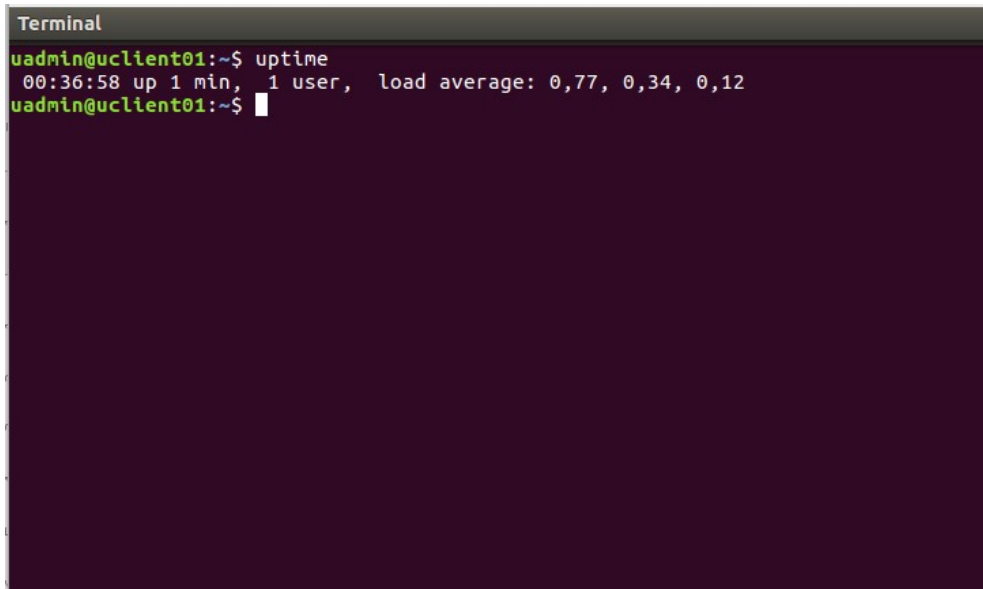


P12c. Procesos

Los **procesos** en Linux (o en cualquier sistema operativo) son **programas que están en ejecución** y se identifican mediante el **PID**. Es importante responder a las cuestiones que se plantean.

1. Comando **uptime**



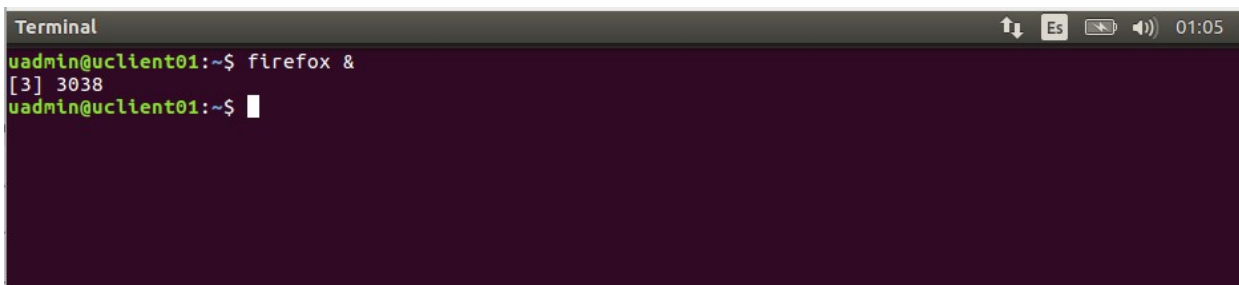
```
Terminal
uadmin@uclient01:~$ uptime
00:36:58 up 1 min, 1 user, load average: 0,77, 0,34, 0,12
uadmin@uclient01:~$
```

¿qué valores nos muestra el comando **uptime**?

Ejecuta la orden **top** y compara con el comando **uptime** ¿y la orden **w**?

2. La orden **ps tree** muestra los procesos. Ejecutando la orden **ps** podemos ver los **PID** y el **PPID** son efectivamente los números de los procesos padres e hijos. Lancemos una orden **firefox &** y anotemos el número que resulta.

¿qué significa el símbolo **&**?



```
Terminal
uadmin@uclient01:~$ firefox &
[3] 3038
uadmin@uclient01:~$
```

3. A continuación ejecutamos **ps -lax** y anotamos el valor PID de nano ¿que PPID supones que le corresponde?

```

ucient01 (finS4) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminal
4 0 2239 1 20 0 620508 28924 - SL ? 0:00 /usr/lib/x86_64-linux-gnu/fwupd/fw
0 1000 2275 1636 20 0 353356 4908 poll_s SL ? 0:00 /usr/lib/gvfs/gvfsd-trash --spawne
0 1000 2315 1886 20 0 412804 11180 poll_s SL ? 0:00 zeitgeist-datahub
0 1000 2322 1636 20 0 4508 516 wait S ? 0:00 /bin/sh -c /usr/lib/x86_64-linux-g
0 1000 2326 2322 20 0 344628 4892 poll_s SL ? 0:00 /usr/bin/zeitgeist-daemon
0 1000 2334 1636 20 0 312640 7652 poll_s SL ? 0:00 /usr/lib/x86_64-linux-gnu/zeitgeis
0 1000 2364 1636 20 0 647848 21604 poll_s SL ? 0:03 /usr/lib/gnome-terminal/gnome-term
0 1000 2371 2364 20 0 16664 3308 wait Ss pts/6 0:00 bash
0 1000 2406 1886 20 0 584452 17148 poll_s SL ? 0:00 update-notifier
0 1000 2437 1886 20 0 585692 7252 poll_s SL ? 0:00 /usr/lib/x86_64-linux-gnu/deja-dup
4 0 2608 1 20 0 266584 4444 - Ssl ? 0:00 /usr/sbin/cups-browsed
0 1000 2765 1636 20 0 355740 5448 poll_s SL ? 0:00 /usr/lib/gvfs/gvfsd-network --spaw
0 1000 2796 1636 20 0 364428 5692 poll_s SL ? 0:00 /usr/lib/gvfs/gvfsd-dnssd --spawne
1 0 2861 2 20 0 0 0 - S ? 0:00 [kworker/0:1]
0 1000 2903 2371 20 0 16760 1936 signal T pts/6 0:00 nano
0 1000 2943 2371 20 0 982964 211400 poll_s SL pts/6 0:21 /usr/lib/firefox/firefox
0 1000 2973 1636 20 0 55868 4516 poll_s S ? 0:00 /usr/lib/x86_64-linux-gnu/gconf/gc
1 0 3157 2 20 0 0 0 - S ? 0:01 [kworker/0:2]
1 0 3194 2 20 0 0 0 - S ? 0:00 [kworker/u2:0]
1 0 3204 2 20 0 0 0 - S ? 0:00 [kworker/u2:2]
1 0 3481 2 20 0 0 0 - S ? 0:00 [kworker/u2:1]
1 0 3482 2 20 0 0 0 - S ? 0:00 [kworker/0:0]
0 1000 3527 2371 20 0 31448 1608 - R+ pts/6 0:00 ps -lax
uadmin@ucient01:~$

```

Ahora ejecutamos **pstree** y comprobamos el nombre del proceso padre

```

graph TD
    update-notifier --> gmain
    update-notifier --> dconf-worker[dconf worker]
    update-notifier --> gdbus
    zeitgeist-datahub --> gmain
    zeitgeist-datahub --> gdbus
    zeitgeist-datahub --> pool
    dconf-worker --> gdbus
    dconf-worker --> gmain
    gnome-terminal --> bash
    gnome-terminal --> firefox
    bash --> Cache-I0[Cache I/O]
    bash --> Cache2-I0[Cache2 I/O]
    bash --> Compositor
    bash --> DNS-Resolver-1[DNS Resolver #1]
    bash --> DNS-Resolver-2[DNS Resolver #2]
    bash --> DNS-Resolver-3[DNS Resolver #3]
    bash --> DNS-Resolver-4[DNS Resolver #4]
    bash --> DNS-Resolver-5[DNS Resolver #5]
    bash --> DNS-Resolver-7[DNS Resolver #7]

```

Realiza el mismo proceso con el programa **nautilus &** ¿qué valores de PID PPID y proceso padre nos resulta?

- Para ver los procesos de un usuario **ps -lau nombreusuario** y para ver los datos de un procesos concreto **ps -aux | nombreproceso**

```

ucient01 (finS4) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminal Terminal Ficheiro Editar Ver Buscar Terminal Axuda
uadmin@ucient01:~$ ps -lax | grep firefox
0 1000 2943 2371 20 0 983284 212524 poll_s SL pts/6 0:24 /usr/lib/firefox/firefox
0 1000 3608 2371 20 0 16760 1020 pipe_w S+ pts/6 0:00 grep --color=auto firefox
uadmin@ucient01:~$

```

5. Vamos a crear el siguiente script llamado **/tmp/bucle**. Básicamente lanza un echo fichero **/dev/null** que es un fichero que borra o descarta continuamente todo lo que llega a él.

```
#!/bin/bash
echo 'nada' >/dev/null
exec /tmp/bucle
```

A continuación lanzamos tres terminales

En la **primera terminal** ejecutamos el comando **top** que nos indica el estado del sistema.

En la **segunda terminal** lanzamos el **script**.

```
top - 15:18:30 up 1:24, 3 users, load average: 1.01, 0.53, 0.28
Tasks: 147 total, 2 running, 140 sleeping, 4 stopped, 1 zombie
Cpu(s): 53.8%us, 46.2%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 508000k total, 495668k used, 12332k free, 28840k buffers
Swap: 522236k total, 0k used, 522236k free, 235328k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 2054 root        20   0  6028   816   692  R   91.6   0.2   1:54.70  bucle
 1961 franchises 20   0  2632  1148   860  R    1.8   0.2   0:05.38  top
```

Si volvemos a la anterior terminal observamos como se está ejecutando el script como un **proceso**.

```
top - 15:18:30 up 1:24, 3 users, load average: 1.01, 0.53, 0.28
Tasks: 147 total, 2 running, 140 sleeping, 4 stopped, 1 zombie
Cpu(s): 53.8%us, 46.2%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 508000k total, 495668k used, 12332k free, 28840k buffers
Swap: 522236k total, 0k used, 522236k free, 235328k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 2054 root        20   0  6028   816   692  R   91.6   0.2   1:54.70  bucle
 1961 franchises 20   0  2632  1148   860  R    1.8   0.2   0:05.38  top
```

Con la combinación **CTRL+Z** para de ejecutarse el bucle y comprobamos los cambios en **top** hasta que el proceso “pasa al estado en espera” (recordad *estado de los procesos* de la Unidad 1)

```
uadmin@uclient01:~$ chmod +x /tmp/bucle.sh
uadmin@uclient01:~$ sh /tmp/bucle.sh
^Z
[4]+ Detido sh /tmp/bucle.sh
uadmin@uclient01:~$
```

```
Terminal
top - 11:36:07 up 2:27, 1 user, load average: 0,30, 0,46, 0,27
Tasks: 191 total, 1 running, 187 sleeping, 3 stopped, 0 zombie
%Cpu(s): 0,7 us, 0,3 sy, 0,0 ni, 99,0 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
KiB Mem : 1016160 total, 75824 free, 634864 used, 305472 buff/cache
KiB Swap: 1046524 total, 827140 free, 219384 used. 201320 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR S %CPU  %MEM    TIME+  COMMAND
 1930 uadmin    20   0 1171836 124404 42696 S   1,0 12,2 11:07.21 compiz
 1353 root      20   0 312824 61676 18700 S   0,3 6,1 1:01.25 Xorg
 3951 uadmin    20   0 36096 3436 2896 R   0,3 0,3 0:00.44 top
   1 root     20   0 185348 4228 2788 S   0,0 0,4 0:01.65 systemd
   2 root     20   0 0 0 0 S   0,0 0,0 0:00.00 kthreadd
   3 root     20   0 0 0 0 S   0,0 0,0 0:00.92 ksoftirqd/0
   5 root     0 -20 0 0 0 S   0,0 0,0 0:00.00 kworker/0:0H
   7 root     20   0 0 0 0 S   0,0 0,0 0:00.63 rcu_sched
   8 root     20   0 0 0 0 S   0,0 0,0 0:00.00 rcu_bh
   9 root     rt    0 0 0 0 S   0,0 0,0 0:00.00 migration/0
```

Ejecutamos la orden **fg**.

Comprobando otra vez el terminal donde se ejecuta **top** contesta:

¿cuál es el resultado de esta orden? ¿cuál es el valor de PID?

Compara el PID con el que tenía antes ¿qué significa?

```
uadmin@uclient01:~$ fg
sh /tmp/bucle.sh
```

6. Cambiamos la velocidad de refresco de **top**. Para ello pulsamos la tecla **d** y modificamos el valor, por ejemplo, a 2 s.

```
top - 11:45:11 up 2:30, 1 user, load average: 1,29, 0,50, 0,57
Tasks: 191 total, 2 running, 187 sleeping, 2 stopped, 0 zombie
%Cpu(s): 46,5 us, 53,0 sy, 0,0 ni, 0,0 id, 0,0 wa, 0,0 hi, 0,5 si, 0,0 st
KiB Mem : 1016160 total, 77480 free, 636444 used, 302236 buff/cache
KiB Swap: 1046524 total, 825748 free, 220776 used. 199856 avail Mem
Change delay from 4,0 to 2
  PID USER      PR  NI  VIRT  RES  SHR S %CPU  %MEM    TIME+  COMMAND
 3920 uadmin    20   0 1356 4 0 R  91,9 0,0 8:35.62 bucle.sh
 1930 uadmin    20   0 1171836 126428 42696 S   4,5 12,4 11:19.07 compiz
 1353 root      20   0 312824 61676 18700 S   1,1 6,1 1:04.04 Xorg
```

También podemos modificar la prioridad de un proceso (comando **nice** en modo terminal). En **top** pulsamos la tecla **r** y le damos el valor por ejemplo 10.

```
KiB Mem : 1016160 total, 77092 free, 639792 used, 299276 buff/cache
KiB Swap: 1046524 total, 824580 free, 221944 used. 196512 avail Mem
PID to renice [default pid = 3920] 10
  PID USER      PR  NI  VIRT  RES  SHR S %CPU  %MEM    TIME+  COMMAND
 3920 uadmin    20   0 5740 540 492 R  95,5 0,1 11:09.56 bucle.sh
 1930 uadmin    20   0 1171836 126424 42696 S   1,5 12,4 11:20.19 compiz
   3 root     20   0 0 0 0 R   0,5 0,0 0:03.56 ksoftirqd/0
 250 root     20   0 45368 2612 2424 S   0,5 0,3 0:00.69 systemd-ude+
 1937 uadmin    20   0 572264 5672 4808 S   0,5 0,6 0:00.55 indicator-p+
```

```

KiB Mem : 1016160 total, 86940 free, 640504 used, 288716 buff/cache
KiB Swap: 1046524 total, 823472 free, 223052 used. 195840 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR  S  %CPU  %MEM     TIME+ COMMAND
 3920 uadmin    39   19    9632     568    504  R  96,0   0,1   14:10.33 bucle.sh
 1930 uadmin    20    0 1171836 126416 42688  S   1,5  12,4   11:22.91 compiz
     3 root      20    0         0         0     0  R   1,0   0,0    0:04.46 ksoftirqd/0
 1017 nobody    20    0   47364     504    488  S   0,5   0,0    0:00.59 dnsmasq

```

¿Cuál es la prioridad ahora?

Si la quiero modificar en modo terminal con otro usuario.

```

Terminal
uadmin@uclient01:~$ sudo su
[sudo] password for uadmin:
root@uclient01:/home/uadmin# cd /tmp
root@uclient01:/tmp# nice -n5 ./bucle.sh

```

Y volvemos a **top**.

```

uclient01 (fnS4) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminal
top - 11:57:11 up 2:48, 1 user, load average: 2,07, 1,64, 1,12
Tasks: 196 total, 5 running, 189 sleeping, 2 stopped, 0 zombie
%Cpu(s): 2,5 us, 52,9 sy, 44,5 ni, 0,0 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
KiB Mem : 1016160 total, 88108 free, 646952 used, 281100 buff/cache
KiB Swap: 1046524 total, 822388 free, 224136 used. 189964 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR  S  %CPU  %MEM     TIME+ COMMAND
 4159 root      25    5   15024     896    788  R  48,5   0,1    1:14.79 bucle.sh
 3920 uadmin    39   19    9632     568    504  R  48,0   0,1   18:43.57 bucle.sh
 1930 uadmin    20    0 1174632 127248 42524  S   2,0  12,5   11:39.65 compiz
     3 root      20    0         0         0     0  R   0,5   0,0    0:06.18 ksoftirqd/0
     1 root      20    0 185348    4212   2772  S   0,0   0,4    0:01.66 systemd
     2 root      20    0         0         0     0  S   0,0   0,0    0:00.00 kthreadd

```

La asignación de la CPU es mayor en el proceso de mayor prioridad.

Podemos cambiar la prioridad otra vez con el comando **renice**.

```

File Machine View Input Devices Help
Terminal
uadmin@uclient01:/tmp$ renice 10 1930
1930 (process ID) old priority 0, new priority 10
uadmin@uclient01:/tmp$

```

7. Otra opción es matar los procesos en **top** para ello escribimos **k**.

```
Tasks: 202 total, 4 running, 191 sleeping, 7 stopped, 0 zombie
%Cpu(s): 0,0 us, 56,6 sy, 43,4 ni, 0,0 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
KiB Mem : 1016160 total, 120856 free, 649332 used, 245972 buff/cache
KiB Swap: 1046524 total, 798092 free, 248432 used. 188020 avail Mem
PID to signal/kill [default pid = 4218]
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
4218 root 39 19 15032 912 800 R 95,4 0,1 71:13.63 bucle.sh
1930 uadmin 30 10 1171428 122260 40728 S 2,3 12,0 34:03.56 compiz
1353 root 20 0 315384 60304 17328 S 0,7 5,9 2:13.69 Xorg
2943 uadmin 20 0 993268 217760 45052 S 0,7 21,4 0:55.14 firefox
3 root 20 0 0 0 0 R 0,3 0,0 0:46.07 ksoftirqd/0
```

Por defecto matar el primer proceso pero podemos modificarlo escribiendo el PID del proceso que nos interese.

En línea de comandos sería con el comando **kill**:

```
Terminal
root@uclient01:/tmp# ./bucle.sh &
[3] 5333
root@uclient01:/tmp# kill -9 5333
root@uclient01:/tmp#
```

8. Para estudiar el uso de CPU y Memoria de los procesos tenemos los comandos **mpstat** y **vmstat** respectivamente. Para ello necesitamos instalar el paquete **sysstat**

```
uadmin@uclient01:/tmp$ sudo apt-get install sysstat
[sudo] password for uadmin:
Sorry, try again.
[sudo] password for uadmin:
Lendo as listas de paquetes... Feito
Construindo a árvore de dependencias
Lendo a informação do estado... Feito
Os seguintes paquetes foron instalados automaticamente e xa non son necesarios:
 linux-headers-4.4.0-21 linux-headers-4.4.0-21-generic linux-image-4.4.0-21-generic
 linux-image-extra-4.4.0-21-generic ubuntu-core-launcher
Use «sudo apt autoremove» para eliminálos.
Paquetes suxeridos:
 isag
Os seguintes paquetes NOVOS hanse instalar:
 sysstat
```

A continuación para que el sistema tengo tareas a realizar y acceso al disco ejecutamos la siguiente orden:

```
find / -type f -exec cat {} > dev/null \
```

Ejecutamos **mpstat** y posteriormente **mvstat**

Antes:

```
root@francesco-virtual-machine:/home/francesco# mpstat
Linux 2.6.38-8-generic (francesco-virtual-machine) 03/12/11 _i686_ (
1 CPU)

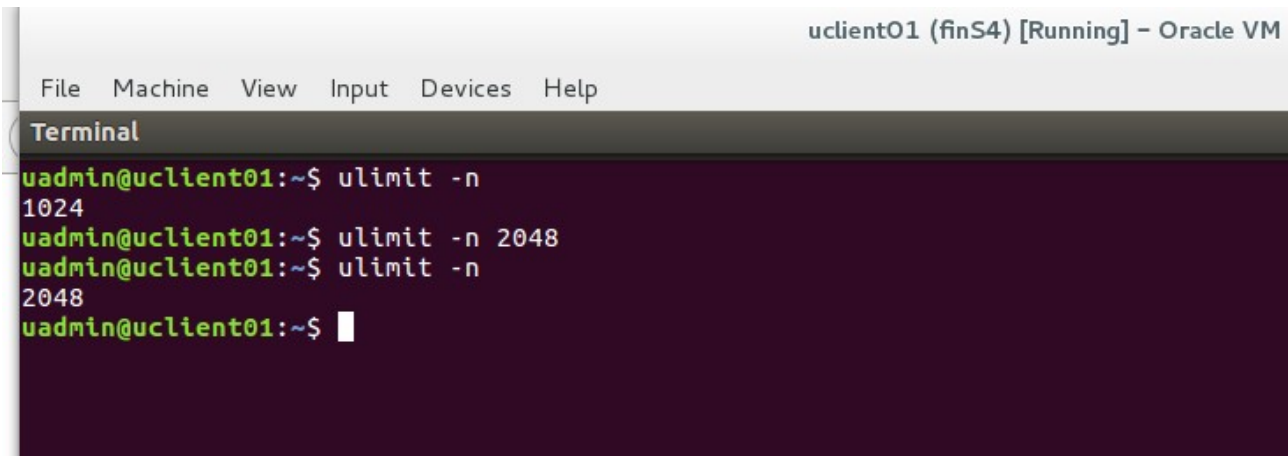
15:48:03 CPU %usr %nice %sys %iowait %irq %soft %steal %guest
%idle
15:48:03 all 7,84 4,68 14,02 9,75 0,00 0,30 0,00 0,00
63,40
root@francesco-virtual-machine:/home/francesco# vmstat
procs -----memory----- --swap-- -----io----- -system-- ----cpu----
r b swpd free buff cache si so bi bo in cs us sy id wa
2 0 516 9172 15452 166956 0 1 1695 212 201 361 13 14 63 10
root@francesco-virtual-machine:/home/francesco#
```

Después:

```
root@francesco-virtual-machine:/home/francesco# mpstat
Linux 2.6.38-8-generic (francesco-virtual-machine) 03/12/11 _i686_ (
1 CPU)

15:48:48 CPU %usr %nice %sys %iowait %irq %soft %steal %guest
%idle
15:48:48 all 7,68 9,22 14,07 9,35 0,00 0,34 0,00 0,00
59,35
root@francesco-virtual-machine:/home/francesco# vmstat
procs -----memory----- --swap-- -----io----- -system-- ----cpu----
r b swpd free buff cache si so bi bo in cs us sy id wa
2 1 1964 6336 14152 162220 0 3 2132 237 214 368 17 14 59 9
root@francesco-virtual-machine:/home/francesco#
```

9. Finalmente está el comando **ulimit** que limita el número de procesos que puede ejecutar un usuario del sistema.



```
uclient01 (finS4) [Running] - Oracle VM
File Machine View Input Devices Help
Terminal
uadmin@uclient01:~$ ulimit -n
1024
uadmin@uclient01:~$ ulimit -n 2048
uadmin@uclient01:~$ ulimit -n
2048
uadmin@uclient01:~$
```

El comando **ulimit -n** nos indica el número máximo de procesos que puede ejecutar un usuario. Con el segundo lo modificamos.

En el fichero **/etc/security/limits.conf** podemos modificarlos permanentemente

