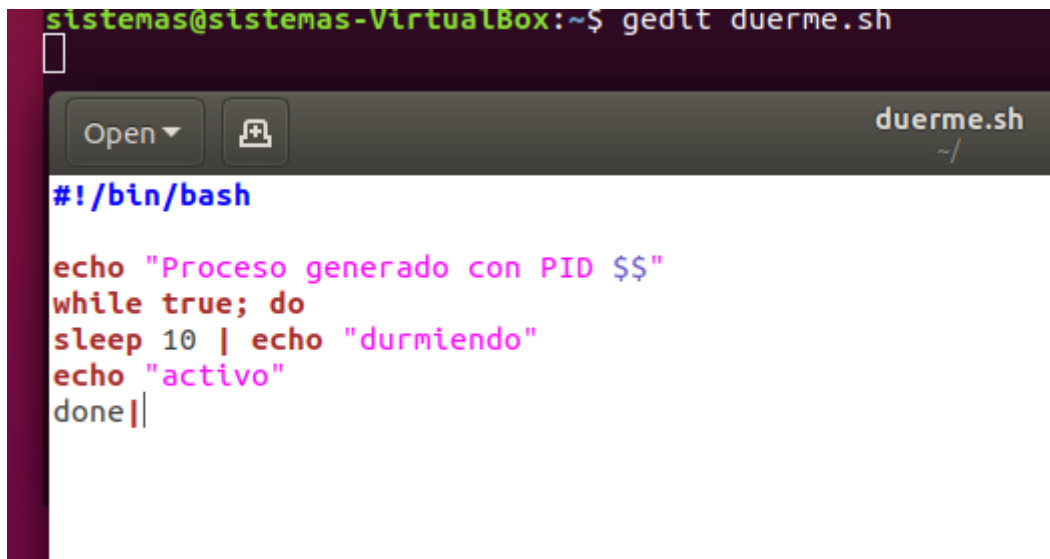


## PRÁCTICA 4.5 – KEVIN ROSALES

### CONTROL DE PROCESOS

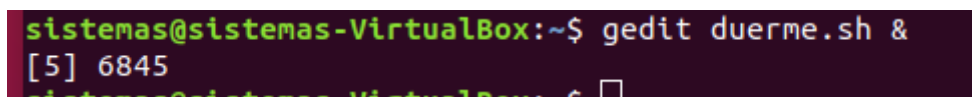
1. Crea con gedit un fichero llamado “duerme.sh” en tu home con el siguiente contenido:

```
#!/bin/bash
echo "Proceso generado con PID $$"
while true; do
sleep 10 | echo "durmiendo"
echo "activo"
done
```



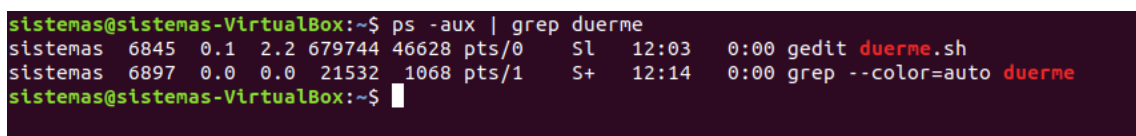
The screenshot shows a terminal window with the prompt `sistemas@sistemas-VirtualBox:~$`. The user has run `gedit duerme.sh`, which has opened a gedit editor window titled `duerme.sh`. The editor shows the script content: `#!/bin/bash`, `echo "Proceso generado con PID $@"`, `while true; do`, `sleep 10 | echo "durmiendo"`, `echo "activo"`, and `done`. The cursor is at the end of the `done` line.

2. Dale permiso de ejecución al fichero duerme ejecutando “`chmod +x duerme.sh`” y ejecuta el fichero duerme desde el terminal en segundo plano.



The screenshot shows a terminal window with the prompt `sistemas@sistemas-VirtualBox:~$`. The user has run `gedit duerme.sh &`, and the terminal shows the response `[5] 6845`, indicating the script is running in the background with PID 6845.

3. Abre una pestaña nueva del terminal y ejecuta el comando “`ps -aux | grep duerme`”. ¿Dónde indica el PID del proceso?



The screenshot shows a terminal window with the prompt `sistemas@sistemas-VirtualBox:~$`. The user has run `ps -aux | grep duerme`, and the output is displayed in two lines:

USER	PID	%CPU	%MEM	VSZ	SSZ	PTS	ST	TIME	COMMAND
sistemas	6845	0.1	2.2	679744	46628	pts/0	Sl	12:03	0:00 gedit <b>duerme.sh</b>
sistemas	6897	0.0	0.0	21532	1068	pts/1	S+	12:14	0:00 grep --color=auto <b>duerme</b>

The prompt is now `sistemas@sistemas-VirtualBox:~$`.

El kernel asigna un número de identificación PID a todos los procesos según se van creando.

4. Ejecuta “*ps tree*” y localiza el programa “*duerme.sh*”, indica quien es su proceso padre y cual es su proceso hijo. ¿Cuál es el proceso padre de todos?

```
sistemas@sistemas-VirtualBox:~$ ps tree
systemd├─ModemManager──2*[{ModemManager}]
      │├─NetworkManager├─dhclient
      ││                └─2*[{NetworkManager}]
      │├─2*[VBoxClient──VBoxClient]
      │├─2*[VBoxClient──VBoxClient──{VBoxClie
      ││    │├─VBoxClient──VBoxClient──2*[{VBoxClie
      ││    │└─VBoxService──7*[{VBoxService}]
      │├─accounts-daemon──2*[{accounts-daemon}]
      │├─acpid
      │├─avahi-daemon──avahi-daemon
      │├─boltd──2*[{boltd}]
      │├─colord──2*[{colord}]
      │├─cron
      │├─cups-browsed──2*[{cups-browsed}]
      │├─cupsd
      │├─dbus-daemon
      │├─fwupd──4*[{fwupd}]
      │└─gdm3├─gdm-session-wor
              └─gdm-wayland-s
```

5. Cambia la prioridad del proceso duerme para que tenga una prioridad (nice) 5. Comprueba la prioridad de duerme con el comando “*ps -lax*”. ¿Ahora este proceso tiene más o menos prioridad para competir por la CPU?

Ahora tiene menos prioridad , ya que a mas alto el numero menor es la prioridad

6. Cambia la prioridad de duerme a -5 y vuelve a comprobar la prioridad con el comando *top*. ¿Por qué has tenido que usar *sudo* para cambiar la prioridad?

```
sistemas@sistemas-VirtualBox:~$ sudo renice -n -5 -p 6845
6845 (process ID) old priority 10, new priority -5
sistemas@sistemas-VirtualBox:~$
```

7. Instala y ejecuta *htop* mostrando los procesos ordenados por prioridad (nice).

No consigo ordenadr por prioridad nice

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPUS	MEM%	TIME+	Command
2220	sistemas	20	0	2985M	294M	62368	S	7.4	14.8	2:14.54	/usr/bin/gnome-shell
2033	sistemas	20	0	463M	102M	48644	S	7.4	5.2	0:43.27	/usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /run/user
7675	sistemas	20	0	34800	5140	3576	R	4.0	0.3	0:04.01	htop
7881	sistemas	20	0	784M	30896	19748	S	2.7	1.5	0:20.41	/usr/lib/gnome-terminal/gnome-terminal-server
2039	sistemas	20	0	463M	102M	48644	S	0.7	5.2	0:08.11	/usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /run/user
1	root	20	0	220M	9080	6336	S	0.0	0.4	0:14.36	/lib/systemd/systemd --system --deserialize 41
2172	sistemas	20	0	123M	1072	704	S	0.0	0.1	0:16.70	/usr/bin/VBoxClient --draganddrop
2174	sistemas	20	0	123M	1072	704	S	0.0	0.1	0:16.70	/usr/bin/VBoxClient --draganddrop
2244	sistemas	20	0	368M	7928	6228	S	0.0	0.4	0:03.52	ibus-daemon --xim --panel disable
2246	sistemas	20	0	368M	7928	6228	S	0.0	0.4	0:02.31	ibus-daemon --xim --panel disable
2772	sistemas	20	0	215M	6020	5176	S	0.0	0.3	0:01.08	/usr/lib/ibus/ibus-engine-simple
2779	sistemas	20	0	215M	6020	5176	S	0.0	0.3	0:00.79	/usr/lib/ibus/ibus-engine-simple
7886	sistemas	20	0	784M	30896	19748	S	0.0	1.5	0:00.53	/usr/lib/gnome-terminal/gnome-terminal-server
497	messagebu	20	0	51888	5628	3352	S	0.0	0.3	0:09.50	/usr/bin/dbus-daemon --system --address=systemd: --
1645	gdm	20	0	513M	15024	9828	S	0.0	0.7	0:00.34	/usr/lib/gnome-settings-daemon/gsd-power
27685	root	19	-1	94840	13656	12944	S	0.0	0.7	0:00.17	/lib/systemd/systemd-journald
27682	systemd-r	20	0	70632	4996	4444	S	0.0	0.2	0:00.15	/lib/systemd/systemd-resolved
18604	root	20	0	34256	3980	2848	S	0.0	0.2	0:00.13	/lib/systemd/systemd-udevd
4717	root	20	0	296M	10832	9404	S	0.0	0.5	0:00.03	/usr/sbin/cups-browsed
4722	root	20	0	296M	10832	9404	S	0.0	0.5	0:00.00	/usr/sbin/cups-browsed
4721	root	20	0	296M	10832	9404	S	0.0	0.5	0:00.00	/usr/sbin/cups-browsed
4716	root	20	0	105M	8172	6884	S	0.0	0.4	0:00.38	/usr/sbin/cupsd -l
2877	root	20	0	554M	20076	6992	S	0.0	1.0	0:01.21	/usr/lib/fwupd/fwupd
2883	root	20	0	554M	20076	6992	S	0.0	1.0	0:00.08	/usr/lib/fwupd/fwupd
2882	root	20	0	554M	20076	6992	S	0.0	1.0	0:00.00	/usr/lib/fwupd/fwupd
2881	root	20	0	554M	20076	6992	S	0.0	1.0	0:00.00	/usr/lib/fwupd/fwupd
2878	root	20	0	554M	20076	6992	S	0.0	1.0	0:00.12	/usr/lib/fwupd/fwupd
2373	sistemas	20	0	496M	6920	5200	S	0.0	0.3	0:00.04	/usr/lib/gnome-settings-daemon/gsd-printer
2396	sistemas	20	0	496M	6920	5200	S	0.0	0.3	0:00.00	/usr/lib/gnome-settings-daemon/gsd-printer
2393	sistemas	20	0	496M	6920	5200	S	0.0	0.3	0:00.00	/usr/lib/gnome-settings-daemon/gsd-printer
2250	sistemas	20	0	347M	13520	8756	S	0.0	0.7	0:00.30	/usr/lib/ibus/ibus-x11 --kill-daemon
2261	sistemas	20	0	347M	13520	8756	S	0.0	0.7	0:00.00	/usr/lib/ibus/ibus-x11 --kill-daemon
2260	sistemas	20	0	347M	13520	8756	S	0.0	0.7	0:00.00	/usr/lib/ibus/ibus-x11 --kill-daemon
2230	sistemas	9	-11	1144M	10752	7492	S	0.0	0.5	0:00.73	/usr/bin/pulseaudio --start --log-target=syslog
2232	sistemas	-6	0	1144M	10752	7492	S	0.0	0.5	0:00.02	/usr/bin/pulseaudio --start --log-target=syslog
2231	sistemas	-6	0	1144M	10752	7492	S	0.0	0.5	0:00.52	/usr/bin/pulseaudio --start --log-target=syslog
2171	sistemas	20	0	59684	316	0	S	0.0	0.0	0:00.00	/usr/bin/VBoxClient --draganddrop
2173	sistemas	20	0	123M	1072	704	S	0.0	0.1	0:00.00	/usr/bin/VBoxClient --draganddrop

8. Mata el proceso “duerme.sh” con la orden kill.

11	SIGSEGV	522	root	20	0	301M	0800	0232	S	0.0	0.3	0:00.38	/usr/lib/accountsserv
12	SIGUSR2	2354	sistemas	20	0	653M	14952	9604	S	0.0	0.7	0:01.11	/usr/lib/gnome-setting
13	SIGPIPE	6845	sistemas	15	-5	663M	46628	30412	S	0.0	2.3	0:01.07	gedit duerme.sh
14	SIGALRM	7886	sistemas	20	0	785M	30896	19748	S	0.0	1.5	0:00.54	/usr/lib/gnome-termina
15	SIGTERM	2877	root	20	0	554M	20076	6992	S	0.0	1.0	0:01.23	/usr/lib/fwupd/fwupd
16	SIGSTKFLT	658	kernoops	20	0	56940	76	0	S	0.0	0.0	0:00.39	/usr/sbin/kerneloops
17	SIGCHLD	656	kernoops	20	0	56940	1792	1688	S	0.0	0.1	0:00.40	/usr/sbin/kerneloops

9. Vuelve a ejecutar “duerme.sh” en segundo plano y para que no se cierre aunque cierres tu sesión. Cierra la sesión de tu usuario, vuelve a iniciar sesión, comprueba si el proceso “duerme.sh” sigue ejecutándose. ¿Donde se almacena la salida del script “duerme.sh”?

Se almacena en /etc

10. Instala “at” y crea una tarea programada para que dentro de 5 minutos a partir de ahora, muestre una notificación del sistema con “notify-send” con el mensaje “Entrega de la práctica ya disponible” y abra automáticamente el navegador “firefox” con la página de Aules. AYUDA: para lanzar una aplicación gráfica hay que hacer antes “export DISPLAY=:0”.

```
sistemas@sistemas-VirtualBox:~$ at now +5 minute
warning: commands will be executed using /bin/sh
at> notify-send"Entrega la practica disponible"
at> export DISPLAY=:0
at> firefox
at> <EOT>
job 4 at Tue May 5 13:24:00 2020
sistemas@sistemas-VirtualBox:~$
```

11. Crea una tarea programada con watch para que cada 5 segundos ejecute “free -h”.

```
sistemas@sistemas-VirtualBox:~$ watch -n 5 free -h
[4]+  Stopped                  watch -n 5 free -h
sistemas@sistemas-VirtualBox:~$
```

12. Instala “rkhunter” y crea una tarea programada con crontab para el usuario “root” para que ejecute “rkhunter -c” de lunes a viernes a las 22:00. Para comprobar que funciona cambia la hora de la tarea programada para dentro de 2 minutos, comprueba si se ha actualizado el fichero “/var/log/rkhunter.log”.