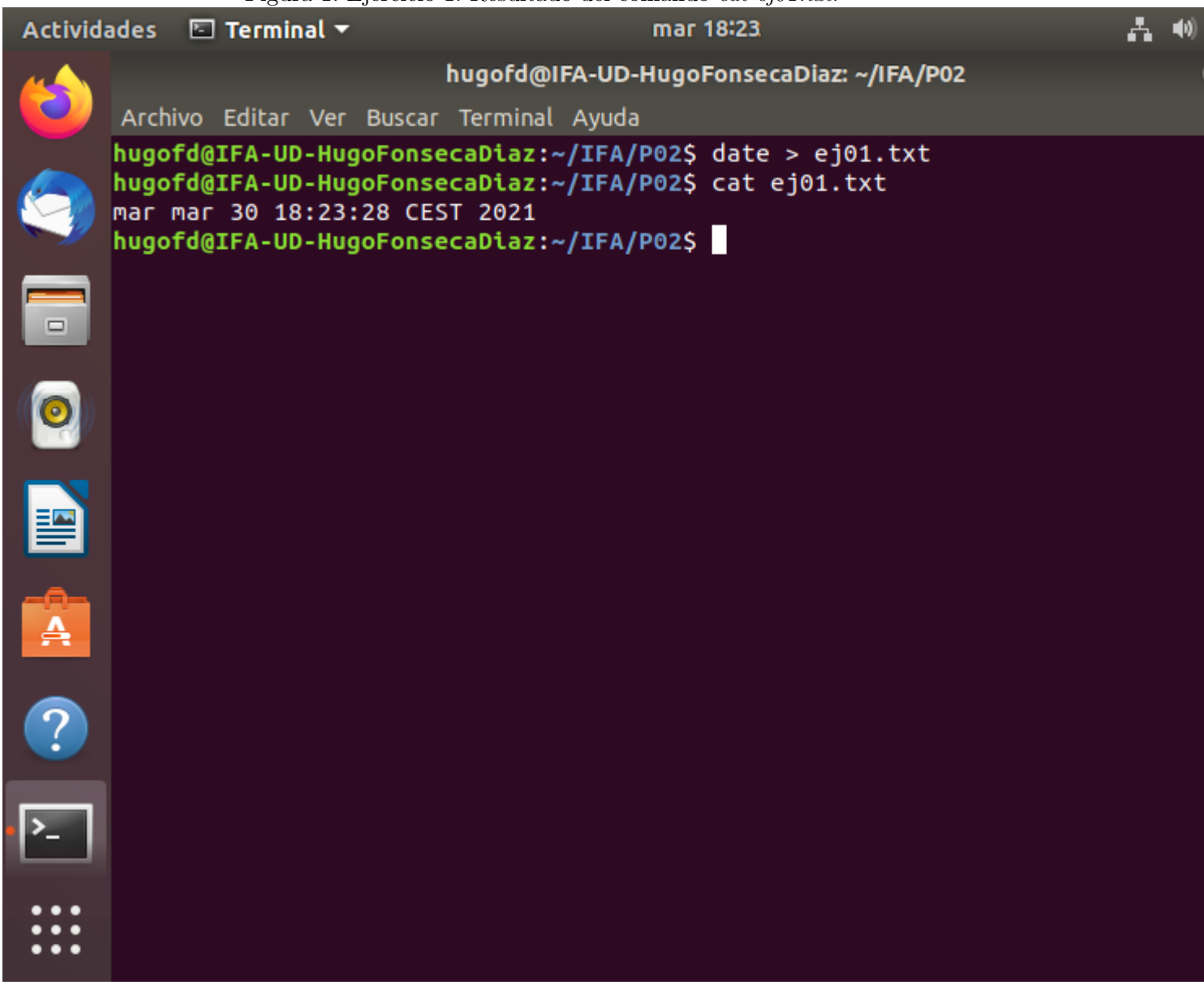
Escuela de Ingeniería Informática. Universidad de Oviedo.

20 de junio de 2021

1. Ejercicio 1

Se guarda la fecha y hora del sistema en el archivo `ej01.txt` con el comando `date > ej01.txt`. Se muestra ese archivo con el comando `cat`.

Figura 1: Ejercicio 1: Resultado del comando *cat ej01.txt*.



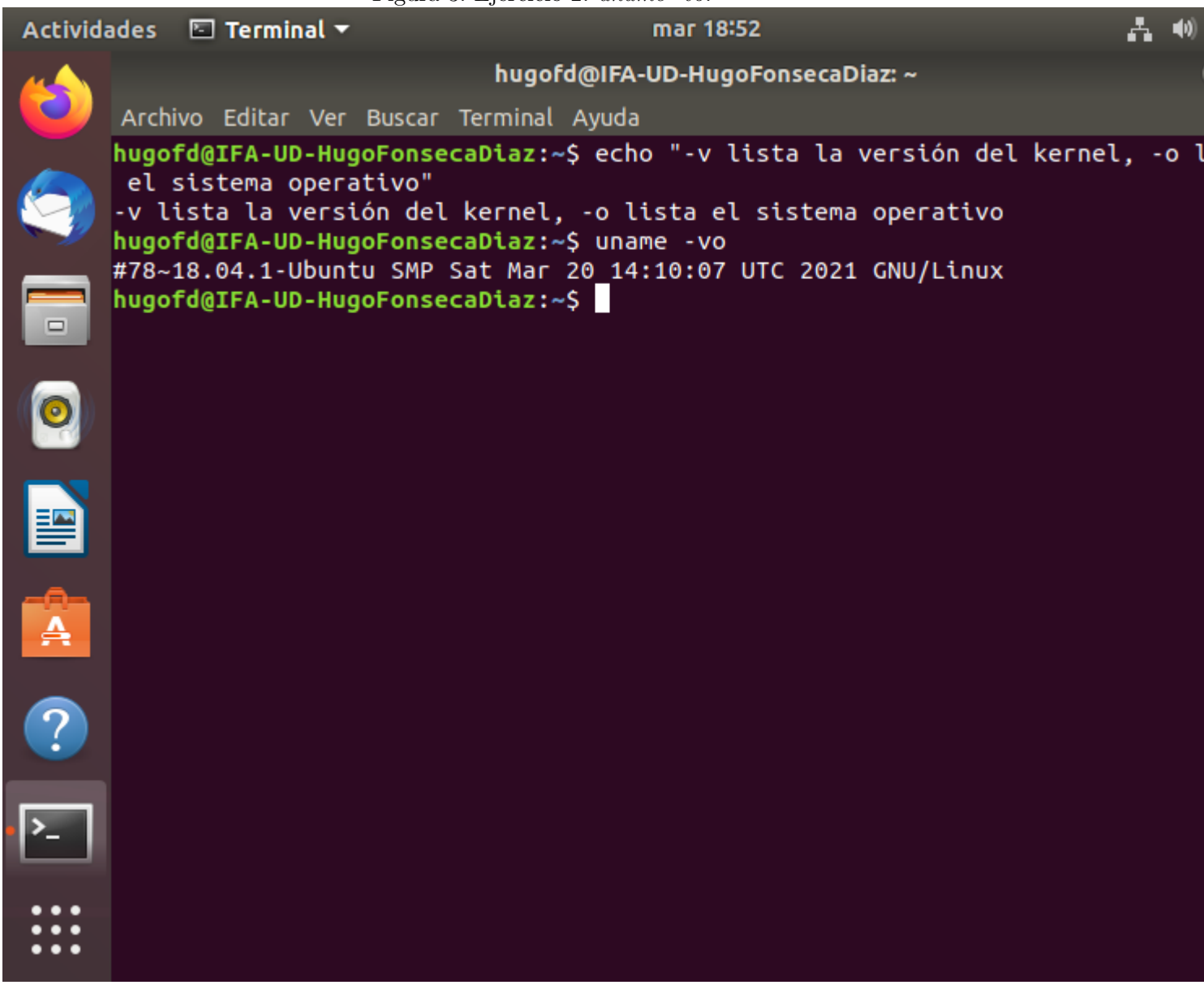
Se accede al sitio web `https://time.is/es/Spain` y se comprueba que la hora es la misma.

Figura 2: Ejercicio 1: Hora en el sitio web *time.is*.

2. Ejercicio 2

Se utiliza el comando `uname` con las opciones `v` (lista la versión del kernel) y `o` (lista el nombre del sistema operativo).

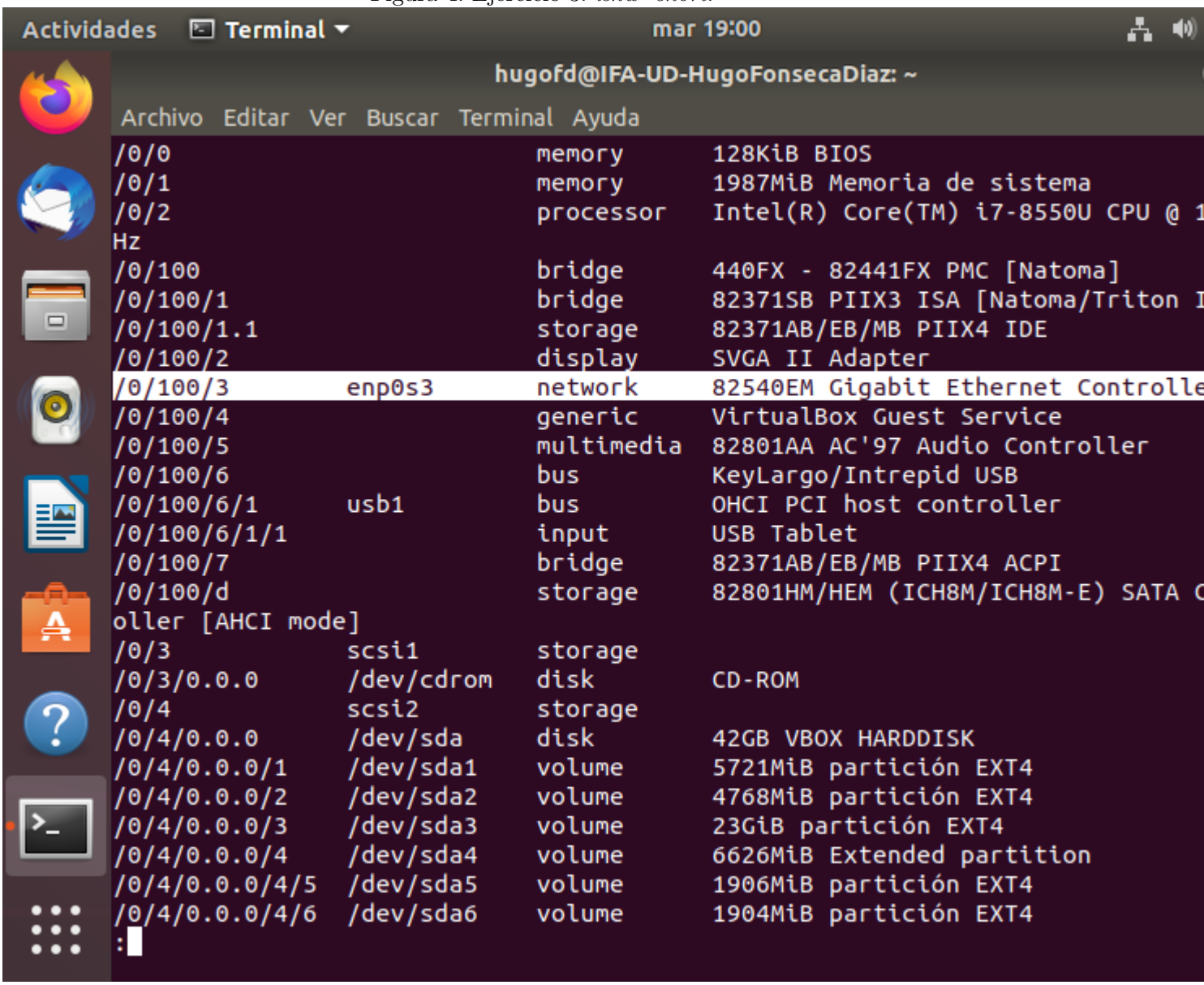
Figura 3: Ejercicio 2: *uname -vo*.



3. Ejercicio 3

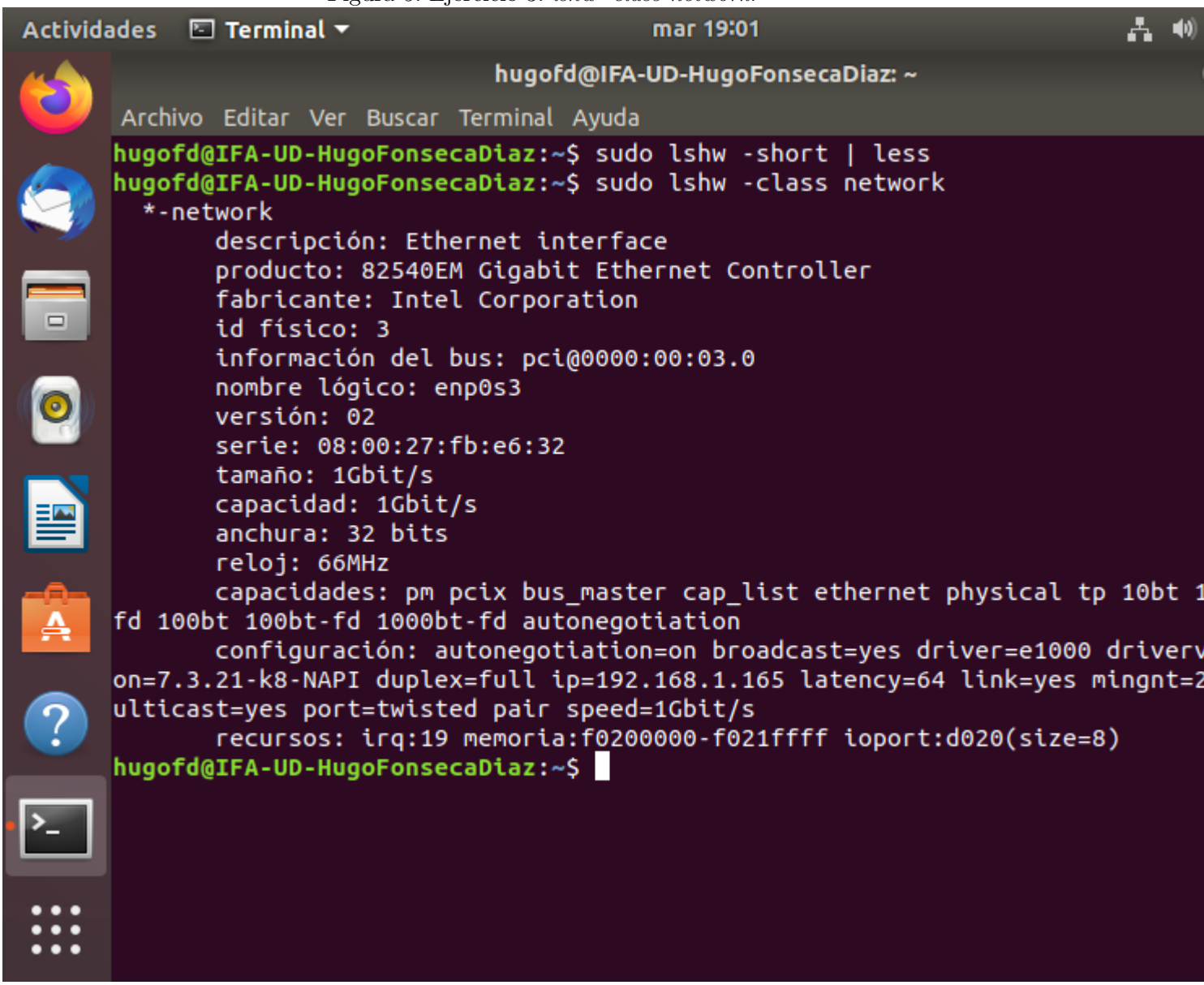
Se utiliza el comando `lshw`, primero con la flag `short` para encontrar el nombre de la clase de los dispositivos de red.

Figura 4: Ejercicio 3: *lshw -short*.



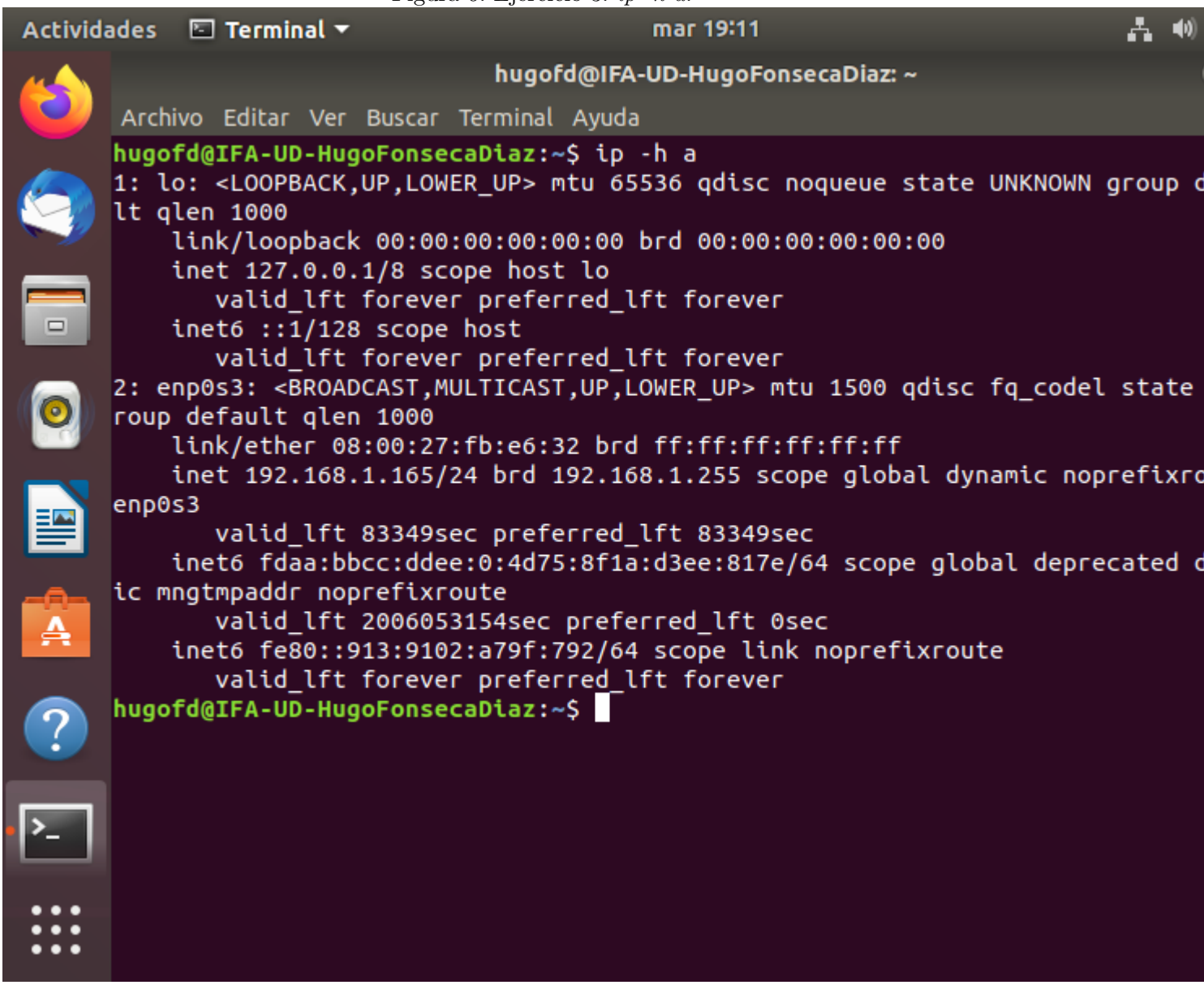
Una vez se sabe que el nombre de la clase de los dispositivos de red es `network`, se utiliza el comando `lshw` con la flag `-class network`.

Figura 5: Ejercicio 3: *lshw -class network*.



También puede utilizarse el comando `ip -h a` para mostrar más información sobre el dispositivo de red `enp0s3`.

Figura 6: Ejercicio 3: *ip -h a*.



4. Ejercicio 4

Se utiliza el comando `netstat` del paquete `net-tools`. Su flag `a` permite ver todos los sockets, por lo que `sudo netstat -a > ej04.txt` guarda la información de los sockets activos y no activos en un fichero de texto. También son interesantes sus flags `n` (se muestran las direcciones numéricamente), `p` (se muestran los procesos pertenecientes a los sockets), `t` (tcp) y `u` (udp).

Figura 7: Ejercicio 4: *cat ej04.txt | less*.

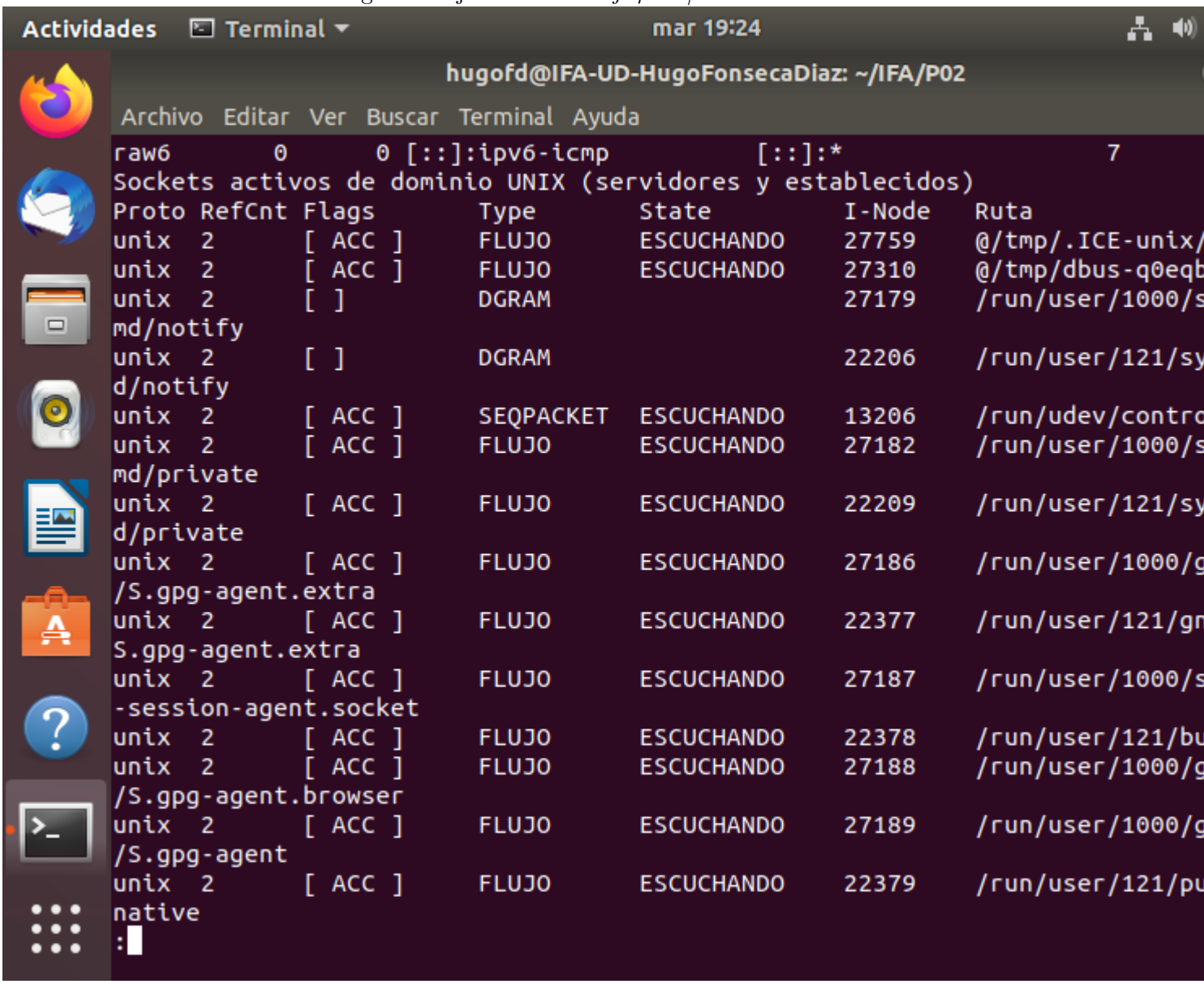
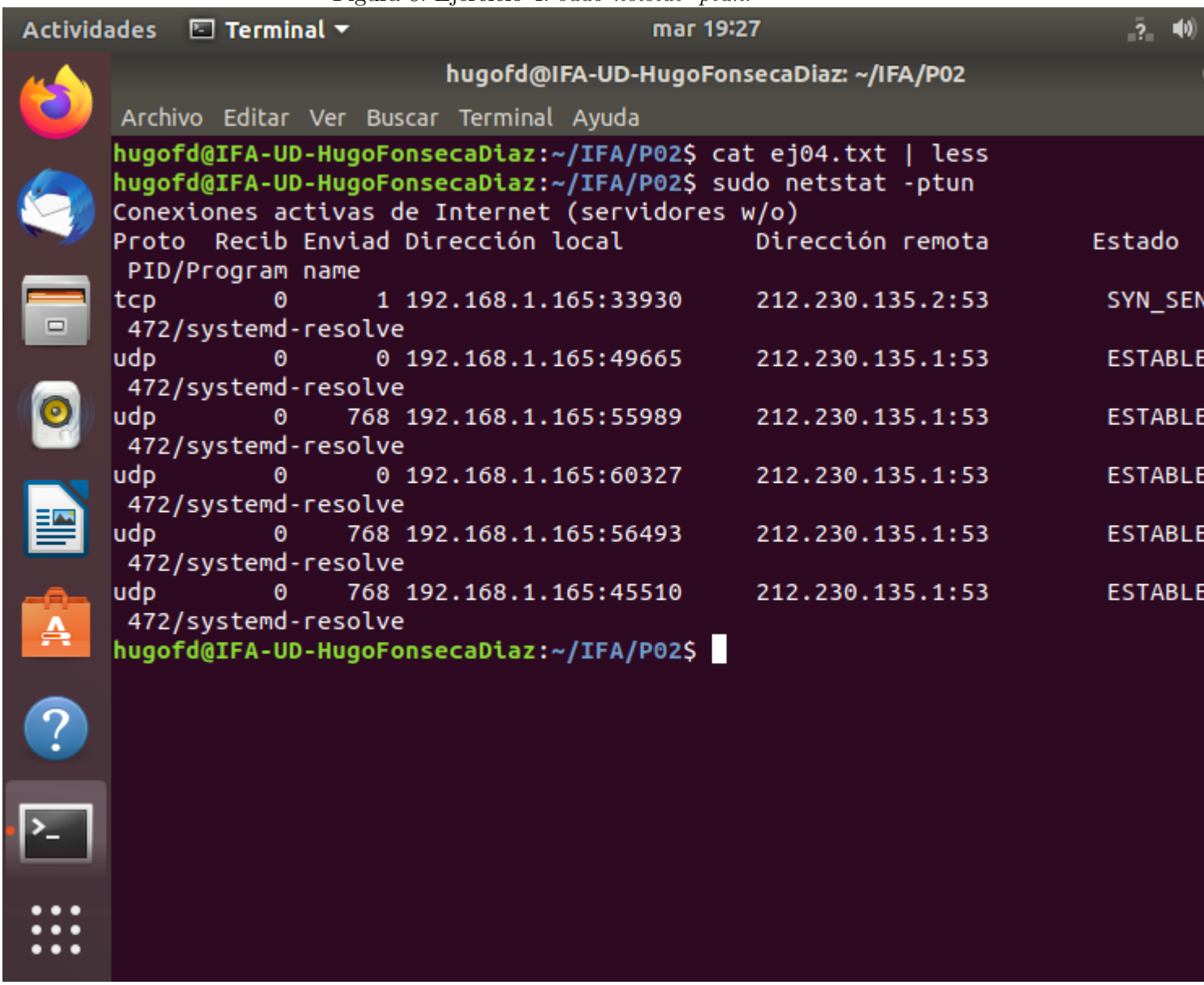
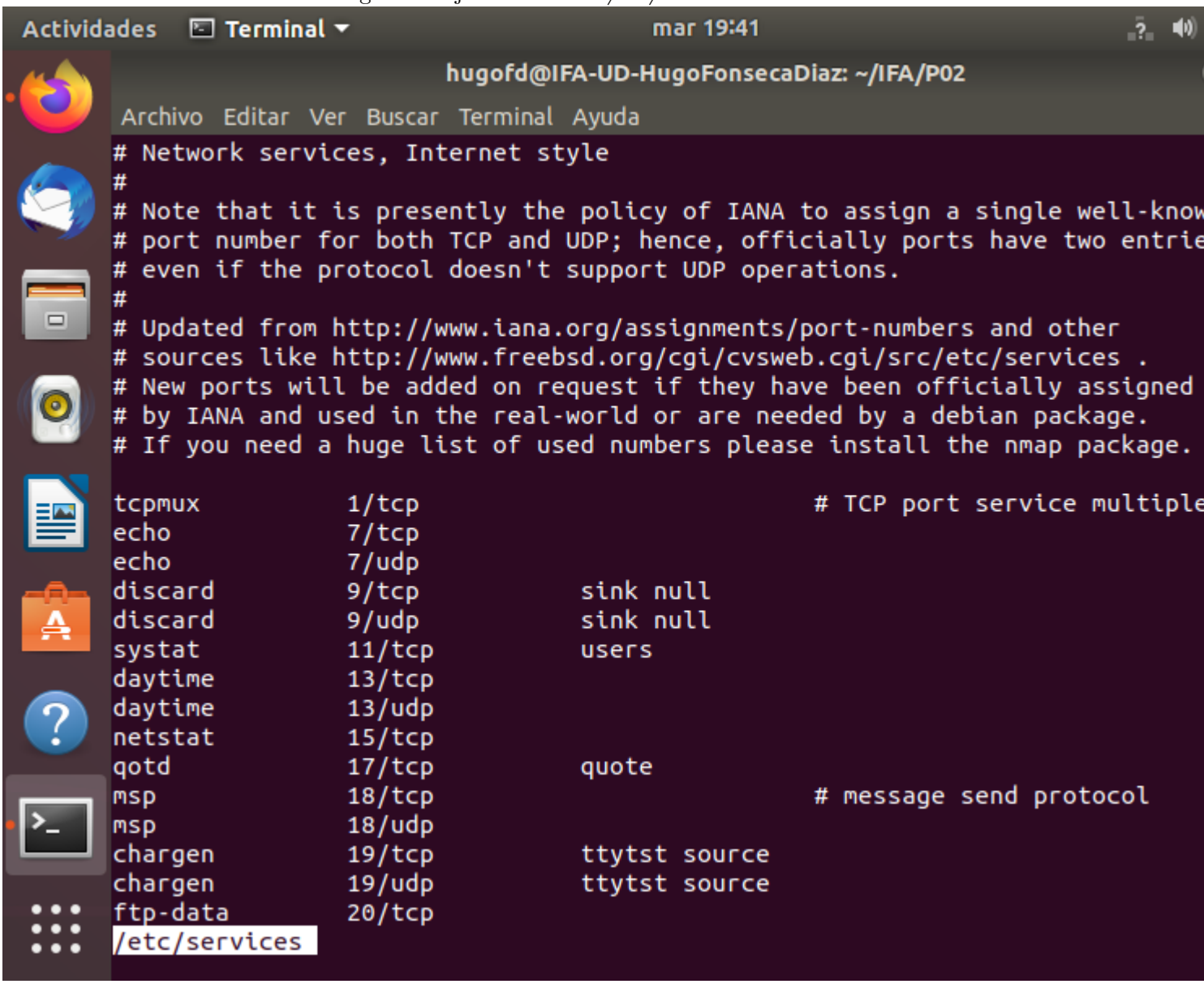


Figura 8: Ejercicio 4: *sudo netstat -ptun*.



También se puede ver información de los servicios de red en `/etc/services`.

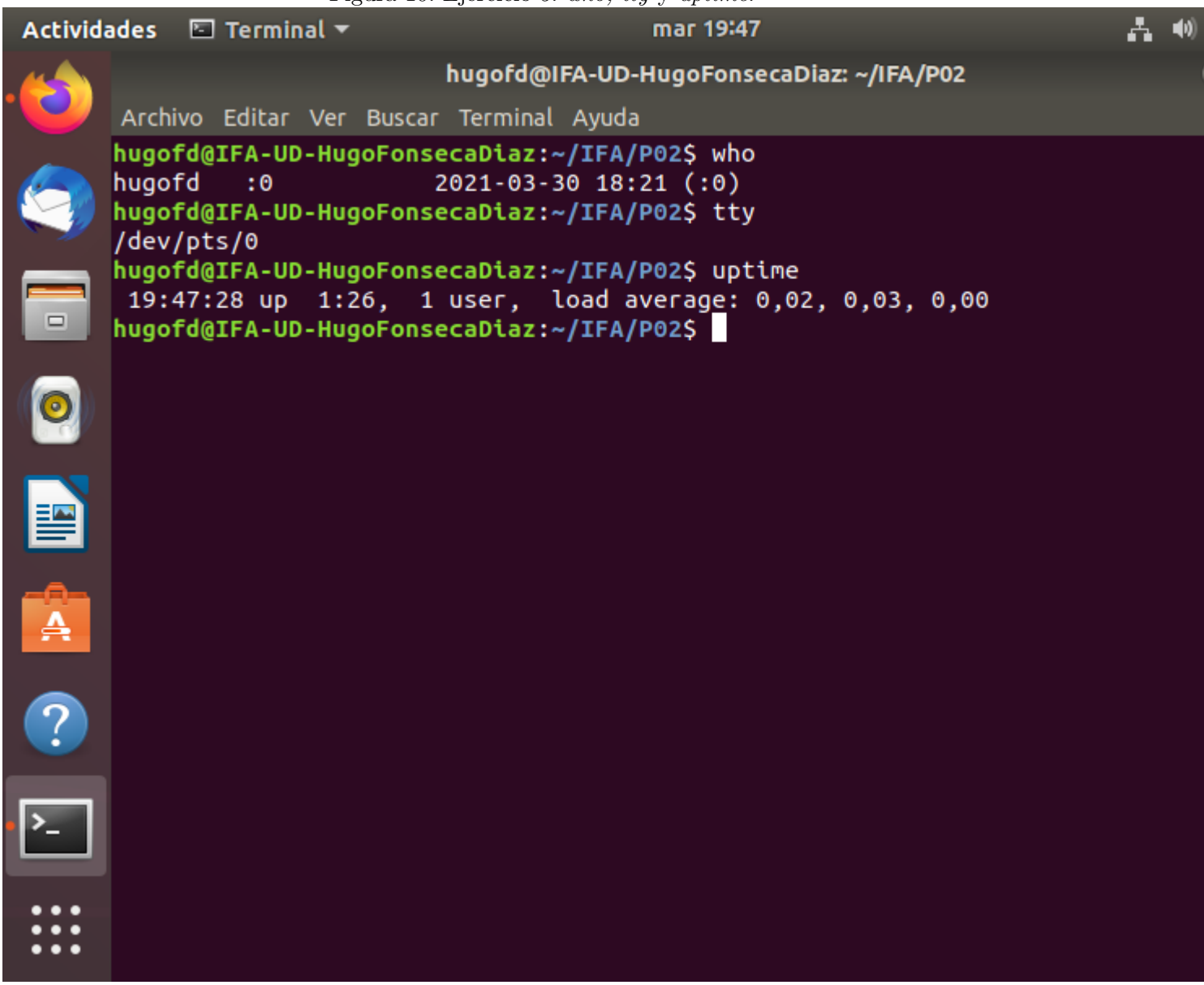
Figura 9: Ejercicio 4: *less /etc/services*.



5. Ejercicio 5

Para resolver este ejercicio se usan tres comandos: `who` muestra los usuarios conectados y la terminal en la que están, `tty` muestra la terminal conectada actualmente al standard input y `uptime` muestra el tiempo que ha pasado desde el arranque del sistema.

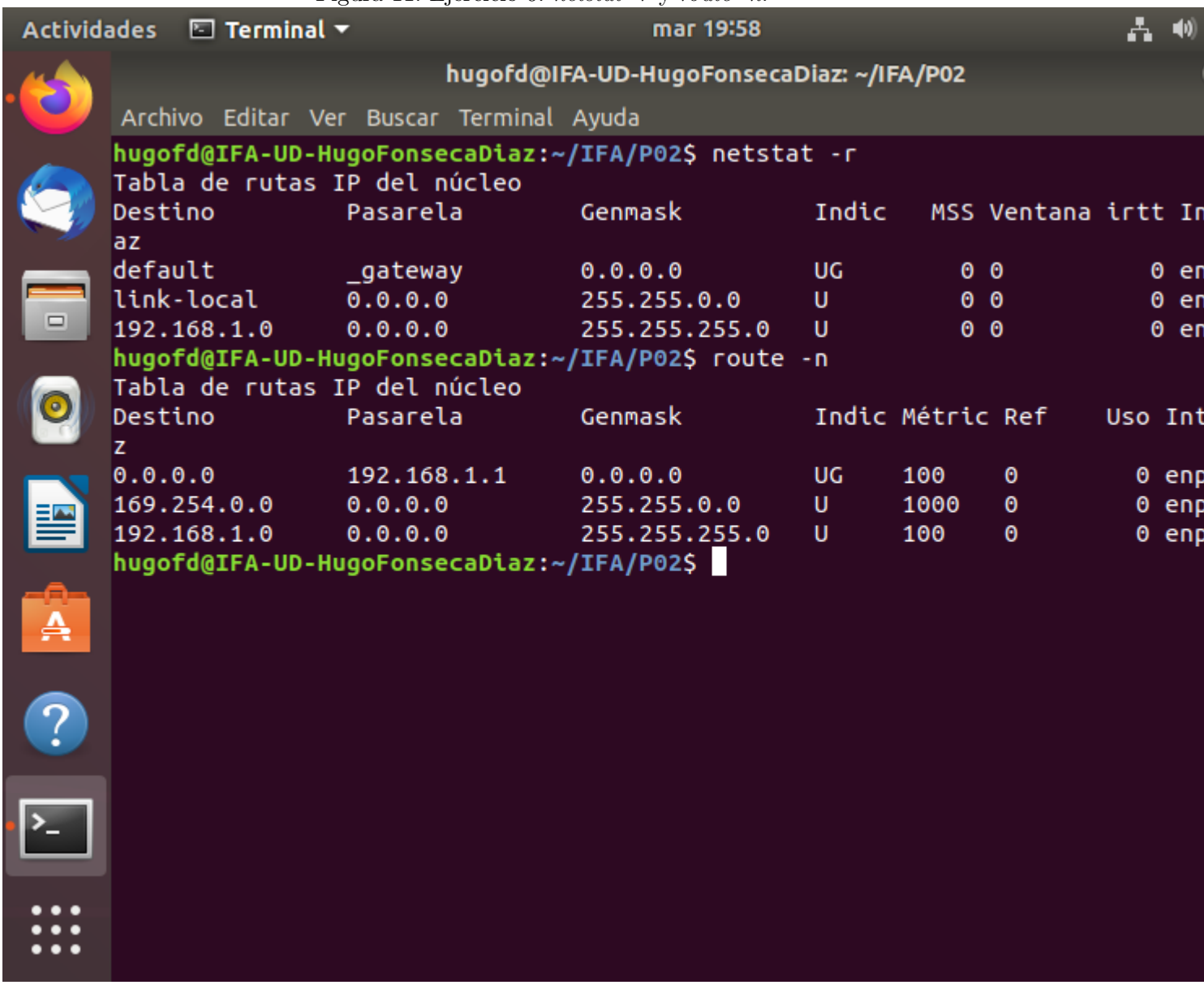
Figura 10: Ejercicio 5: *who*, *tty* y *uptime*.



6. Ejercicio 6

Existen al menos dos opciones de mostrar la información sobre la tabla de enrutamiento: mediante el comando `netstat` con su flag `r` (que muestra la tabla de enrutamiento) o usando el comando `route` con su flag `n` (que muestra las direcciones de red de forma numérica).

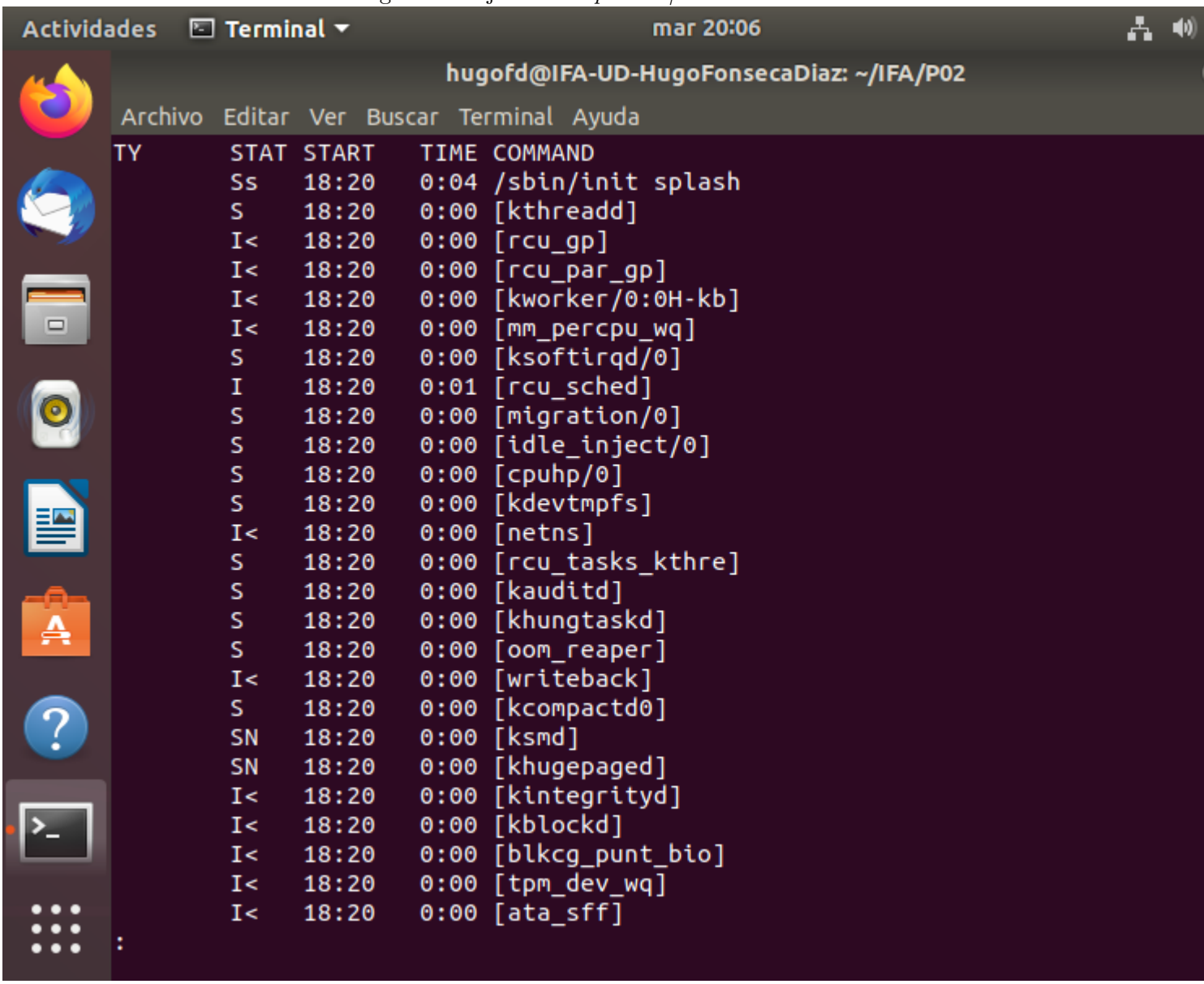
Figura 11: Ejercicio 6: *netstat -r* y *route -n*.



7. Ejercicio 7

Se usa el comando `ps`. Dicho comando puede utilizarse siguiendo tres sintaxis: la de UNIX, la de BSD o la de GNU. Para mostrar todos los procesos del sistema con sintaxis de UNIX podría usarse `ps -eF`. Con sintaxis de BSD se puede usar `ps axu`. Para que se muestre el nombre del proceso sin cortarse se puede pasar el resultado del comando `ps` al comando `less` con una pipe de UNIX.

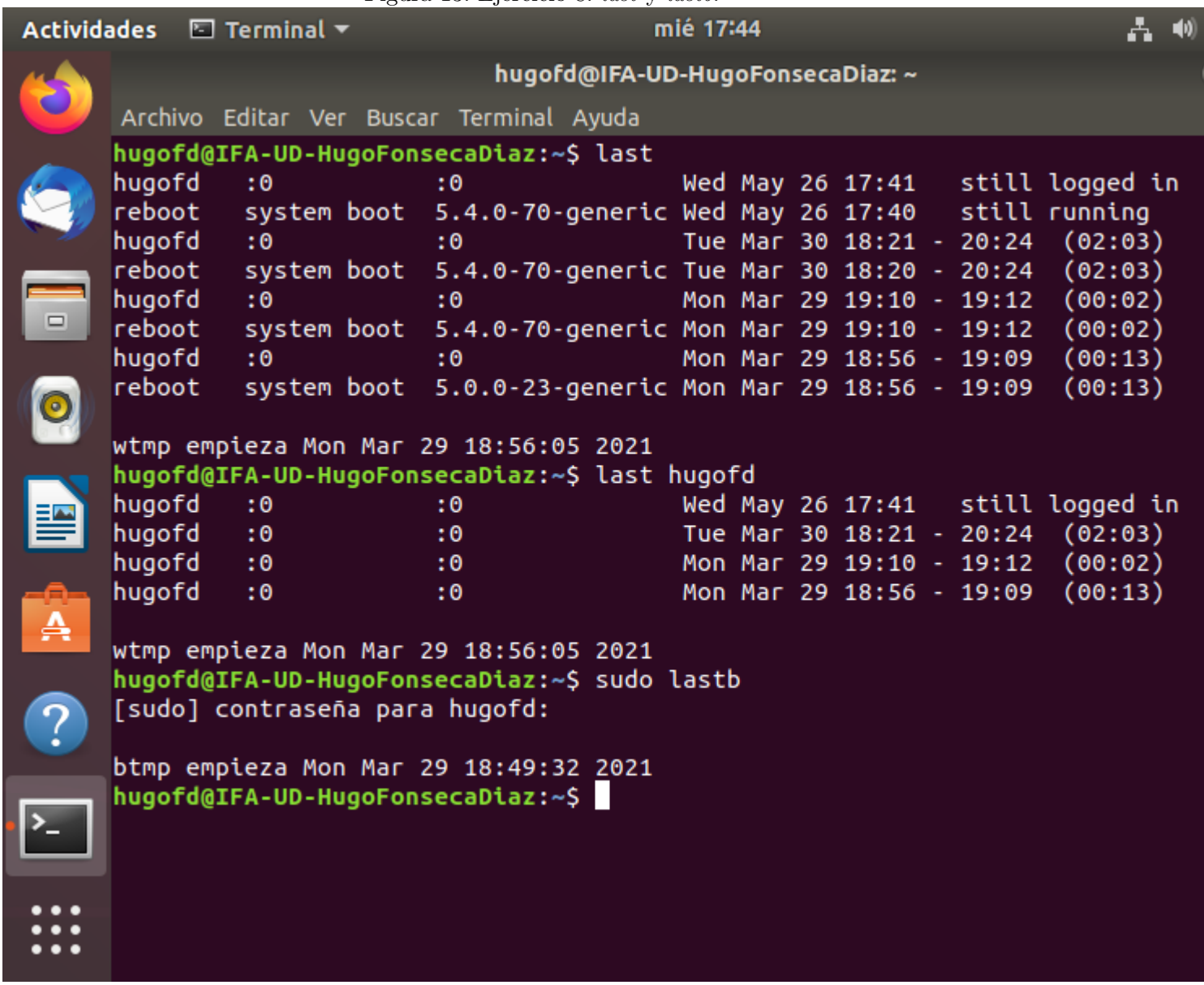
Figura 12: Ejercicio 7: *ps axu | less*.



8. Ejercicio 8

Se usarán los comandos `last` y `lastb`. El primero se utiliza para sacar la información de los accesos de todos los usuarios al sistema, incluyendo también un ejemplo de uso para un usuario concreto. El segundo es un comando similar pero buscando en */var/log/btmp*, lo que muestra intentos fallidos de acceso al sistema.

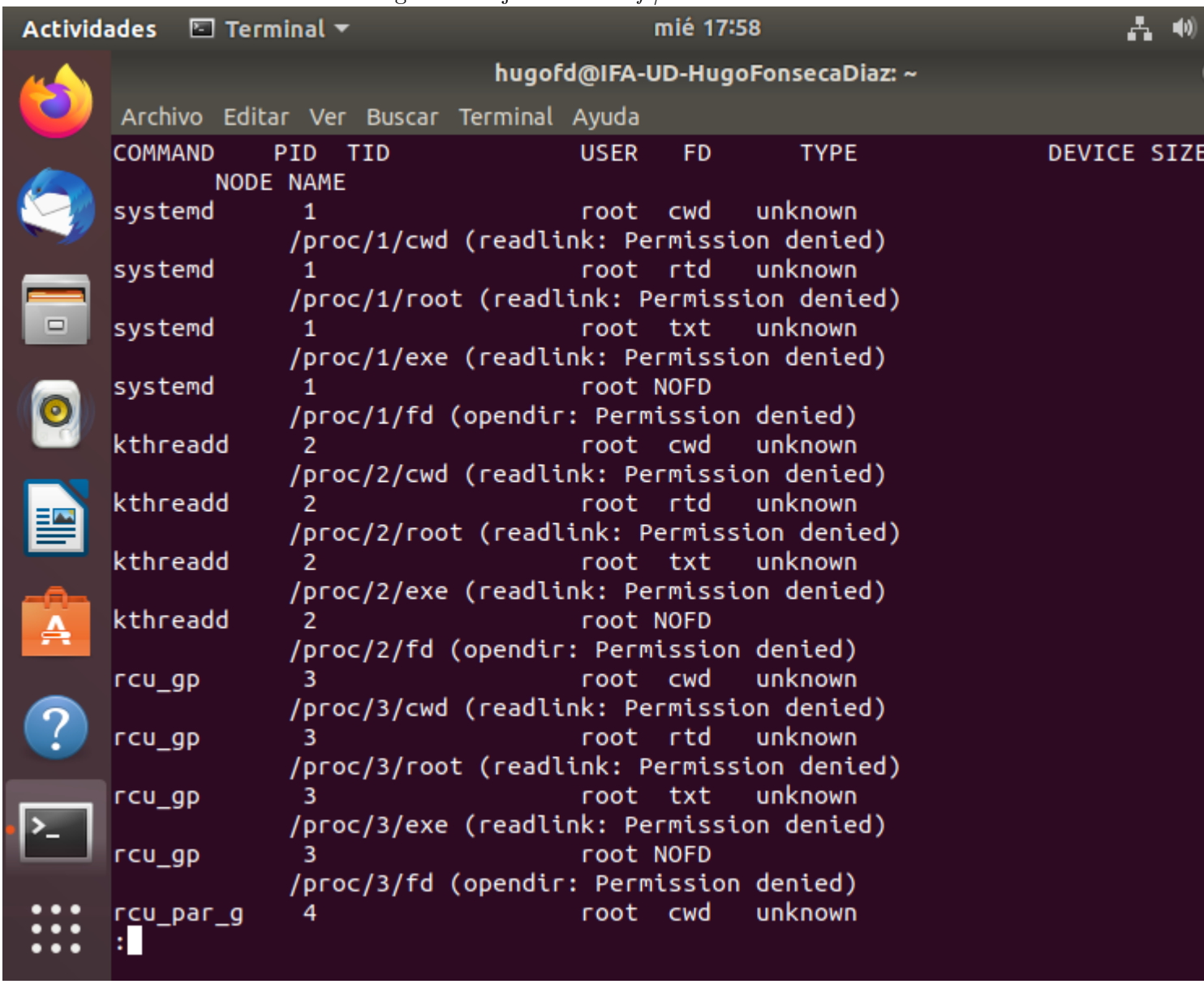
Figura 13: Ejercicio 8: *last* y *lastb*.



9. Ejercicio 9

Se utiliza el comando `lsof`, cuya salida está pensada para ser la entrada de otro programa que la parsee. Se hace una pipe de Unix con el comando `less` para poder visualizar la salida del comando.

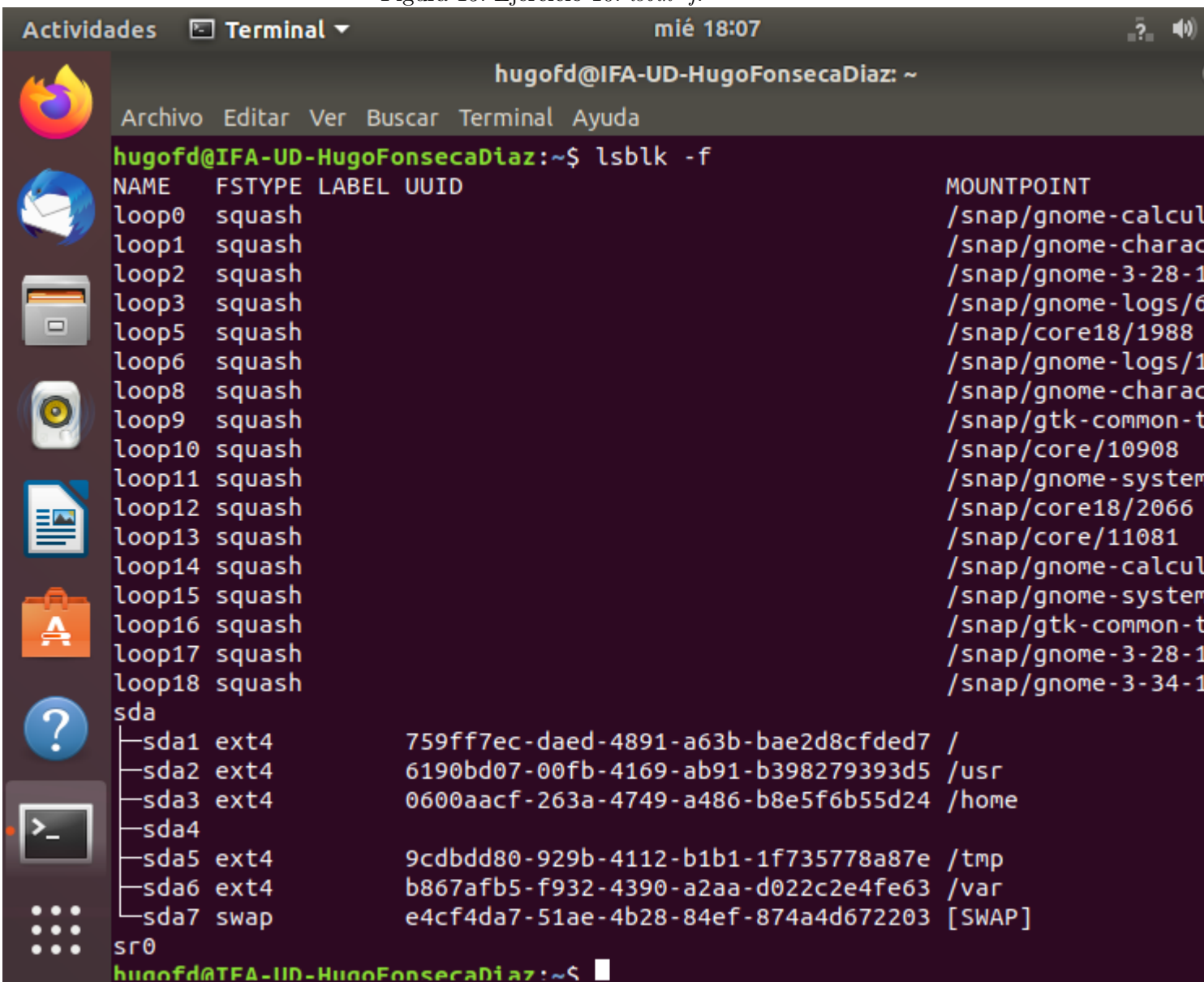
Figura 14: Ejercicio 9: *lsof | less*.



10. Ejercicio 10

Se puede usar el comando `lsblk` con la opción `f`. El comando muestra información de los dispositivos del sistema y la opción *f* muestra los sistemas de ficheros de los mismos.

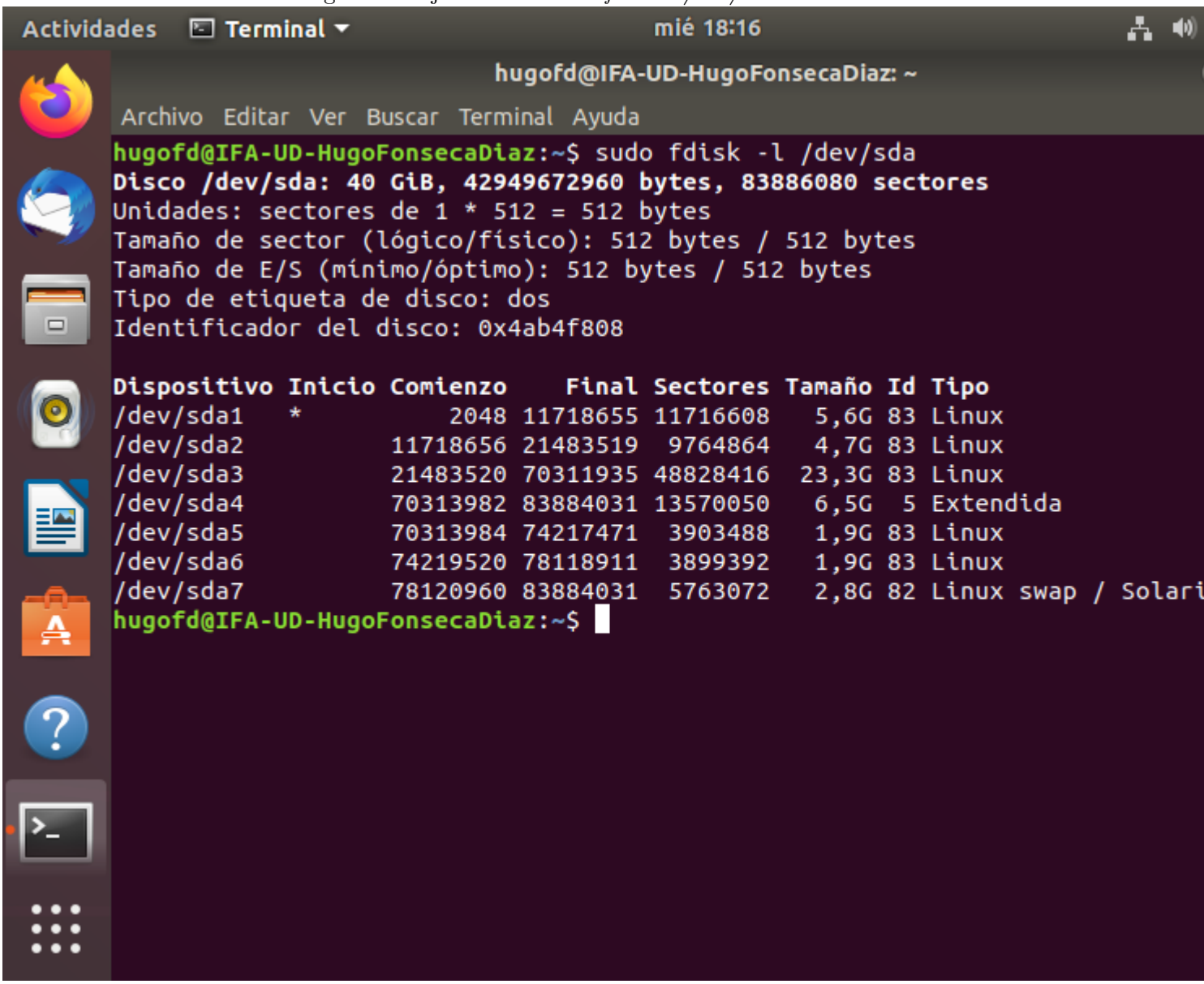
Figura 15: Ejercicio 10: *lsblk -f*.



11. Ejercicio 11

Para mostrar las particiones del disco *sda* junto a sus sectores de inicio y fin, se utiliza el comando `fdisk` con la opción `l`, que lista dichas particiones, y pasándole como parámetro el disco que queremos inspeccionar (en este caso */dev/sda*). No es necesario especificarle que las unidades del tamaño sean sectores puesto que es el comportamiento por defecto.

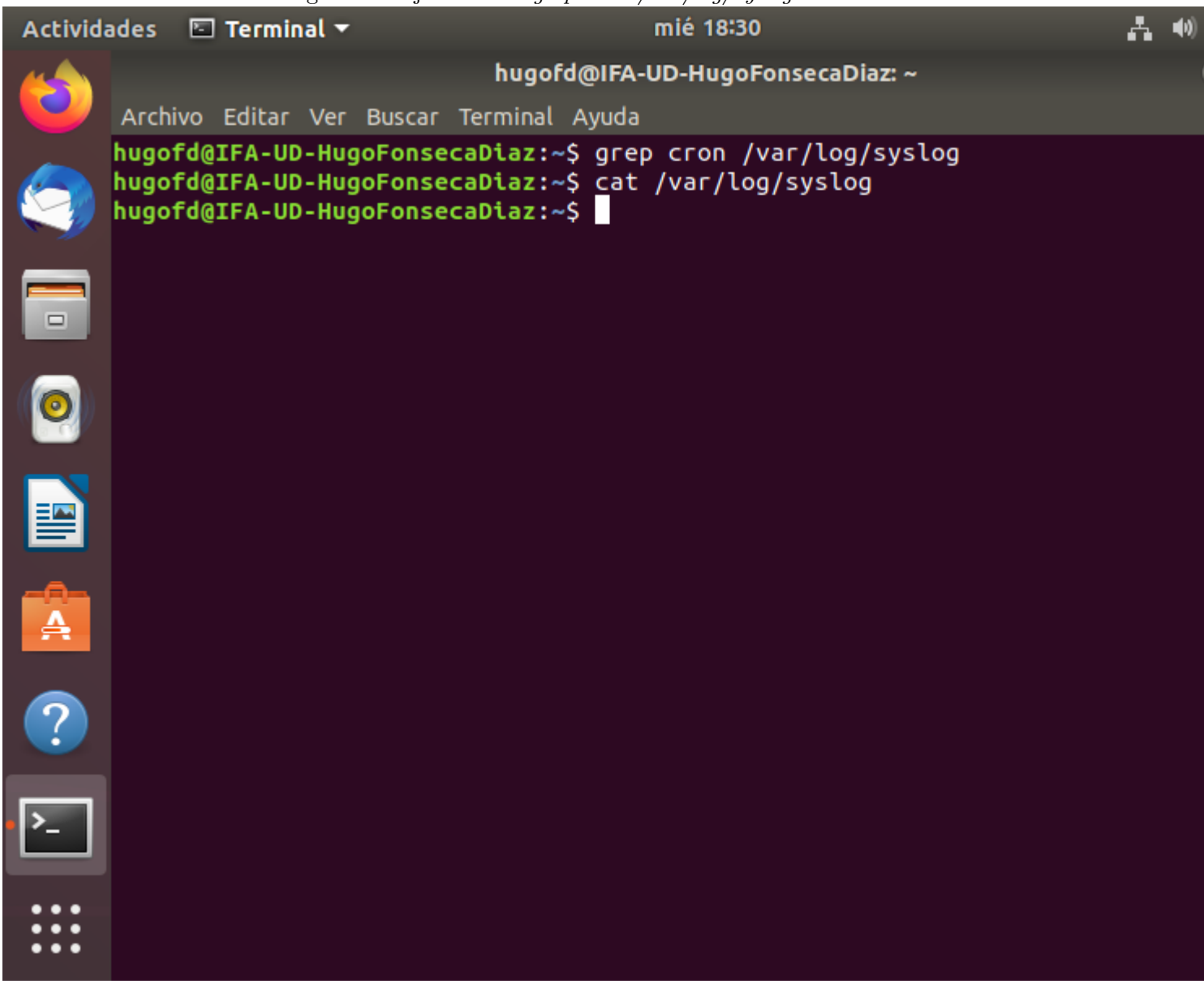
Figura 16: Ejercicio 11: *sudo fdisk -l /dev/sda.*



12. Ejercicio 12

El fichero donde el kernel almacena las acciones realizadas por `cron` se encuentra en */var/log/syslog*. Puede hacerse un `grep` con la string *cron* en dicho archivo para visualizar las acciones, sin embargo, debido al poco espacio en la partición */var*, en nuestro caso ese archivo está vacío.

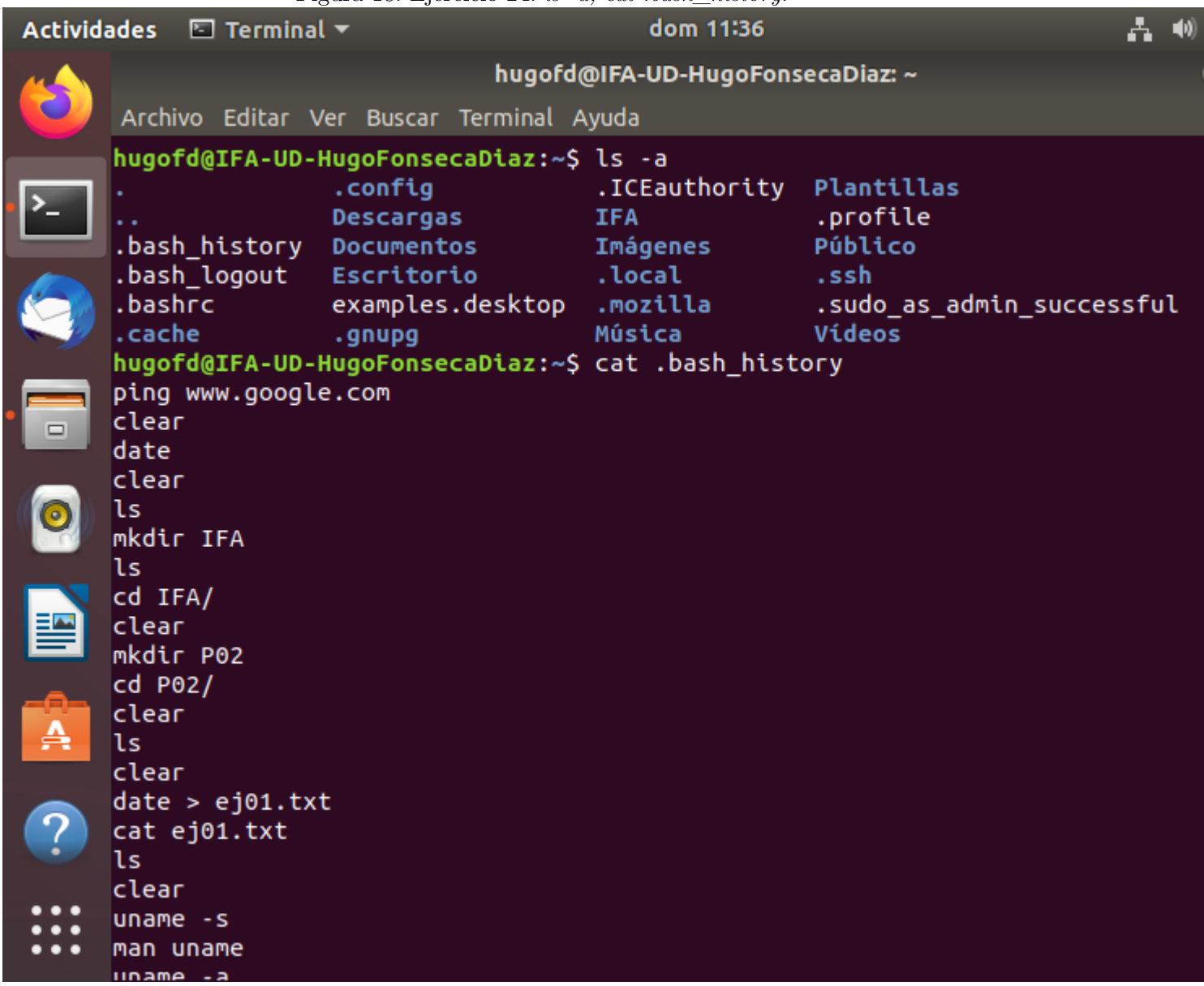
Figura 17: Ejercicio 12: *grep cron /var/log/syslog*.



13. Ejercicio 24

El historial de comandos de `bash` del usuario se encuentra en un fichero oculto de su carpeta *HOME* llamado *bash_history*. Se puede utilizar el comando `ls` con la flag *a* para listar todos los archivos, incluidos los ocultos, para comprobar que efectivamente existe el archivo del historial.

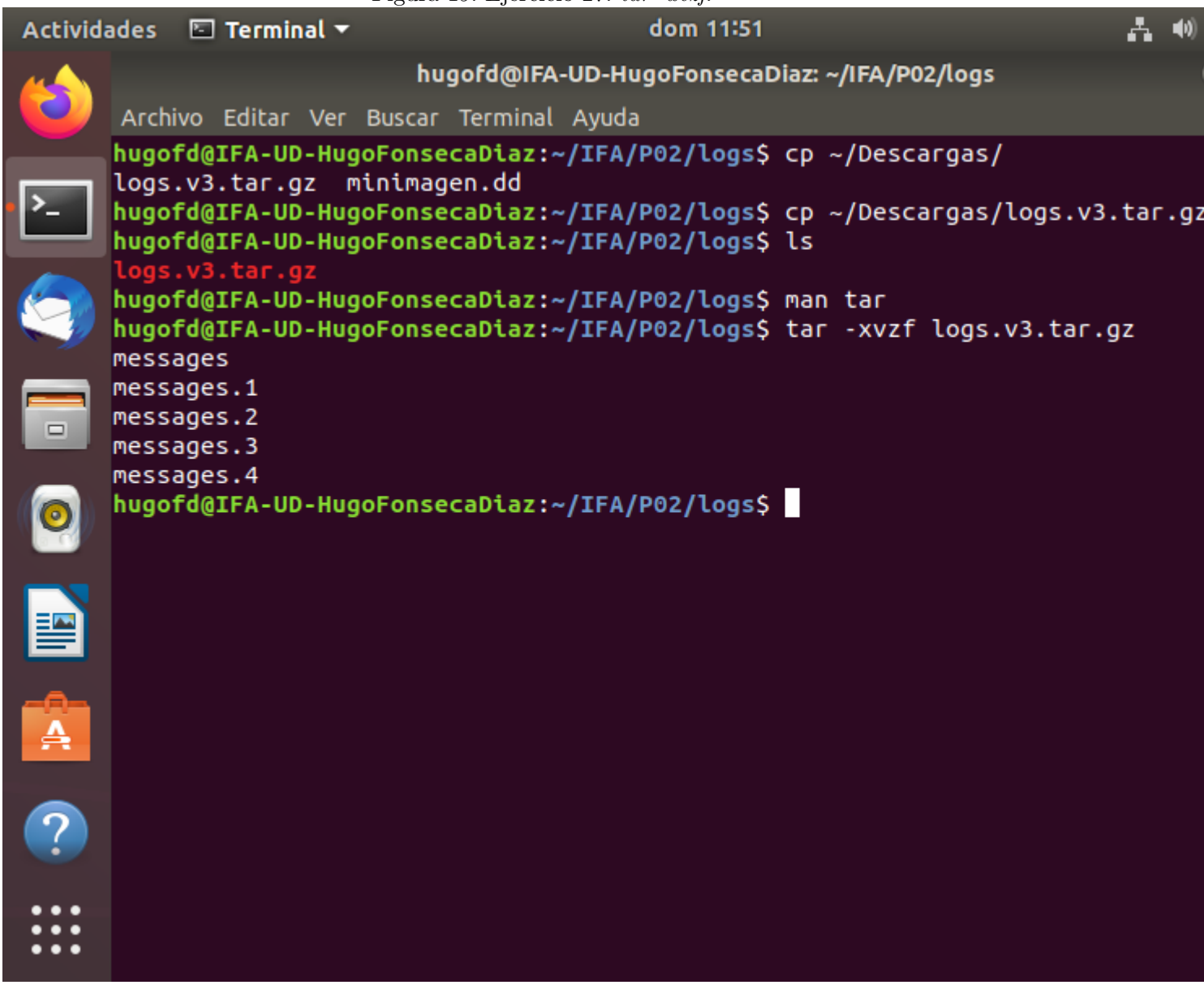
Figura 18: Ejercicio 24: *ls -a; cat .bash_history.*



14. Ejercicio 27

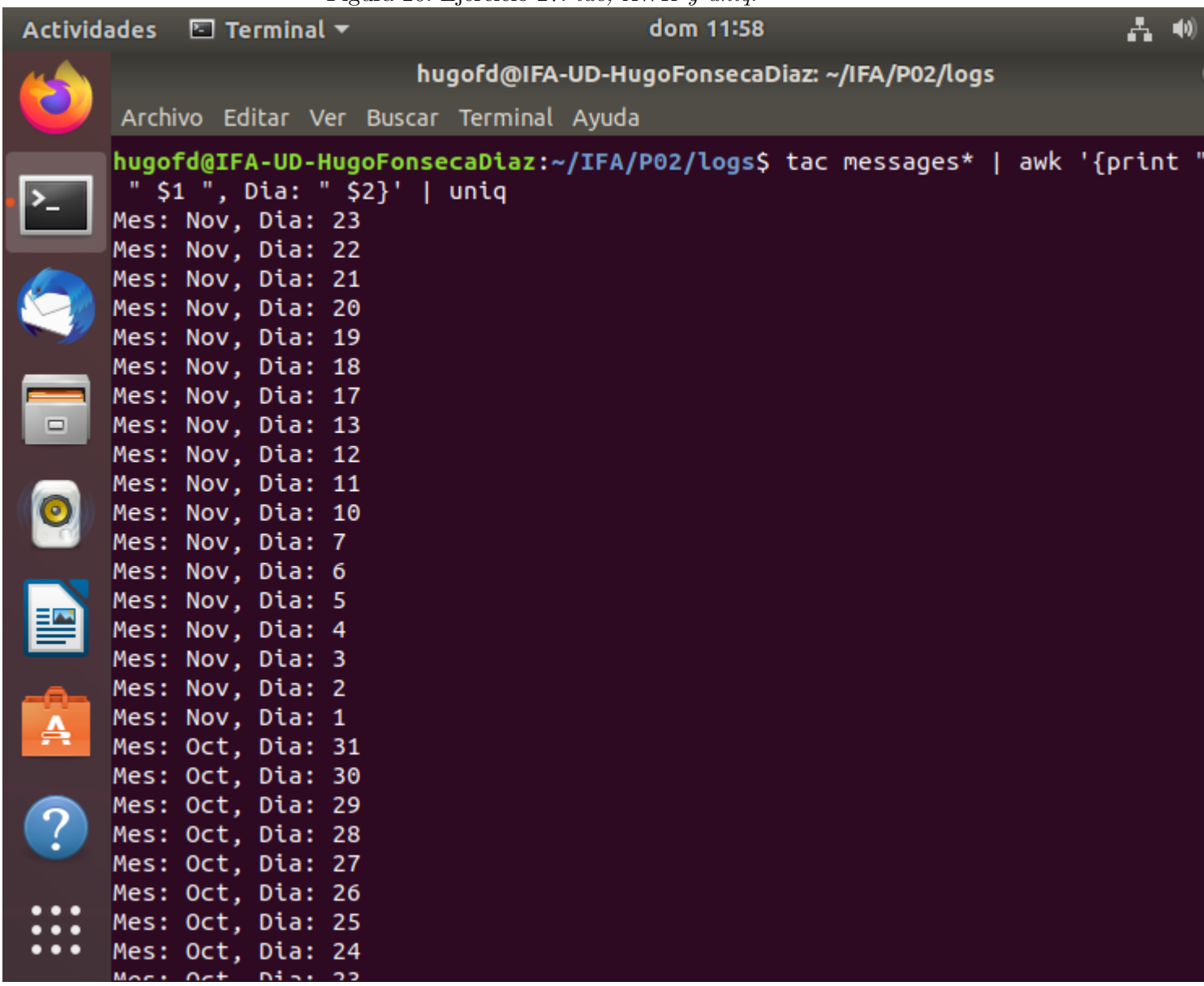
Se descomprime el archivo con el comando `tar` y las flags *xvzf*, siendo x una indicación de que se quiere extraer los contenidos del archivo comprimido, v para que lo haga de manera verbosa, z para indicarle al comando que el archivo es un zip y f para pasarle el fichero que se desea extraer al comando.

Figura 19: Ejercicio 27: *tar -xvzf*.



Una vez descomprimidos los ficheros de texto, se procede a utilizar tres nuevas herramientas. Se usa `tac` para concatenar ficheros de forma inversa (es el comando `cat` invertido), el lenguaje de programación AWK para procesar texto y el comando `uniq` para omitir líneas repetidas.

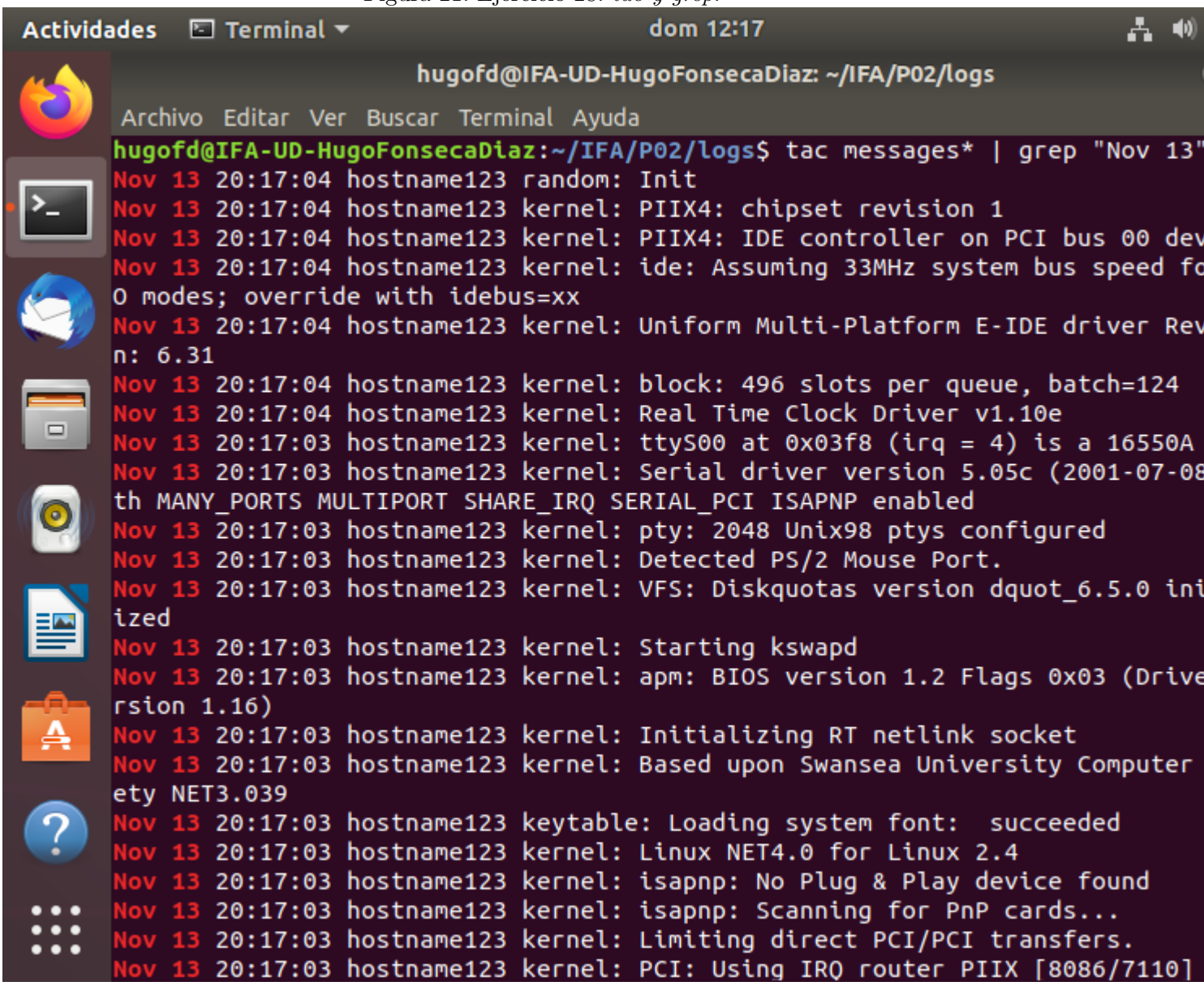
Figura 20: Ejercicio 27: *tac, AWK y uniq.*



15. Ejercicio 28

Se usan los comandos `tac` y `grep`. El primero se usa para concatenar inversamente los ficheros de los mensajes y el segundo para buscar las líneas donde aparece la cadena de texto *"Nov 13"*.

Figura 21: Ejercicio 28: *tac y grep*.



Referencias