

Práctica 6

Iniciar el snapshot o punto de salvado llamado “**Práctica4v21_SOM**” en la máquina virtual de Ubuntu 20.04.

Buscar información de procesos y servicios en la máquina virtual Ubuntu 20.04.

1. Comando de terminal que muestran información del sistema y de procesos. Ejecutar y realizar cada uno de los comandos de terminal que se muestran en la siguiente dirección:

<https://blog.desdelinux.net/top-htop-nmon-monitores-del-sistema-en-terminal/>

Con el comando top nos muestra la cantidad de usuarios que están activos, tiempo online, también podemos ver la cantidad de tareas o procesos, la cantidad de ellas que están en ejecución, durmiendo, detenidas o en modo zombie

También tenemos un listado de los procesos, de cada uno se puede saber su PID, el usuario que lo ejecuta, cantidad de la memoria ya sea real o virtual, el porcentaje de CPU que esta consumiendo, tiempo de ejecución y el comando/proceso específico

```
top - 10:39:10 up 24 min,  1 user,  load average: 0,00, 0,02, 0,09
Tareas: 166 total,   1 ejecutar, 164 hibernar,   1 detener,   0 zombie
%Cpu(s):  0,3 usuario,  0,3 sist,  0,0 adecuado, 99,3 inact,  0,0 en espera,  0
MiB Mem : 1987,8 total,  334,7 libre,  655,8 usado,  997,3 búfer/caché
MiB Intercambio:  923,3 total,  923,3 libre,    0,0 usado. 1166,4 dispo
```

PID	USUARIO	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	HORA+	ORDEN
1384	alumno	20	0	3663372	336768	116828	S	1,3	16,5	0:08.90	gnome-+
1244	alumno	20	0	221128	55468	36040	S	0,7	2,7	0:02.67	Xorg
1703	alumno	20	0	817884	52084	39400	S	0,3	2,6	0:01.58	gnome-+
2976	alumno	20	0	14720	4100	3448	R	0,3	0,2	0:00.03	top
1	root	20	0	102156	11556	8328	S	0,0	0,6	0:00.84	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.00	kthrea+
3	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	rcu_pa+
6	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworke+
9	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	mm_per+
10	root	20	0	0	0	0	S	0,0	0,0	0:00.12	ksofti+
11	root	20	0	0	0	0	I	0,0	0,0	0:00.10	rcu_sc+
12	root	rt	0	0	0	0	S	0,0	0,0	0:00.01	migrat+
13	root	-51	0	0	0	0	S	0,0	0,0	0:00.00	idle_i+
14	root	20	0	0	0	0	S	0,0	0,0	0:00.00	cpuhp/0
15	root	20	0	0	0	0	S	0,0	0,0	0:00.00	kdevtm+
16	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	netns
17	root	20	0	0	0	0	S	0,0	0,0	0:00.00	rcu_ta+
18	root	20	0	0	0	0	S	0,0	0,0	0:00.00	kauditd
19	root	20	0	0	0	0	S	0,0	0,0	0:00.00	khungtd
20	root	20	0	0	0	0	S	0,0	0,0	0:00.00	oom_re+
21	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	writeb+

El comando htop nos muestra la información más ordenada, es más simple de interpretar. Nos muestra prácticamente lo mismo que el comando anterior, pero nos ofrece otras opciones como buscar procesos determinados en el listado, verlos ordenados como un árbol (procesos y subprocesos o procesos asociados), filtrar por los que más CPU o RAM consumen, matar procesos

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1384	alumno	20	0	3587M	330M	115M	S	0.7	16.6	0:11.11	/usr/bin/gnome
4338	alumno	20	0	13232	3928	3240	R	0.7	0.2	0:00.13	htop
1168	alumno	9	-11	1962M	20204	15724	S	0.7	1.0	0:00.18	/usr/bin/pulse
4342	alumno	20	0	427M	25908	15424	S	0.7	1.3	0:00.07	/usr/libexec/t
1244	alumno	20	0	222M	57364	37044	S	0.0	2.8	0:03.34	/usr/lib/xorg/
1249	alumno	-6	0	1962M	20204	15724	S	0.0	1.0	0:00.09	/usr/bin/pulse
1703	alumno	20	0	798M	52084	39400	S	0.0	2.6	0:02.08	/usr/libexec/g
1262	alumno	20	0	222M	57364	37044	S	0.0	2.8	0:00.43	/usr/lib/xorg/
4349	alumno	39	19	743M	35952	24528	S	0.0	1.8	0:00.07	/usr/libexec/t
4337	alumno	20	0	3587M	330M	115M	S	0.0	16.6	0:00.04	/usr/bin/gnome
1171	alumno	39	19	574M	24856	16808	S	0.0	1.2	0:00.11	/usr/libexec/t
1177	alumno	20	0	14000	10948	3852	S	0.0	0.5	0:00.26	/usr/bin/dbus-
1550	alumno	20	0	1141M	34656	23320	S	0.0	1.7	0:00.01	/usr/libexec/g
1505	alumno	20	0	1141M	34656	23320	S	0.0	1.7	0:00.18	/usr/libexec/g
1	root	20	0	99M	11556	8328	S	0.0	0.6	0:00.88	/sbin/init spl
223	root	19	-1	54012	19256	17440	S	0.0	0.9	0:00.22	/lib/systemd/s
246	root	20	0	23720	7352	4032	S	0.0	0.4	0:00.36	/lib/systemd/s
493	systemd-r	20	0	24052	12036	8036	S	0.0	0.6	0:00.08	/lib/systemd/s
526	systemd-t	20	0	90420	6352	5476	S	0.0	0.3	0:00.00	/lib/systemd/s
494	systemd-t	20	0	90420	6352	5476	S	0.0	0.3	0:00.05	/lib/systemd/s
531	root	20	0	239M	9532	8544	S	0.0	0.5	0:00.04	/usr/lib/accou
635	root	20	0	239M	9532	8544	S	0.0	0.5	0:00.00	/usr/lib/accou

F1 Help F2 Setup F3 Search F4 Filter F5 Tree F6 SortBy F7 Nice - F8 Nice + F9 Kill F10 Quit

El comando nmon nos permite poder personalizar más la información que nos muestra. Nos da información de forma modular ya que si queremos que solo nos muestre información sobre la CPU presionaremos una tecla, si queremos que también nos muestre por ejemplo la RAM presionamos otra tecla (tecla c y m)

```
alumno@Ubuntu20: ~
nmon-16k [H for help] Hostname=Ubuntu20 Refresh= 2secs 10:48.45

-----
          For help type H or ...
          nmon -? - hint
          nmon -h - full details

          To stop nmon type q to Quit

-----

Use these keys to toggle statistics on/off:
  c = CPU          l = CPU Long-term      - = Faster screen updates
  C = " WideView  U = Utilisation          + = Slower screen updates
  m = Memory       V = Virtual memory     j = File Systems
  d = Disks        n = Network            . = only busy disks/procs
  r = Resource     N = NFS                h = more options
  k = Kernel       t = Top-processes      q = Quit
```

Aquí vemos que nos pone las teclas que debemos pulsar según lo que queramos observar, como ejemplo pongo c y m

```
nmon-16k [H for help] Hostname=Ubuntu20 Refresh= 2secs 11:25.53
CPU Utilisation
-----+-----
CPU User% Sys% Wait% Idle|0          |25          |50          |75          |100
  1  0.0   0.0   0.0 100.0|>
-----+-----
```

```
nmon-16k Hostname=Ubuntu20 Refresh= 2secs 11:26.19
Memory and Swap
-----
PageSize:4KB RAM-Memory Swap-Space High-Memory Low-Memory
Total (MB)    1987.8    923.3    - not in use - not in use
Free (MB)     129.7    923.3
Free Percent   6.5%    100.0%
Linux Kernel Internal Memory (MB)
      Cached= 1009.5    Active= 1034.6
Buffers= 53.3 Swapcached= 0.0 Inactive = 641.8
Dirty  = 0.1 Writeback = 0.0 Mapped  = 271.3
Slab   = 130.1 Commit_AS = 3223.2 PageTables= 10.7
```

2. Terminal de Ubuntu 20.04

Utilizar todos los comandos de procesos que se describen en el documento "ControlProcesos.pdf"

- Listar todos los procesos
Utilizamos el comando `ps`, `pstree`

```
alumno@Ubuntu20:~$ ps
```

PID	TTY	TIME	CMD
1711	pts/0	00:00:00	bash
2508	pts/0	00:00:00	top
2976	pts/0	00:00:00	top
4338	pts/0	00:00:00	htop
5346	pts/0	00:00:00	nmon
5738	pts/0	00:00:00	nmon
5825	pts/0	00:00:00	ps

```
alumno@Ubuntu20:~$ pstree
```

```
systemd--ModemManager--2*[{ModemManager}]
systemd--NetworkManager--2*[{NetworkManager}]
systemd--accounts-daemon--2*[{accounts-daemon}]
systemd--acpid
systemd--avahi-daemon--avahi-daemon
systemd--colord--2*[{colord}]
systemd--cron
systemd--cups-browsed--2*[{cups-browsed}]
systemd--cupsd
systemd--dbus-daemon
systemd--gdm3--gdm-session-wor--gdm-x-session--Xorg--{Xorg}
systemd--gdm3--gdm-session-wor--gdm-x-session--gnome-session-b--ssh-agent
systemd--gdm3--gdm-session-wor--gdm-x-session--2*[{gnome+
systemd--gdm3--gdm-session-wor--2*[{gdm-x-session}]
systemd--gdm3--2*[{gdm3}]
systemd--gnome-keyring-d--3*[{gnome-keyring-d}]
systemd--2*[{kerneloops}]
systemd--networkd-dispat
systemd--polkitd--2*[{polkitd}]
systemd--rsyslogd--3*[{rsyslogd}]
systemd--rtkit-daemon--2*[{rtkit-daemon}]
systemd--snapd--8*[{snapd}]
systemd--switcheroo-cont--2*[{switcheroo-cont}]
systemd--systemd--(sd-pam)
systemd--at-spi-bus-laun--dbus-daemon
systemd--at-spi-bus-laun--3*[{at-spi-bus-laun}]
```

- Modificar la prioridad de un proceso.
Utilizamos el comando `nice`, `renice`

```
alumno@Ubuntu20:~$ ps
```

PID	TTY	TIME	CMD
1711	pts/0	00:00:00	bash
2508	pts/0	00:00:00	top
2976	pts/0	00:00:00	top
4338	pts/0	00:00:00	htop
5346	pts/0	00:00:00	nmon
5738	pts/0	00:00:00	nmon
5915	pts/0	00:00:00	ps

```
alumno@Ubuntu20:~$ renice -n 10 -p 1711
1711 (process ID) prioridad anterior 0, nueva prioridad 10
alumno@Ubuntu20:~$
```

- Detener/Borrar procesos

```

root@Ubuntu20:/home/alumno# ps
  PID TTY          TIME CMD
  6126 pts/0        00:00:00 sudo
  6127 pts/0        00:00:00 su
  6128 pts/0        00:00:00 bash
  6135 pts/0        00:00:00 ps
root@Ubuntu20:/home/alumno# kill 6126
root@Ubuntu20:/home/alumno#
Sesión terminada, matando el intérprete de órdenes...exit
...matado.
Terminado

```

- Comando pdf
 - Comando sleep → permite bloquear o suspender la ejecución actual por un intervalo de tiempo determinado

```

root@Ubuntu20:/home/alumno# sleep 5s
root@Ubuntu20:/home/alumno#

```

- Comando strace

```

prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}
) = 0
statfs("/sys/fs/selinux", 0x7fff500950d0) = -1 ENOENT (No existe el archivo o e
l directorio)
statfs("/selinux", 0x7fff500950d0)      = -1 ENOENT (No existe el archivo o el
directorio)
brk(NULL)                               = 0x55e67ca28000
brk(0x55e67ca49000)                     = 0x55e67ca49000
openat(AT_FDCWD, "/proc/filesystems", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0444, st_size=0, ...}) = 0
read(3, "nodev\tsysfs\nnodev\ttmpfs\nnodev\tbd...", 1024) = 360
read(3, "", 1024)                       = 0
close(3)                                = 0
access("/etc/selinux/config", F_OK)      = -1 ENOENT (No existe el archivo o el
directorio)
openat(AT_FDCWD, "/usr/lib/locale/locale-archive", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=8378608, ...}) = 0
mmap(NULL, 8378608, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f4cd1ee6000
close(3)                                = 0
ioctl(1, TCGETS, {B38400 opost isig icanon echo ...}) = 0
ioctl(1, TIOCGWINSZ, {ws_row=29, ws_col=79, ws_xpixel=0, ws_ypixel=0}) = 0
openat(AT_FDCWD, ".", O_RDONLY|O_NONBLOCK|O_CLOEXEC|O_DIRECTORY) = 3
fstat(3, {st_mode=S_IFDIR|0755, st_size=4096, ...}) = 0
getdents64(3, /* 20 entries */, 32768) = 640
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/gconv/gconv-modules.cache", O_RDONLY
) = 4
fstat(4, {st_mode=S_IFREG|0644, st_size=27002, ...}) = 0

```

- Comando jobs → utilizado para listar los procesos que estén ejecutando en segundo plano o en primer plano

```

root@Ubuntu20:/home/alumno# jobs -l
[1]+  6349 Parado                  sleep 2976
root@Ubuntu20:/home/alumno#

```


- Comando pgrep → devuelve el ID de los procesos cuyo nombre coincida con la expresión que se busca

```
root@Ubuntu20:/home/alumno# pgrep -u alumno bash
1711
root@Ubuntu20:/home/alumno#
```

- Comando ulimit → puede establecer o informar el límite de los recursos del usuario actual

```
root@Ubuntu20:/home/alumno# ulimit
unlimited
```

- Comando uptime → podemos ver la hora o el tiempo que lleva en ejecución el sistema

```
root@Ubuntu20:/home/alumno# uptime
12:24:14 up 2:10, 1 user, load average: 0,07, 0,03, 0,01
root@Ubuntu20:/home/alumno#
```

Comando free → nos permite obtener el consumo de memoria.

```
root@Ubuntu20:/home/alumno# free
              total        usado       libre  compartido búfer/caché  disponible
Memoria:    2035504    1060432    163424         20396        811648        79648
Swap:        945416         1548     943868
```

- Comando uname → fin es el de mostrar información del sistema operativo como: la versión del mismo, kernel y detalles del equipo entre otras posibilidades.

```
root@Ubuntu20:/home/alumno# uname -a
Linux Ubuntu20 5.4.0-48-generic #52-Ubuntu SMP Thu Sep 10 10:58:49 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
```