

INGENIERÍA DE SERVIDORES (2016-2017)
GRADO EN INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE GRANADA

Memoria Práctica 3



Iván Rodríguez Millán

8 de diciembre de 2016

Índice

1 ¿Qué archivo le permite ver qué programas se han instalado con el gestor de paquetes? ¿Qué significan las terminaciones .1.gz o .2.gz de los archivos en ese directorio?	5
1.1 Respuesta A):	5
1.2 Respuesta B):	8
2 ¿Qué archivo ha de modificar para programar una tarea? Escriba la línea necesaria para ejecutar una vez al día una copia del directorio home/ivan/codigo a home/ivan/seguridad/\$fecha donde \$fecha es la fecha actual(puede usar el comando date).	8
2.1 Respuesta :	8
3 Pruebe a ejecutar el comando dmesg, conectar un dispositivo USB y vuelva a ejecutar el comando. Copie y pegue la salida del comando. (Considere usar dmesg tail). Comente qué observa en la información mostrada.	9
3.1 Respuesta :	9
4 Ejecute el monitor de “System Performance” y muestre el resultado. Incluya capturas de pantalla comentando la información que aparece.	11
4.1 Respuesta :	11
5 Cree un recopilador de datos definido por el usuario (modo avanzado) que incluya tanto el contador de rendimiento como los datos de seguimiento: Todos los referentes al procesador, al proceso y al servicio web. Intervalo de muestra 15 segundos. Almacene el resultado en el directorio Escritorio/logs. Incluya las capturas de pantalla de cada paso.	16
5.1 Respuesta :	16
6 Visite la web (MUNIN) del proyecto y acceda a la demo que proporcionan donde se muestra cómo monitorizan un servidor. Monitorice varios parámetros y haga capturas de pantalla de lo que está mostrando comentando qué observa.	20
6.1 Respuesta :	20
7 Escriba un breve resumen sobre alguno de los artículos donde se muestra el uso de strace o busque otro y coméntelo.	22
7.1 Respuesta :	22
8 Escriba un script en Python o PHP y analice su comportamiento usando el profiler presentado.	25
8.1 Respuesta :	25

9 Acceda a la consola mysql (o a través de phpMyAdmin) y muestre el resultado de mostrar el “profile” de una consulta (la creación de la BD y la consulta la puede hacer libremente).	27
9.1 Respuesta :	27

Índice de figuras

1.1. Comando para acceder de forma cómo al fichero con el historial del gestor APT.	5
1.2. Ejemplo del fichero history.log, mostrando información sobre la instalación de gedit.	6
1.3. Comando para acceder de forma cómo al fichero con el historial del gestor YUM.	7
1.4. Ejemplo del fichero yum.log	7
2.1. Fichero crontab	8
3.1. Aplicamos el comando dmesg tail, para ver las diferencias de tener un USB insertado a no tenerlo.	9
3.2. Aplicamos el comando dmesg grep -i USB.	10
3.3. Continuación del comando dmesg grep -i USB.	10
4.1. Monitorización en tiempo real del porcentaje de procesador usado.	11
4.2. Monitorización en tiempo real del porcentaje de procesador usado, y memoria.	12
4.3. Datos recopilados del diagnóstico realizado mediante el system performance.	13
4.4. Desglose de información con respecto al procesador(CPU).	13
4.5. Gráfica de los datos recopilados.	14
4.6. Datos sobre la memoria recopilados.	15
4.7. Datos sobre los procesos recopilados.	15
4.8. Datos sobre el procesador recopilados.	16
5.1. Inicio creación de un recopilador de datos.	17
5.2. Indicando performance counter alert(alerta de contador del rendimiento). .	17
5.3. Añadiendo los contadores de seguimiento , proceso, procesador y servicio web.	18
5.4. Cambiamos el directorio donde se guardarán los resultados.	18
5.5. Indicamos que los intervalos de muestras sean cada 15 segundo.	19
6.1. Monitorización de la utilización de disco de la última semana.	20
6.2. Monitorización del uso de la CPU de la última semana.	21
6.3. Monitorización de los cambios de contexto e interrupciones producidos en la última semana.	21
6.4. Monitorización del tráfico de bytes por segundo a través de las diferentes interfaces de red en el último año.	22
7.1. Ejecución del comando strace teniendo permisos de lectura sobre el fichero de prueba.	23

7.2. Ejecución del comando strace sin tener permisos de lectura sobre el fichero de prueba.	24
8.1. Analizando el comportamiento de un script recursivo realizado en Python con el profiler cProfiler.	25
8.2. Analizando el comportamiento de un script iterativo realizado en Python con el profiler cProfiler.	26
9.1. Acceso a mysql y creación de una nueva base de datos.	27
9.2. Creación de una tabla de prueba e inserción de datos.	28
9.3. Verificación del número perteneciente a la consulta realizada, para mostrar el profile.	28
9.4. Mostrando el profile de una consulta realizada a la base de datos creada como ejemplo.	29

Índice de tablas

1. **¿Qué archivo le permite ver qué programas se han instalado con el gestor de paquetes? ¿Qué significan las terminaciones .1.gz o .2.gz de los archivos en ese directorio?**

- 1.1. **Respuesta A):**

En el caso de Ubuntu Server, en donde usamos el gestor de paquetes apt, el archivo en donde se puede ver el historial de los programas que se han instalado es “/var/log/apt/history.log”. [1]

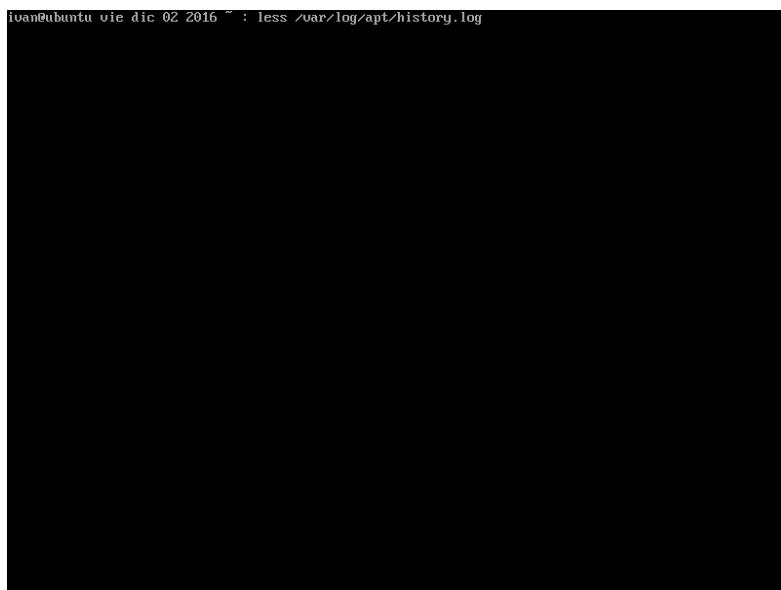


Figura 1.1: Comando para acceder de forma cómo al fichero con el historial del gestor APT.

```

Start-Date: 2016-11-08 11:07:49
CommandLine: apt install gedit
Install: zenity:amd64 (3.8.0-1ubuntu1, automatic), gir1.2-atk-1.0:amd64 (2.10.0-2ubuntu2, automatic)
, libhunspell-1.3-0:amd64 (1.3.2-6ubuntu2.1, automatic), python-dbus:amd64 (1.2.0-2build2, automatic
), libgush2:amd64 (0.1.6-5, automatic), libjack-jackd2-0:amd64 (1.9.9.5-20130622git7de15e7a-1ubuntu1
, automatic), libgeoclue0:amd64 (0.12.99-3ubuntu1, automatic), libxcb-driz2-0:amd64 (1.10-2ubuntu1, a
utomatic), liblcms2-2:amd64 (2.5-0ubuntu4.1, automatic), libvorbisfile3:amd64 (1.3.2-1.3ubuntu1, aut
omatic), libpangoftf-1.0-0:amd64 (1.36.3-1ubuntu1.1, automatic), glib-networking-common:amd64 (2.40
.0-1ubuntu0.1, automatic), libfixes3:amd64 (5.0.1-1ubuntu1.1, automatic), libatk1.0-0:amd64 (2.10.0-
2ubuntu2, automatic), libxft2:amd64 (2.3.1-2, automatic), libjpeg-turbo8:amd64 (1.3.0-2ubuntu2, auto
matic), libasound2-data:amd64 (1.0.27.2-3ubuntu7, automatic), libpcaccess0:amd64 (0.13.2-1, automati
c), zeitgeist-core:amd64 (0.9.14-2ubuntu4.1, automatic), aspell:amd64 (0.60.7-20110707-1ubuntu1, au
tomatic), libgraphite2-3:amd64 (1.3.6-1ubuntu0.14.04.1, automatic), libsane-common:amd64 (1.0.23-3ub
untu3.1, automatic), libaspell115:amd64 (0.60.7-20110707-1ubuntu1, automatic), libice6:amd64 (1.0.8-2
, automatic), libltdl7:amd64 (2.4.2-1.7ubuntu1, automatic), libsoup2.4-1:amd64 (2.44.2-2ubuntu2.1, a
utomatic), libpeas-common:amd64 (1.8.1-2ubuntu2, automatic), libx11-xcb1:amd64 (1.6.2-1ubuntu2, auto
matic), libgstreamer-plugins-base1.0-0:amd64 (1.2.4-1ubuntu2, automatic), libxi6:amd64 (1.7.1.901-1
ubuntu1.1, automatic), at-sp1-core:amd64 (2.10.2.1s.2.10.1-1ubuntu1, automatic), libicu52:amd64 (52
.1-3ubuntu0.4, automatic), gstreamer1.0-plugins-base:amd64 (1.2.4-1ubuntu2, automatic), libcanberra
0:amd64 (0.30-1ubuntu3, automatic), libpango-1.0-0:amd64 (1.36.3-1ubuntu1.1, automatic), libdrm-inte
grate1:amd64 (2.4.67-1ubuntu0.14.04.1, automatic), libharfbuzz-icu0:amd64 (0.9.27-1ubuntu1.1, automatic
), libxcursor1:amd64 (1.1.14-1, automatic), yelp-xs1:amd64 (3.10.1-1, automatic), libjavascriptcorejk
-3.0-0:amd64 (2.4.10-1ubuntu0.14.04.1, automatic), gir1.2-pango-1.0:amd64 (1.36.3-1ubuntu1.1, autom
atic), libuebp5:amd64 (0.4.0-4, automatic), gedit-common:amd64 (3.10.4-2ubuntu4, automatic), gir1.2-
gtk-3.0:amd64 (3.10.8-1ubuntu1.6, automatic), libblu3m3.4:amd64 (3.4-1ubuntu3, automatic), libgdk-pix
buf2.0-common:amd64 (2.30.7-1ubuntu1.6, automatic), libgl1-nesa-dri:amd64 (10.1.3-1ubuntu0.6, automa
tic), libfontconfig1:amd64 (2.11.0-0ubuntu4.2, automatic), libsm6:amd64 (1.2.1-2, automatic), gir1.2-
peas-1.0:amd64 (1.8.1-2ubuntu2, automatic), gstreamer1.0-x:amd64 (1.2.4-1ubuntu2, automatic), libc
airohung1:amd64 (1.0.6-1, automatic), libcup2:amd64 (1.7.2-2ubuntu1.7, automatic), libcdparanoia0:am
d64 (3.10.2+debian-1, automatic), libtiff5:amd64 (4.0.3-7ubuntu0.4, automatic), libtagic2a:amd64 (1
.9.1-2, automatic), libxkbcommon0:amd64 (0.4.1-1ubuntu1, automatic), libdrm-radeon1:amd64 (2.4.67-1u
buntu0.14.04.1, automatic), libgudev1-0.0:amd64 (204-5ubuntu20.19, automatic), libvisual-0.4-plugin
s:amd64 (0.4.0.dsfg.1-7build1, automatic), libiecii1284-3:amd64 (0.2.11-12, automatic), libjson-glib-
1.0-common:amd64 (0.16.2-1ubuntu1, automatic), libpangocairo-1.0-0:amd64 (1.36.3-1ubuntu1.1, automati
c), fontconfig:amd64 (2.11.0-1ubuntu4.2, automatic), libxtst6:amd64 (1.2.2-1, automatic), notificat
:

```

Figura 1.2: Ejemplo del fichero history.log, mostrando información sobre la instalación de gedit.

Mientras que en el caso de CentOS 7, al usar el gestor de paquetes yum, el archivo es “/var/log/yum.log”. [2]

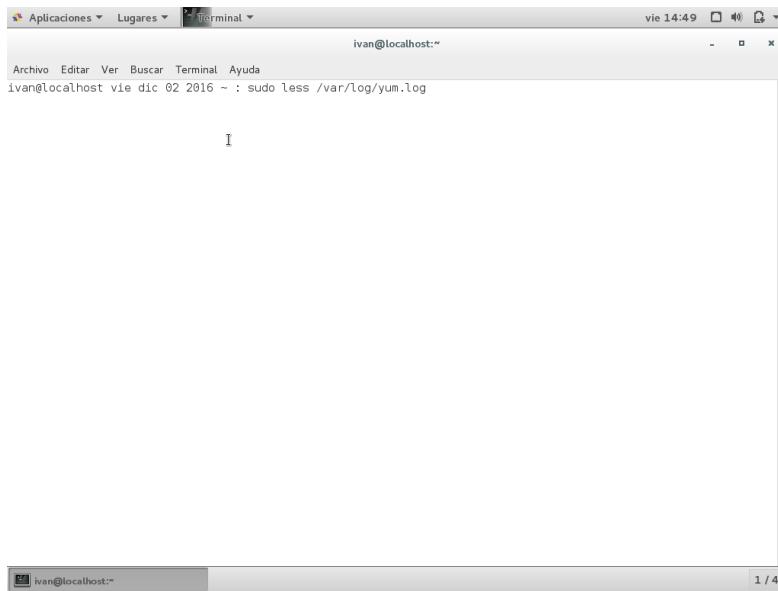


Figura 1.3: Comando para acceder de forma cómo al fichero con el historial del gestor YUM.

```
Aplicaciones ▾ Lugares ▾ Terminal ▾ ivan@localhost:~ vie 14:50 □ 40 □ x
ivan@localhost:~ Archivo Editar Ver Buscar Terminal Ayuda
ivan@localhost vie dic 02 2016 ~ : sudo less /var/log/yum.log
1 / 4
```

```
Nov 03 12:56:19 Installed: perl-HTTP-Message-6.06-6.el7.noarch
Nov 03 12:56:19 Installed: perl-HTTP-Cookies-6.01-5.el7.noarch
Nov 03 12:56:19 Installed: perl-HTML-PgServer-3.71-4.el7.x86_64
Nov 03 12:56:19 Installed: perl-HTTP-Daemon-6.01-5.el7.noarch
Nov 03 12:56:19 Installed: perl-HTTP-Negotiate-6.01-5.el7.noarch
Nov 03 12:56:19 Installed: perl-File-Listing-6.04-7.el7.noarch
Nov 03 12:56:20 Installed: sgml-common-0.6.3-39.el7.noarch
Nov 03 12:56:20 Installed: dobook-dtds-1.0.60.el7.noarch
Nov 03 12:56:22 Installed: dobook-style-xsl-1.78.1-3.el7.noarch
Nov 03 12:56:23 Installed: perl-Net-LIBIDN-0.12-15.el7.x86_64
Nov 03 12:56:23 Installed: perl-IO-Socket-SSL-1.94-3.el7.noarch
Nov 03 12:56:23 Installed: perl-Net-HTTP-6.06-2.el7.noarch
Nov 03 12:56:23 Installed: perl-libwww-perl-6.05-2.el7.noarch
Nov 03 12:56:23 Installed: icoutils-0.3.0-3.el7.x86_64
Nov 03 12:56:23 Installed: libmng-1.0.10-4.el7.x86_64
Nov 03 12:56:26 Installed: libcurl-7.14.0-5.el7.x86_64
Nov 03 12:56:26 Installed: phonon-4.6.0-10.el7.x86_64
Nov 03 12:56:26 Installed: 2:phonon-backend-gstreamer-4.6.3-3.el7.x86_64
Nov 03 12:56:26 Installed: compat-poppler-022-qt-0.22.5-4.el7.x86_64
Nov 03 12:56:26 Installed: dbusmenu-qt-0.9.2-7.el7.x86_64
Nov 03 12:56:26 Installed: qimageblitz-0.0.6-7.el7.x86_64
Nov 03 12:56:26 Installed: gamln-0.1.10-16.el7.x86_64
Nov 03 12:56:26 Installed: 6:kdelibs-ktexteditor-4.14.8-5.el7_2.x86_64
Nov 03 12:56:30 Installed: 6:kdelibs-4.14.8-5.el7_2.x86_64
Nov 03 12:56:30 Installed: kaktivites-4.10.5-3.el7.x86_64
Nov 03 12:56:31 Installed: nepomuk-core-4.10.5-5.el7.x86_64
Nov 03 12:56:31 Installed: nepomuk-core-libs-4.10.5-5.el7.x86_64
Nov 03 12:56:31 Installed: kdepinilbs-kxmlrpcclient-4.10.5-4.el7.x86_64
Nov 03 12:56:32 Installed: kdepinilbs-4.10.5-4.el7.x86_64
Nov 03 12:56:32 Installed: kactivities-4.10.5-3.el7.x86_64
Nov 03 12:56:32 Installed: kde-runtime-drkongi-4.10.5-6.el7.x86_64
Nov 03 12:56:34 Installed: kde-runtime-4.10.5-8.el7.x86_64
Nov 03 12:56:34 Installed: kde-runtime-libs-4.10.5-8.el7.x86_64
:
```

Figura 1.4: Ejemplo del fichero yum.log

1.2. Respuesta B):

Dado que el sistema puede llegar a generar ficheros log demasiado extensos, y por lo tanto crear serios inconvenientes en el almacenamiento de nuestro servidor, existe un demonio cron (cron.daily)[3] , que todos los días ejecuta un script (logrotate) que permite automatizar la rotación, compresión, borrado y envío de ficheros log. [4]

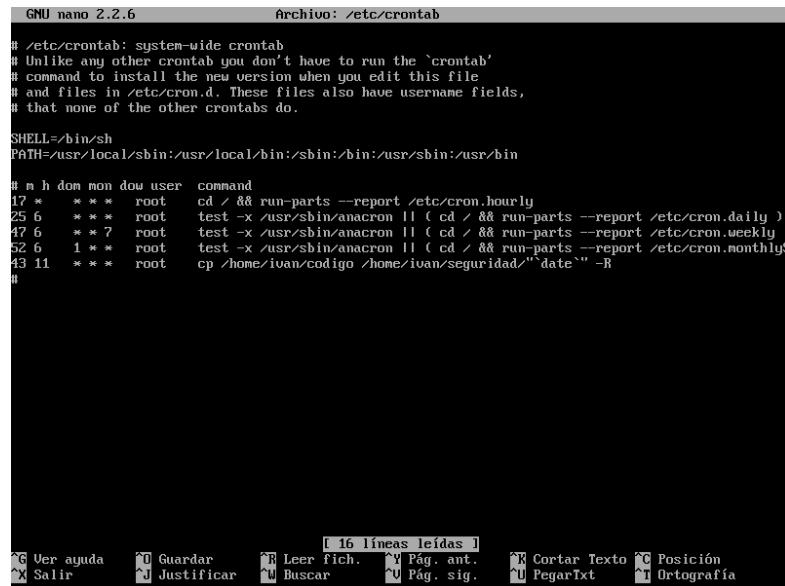
Así pues respondiendo a la pregunta el script “logrotate” los comprime con la extensión “nº.gz” , así cuanto mayor sea el número más viejo será el archivo.[5]

2. ¿Qué archivo ha de modificar para programar una tarea? Escriba la línea necesaria para ejecutar una vez al día una copia del directorio home/ivan/codigo a home/ivan/seguridad/\$fecha donde \$fecha es la fecha actual(puede usar el comando date).

2.1. Respuesta :

Si se quiere programar una tarea se tienen dos opciones, o bien modificar el archivo /etc/crontab ó escribir archivos crontab adicionales en el directorio /etc/cron.d . [6]

A continuación se muestra un ejemplo de como realizar una copia de seguridad de un directorio codigo a otro directorio seguridad y dentro de seguridad ir guardando todos los días por fecha de copia.



```
GNU nano 2.2.6          Archivo: /etc/crontab

# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *      * * *    root    cd / && run-parts --report /etc/cron.hourly
25 6      * * *    root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6      * * 7    root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6      1 * * *    root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
43 11     * * *    root    cp /home/ivan/codigo /home/ivan/seguridad/"`date`"-R
#



[ 16 líneas leídas ]
```

GNU nano 2.2.6 Archivo: /etc/crontab

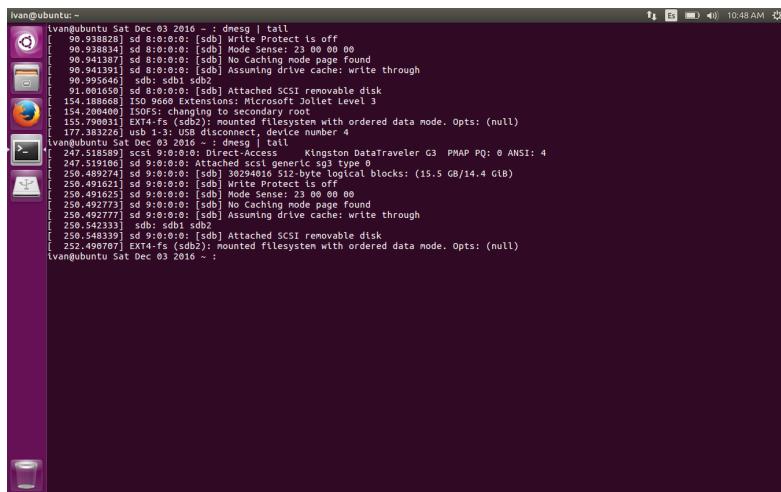
^G Ver ayuda ^O Guardar ^R Leer fich. ^Y Pág. ant. ^K Cortar Texto ^C Posición
^X Salir ^J Justificar ^U Buscar ^V Pág. sig. ^U PegarTxt ^T Ortografía

Figura 2.1: Fichero crontab

- 3. Pruebe a ejecutar el comando dmesg, conectar un dispositivo USB y vuelva a ejecutar el comando. Copie y pegue la salida del comando. (Considere usar dmesg | tail). Comente qué observa en la información mostrada.**

3.1. Respuesta :

El comando dmesg [7] es usado para examinar las señales del kernel. En nuestro caso vamos a comprobar el funcionamiento del comando viendo las diferencias de no tener insertado un USB a si tenerlo insertado.



```
ivan@ubuntu:~$ dmesg | tail
[ 98.938826] sd 8:0:0:0: [sdb] Write Protect is off
[ 98.938834] sd 8:0:0:0: [sdb] Mode Sense: 23 00 00 00
[ 98.941387] sd 8:0:0:0: [sdb] No Caching mode page found
98.941391] sd 8:0:0:0: [sdb] Assuming drive cache: write through
[ 98.941391] sd 8:0:0:0: [sdb] Attached SCSI removable disk
91.600159] sd 8:0:0:0: [sdb] Attached SCSI removable disk
154.188668] ISO 9660 Extensions Microsoft Jollet Level 3
154.208440] ISO9660 changing secondary root
154.208440] ISO 9660: /dev/sdb1 mounted with ordered data mode. Opts: (null)
177.383226] usb 1:3: USB disconnect, device number 4
ivan@ubuntu:~$ dmesg | tail
247.516113] scsi 0:0:0:0: Direct-Access Kingston DataTraveler G3 PQ: 0 ANSI: 4
247.516113] sd 0:0:0:0: [sdb] Attached SCSI generic sg3 type 0
247.516113] sd 0:0:0:0: [sdb] 30294016 512-byte logical blocks: (15.5 GB/14.4 GiB)
250.492974] sd 9:0:0:0: [sdb] 512-byte logical blocks: (15.5 GB/14.4 GiB)
250.491621] sd 9:0:0:0: [sdb] Write Protect is off
250.491621] sd 9:0:0:0: [sdb] Mode Sense: 23 00 00 00
[ 250.492773] sd 9:0:0:0: [sdb] No Caching mode page found
[ 250.492773] sd 9:0:0:0: [sdb] Assuming drive cache: write through
[ 250.492333] sdb: sdb1 sdb2
[ 250.492333] sd 9:0:0:0: [sdb] Attached SCSI removable disk
[ 252.499787] EXT4-fs (sdb2) mounted filesystem with ordered data mode. Opts: (null)
ivan@ubuntu:~$
```

Figura 3.1: Aplicamos el comando dmesg | tail, para ver las diferencias de tener un USB insertado a no tenerlo.

En la imagen de arriba se aplica el comando en primer lugar sin el USB insertado, y en segundo lugar con el USB insertado. Se puede apreciar como nos muestra la marca del USB en este caso Kingston. Nos muestra también el controlador SCSI(Small computer system interface) Más abajo nos muestra también la memoria que posee el dis, positivo en Gigas. También nos indica el sistema de archivos que tiene el dispositivo (EXT4). También tenemos otro comando como dmesg | grep -i USB, para mostrar información de los dispositivos USB entre otros. En donde por ejemplo se nos muestra el numero de serie del dispositivo, marca del dispositivo, tipo de producto (DataTraveler G3). A parte también se nos muestra información por ejemplo de la CAM(FaceTime HD Camera).

```

[ivan@ubuntu sat Dec 03 2016 ~] dmesg | grep -i usb
0.0648802] ACPI: bus type USB registered
0.0648860] usbf: registered new interface driver usbf
0.0648870] usbc: registered new interface driver usbc
0.0648848] usbc: registered new device driver usb
0.065764] ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
0.065952] ehci_hcd 0000:00:1d.0: EHCI bus registered, assigned bus number 1
0.065952] ehci_hcd 0000:00:1d.0: EHCI 1.00
0.074426] usb usb1: New USB device found, idVendor=1d0b, idProduct=0002
0.074438] usb usb1: New USB device strings: Mfr=3, Product=2, SerialNumber=1
0.074439] usb usb1: Manufacturer: Linux 4.4.0-31-generic ehci_hcd
0.074442] usb usb1: SerialNumber: 0000:00:1d.7
0.074522] hub 1-0:1.0: USB hub found
0.074522] hub 1-0:1.0: 1 ports available
0.074598] ehci_hcd: Universal Host Controller Interface driver
0.075016] ehci_hcd 0000:00:1d.0: new USB bus registered, assigned bus number 2
0.075246] usb usb2: New USB device found, idVendor=1d0b, idProduct=0001
0.075246] usb usb2: New USB device strings: Mfr=3, Product=2, SerialNumber=1
0.075254] usb usb2: Product: UHCI Host Controller
0.075256] usb usb2: Manufacturer: Linux 4.4.0-31-generic ehci_hcd
0.075268] usb usb2: SerialNumber: 0000:00:1d.0
0.075268] hub 1-0:1.0: USB hub found
0.075478] ehci_hcd 0000:00:1d.0: new USB bus registered, assigned bus number 3
0.075658] usb usb3: New USB device found, idVendor=1d0b, idProduct=0002
0.075662] usb usb3: New USB device strings: Mfr=3, Product=2, SerialNumber=1
0.075662] usb usb3: Product: UHCI Host Controller
0.075678] usb usb3: Manufacturer: Linux 4.4.0-31-generic ehci_hcd
0.075678] usb usb3: SerialNumber: 0000:00:1d.6
0.075678] hub 1-0:1.0: USB hub found
0.075678] ehci_hcd 0000:00:1d.0: new USB bus registered, assigned bus number 4
0.075992] usb usb4: We don't know the algorithms for LPM for this host, disabling LPM.
0.076136] usb usb4: New USB device found, idVendor=1d0b, idProduct=0003
0.076144] usb usb4: New USB device strings: Mfr=3, Product=2, SerialNumber=1
0.076144] usb usb4: Product: UHCI Host Controller
0.076148] usb usb4: Manufacturer: Linux 4.4.0-31-generic ehci_hcd
0.076150] usb usb4: SerialNumber: 0000:00:1d.6
0.076150] hub 1-0:1.0: USB hub found
0.076227] usb 1-1: new high-speed USB device number 2 using ehci-pci
1.318830] usb 1-1: New USB device found, idVendor=203a, idProduct=ffff
1.318832] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
1.318832] usb 1-1: Product: FaceTime HD Camera
1.318834] usb 1-1: New USB device found, idVendor=203a, idProduct=ffff
1.318834] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
1.318834] usb 1-1: Product: Parallels
1.318836] usb 1-1: SerialNumber: D3H54667MZBG1HPAA
4.920497] uscore: registered new interface driver uvcvideo

```

Figura 3.2: Aplicamos el comando dmesg | grep -i USB.

```

[ivan@ubuntu:~]
0.876136] usb usb4: New USB device found, idVendor=1d0b, idProduct=0003
0.876140] usb usb4: New USB device strings: Mfr=3, Product=2, SerialNumber=1
0.876144] usb usb4: Product: UHCI Host Controller
0.876148] usb usb4: Manufacturer: Linux 4.4.0-31-generic ehci-hcd
0.876150] usb usb4: SerialNumber: 0000:00:1d.6
0.876226] hub 4-0:1.0: USB hub found
0.876226] hub 4-0:1.0: 1 ports available
1.138830] usb 1-1: New high-speed USB device number 2 using ehci-pci
1.138830] usb 1-1: New USB device found, idVendor=203a, idProduct=ffff
1.318832] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
1.318832] usb 1-1: Product: FaceTime HD Camera
1.318834] usb 1-1: New USB device found, idVendor=203a, idProduct=ffff
1.318834] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
1.318834] usb 1-1: Product: Parallels
1.318836] usb 1-1: SerialNumber: D3H54667MZBG1HPAA
4.920497] uscore: registered new interface driver uvcvideo
4.920501] usb Vl805: Class driver v1.1
4.920501] usb Vl805: New USB device found, idVendor=1d0b, idProduct=0003
4.920501] usb Vl805: New USB device number 3 using ehci-pci
0.466632] usb 1-2: New USB device found, idVendor=203a, idProduct=ffff
0.466636] usb 1-2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
0.466638] usb 1-2: Product: Virtual Printer (Print to PDF (Mac Desktop))
0.466640] usb 1-2: Manufacturer: Parallels
0.466641] usb 1-2: SerialNumber: TAG11087acaa0
6.491042] usblp 1-2:1.0: usb1p0 USB Unidirectional printer dev 3 if 0 alt 0 proto 1 vid 0x203a pid 0xFFFFA
6.491042] usblp 1-2:1.0: registered new interface driver usb1p0
11.604976] usblp 1-2:1.0: usb1p0 USB Unidirectional printer dev 3 if 0 alt 0 proto 1 vid 0x203a pid 0xFFFFA
86.478916] usb 1-3: new high-speed USB device number 4 using ehci-pci
86.478916] usb 1-3: New USB device found, idVendor=0951, idProduct=1643
86.478916] usb 1-3: New USB device number 5 using ehci-pci
86.478916] usb 1-3: Product: Kingston DataTraveler G3
86.616089] usb 1-3: Manufacturer: Kingston DataTraveler G3
86.616091] usb 1-3: SerialNumber: 00000000000000000000000000000000
86.631752] usb-storage 1-3:1.0: usb-storage device detected
86.632317] scsi host8: usb-storage 1-3:1.0
86.632402] usb-core: registered new interface driver usb-storage
86.632402] usb-core: New USB device found, idVendor=1d0b, idProduct=0003
177.383226] usb 1-3: USB disconnect, device number 4
246.393652] usb 1-3: new high-speed USB device number 5 using ehci-pci
246.441159] usb 1-3: New USB device found, idVendor=0951, idProduct=1643
246.441163] usb 1-3: New USB device strings: Mfr=1, Product=2, SerialNumber=3
246.441163] usb 1-3: Product: DataTraveler G3
246.441165] usb 1-3: Manufacturer: Kingston DataTraveler G3
246.441165] usb 1-3: SerialNumber: 00000000000000000000000000000000
246.447188] scsi host9: usb-storage 1-3:1.0
246.447188] usb-storage 1-3:1.0: usb-storage device detected
246.447188] scsi host9: usb-storage 1-3:1.0
291.164747] usb 1-3: USB disconnect, device number 5

```

Figura 3.3: Continuación del comando dmesg | grep -i USB.

- 4. Ejecute el monitor de “System Performance” y muestre el resultado. Incluya capturas de pantalla comentando la información que aparece.**

4.1. Respuesta :

En primer lugar entramos al monitor Perform, y una vez dentro y en la pestaña de herramientas de monitorización, nos encontramos con el performance monitor, que básicamente consiste en un monitor en tiempo real. Lo primero que nos encontramos es que se realiza una monitorización del porcentaje del procesador.

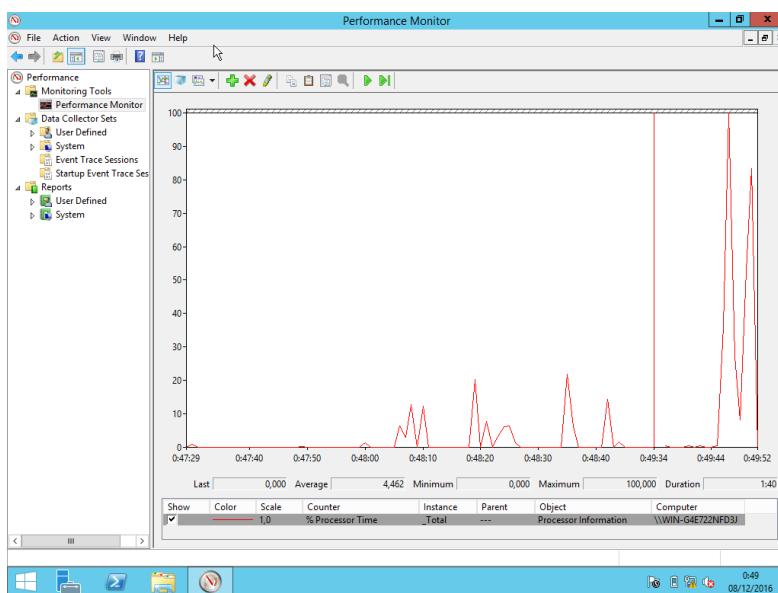


Figura 4.1: Monitorización en tiempo real del porcentaje de procesador usado.

A esto le podemos añadir tantos datos como queramos, para ello simplemente debemos irnos a la cruz verde, y añadir los datos a medir. Probamos a introducir datos referentes a la memoria.

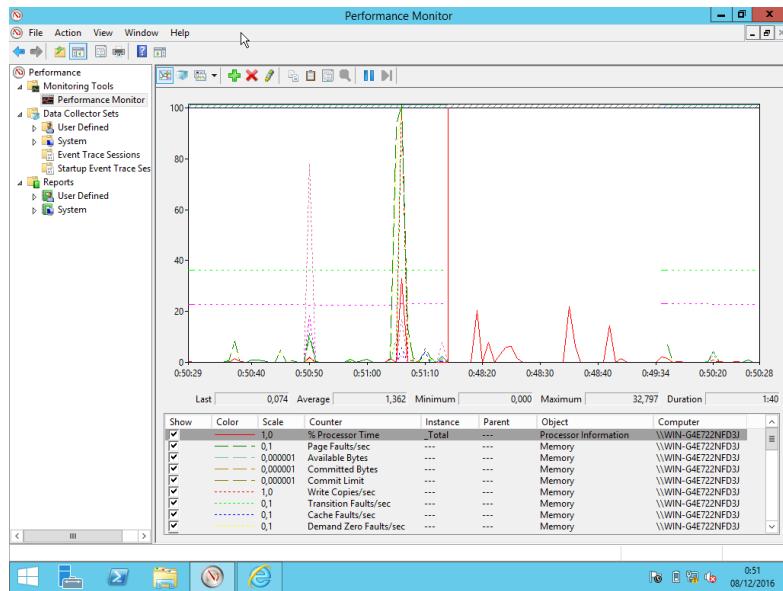


Figura 4.2: Monitorización en tiempo real del porcentaje de procesador usado, y memoria.

Y comprobamos, que por ejemplo se produce un pico (Línea discontinua en verde), este pico se produce justo cuando abrimos el explorador, hace referencia a una falta de página.

Junto con el pico de la falta de página, también se percibe aunque en menor medida, una pequeña subida en el porcentaje de uso del procesador (como es normal al iniciarse cualquier proceso en el sistema).

A continuación se muestra un ejemplo de ejecución de un recopilador de datos.

En nuestro caso lanzamos el recopilador de datos generado por el sistema (System Performance). Tras esperar 60 segundos, nos muestra los datos recopilados, que en nuestro caso son los siguientes:

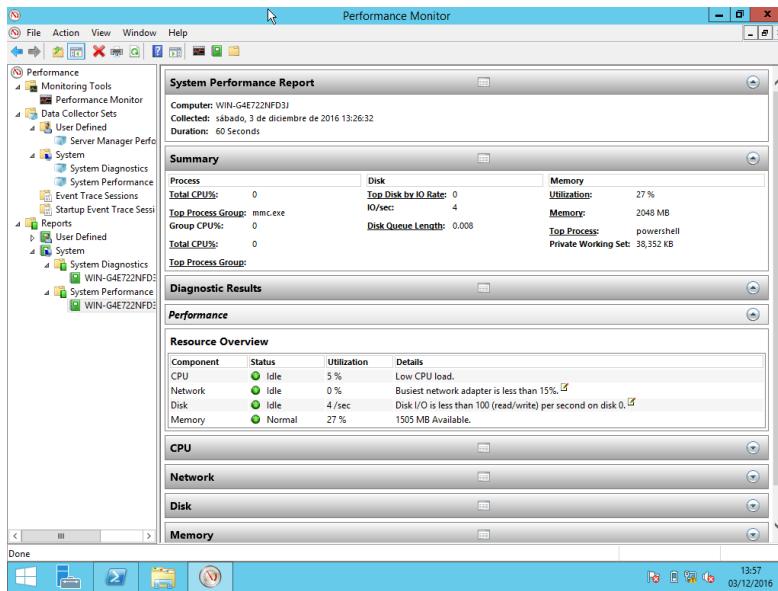


Figura 4.3: Datos recopilados del diagnóstico realizado mediante el system performance.

En primer lugar se nos muestra un sumario con datos del procesador, disco y memoria. Si pulsamos en el texto mas inferior que detalla algo de información, vemos que nos lleva a otra parte en donde nos detalla perfectamente todo. Por ejemplo para el caso de la CPU:

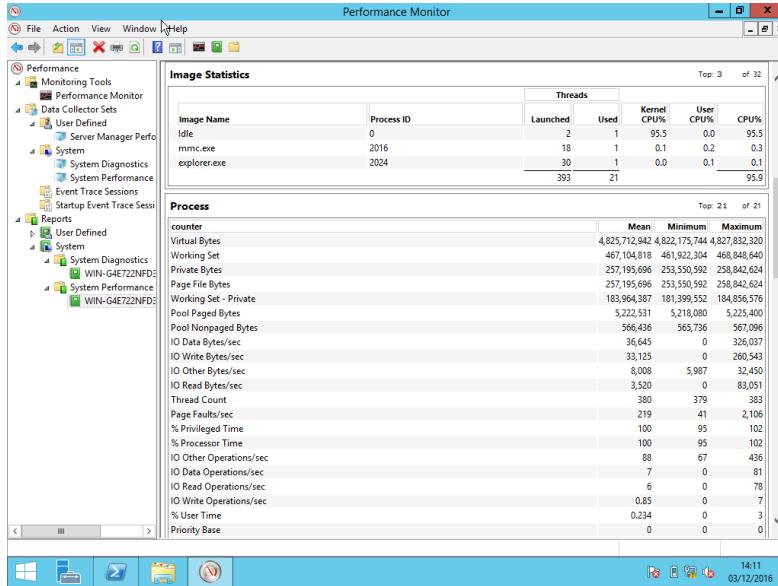


Figura 4.4: Desglose de información con respecto al procesador(CPU).

Vemos como nos muestran los procesos acompañados de su ID, que han usado la CPU , y el porcentaje de esta que han usado. También vemos las hebras que se han lanzado y se han usado. Así como también se puede observar el porcentaje de CPU del kernel y del usuario para cada proceso. Se puede observar como el idle(ocioso) se lleva el 95.5 % de porcentaje de CPU, por ello podemos admitir que el sistema está prácticamente inactivo. Por otro lado y dirigiéndonos a la imagen 4.3 observamos que la memoria esta en un 27 % de utilización, del total de 2GiB.

Más abajo el propio recopilador de datos nos informa sobre el uso de los principales componentes del servidor como la CPU, la red, la memoria y el disco. Y nos deja caer información como el poco uso de la CPU, el prácticamente nulo uso de la red debido a ese 0 %, en disco el número de entradas/salidas (escritura/lectura) es de 4 por segundo, y en memoria el uso es más normal con unos 1505 MiB disponibles del total de 2GiB.

El recopilador también nos trae una gráfica con muchísimas componentes.

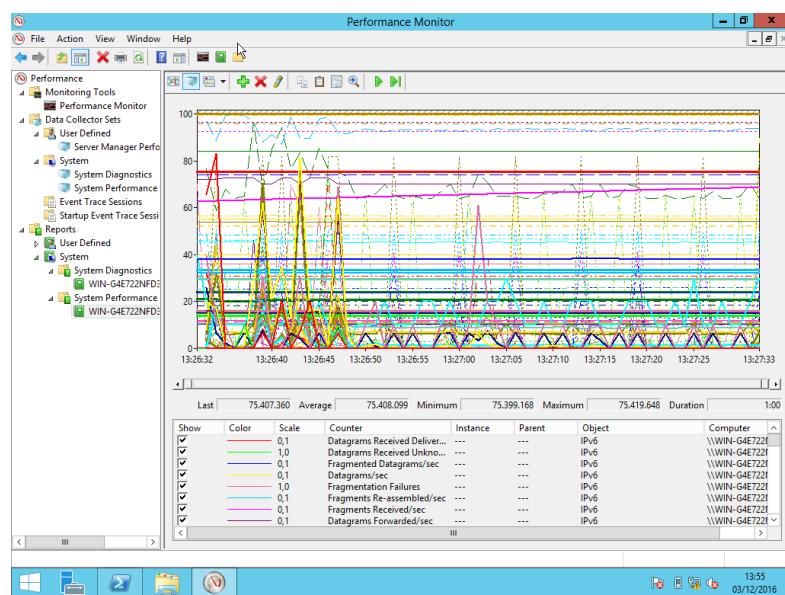


Figura 4.5: Gráfica de los datos recopilados.

Como bien se puede apreciar en la imagen la gráfica es poco legible, debido a la cantidad de componentes que se introducen en ella, pero hay una opción para llevarnos esta gráfica a datos, y así poder razonar sobre ellos.

Por ejemplo podemos mostrar los datos sobre la memoria, en ella se puede ver los Bytes disponibles, los KiloBytes disponibles y los MegaBytes disponibles. También los Bytes de cache, las faltas de páginas por segundo, el porcentaje de Bytes en uso, las entradas en las tablas de páginas libres del sistema, etc:

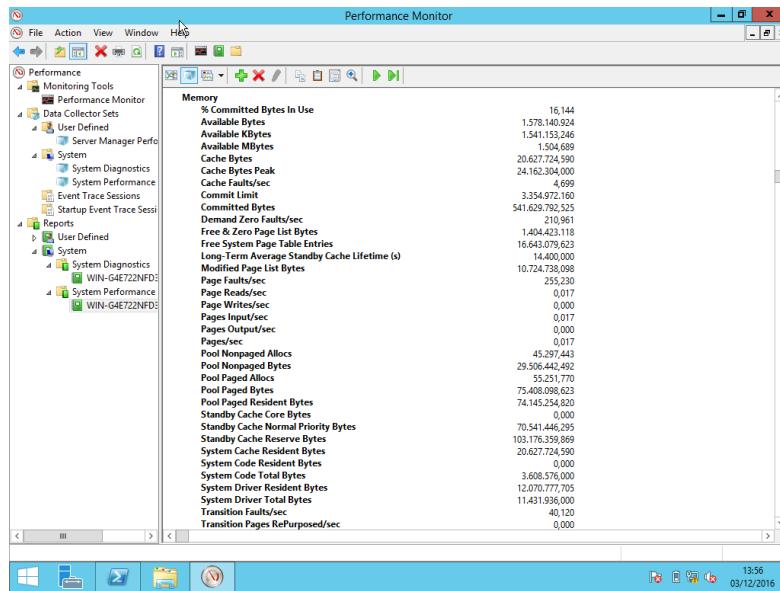


Figura 4.6: Datos sobre la memoria recopilados.

También podemos mostrar información sobre los procesos:

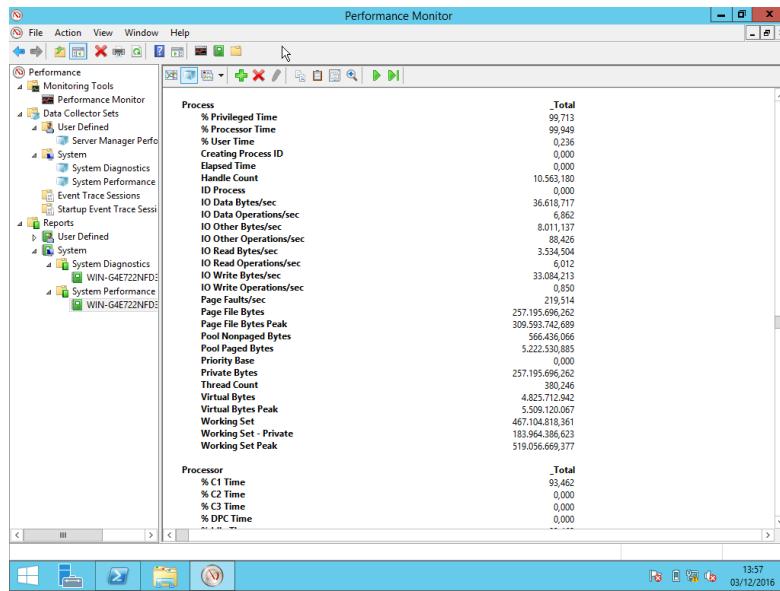


Figura 4.7: Datos sobre los procesos recopilados.

Por último mostramos información sobre el procesador, el porcentaje de cada procesador, el porcentaje ocioso, interrupciones por segundo, etc.

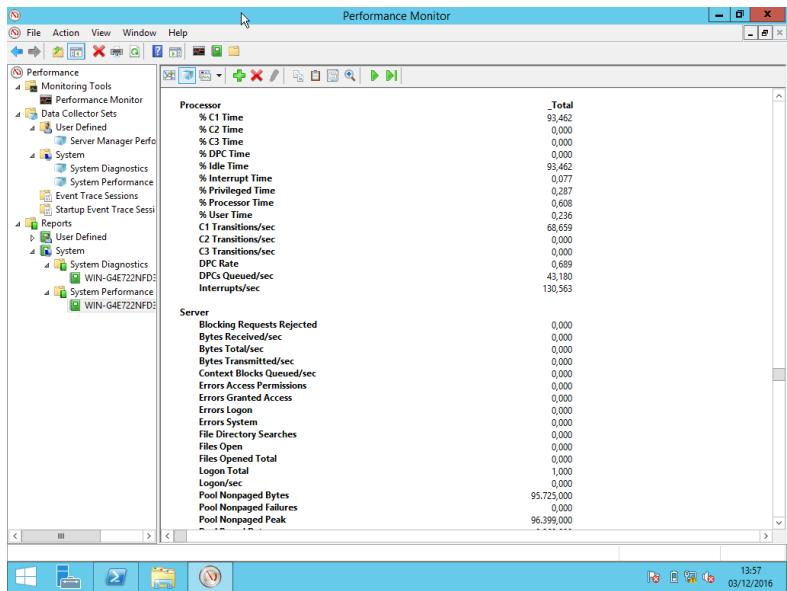


Figura 4.8: Datos sobre el procesador recopilados.

5. Cree un recopilador de datos definido por el usuario (modo avanzado) que incluya tanto el contador de rendimiento como los datos de seguimiento:

Todos los referentes al procesador, al proceso y al servicio web.

Intervalo de muestra 15 segundos.

Almacene el resultado en el directorio Escritorio/logs.

Incluya las capturas de pantalla de cada paso.

5.1. Respuesta :

A continuación se muestra el proceso de creación de un recopilador de datos definido por el usuario[8] :

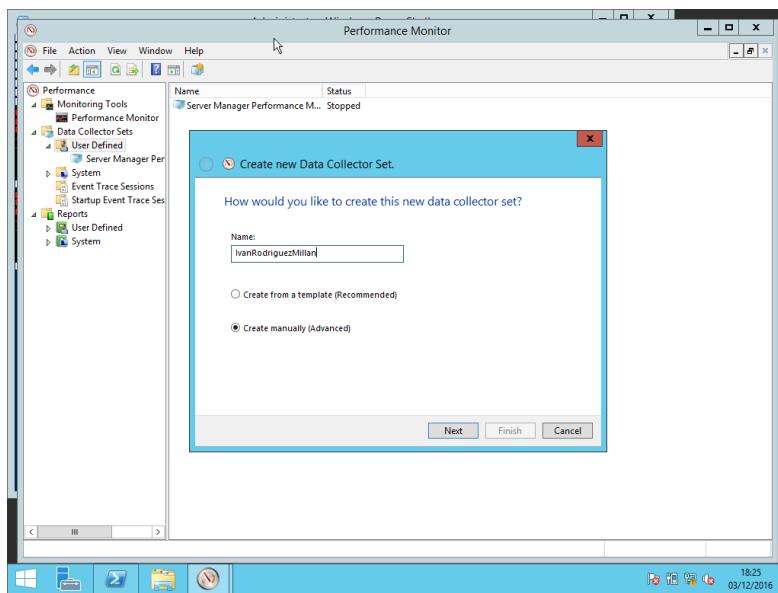


Figura 5.1: Inicio creación de un recopilador de datos.

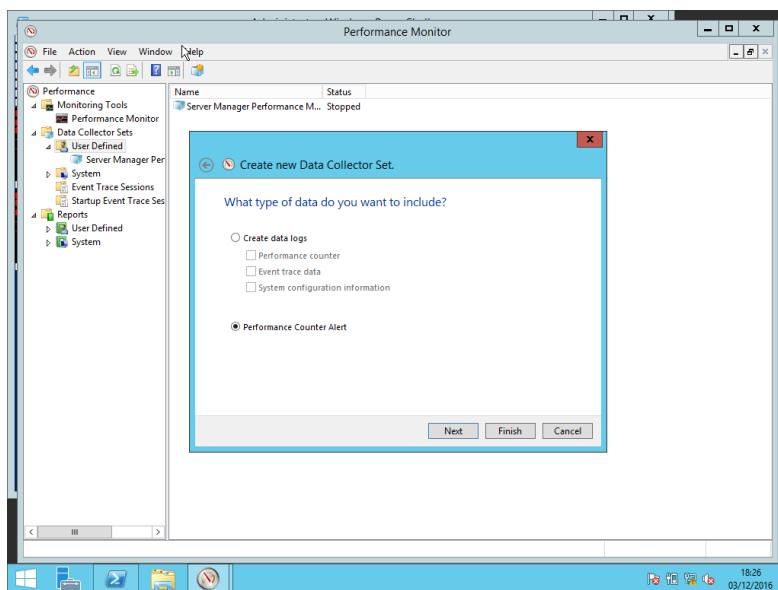


Figura 5.2: Indicando performance counter alert(alerta de contador del rendimiento).

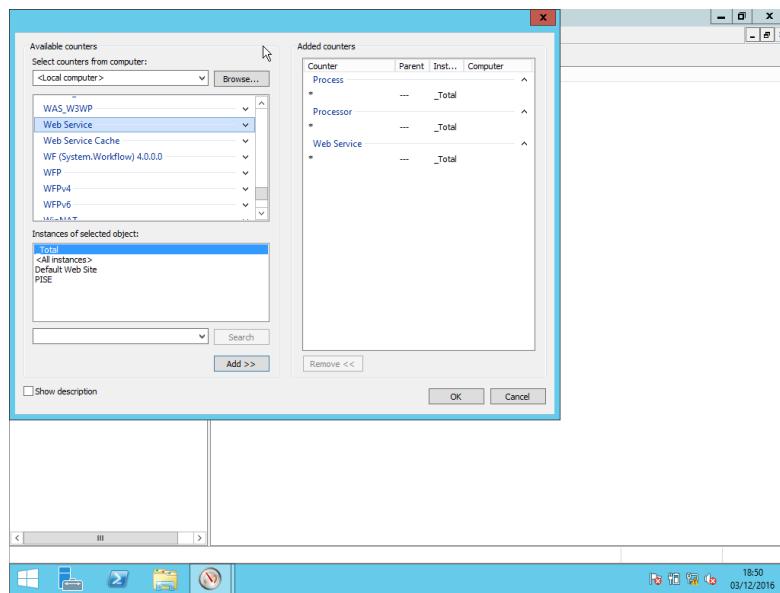


Figura 5.3: Añadiendo los contadores de seguimiento , proceso, procesador y servicio web.

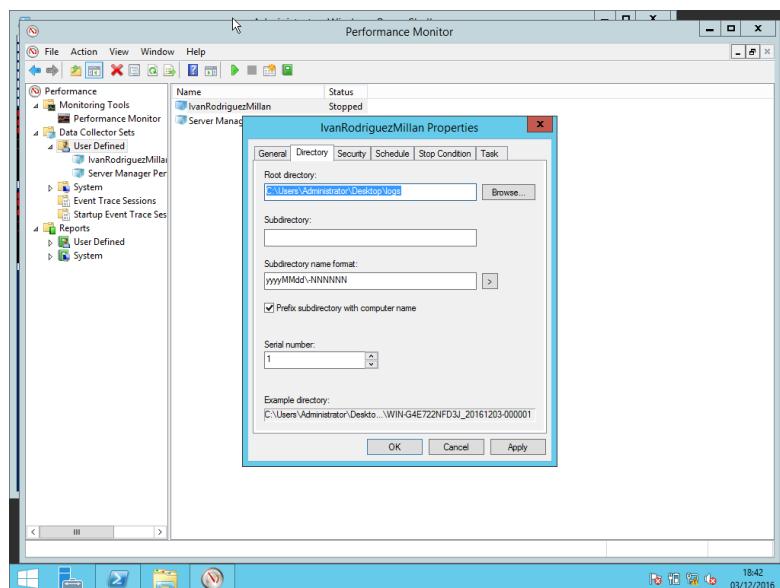


Figura 5.4: Cambiamos el directorio donde se guardarán los resultados.

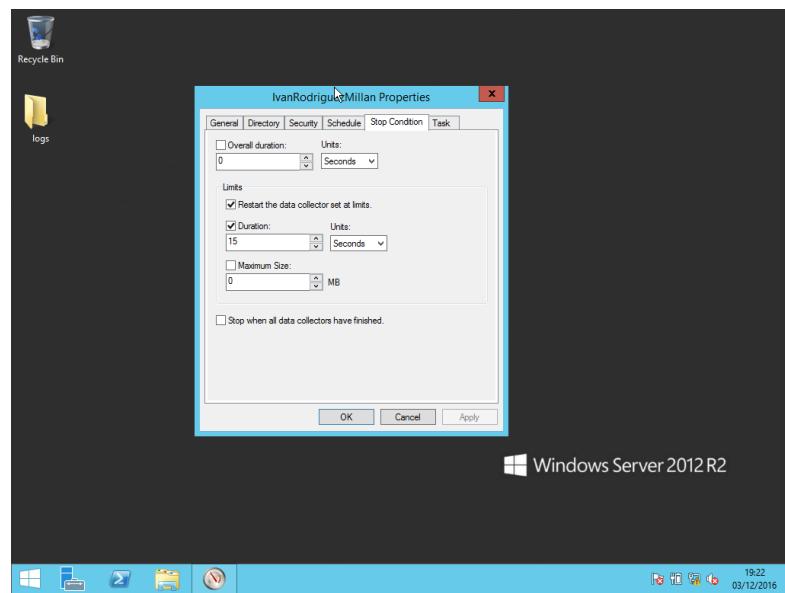


Figura 5.5: Indicamos que los intervalos de muestras sean cada 15 segundo.

- 6. Visite la web (MUNIN) del proyecto y acceda a la demo que proporcionan donde se muestra cómo monitorizan un servidor. Monitorice varios parámetros y haga capturas de pantalla de lo que está mostrando comentando qué observa.**

6.1. Respuesta :

En primer lugar muestro una imagen con la monitorización de la utilización de disco de la última semana. Se puede apreciar como a lo largo de la semana apenas supera el 0 porciento de utilización de disco, salvo el domingo en donde se aprecia un pico de más de un 10 porciento de utilización.



Figura 6.1: Monitorización de la utilización de disco de la última semana.

En segundo lugar se muestra una imagen sobre el uso de CPU en la última semana. En ella se puede apreciar como la mayor parte del tiempo ha estado ociosa, el sistema apenas ha hecho uso, y el usuario/consumidor apenas ha llegado al 20 porciento de uso.

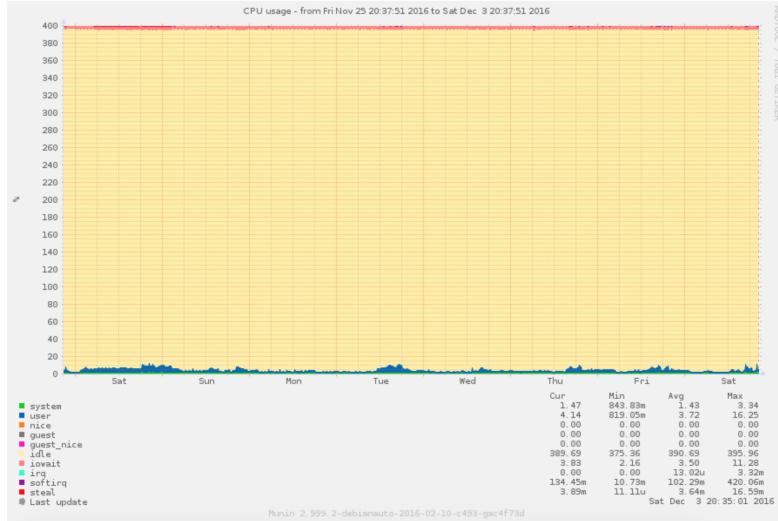


Figura 6.2: Monitorización del uso de la CPU de la última semana.

En tercer lugar se muestra una imagen con los cambios de contexto e interrupciones producidas en la última semana. Podemos ver como a lo largo de la semana se producen de media unas 131.14 interrupciones y unos 141.09 cambios de contexto de media. Aunque el domingo apreciamos un gran pico de interrupciones llegando desde las 100 hasta las 600, y en cambios de contexto desde los 100 hasta los 1300.

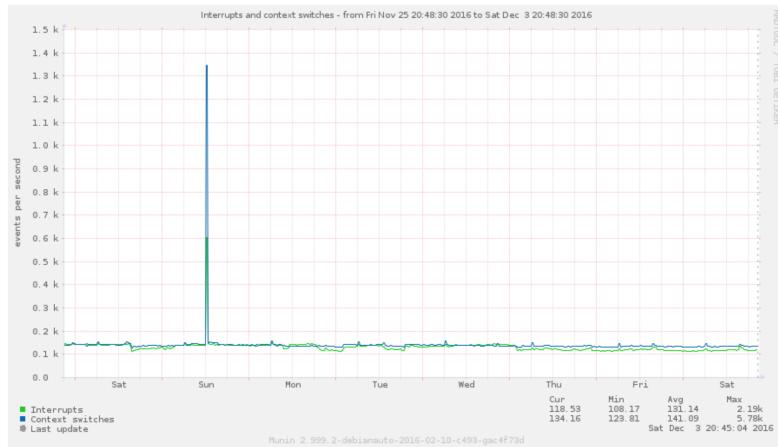


Figura 6.3: Monitorización de los cambios de contexto e interrupciones producidos en la última semana.

Y por cuarto lugar muestro una imagen del tráfico en las diferentes interfaces de red(local y eth0). Se ve como de media la interfaz eth0 es por la que más bytes por segundo se en-

vían, y por la interfaz local es por la que más bytes por segundo se reciben. Curiosamente cuando más datos se envían por a través de la interfaz eth0 es en el mes de Diciembre (Concretamente a finales), esto puede ser debido a ser los días de fin de año. También a comienzos de año (Enero y Febrero) se ve un gran aumento de la actividad con respecto al resto de año.

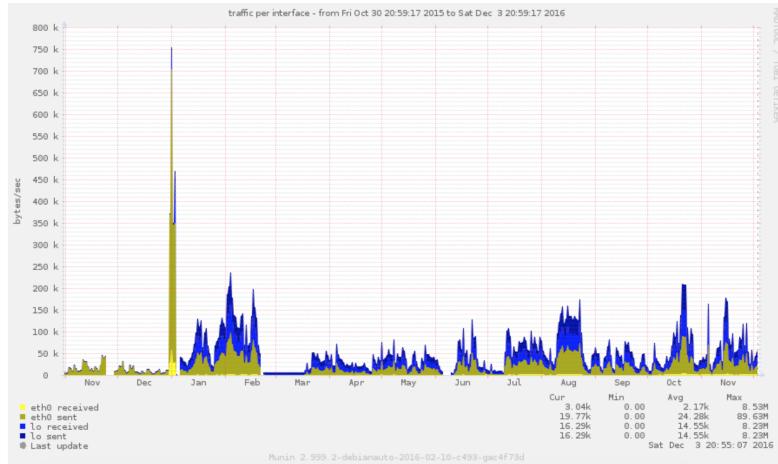


Figura 6.4: Monitorización del tráfico de bytes por segundo a través de las diferentes interfaces de red en el último año.

7. Escriba un breve resumen sobre alguno de los artículos donde se muestra el uso de strace o busque otro y coméntelo.

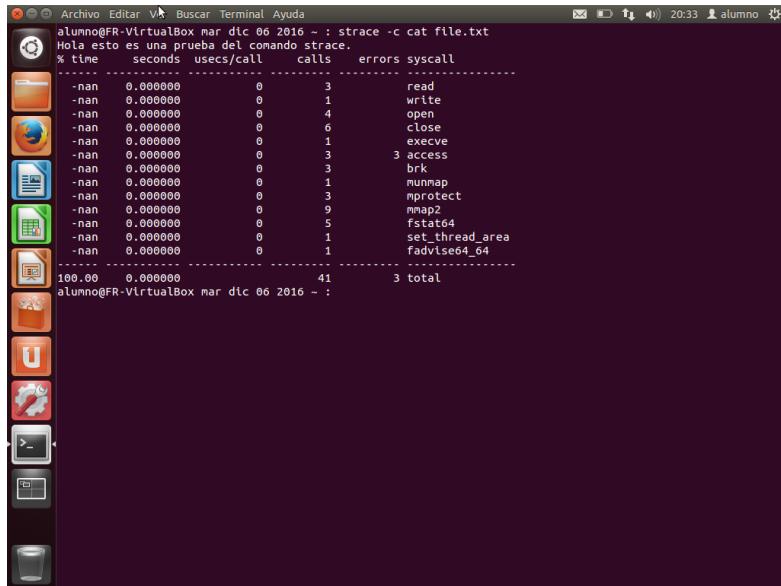
7.1. Respuesta :

Muy a menudo ocurre que los procesos finalizan de manera incorrecta y no se tiene información de que problema hubo, ni tampoco información útil de como resolver los errores. Para los administradores de Ubuntu hay una solución, y esta es strace. Strace te muestra lo que el programa está haciendo internamente en cada paso.

Aunque pueda parecer un depurador, no lo es. Y tampoco es una herramienta del programador. Es una herramienta del sistema que sirve para supervisar y visualizar las llamadas y señales del sistema. El uso de strace se lleva a cabo mediante comandos del bash. El uso se hace bastante sencillo, pues solamente tendrás que adjuntar strace al proceso, y automáticamente se te mostrarán las llamadas del sistema y señales resultantes del proceso.

Vamos a realizar una prueba bastante sencilla de entender. Nos creamos un fichero de texto, y probamos realizar un cat fichero.txt con strace, pero a su vez, para ver las di-

ferencias de cuando nos da error a cuando no, en un caso vamos a tener permisos de lectura y en otro caso no.



The screenshot shows a terminal window titled "alumno@FR-VirtualBox mar dic 06 2016 ~ : strace -c cat file.txt". The window contains the following text:

```
Hola esto es una prueba del comando strace.  
% time    seconds  usecs/call   calls  errors syscall  
-----  
-nan  0.000000      0       3      read  
-nan  0.000000      0       1      write  
-nan  0.000000      0       4      open  
-nan  0.000000      0       6      close  
-nan  0.000000      0       1      execve  
-nan  0.000000      0       3      3 access  
-nan  0.000000      0       3      brk  
-nan  0.000000      0       1      munmap  
-nan  0.000000      0       3      mprotect  
-nan  0.000000      0       9      mmap2  
-nan  0.000000      0       5      fstat64  
-nan  0.000000      0       1      set_thread_area  
-nan  0.000000      0       1      fadvise64_64  
-----  
100.00 0.000000      41      3 total
```

Figura 7.1: Ejecución del comando strace teniendo permisos de lectura sobre el fichero de prueba.

En el primer caso tenemos permisos de lectura y vemos que en la fila de open no nos da ningún error. Lo ejecutamos con el comando -c, para resumir la información saliente, ya que si lo hiciésemos de la forma tradicional (sin el comando -c), nos saldría una cantidad de información que para el ejemplo es innecesaria.

The screenshot shows a terminal window titled "Terminal" with the command "strace -c cat file.txt" running. The output shows various system calls and their statistics. The last line of the output is "100.00 0.00000 65 15 total".

	% time	seconds	usecs/call	calls	errors	syscall
-nan	0.000000	0	1	3		read
-nan	0.000000	0	4	4		write
-nan	0.000000	0	18	12		open
-nan	0.000000	0	8	8		close
-nan	0.000000	0	1	1		execve
-nan	0.000000	0	3	3		access
-nan	0.000000	0	3	3		brk
-nan	0.000000	0	2	2		munmap
-nan	0.000000	0	3	3		mprotect
-nan	0.000000	0	12	12		mmap2
-nan	0.000000	0	7	7		fstat64
-nan	0.000000	0	1	1		set_thread_area
	100.00	0.00000	65	15	total	

Figura 7.2: Ejecución del comando strace sin tener permisos de lectura sobre el fichero de prueba.

En el segundo caso probamos a quitarle permisos de lectura, y efectivamente nos salta con un error en la fila de open. Y es que no podemos abrir un fichero si sobre él no tenemos permisos.

8. Escriba un script en Python o PHP y analice su comportamiento usando el profiler presentado.

8.1. Respuesta :

En mi caso, voy a realizar lo siguiente, en primer lugar, creamos un script para calcular cualquier número de Fibonacci de manera recursiva, y a esto le pasaremos el profiler mencionado en [9], seguidamente creamos otro script para calcular cualquier número de Fibonacci también, pero esta vez de manera iterativa. Es conocido el hecho de que el algoritmo de Fibonacci es más rápido de forma iterativa que de forma recursiva debido a la acción de ir guardando los valores anteriores al número a calcular. Así nos deben de dar tiempos bastante dispares para un valor de fibonacci medianamente alto como es el 40 (en Fibonacci el 102334155).

```
root@ubuntu dom dic 04 2016 ~ : cat fibo.py
#!/usr/bin/env python
import cProfile
def fib(n):
    if n==0 or n==1:
        return n
    else:
        return fib(n-1) + fib(n-2)
cProfile.run('print fib(40)')
root@ubuntu dom dic 04 2016 ~ : python fibo.py
102334155
331160283 function calls (3 primitive calls) in 93.100 seconds

Ordered by: standard name

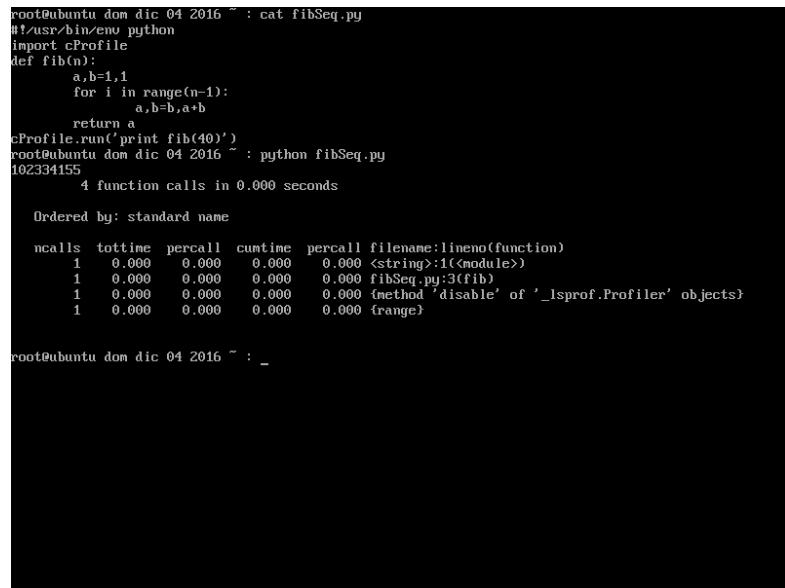
   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
      1    0.000    0.000   93.100   93.100 <string>:1(<module>)
331160281/1   93.100    0.000   93.100   93.100 fibo.py:3(fib)
      1    0.000    0.000     0.000     0.000 {method 'disable' of '_lsprof.Profiler' objects}

root@ubuntu dom dic 04 2016 ~ :
```

Figura 8.1: Analizando el comportamiento de un script recursivo realizado en Python con el profiler cProfiler.

En esta imagen, en la salida que proporciona el profiler cProfiler se pueden apreciar varias cosas, en primer lugar, se indica que 331160283 llamadas fueron monitorizadas, y del total 3 fueron primitivas. En total el programa duró 93.1 segundos. ncalls es el número de llamadas que se realizan al programa en cuestión, cuando aparecen dos números, viene a indicarnos recursividad en el programa, siendo el segundo valor el número de llamadas primitivas. tottime es el tiempo total que se le ha dedicado a la función dada. percall es el tottime dividido entre ncalls. cumtime es el tiempo acumulado en la función dada y las funciones inferiores. Y el último percall es el cociente de cumtime entre las llamadas primitivas.

En la siguiente imagen se muestra lo mismo pero de forma iterativa.



```
root@ubuntu dom dic 04 2016 ~ : cat fibSeq.py
#!/usr/bin/env python
import cProfile
def fib(n):
    a,b=1,1
    for i in range(n-1):
        a,b=b,a+b
    return a
cProfile.run('print fib(40)')
root@ubuntu dom dic 04 2016 ~ : python fibSeq.py
102334155
    4 function calls in 0.000 seconds

Ordered by: standard name

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
         1    0.000    0.000    0.000    0.000 <string>:1(<module>)
         1    0.000    0.000    0.000    0.000 fibSeq.py:3(fib)
         1    0.000    0.000    0.000    0.000 {method'>.disable' of '_lsprof.Profiler' objects}
         1    0.000    0.000    0.000    0.000 {range}

root@ubuntu dom dic 04 2016 ~ :
```

Figura 8.2: Analizando el comportamiento de un script iterativo realizado en Python con el profiler cProfiler.

En este caso vemos que se monitorizan 4 llamadas, y duraron 0 segundos (Obviamente no duró 0 segundos, simplemente el programa aproxima a 0 el valor que parece haber sido ínfimo).

Se puede apreciar la diferencia claramente entre ambos script y sobre todo la forma de programar un mismo problema como es el de Fibonacci tardando en el primero algo más de 1 minuto y medio, mientras que en el segundo apenas tarda en calcular el valor final.

- 9. Acceda a la consola mysql (o a través de phpMyAdmin) y muestre el resultado de mostrar el “profile” de una consulta (la creación de la BD y la consulta la puede hacer libremente).**

9.1. Respuesta :

En primer lugar entramos a mysql y creamos una nueva base de datos [10] para realizar el ejercicio.

```
root@ubuntu dom dic 04 2016 ~ : mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 58
Server version: 5.5.53-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> create database miprueba;
```

Figura 9.1: Acceso a mysql y creación de una nueva base de datos.

En segundo lugar nos colocamos en esa base de datos con el comando “use” , creamos una tabla [11] de prueba e insertamos unos datos. [12]

```

mysql> use nprueba
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> create table prueba_tabla (id_prueba int);
Query OK, 0 rows affected (0.00 sec)

mysql> insert into prueba_tabla (id_prueba) values (1);
Query OK, 1 row affected (0.01 sec)

mysql> insert into prueba_tabla (id_prueba) values (2);
Query OK, 1 row affected (0.01 sec)

mysql> insert into prueba_tabla (id_prueba) values (3);
Query OK, 1 row affected (0.00 sec)

mysql> insert into prueba_tabla (id_prueba) values (4);
Query OK, 1 row affected (0.00 sec)

mysql> insert into prueba_tabla (id_prueba) values (5);
Query OK, 1 row affected (0.00 sec)

mysql> insert into prueba_tabla (id_prueba) values (6);
Query OK, 1 row affected (0.00 sec)

mysql> insert into prueba_tabla (id_prueba) values (7);
Query OK, 1 row affected (0.00 sec)

mysql> insert into prueba_tabla (id_prueba) values (8);
Query OK, 1 row affected (0.00 sec)

mysql> insert into prueba_tabla (id_prueba) values (9);
Query OK, 1 row affected (0.00 sec)

mysql>

```

Figura 9.2: Creación de una tabla de prueba e inserción de datos.

```

mysql> show profiles;
+-----+-----+-----+
| Query_ID | Duration | Query      |
+-----+-----+-----+
| 1 | 0.00394000 | show tables |
| 2 | 0.00031375 | describe prueba_tabla |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from prueba_tabla;
+-----+
| id_prueba |
+-----+
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| 7 |
| 8 |
| 9 |
+-----+
9 rows in set (0.00 sec)

mysql> show profiles;
+-----+-----+-----+
| Query_ID | Duration | Query      |
+-----+-----+-----+
| 1 | 0.00394000 | show tables |
| 2 | 0.00031375 | describe prueba_tabla |
| 3 | 0.00006000 | select * from prueba_tabla |
+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>

```

Figura 9.3: Verificación del número perteneciente a la consulta realizada, para mostrar el profile.

Y por ultimo realizamos una consulta [13] a esta tabla y usando el comando “show profiles” verificamos que número se le a asignado a esta consulta, para posteriormente mostrar el profile.

```
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| 7 |
| 8 |
| 9 |
+----+
9 rows in set (0.00 sec)

mysql> show profiles;
+-----+-----+-----+
| Query_ID | Duration | Query          |
+-----+-----+-----+
| 1 | 0.003941000 | show tables      |
| 2 | 0.00031375 | describe prueba_tabla |
| 3 | 0.00006000 | select * from prueba_tabla |
+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> show profile cpu for query 3;
+-----+-----+-----+-----+
| Status           | Duration | CPU_user | CPU_system |
+-----+-----+-----+-----+
| starting         | 0.000030 | 0.000000 | 0.000000 |
| Waiting for query cache lock | 0.000003 | 0.000000 | 0.000000 |
| checking query cache for query | 0.000008 | 0.000000 | 0.000000 |
| checking privileges on cached | 0.000002 | 0.000000 | 0.000000 |
| checking permissions       | 0.000006 | 0.000000 | 0.000000 |
| sending cached result to client | 0.000009 | 0.000000 | 0.000000 |
| logging slow query        | 0.000002 | 0.000000 | 0.000000 |
| cleaning up        | 0.000003 | 0.000000 | 0.000000 |
+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql> _
```

Figura 9.4: Mostrando el profile de una consulta realizada a la base de datos creada como ejemplo.

Referencias

- [1] “https://www.debian.org/doc/manuals/debian-reference/ch02.en.html#_verification_of_installed_package_files,” consultado el 1 de Diciembre de 2016.
- [2] “https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-managing_log_files_in_a_graphical_environment.html#s2-logfiles-viewing,” consultado el 1 de Diciembre de 2016.
- [3] “https://debian-handbook.info/browse/es-ES/stable/sect_task-scheduling-cron-atd.html,” consultado el 2 de Diciembre de 2016.
- [4] “<http://manpages.ubuntu.com/manpages/precise/man8/logrotate.8.html>,” consultado el 2 de Diciembre de 2016.
- [5] “https://debian-administration.org/article/117/Creating_logfile_archives_with_logrotate,” consultado el 2 de Diciembre de 2016.
- [6] “https://debian-handbook.info/browse/es-ES/stable/sect_task-scheduling-cron-atd.html,” consultado el 2 de Diciembre de 2016.
- [7] “<http://manpages.ubuntu.com/manpages/wily/man1/dmesg.1.html>,” consultado el 3 de Diciembre de 2016.
- [8] “[https://technet.microsoft.com/es-es/library/cc722414\(v=ws.11\).aspx](https://technet.microsoft.com/es-es/library/cc722414(v=ws.11).aspx),” consultado el 3 de Diciembre de 2016.
- [9] “<https://docs.python.org/2/library/profile.html>,” consultado el 4 de Diciembre de 2016.
- [10] “<http://dev.mysql.com/doc/refman/5.7/en/create-database.html>,” consultado el 4 de Diciembre de 2016.
- [11] “<http://dev.mysql.com/doc/refman/5.7/en/create-table.html>,” consultado el 4 de Diciembre de 2016.
- [12] “<http://dev.mysql.com/doc/refman/5.7/en/insert.html>,” consultado el 4 de Diciembre de 2016.
- [13] “<http://dev.mysql.com/doc/refman/5.7/en/select.html>,” consultado el 4 de Diciembre de 2016.