

IFA. Práctica de laboratorio 02

Hugo Fonseca Díaz
email `uo258318@uniovi.es`

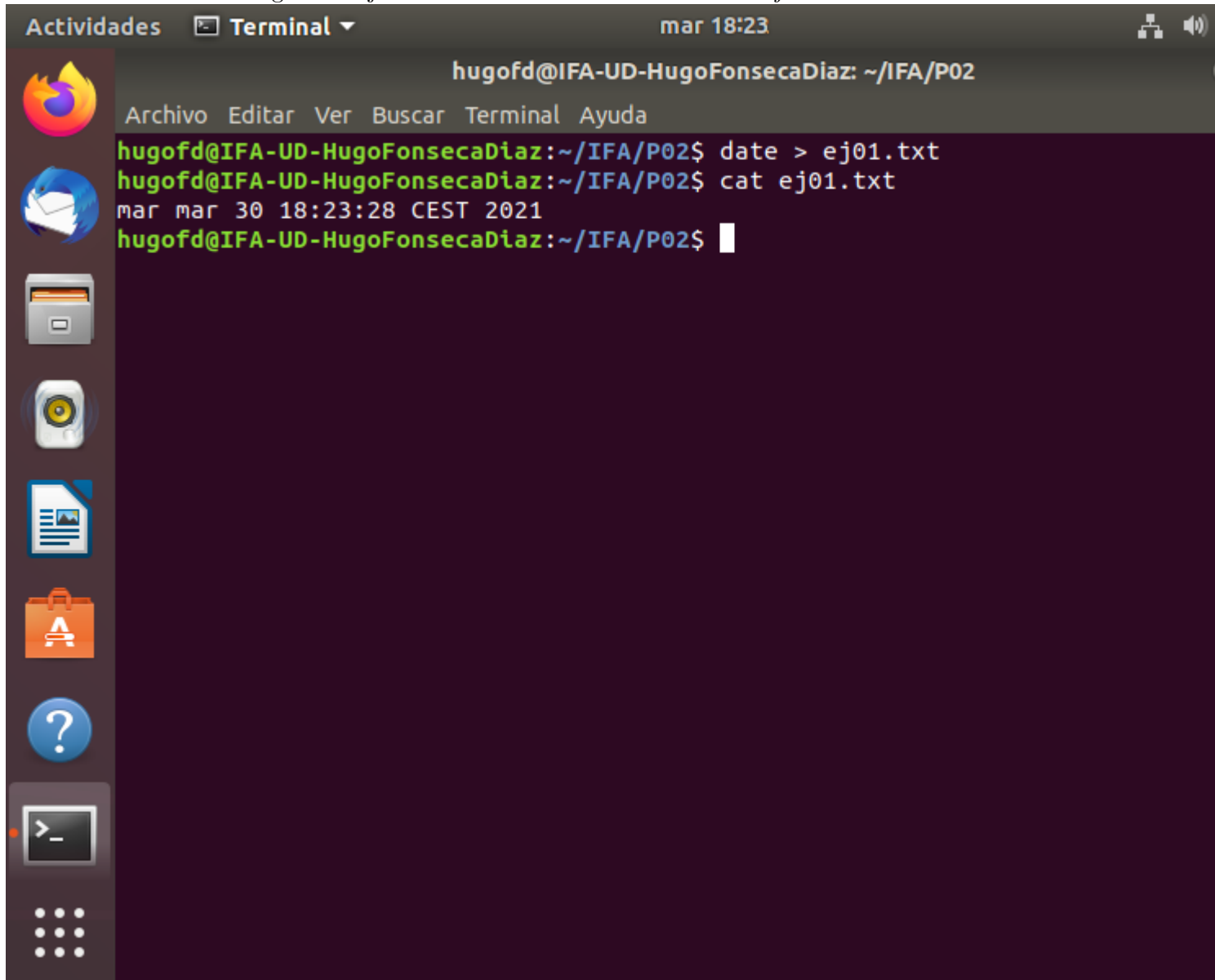
Escuela de Ingeniería Informática. Universidad de Oviedo.

26 de mayo 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`.



The image shows a terminal window titled "Terminal" with a menu bar containing "Archivo", "Editar", "Ver", "Buscar", "Terminal", and "Ayuda". The prompt is "hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02". The user enters the command `date > ej01.txt`, followed by `cat ej01.txt`, which outputs `mar mar 30 18:23:28 CEST 2021`. The terminal window is part of a desktop environment with a sidebar on the left containing icons for Firefox, a mail client, a file manager, a music player, a document viewer, a shopping bag, a help icon, and a terminal icon. The top bar shows "Actividades", "Terminal", and the time "mar 18:23".

```
hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ date > ej01.txt
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ cat ej01.txt
mar mar 30 18:23:28 CEST 2021
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$
```

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

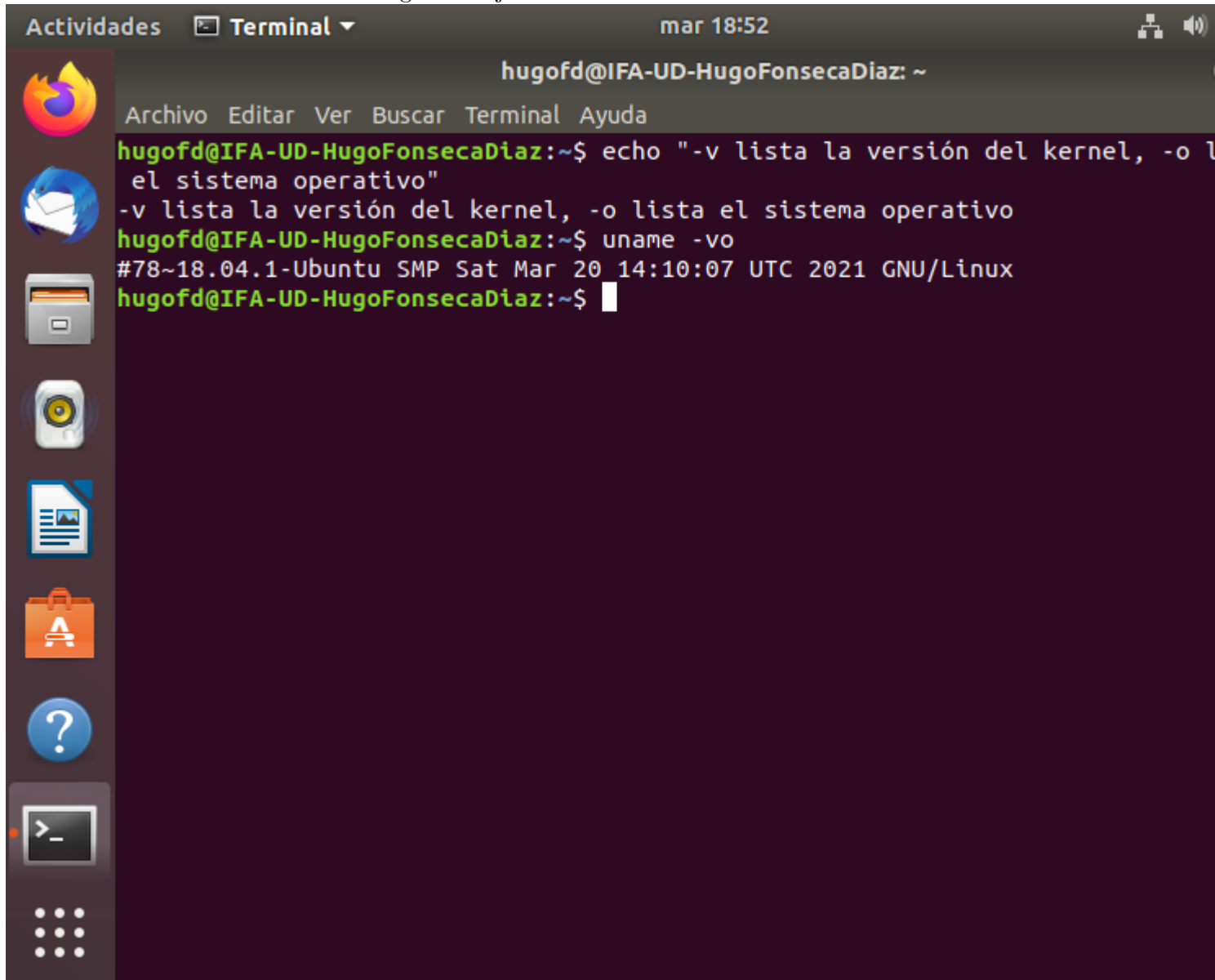
Figura 2: Ejercicio 1: Hora en el sitio web *time.is*.



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.

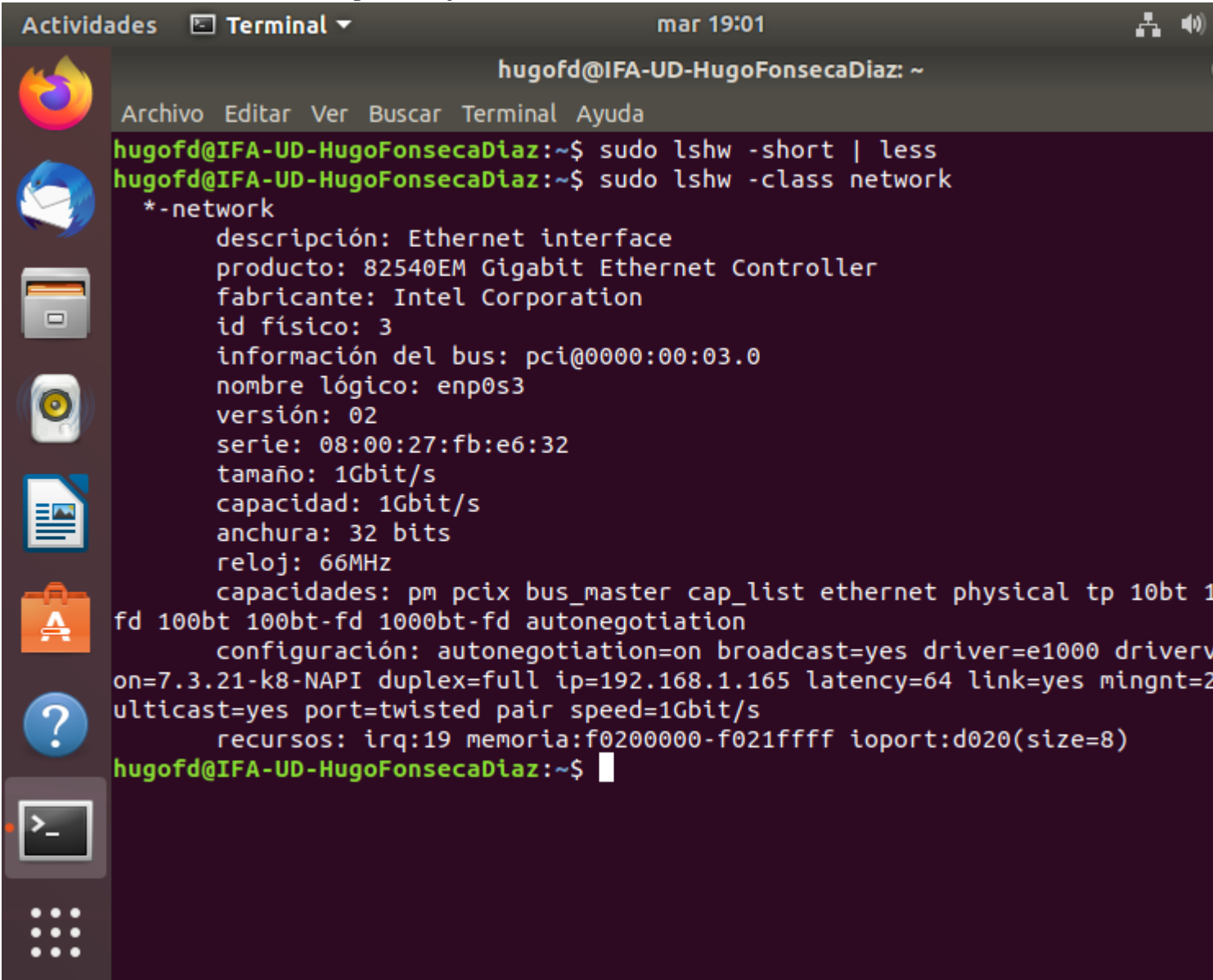
Figura 4: Ejercicio 3: *lshw -short*.

```

hugofd@IFA-UD-HugoFonsecaDiaz: ~
Archivo Editar Ver Buscar Terminal Ayuda
/0/0          memory      128KiB BIOS
/0/1          memory      1987MiB Memoria de sistema
/0/2          processor   Intel(R) Core(TM) i7-8550U CPU @ 1
Hz
/0/100        bridge      440FX - 82441FX PMC [Natoma]
/0/100/1      bridge      82371SB PIIX3 ISA [Natoma/Triton I
/0/100/1.1    storage     82371AB/EB/MB PIIX4 IDE
/0/100/2      display     SVGA II Adapter
/0/100/3      enp0s3      network     82540EM Gigabit Ethernet Controlle
/0/100/4      generic     VirtualBox Guest Service
/0/100/5      multimedia  82801AA AC'97 Audio Controller
/0/100/6      bus         KeyLargo/Intrepid USB
/0/100/6/1    usb1        bus         OHCI PCI host controller
/0/100/6/1/1  input       USB Tablet
/0/100/7      bridge      82371AB/EB/MB PIIX4 ACPI
/0/100/d      storage     82801HM/HEM (ICH8M/ICH8M-E) SATA C
oller [AHCI mode]
/0/3          scsi1       storage     disk
/0/3/0.0.0    /dev/cdrom  disk        CD-ROM
/0/4          scsi2       storage     disk
/0/4/0.0.0    /dev/sda    disk        42GB VBOX HARDDISK
/0/4/0.0.0/1  /dev/sda1   volume     5721MiB partici3n EXT4
/0/4/0.0.0/2  /dev/sda2   volume     4768MiB partici3n EXT4
/0/4/0.0.0/3  /dev/sda3   volume     23GiB partici3n EXT4
/0/4/0.0.0/4  /dev/sda4   volume     6626MiB Extended partition
/0/4/0.0.0/4/5 /dev/sda5   volume     1906MiB partici3n EXT4
/0/4/0.0.0/4/6 /dev/sda6   volume     1904MiB partici3n EXT4
:
  
```

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*.

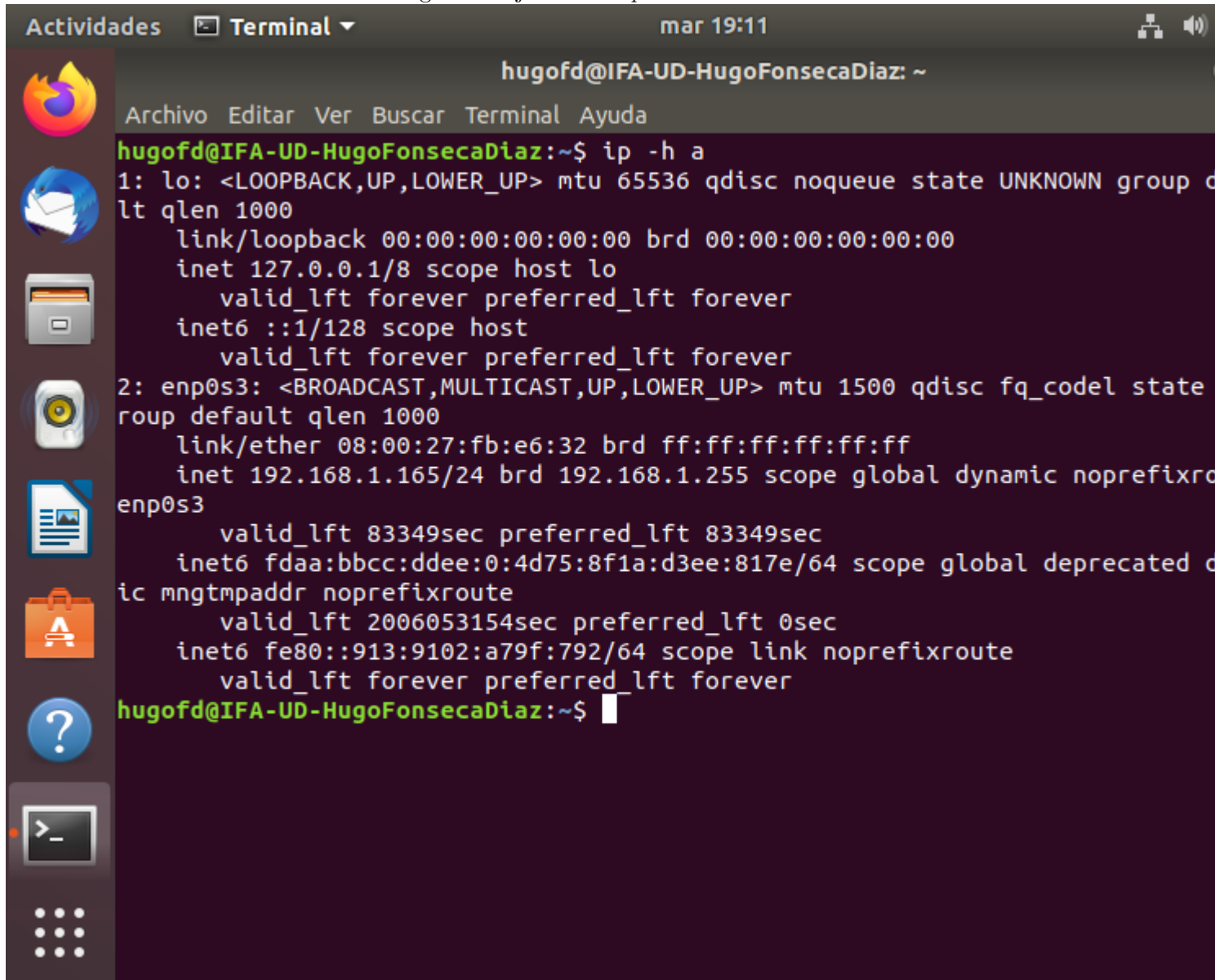


The image shows a terminal window titled "Terminal" with the user "hugofd@IFA-UD-HugoFonsecaDiaz". The terminal displays the command `sudo lshw -short | less` followed by `sudo lshw -class network`. The output shows details for the Ethernet interface `enp0s3`, including its manufacturer (Intel Corporation), speed (1Gbit/s), and various configuration parameters like `autonegotiation=on` and `duplex=full`.

```
hugofd@IFA-UD-HugoFonsecaDiaz: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
hugofd@IFA-UD-HugoFonsecaDiaz:~$ sudo lshw -short | less  
hugofd@IFA-UD-HugoFonsecaDiaz:~$ sudo lshw -class network  
*-network  
   descripción: Ethernet interface  
   producto: 82540EM Gigabit Ethernet Controller  
   fabricante: Intel Corporation  
   id físico: 3  
   información del bus: pci@0000:00:03.0  
   nombre lógico: enp0s3  
   versión: 02  
   serie: 08:00:27:fb:e6:32  
   tamaño: 1Gbit/s  
   capacidad: 1Gbit/s  
   anchura: 32 bits  
   reloj: 66MHz  
   capacidades: pm pcix bus_master cap_list ethernet physical tp 10bt 1  
fd 100bt 100bt-fd 1000bt-fd autonegotiation  
   configuración: autonegotiation=on broadcast=yes driver=e1000 driverv  
on=7.3.21-k8-NAPI duplex=full ip=192.168.1.165 latency=64 link=yes mingnt=2  
ulticast=yes port=twisted pair speed=1Gbit/s  
   recursos: irq:19 memoria:f0200000-f021ffff ioport:d020(size=8)  
hugofd@IFA-UD-HugoFonsecaDiaz:~$
```

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

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



```
hugofd@IFA-UD-HugoFonsecaDiaz: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
hugofd@IFA-UD-HugoFonsecaDiaz:~$ ip -h a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group d  
lt qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state  
roup default qlen 1000  
    link/ether 08:00:27:fb:e6:32 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.1.165/24 brd 192.168.1.255 scope global dynamic noprefixro  
enp0s3  
        valid_lft 83349sec preferred_lft 83349sec  
    inet6 fdad:bbcc:ddee:0:4d75:8f1a:d3ee:817e/64 scope global deprecated d  
ic mngtmpaddr noprefixroute  
        valid_lft 2006053154sec preferred_lft 0sec  
    inet6 fe80::913:9102:a79f:792/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
hugofd@IFA-UD-HugoFonsecaDiaz:~$
```

4. Ejercicio 4

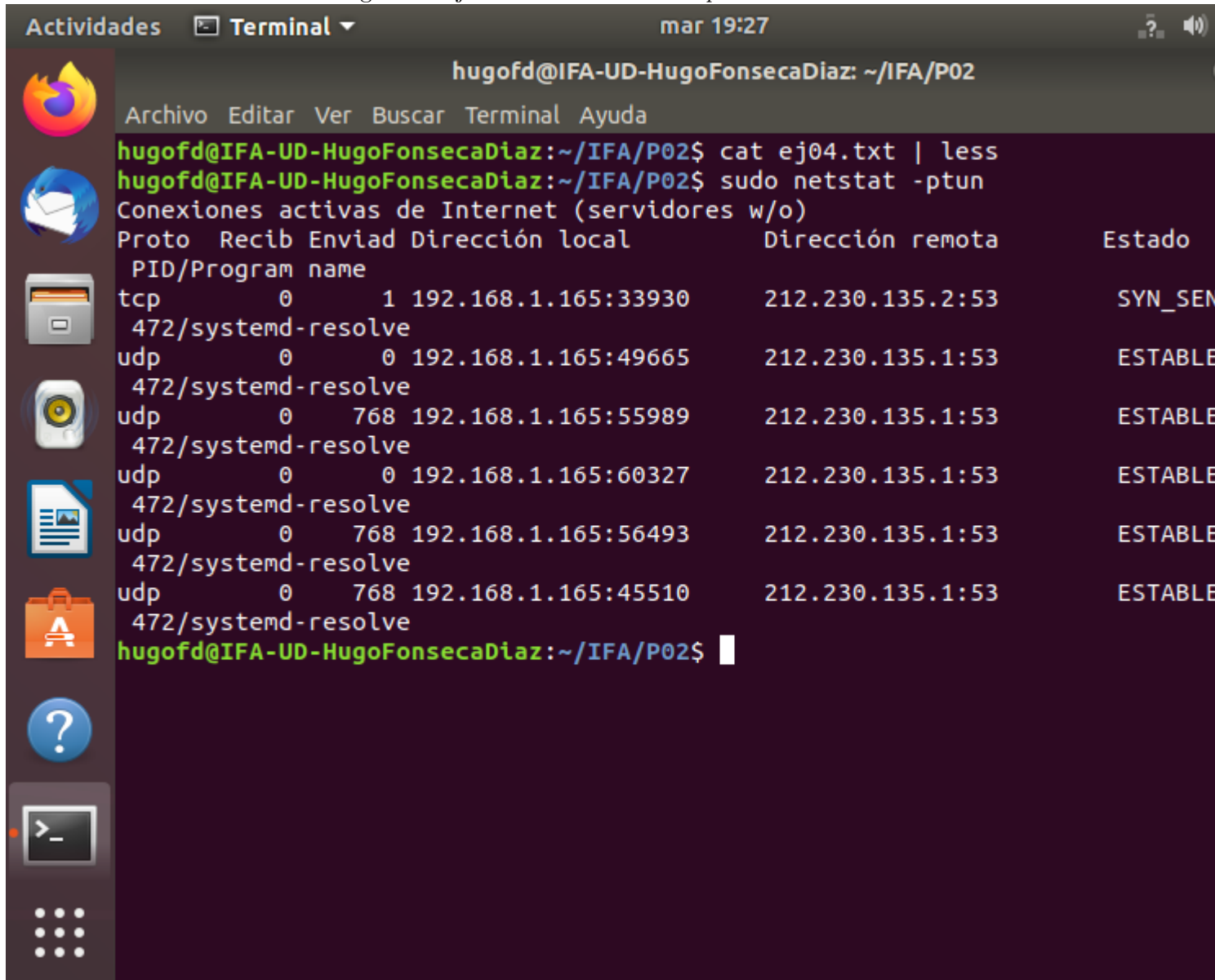
Se utiliza el comando `netstat` del paquete `net-tools`. Su flag `a` permite ver todos los sockets, por lo que `sudo netstat -a > ej04.txt` guarda la información de los sockets activos y no activos en un fichero de texto. También son interesantes sus flags `n` (se muestran las direcciones numéricamente),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

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

```

hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02
Archivo Editar Ver Buscar Terminal Ayuda
raw6      0      0 [::]:ipv6-icmp      [::]:*      7
Sockets activos de dominio UNIX (servidores y establecidos)
Proto RefCnt Flags      Type      State      I-Node      Ruta
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 27759      @/tmp/.ICE-unix/
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 27310      @/tmp/dbus-q0eqb
unix  2      [ ]        DGRAM      27179      /run/user/1000/s
md/notify
unix  2      [ ]        DGRAM      22206      /run/user/121/sy
d/notify
unix  2      [ ACC ]      SEQPACKET  ESCUCHANDO 13206      /run/udev/control
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 27182      /run/user/1000/s
md/private
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 22209      /run/user/121/sy
d/private
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 27186      /run/user/1000/g
/S.gpg-agent.extra
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 22377      /run/user/121/gn
S.gpg-agent.extra
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 27187      /run/user/1000/s
-session-agent.socket
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 22378      /run/user/121/bu
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 27188      /run/user/1000/g
/S.gpg-agent.browser
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 27189      /run/user/1000/g
/S.gpg-agent
unix  2      [ ACC ]      FLUJO      ESCUCHANDO 22379      /run/user/121/pu
native
:
  
```

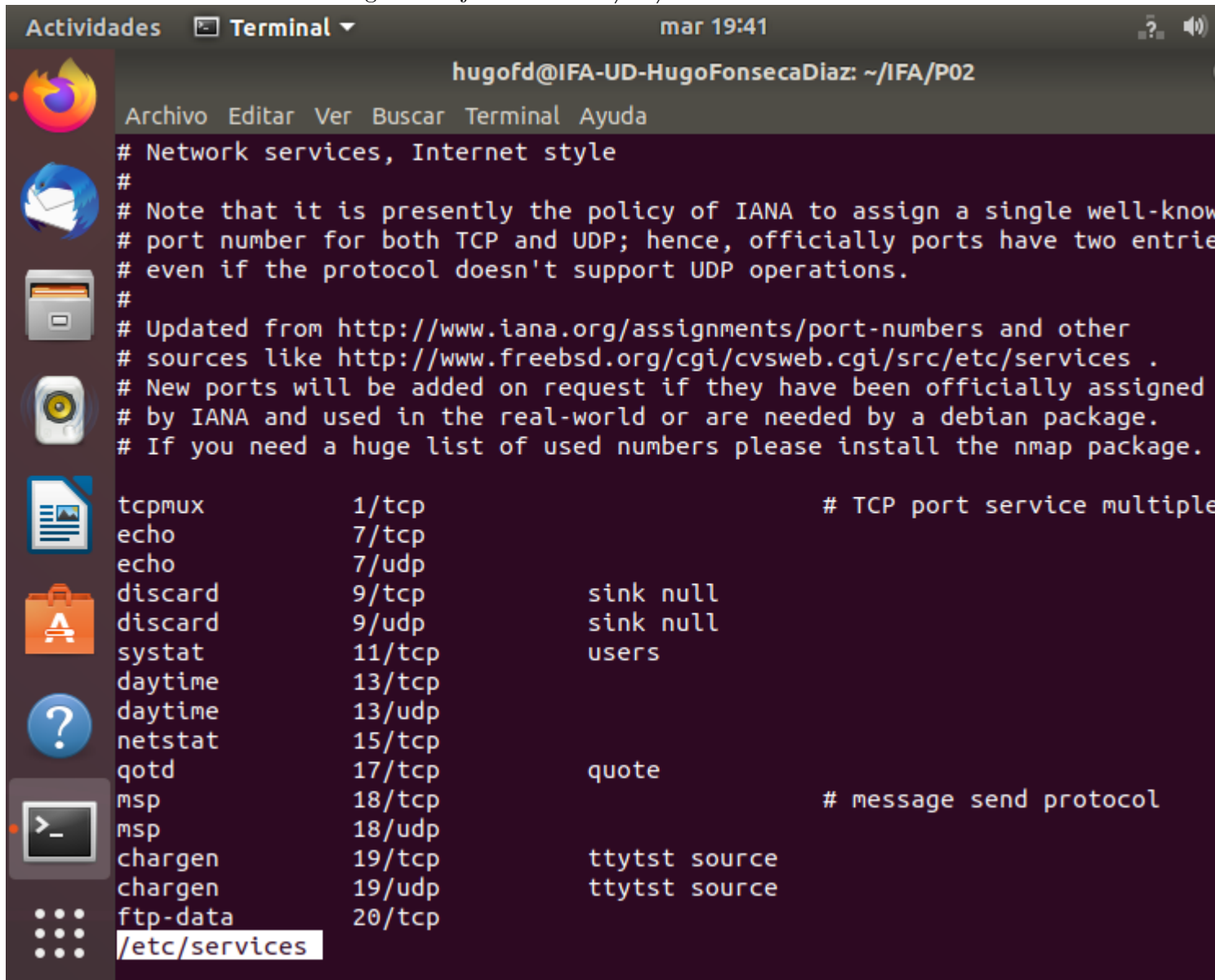

Figura 8: Ejercicio 4: *sudo netstat -ptun*.



```
hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02
Archivo Editar Ver Buscar Terminal Ayuda
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ cat ej04.txt | less
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ sudo netstat -ptun
Conexiones activas de Internet (servidores w/o)
Proto  Recib Envia Dirección local      Dirección remota      Estado
PID/Program name
tcp    0      1 192.168.1.165:33930  212.230.135.2:53      SYN_SEM
472/systemd-resolve
udp    0      0 192.168.1.165:49665  212.230.135.1:53      ESTABLE
472/systemd-resolve
udp    0    768 192.168.1.165:55989  212.230.135.1:53      ESTABLE
472/systemd-resolve
udp    0      0 192.168.1.165:60327  212.230.135.1:53      ESTABLE
472/systemd-resolve
udp    0    768 192.168.1.165:56493  212.230.135.1:53      ESTABLE
472/systemd-resolve
udp    0    768 192.168.1.165:45510  212.230.135.1:53      ESTABLE
472/systemd-resolve
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$
```

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

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



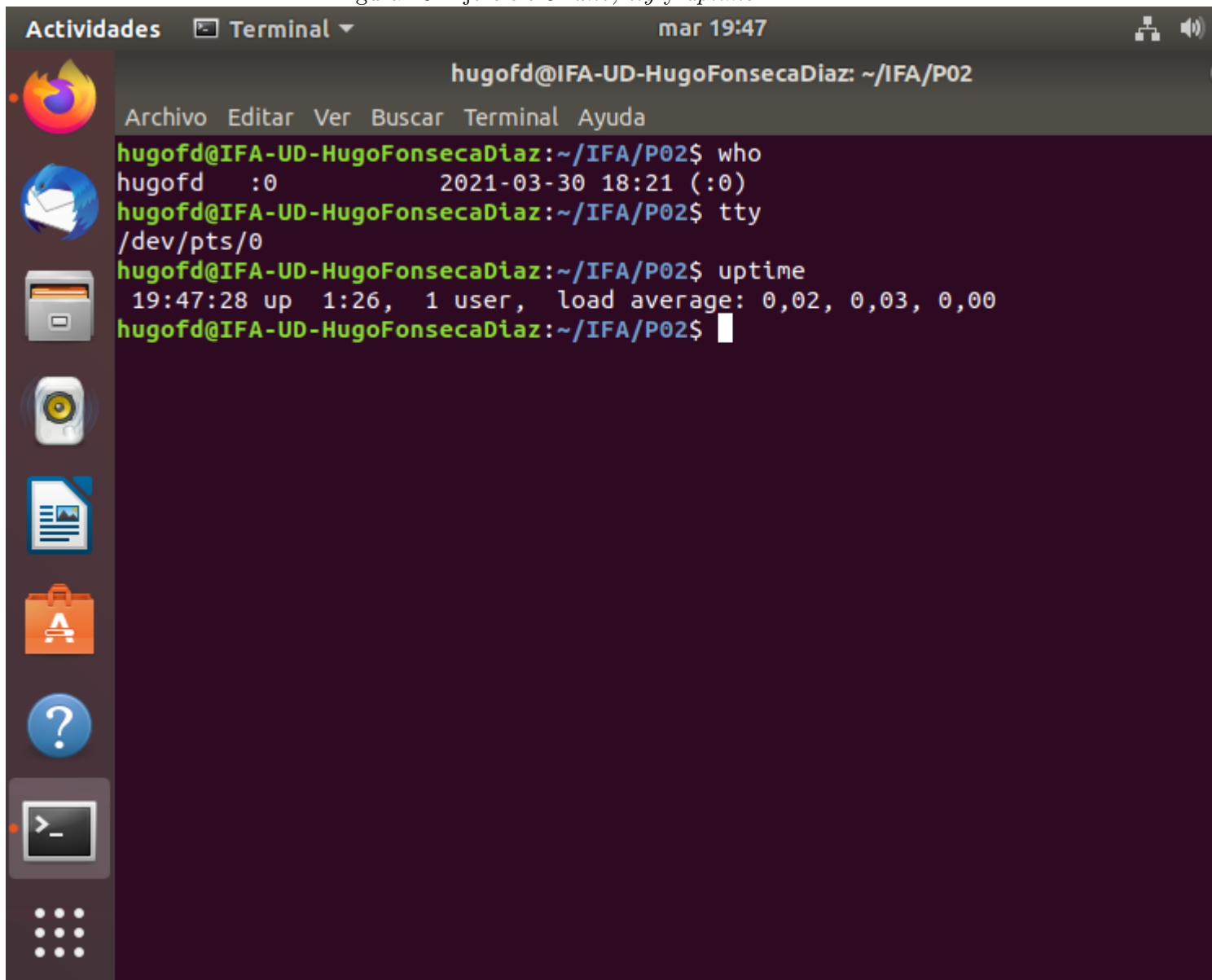
```
hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02
# Network services, Internet style
#
# Note that it is presently the policy of IANA to assign a single well-known
# port number for both TCP and UDP; hence, officially ports have two entries
# even if the protocol doesn't support UDP operations.
#
# Updated from http://www.iana.org/assignments/port-numbers and other
# sources like http://www.freebsd.org/cgi/cvsweb.cgi/src/etc/services .
# New ports will be added on request if they have been officially assigned
# by IANA and used in the real-world or are needed by a debian package.
# If you need a huge list of used numbers please install the nmap package.

tcpmux      1/tcp          # TCP port service multiplex
echo        7/tcp
echo        7/udp
discard     9/tcp          sink null
discard     9/udp          sink null
sysstat     11/tcp         users
daytime     13/tcp
daytime     13/udp
netstat     15/tcp
qotd        17/tcp          quote
msp         18/tcp          # message send protocol
msp         18/udp
chargen     19/tcp          ttytst source
chargen     19/udp          ttytst source
ftp-data    20/tcp
```

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*.



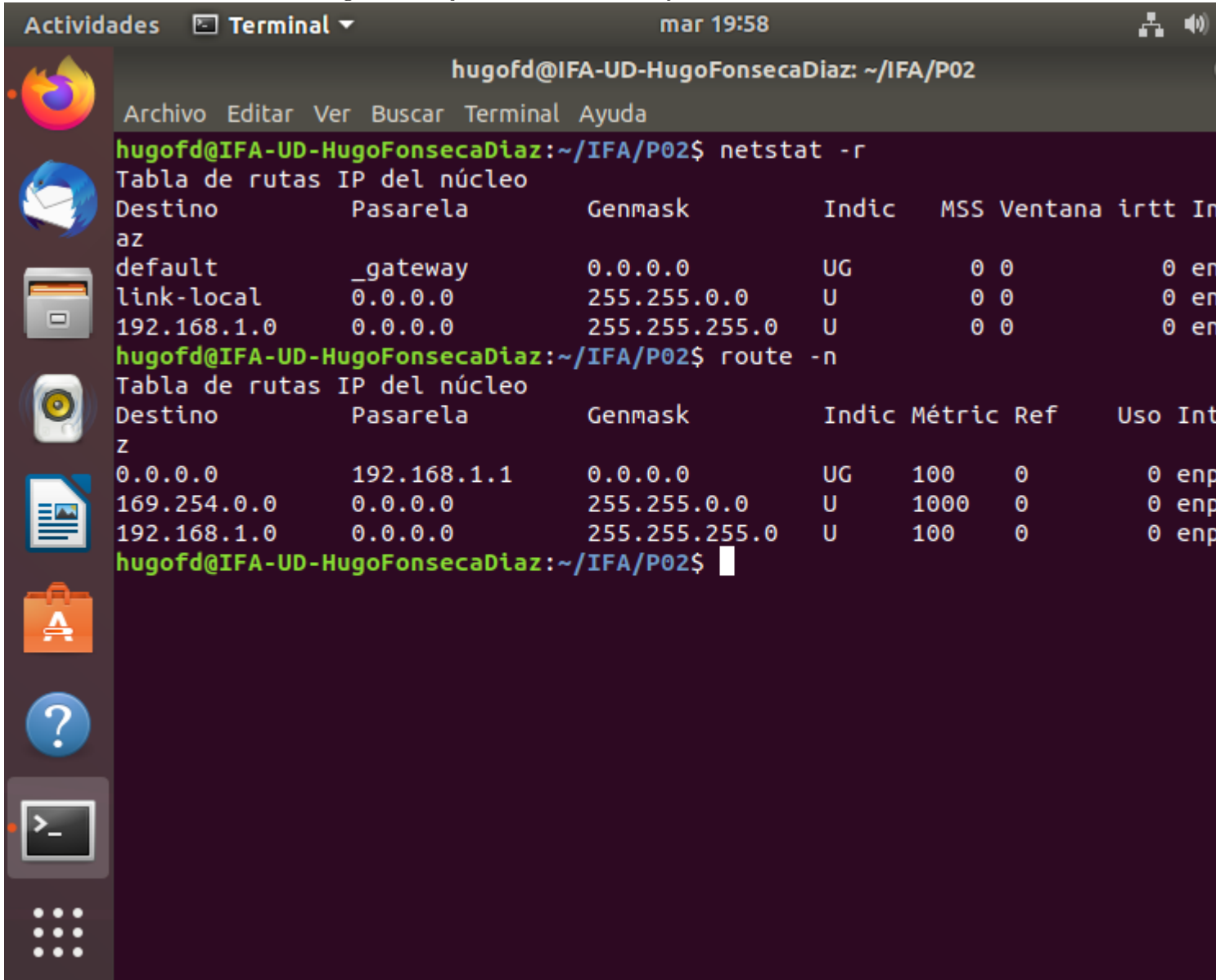
The image shows a terminal window titled "hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02". The window has a menu bar with "Archivo", "Editar", "Ver", "Buscar", "Terminal", and "Ayuda". The terminal output is as follows:

```
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ who
hugofd    :0                2021-03-30 18:21 (:0)
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ tty
/dev/pts/0
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ uptime
 19:47:28 up  1:26,  1 user,  load average: 0,02, 0,03, 0,00
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$
```

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*.



The screenshot shows a terminal window titled "hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02". The terminal displays the output of two commands: `netstat -r` and `route -n`. The `netstat -r` output shows the IP routing table with columns: Destino, Pasarela, Genmask, Indic, MSS, Ventana, irtt, and In. The `route -n` output shows the kernel routing table with columns: Destino, Pasarela, Genmask, Indic, Métric, Ref, Uso, and Int.

```
hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02
Archivo Editar Ver Buscar Terminal Ayuda
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ netstat -r
Tabla de rutas IP del núcleo
Destino      Pasarela      Genmask      Indic  MSS  Ventana  irtt  In
az
default      _gateway      0.0.0.0      UG     0 0      0 en
link-local   0.0.0.0      255.255.0.0  U     0 0      0 en
192.168.1.0   0.0.0.0      255.255.255.0 U     0 0      0 en
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$ route -n
Tabla de rutas IP del núcleo
Destino      Pasarela      Genmask      Indic  Métric  Ref    Uso  Int
0.0.0.0      192.168.1.1   0.0.0.0      UG     100     0      0 enp
169.254.0.0   0.0.0.0      255.255.0.0  U     1000    0      0 enp
192.168.1.0   0.0.0.0      255.255.255.0 U     100     0      0 enp
hugofd@IFA-UD-HugoFonsecaDiaz:~/IFA/P02$
```

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 auxu | less*.

```

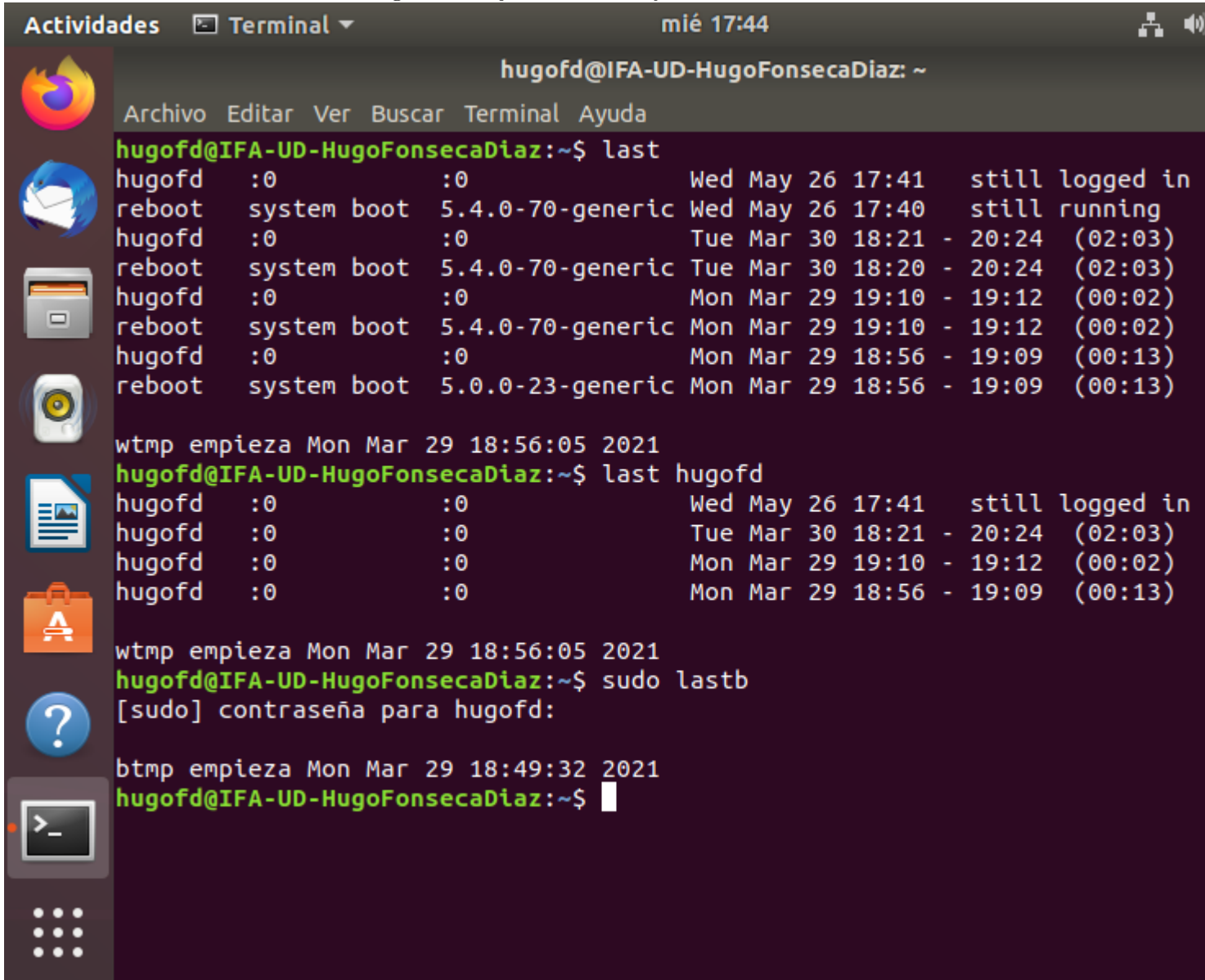
hugofd@IFA-UD-HugoFonsecaDiaz: ~/IFA/P02
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
TY      STAT  START   TIME  COMMAND
Ss      18:20   0:04   /sbin/init splash
S        18:20   0:00   [kthreadd]
I<       18:20   0:00   [rcu_gp]
I<       18:20   0:00   [rcu_par_gp]
I<       18:20   0:00   [kworker/0:0H-kb]
I<       18:20   0:00   [mm_percpu_wq]
S        18:20   0:00   [ksoftirqd/0]
I        18:20   0:01   [rcu_sched]
S        18:20   0:00   [migration/0]
S        18:20   0:00   [idle_inject/0]
S        18:20   0:00   [cpuhp/0]
S        18:20   0:00   [kdevtmpfs]
I<       18:20   0:00   [netns]
S        18:20   0:00   [rcu_tasks_kthre]
S        18:20   0:00   [kauditd]
S        18:20   0:00   [khungtaskd]
S        18:20   0:00   [oom_reaper]
I<       18:20   0:00   [writeback]
S        18:20   0:00   [kcompactd0]
SN       18:20   0:00   [ksmd]
SN       18:20   0:00   [khugepaged]
I<       18:20   0:00   [kintegrityd]
I<       18:20   0:00   [kblockd]
I<       18:20   0:00   [blkcg_punt_bio]
I<       18:20   0:00   [tpm_dev_wq]
I<       18:20   0:00   [ata_sff]

```

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*.



The screenshot shows a terminal window titled "hugofd@IFA-UD-HugoFonsecaDiaz: ~" with a menu bar containing "Archivo", "Editar", "Ver", "Buscar", "Terminal", and "Ayuda". The terminal displays the output of the `last` command, which lists login sessions for the user `hugofd`. The output shows sessions on May 26, March 30, and March 29. The `lastb` command is also executed, showing failed login attempts for the same user on March 29. The terminal window is part of a desktop environment with a sidebar on the left containing icons for Firefox, a mail client, a file manager, a disk utility, a terminal, and a help icon.

```
hugofd@IFA-UD-HugoFonsecaDiaz:~$ last
hugofd      :0                :0                Wed May 26 17:41    still logged in
reboot      system boot      5.4.0-70-generic Wed May 26 17:40    still running
hugofd      :0                :0                Tue Mar 30 18:21 - 20:24 (02:03)
reboot      system boot      5.4.0-70-generic Tue Mar 30 18:20 - 20:24 (02:03)
hugofd      :0                :0                Mon Mar 29 19:10 - 19:12 (00:02)
reboot      system boot      5.4.0-70-generic Mon Mar 29 19:10 - 19:12 (00:02)
hugofd      :0                :0                Mon Mar 29 18:56 - 19:09 (00:13)
reboot      system boot      5.0.0-23-generic Mon Mar 29 18:56 - 19:09 (00:13)

wtmp empieza Mon Mar 29 18:56:05 2021
hugofd@IFA-UD-HugoFonsecaDiaz:~$ last hugofd
hugofd      :0                :0                Wed May 26 17:41    still logged in
hugofd      :0                :0                Tue Mar 30 18:21 - 20:24 (02:03)
hugofd      :0                :0                Mon Mar 29 19:10 - 19:12 (00:02)
hugofd      :0                :0                Mon Mar 29 18:56 - 19:09 (00:13)

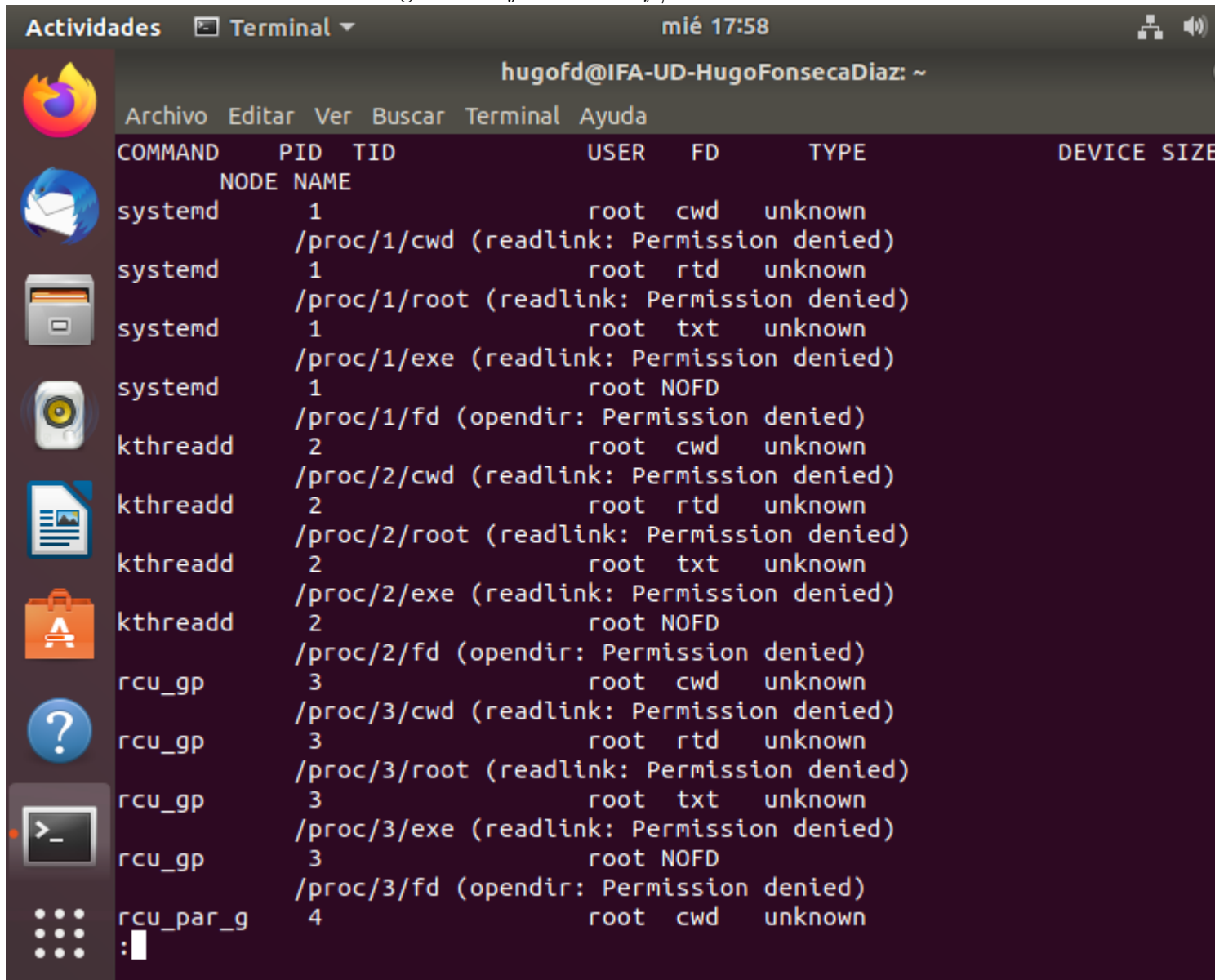
wtmp empieza Mon Mar 29 18:56:05 2021
hugofd@IFA-UD-HugoFonsecaDiaz:~$ sudo lastb
[sudo] contraseña para hugofd:

btmp empieza Mon Mar 29 18:49:32 2021
hugofd@IFA-UD-HugoFonsecaDiaz:~$
```

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*.



```

hugofd@IFA-UD-HugoFonsecaDiaz: ~
COMMAND  PID  TID  USER  FD  TYPE  DEVICE  SIZE
NODE NAME
systemd  1    1    root  cwd  unknown
/proc/1/cwd (readlink: Permission denied)
systemd  1    1    root  rtd  unknown
/proc/1/root (readlink: Permission denied)
systemd  1    1    root  txt  unknown
/proc/1/exe (readlink: Permission denied)
systemd  1    1    root  NOFD
/proc/1/fd (opendir: Permission denied)
kthreadd 2    2    root  cwd  unknown
/proc/2/cwd (readlink: Permission denied)
kthreadd 2    2    root  rtd  unknown
/proc/2/root (readlink: Permission denied)
kthreadd 2    2    root  txt  unknown
/proc/2/exe (readlink: Permission denied)
kthreadd 2    2    root  NOFD
/proc/2/fd (opendir: Permission denied)
rcu_gp   3    3    root  cwd  unknown
/proc/3/cwd (readlink: Permission denied)
rcu_gp   3    3    root  rtd  unknown
/proc/3/root (readlink: Permission denied)
rcu_gp   3    3    root  txt  unknown
/proc/3/exe (readlink: Permission denied)
rcu_gp   3    3    root  NOFD
/proc/3/fd (opendir: Permission denied)
rcu_par_g 4    4    root  cwd  unknown
:

```

Referencias