

**Ingeniería de Servidores (2014-2015)**  
GRADO EN INGENIERÍA INFORMÁTICA  
UNIVERSIDAD DE GRANADA

---

## Memoria Práctica 3

---

Jose Antonio Jiménez Montañés

2 de diciembre de 2014

## Índice

1. a) ¿Qué archivo le permite ver qué programas se han instalado con el gestor de paquetes? b) ¿Qué significan las terminaciones. 1.gz o .2.gz de los archivos en ese directorio? 5
2. Indique qué comandos ha utilizado para realizarlo así como capturas de pantalla del proceso de reconstrucción del RAID. [1] 6
3. ¿Qué archivo ha de modificar para programar una tarea? Escriba la línea necesaria para ejecutar una vez al día una copia del directorio /codigo a /seguridad/\$fecha donde \$fecha es la fecha actual (puede usar el comando date). [2] 7
4. Pruebe a ejecutar el comando, 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. 8
5. Ejecute el monitor de “System Performance” y muestre el resultado. Incluya capturas de pantalla comentando la información que aparece. 9
6. 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. 11
7. Instale alguno de los monitores comentados arriba en su máquina y pruebe a ejecutarlos (tenga en cuenta que si lo hace en la máquina virtual, los resultados pueden no ser realistas). Alternativamente, busque otros monitores para hardware comerciales o de código abierto para Windows y Linux. 17
8. Visite la web del proyecto y acceda a la demo que proporcionan (<http://demo.munin-monitoring.org/>) donde se muestra cómo monitorizan un servidor. Monitoree varios parámetros y haga capturas de pantalla de lo que está mostrando comentando qué observa. 19
9. Instale Nagios en su sistema (el que prefiera) documentando el proceso y muestre el resultado de la monitorización de su sistema comentando qué aparece. [3] 20
10. Con Ganglia haga lo mismo que con Munin. 25

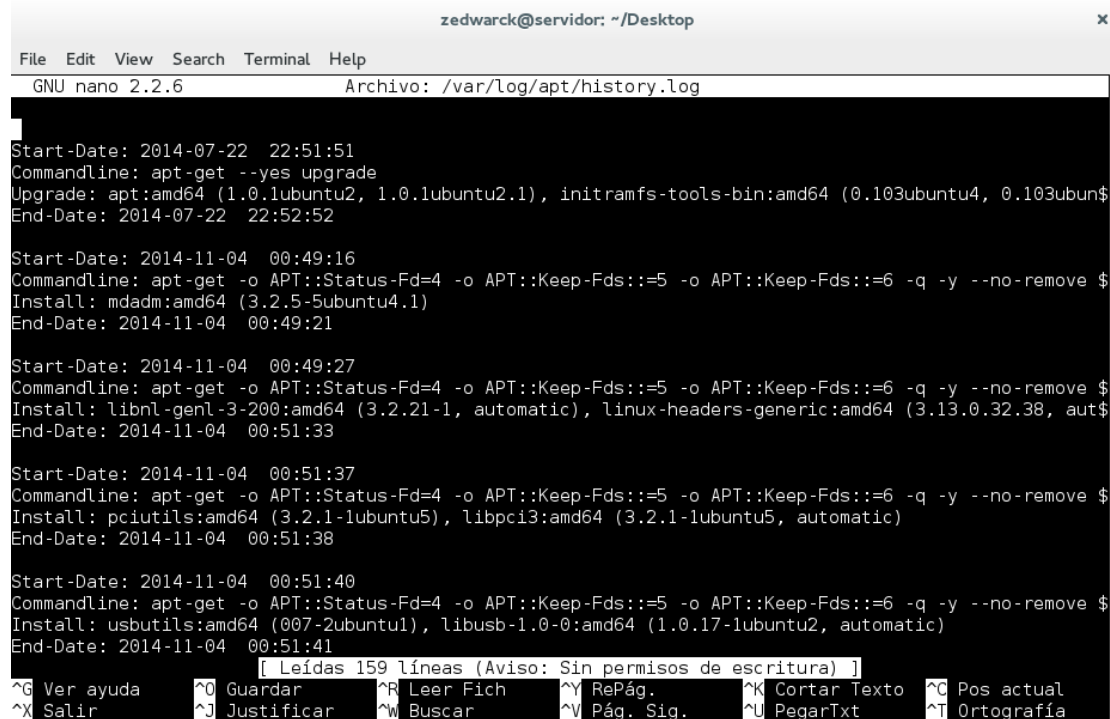
11. Pruebe a instalar este monitor en alguno de sus tres sistemas. Realice capturas de pantalla del proceso de instalación y comente capturas de pantalla del programa en ejecución.[4] 25
12. Pruebe a instalar este monitor en alguno de sus tres sistemas. Realice capturas de pantalla del proceso de instalación y comente las capturas de pantalla del programa de ejecución 27
13. Escriba un breve resumen sobre alguno de los artículos donde se muestra el uso de strace o busque otro y coméntelo 34
14. Desarrolle una página en C o C++ y analice su comportamiento usando valgrind. Visite <http://www.cs.tut.fi/~jkorpela/forms/cgic.html> para ver un ejemplo sencillo de una página web generada por un programa escrito en C. 34
15. 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). 38

## Índice de figuras

1.1. Contenido de history.log . . . . .	5
2.1. Sistema detecta que el RAID falla . . . . .	6
2.2. Sistema detecta nuevo disco en RAID y lo actualiza . . . . .	6
3.1. Configuración del crontab para la tarea diaria. . . . .	7
4.1. Últimos registros Hardware del sistema(Antes de introducir USB). . . . .	8
4.2. Últimos registros Hardware del sistema(Después de introducir USB). . . . .	8
5.1. Esperando a que capture los datos para obtener los resultados. . . . .	9
5.2. Resumen del contenido de datos regidos. . . . .	9
5.3. Menú donde se recogen todos los datos almacenados por categorías. . . . .	10
6.1. Iniciando un recopilador personal. . . . .	11
6.2. Tipo de datos que vamos a incluir en el recopilador. . . . .	12
6.3. Datos que vamos a incluir en el recopilador. . . . .	13
6.4. Intervalo y confirmación de datos a incluir. . . . .	14
6.5. Fichero de salida de datos. . . . .	15
6.6. Finalización de la configuración manual del recopilador de datos. . . . .	16
7.1. xsensors ejecutándose junto con los comandos de opciones disponibles. . . . .	17
7.2. nmon ejecutándose . . . . .	18
7.3. Monitorización de memoria con nmon. . . . .	18
8.1. Monitorización de las interfaces de red desde la webdemo . . . . .	19
8.2. Monitorización del uso de memoria desde la webdemo . . . . .	19
9.1. Prerequisitos de nagios con apt-get. . . . .	20
9.2. Creación de usuario y grupo Nagios. . . . .	21
9.3. Archivos de instalación de Nagios descomprimidos. . . . .	21
9.4. Preconfiguración de Nagios completada. . . . .	22
9.5. Instalación de Nagios completada y servicio corriendo. . . . .	23
9.6. Home Web de Nagios funcionando. . . . .	24
10.1. Monitorización de servidor de pruebas con Munin . . . . .	25
11.1. Pantalla de login de Zabbix. . . . .	25
11.2. Pantalla Principal de Zabbix. . . . .	26
12.1. Pantalla inicial de preinstalación de Cacti. . . . .	27
12.2. Opción de nueva instalación o actualización. . . . .	28
12.3. Comprobación de rutas en la instalación de cacti. . . . .	29
12.4. Login en cacti. . . . .	30
12.5. Cambio de password para admin en cacti. . . . .	31
12.6. Home de cacti. . . . .	32
12.7. Gráficas de muestreo en cacti. . . . .	33
14.1. Resultado en consola de valgrind sobre un programa de multiplicación de matrices. . . . .	37
15.1. Resultado de la consulta en tabla y el tiempo que a tardado en realizarla. . . . .	38
15.2. Resultado de la consulta en formato gráfico. . . . .	39

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

a) En Ubuntu Server esta localizado en: /var/logs/apt/history.log



```
zedwarck@servidor: ~/Desktop
File Edit View Search Terminal Help
GNU nano 2.2.6 Archivo: /var/log/apt/history.log

Start-Date: 2014-07-22 22:51:51
Commandline: apt-get --yes upgrade
Upgrade: apt:amd64 (1.0.1ubuntu2, 1.0.1ubuntu2.1), initramfs-tools-bin:amd64 (0.103ubuntu4, 0.103ubun$
End-Date: 2014-07-22 22:52:52

Start-Date: 2014-11-04 00:49:16
Commandline: apt-get -o APT::Status-Fd=4 -o APT::Keep-Fds::=5 -o APT::Keep-Fds::=6 -q -y --no-remove $
Install: mdadm:amd64 (3.2.5-5ubuntu4.1)
End-Date: 2014-11-04 00:49:21

Start-Date: 2014-11-04 00:49:27
Commandline: apt-get -o APT::Status-Fd=4 -o APT::Keep-Fds::=5 -o APT::Keep-Fds::=6 -q -y --no-remove $
Install: libnl-genl-3-200:amd64 (3.2.21-1, automatic), linux-headers-generic:amd64 (3.13.0.32.38, aut$
End-Date: 2014-11-04 00:51:33

Start-Date: 2014-11-04 00:51:37
Commandline: apt-get -o APT::Status-Fd=4 -o APT::Keep-Fds::=5 -o APT::Keep-Fds::=6 -q -y --no-remove $
Install: pciutils:amd64 (3.2.1-lubuntu5), libpci3:amd64 (3.2.1-lubuntu5, automatic)
End-Date: 2014-11-04 00:51:38

Start-Date: 2014-11-04 00:51:40
Commandline: apt-get -o APT::Status-Fd=4 -o APT::Keep-Fds::=5 -o APT::Keep-Fds::=6 -q -y --no-remove $
Install: usbutils:amd64 (007-2ubuntu1), libusb-1.0-0:amd64 (1.0.17-lubuntu2, automatic)
End-Date: 2014-11-04 00:51:41

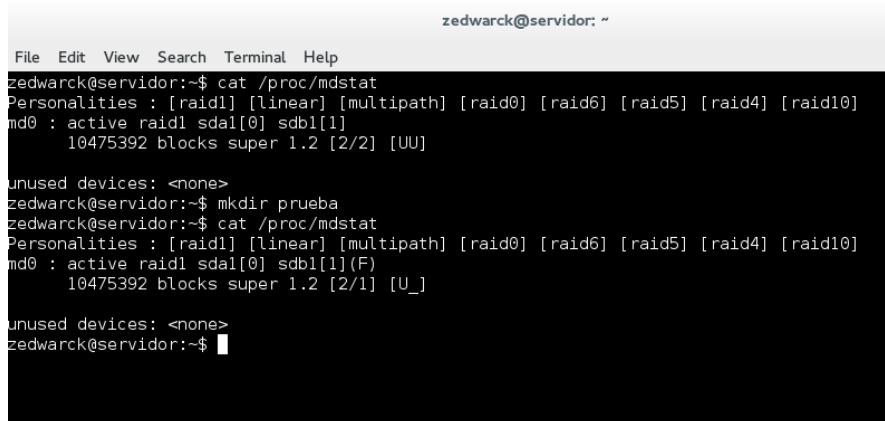
[ Leídas 159 líneas (Aviso: Sin permisos de escritura) ]
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Texto ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Figura 1.1: Contenido de history.log

b) El nivel de compresión gzip utilizado.

## 2. Indique qué comandos ha utilizado para realizarlo así como capturas de pantalla del proceso de reconstrucción del RAID. [1]

Comprobamos primero que funciona el RAID1 y eliminamos el disco2. Comprobamos que detecta que el disco 2 falla: (Lo hemos eliminado)



```
zedwarck@servidor: ~  
File Edit View Search Terminal Help  
zedwarck@servidor:~$ cat /proc/mdstat  
Personalities : [raid1] [linear] [multipath] [raid0] [raid6] [raid5] [raid4] [raid10]  
md0 : active raid1 sdal[0] sdb1[1]  
      10475392 blocks super 1.2 [2/2] [UU]  
  
unused devices: <none>  
zedwarck@servidor:~$ mkdir prueba  
zedwarck@servidor:~$ cat /proc/mdstat  
Personalities : [raid1] [linear] [multipath] [raid0] [raid6] [raid5] [raid4] [raid10]  
md0 : active raid1 sdal[0] sdb1[1](F)  
      10475392 blocks super 1.2 [2/1] [U_]  
  
unused devices: <none>  
zedwarck@servidor:~$
```

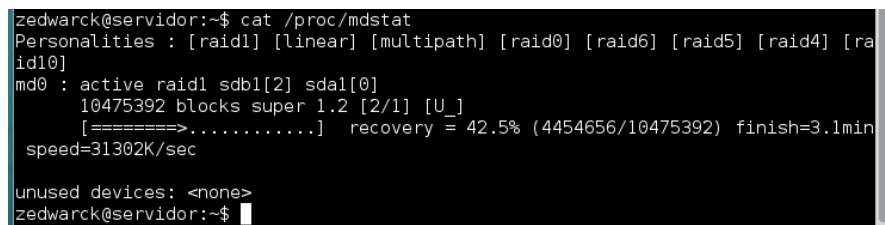
Figura 2.1: Sistema detecta que el RAID falla

Si el disco diera fallos lógicos debería ser retirado con: "sudo mdadm --remove /dev/md0 /dev/sdb1"

De una forma u de otra el siguiente paso sería retirarlo del sistema con: "sudo mdadm --manage /dev/md0 --remove /dev/sdb1"

Luego procedemos a instalar el nuevo disco que ya debería estar en el sistema con: "sudo mdadm --add /dev/md0 /dev/sdb1"

Comprobamos que ahora esta el disco en el sistema RAID y esta actualizando datos:



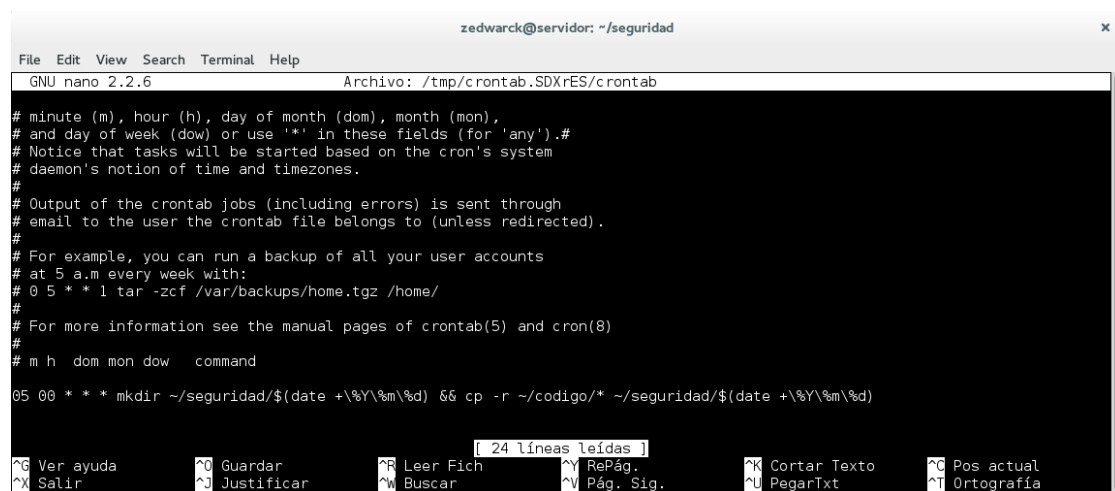
```
zedwarck@servidor:~$ cat /proc/mdstat  
Personalities : [raid1] [linear] [multipath] [raid0] [raid6] [raid5] [raid4] [raid10]  
md0 : active raid1 sdb1[2] sdal[0]  
      10475392 blocks super 1.2 [2/1] [U_]  
      [=====>.....]  recovery = 42.5% (4454656/10475392) finish=3.1min  
      speed=31302K/sec  
  
unused devices: <none>  
zedwarck@servidor:~$
```

Figura 2.2: Sistema detecta nuevo disco en RAID y lo actualiza

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

Se crea o se modifica el archivo de tareas con `crontab -e`, aunque si ya lo hemos creado alguna vez estará situado en: `/var/spool/cron/usuario`

Una vez creado, el demonio cron se ejecuta cada cierto tiempo(a las 5am en nuestro caso) y ejecuta la tarea que le hemos indicado, en nuestro caso: creación de un directorio con la fecha actual y copia del contenido de `/codigo` dentro del directorio creado en `"seguridad"`



The screenshot shows a terminal window titled 'zedwarck@servidor: ~/seguridad'. Inside, the GNU nano 2.2.6 editor is open, editing the file '/tmp/crontab.SDXrES/crontab'. The editor displays the standard crontab header comments and a single cron job entry at the bottom: `05 00 * * * mkdir ~/seguridad/$(date +%Y%m%d) && cp -r ~/codigo/* ~/seguridad/$(date +%Y%m%d)`. The bottom status bar of the nano editor shows various keyboard shortcuts like '^G Ver ayuda', '^O Guardar', etc.

Figura 3.1: Configuración del crontab para la tarea diaria.

**4. Pruebe a ejecutar el comando, 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.**

Usamos el comando: "dmesg | tail -10" para visualizar las 10 ultimas lineas del archivo de logs:

```
File Edit View Search Terminal Help
zedwarck@servidor:~$ dmesg | tail -10
[ 282.021762] disk 1, wo:1, o:1, dev:sdb1
[ 282.026991] md: recovery of RAID array md0
[ 282.026996] md: minimum_guaranteed_speed: 1000 KB/sec/disk.
[ 282.026998] md: using maximum available idle IO bandwidth (but not more than 200000 KB/sec) for recovery.
[ 282.027014] md: using 128k window, over a total of 10475392k.
[ 640.806032] md: md0: recovery done.
[ 640.857606] RAID1 conf printout:
[ 640.857610] --- wd:2 rd:2
[ 640.857613] disk 0, wo:0, o:1, dev:sda1
[ 640.857615] disk 1, wo:0, o:1, dev:sdb1
zedwarck@servidor:~$
```

Figura 4.1: Últimos registros Hardware del sistema(Antes de introducir USB).

Luego insertamos un dispositivo USB y volvemos a visualizar el archivo:

```
zedwarck@servidor:~$ dmesg | tail -20
[ 640.857610] --- wd:2 rd:2
[ 640.857613] disk 0, wo:0, o:1, dev:sda1
[ 640.857615] disk 1, wo:0, o:1, dev:sdb1
[ 5650.591014] usb 1-1: new high-speed USB device number 2 using ehci-pci
[ 5650.823672] usb 1-1: New USB device found, idVendor=0951, idProduct=1643
[ 5650.823678] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 5650.823680] usb 1-1: Product: DataTraveler G3
[ 5650.823682] usb 1-1: Manufacturer: Kingston
[ 5650.823684] usb 1-1: SerialNumber: 001CC05FE8ECFC90B9112502
[ 5650.955708] usb-storage 1-1:1.0: USB Mass Storage device detected
[ 5650.956020] scsi3 : usb-storage 1-1:1.0
[ 5650.956407] usbcore: registered new interface driver usb-storage
[ 5652.032392] scsi 3:0:0:0: Direct-Access Kingston DataTraveler G3 1.00 PQ: 0 ANSI: 4
[ 5652.033205] sd 3:0:0:0: Attached scsi generic sg3 type 0
[ 5652.048253] sd 3:0:0:0: [sdc] 30233588 512-byte logical blocks: (15.4 GB/14.4 GiB)
[ 5652.052356] sd 3:0:0:0: [sdc] Write Protect is off
[ 5652.052363] sd 3:0:0:0: [sdc] Mode Sense: 45 00 00 00
[ 5652.056299] sd 3:0:0:0: [sdc] Write cache: disabled, read cache: enabled, doesn't support DPO or FUA
[ 5652.084367] sdc: sdc1
[ 5652.103282] sd 3:0:0:0: [sdc] Attached SCSI removable disk
zedwarck@servidor:~$
```

Figura 4.2: Últimos registros Hardware del sistema(Después de introducir USB).

Podemos observar como nos da información detallada acerca del dispositivo incluyendo su número de serie, capacidad, marca, modelo, cache, etc.



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

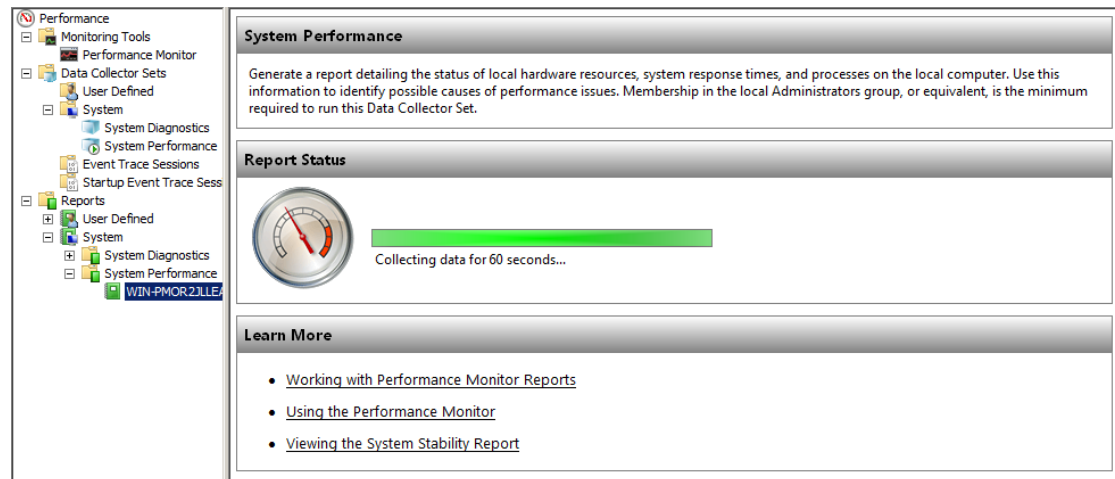


Figura 5.1: Esperando a que capture los datos para obtener los resultados.

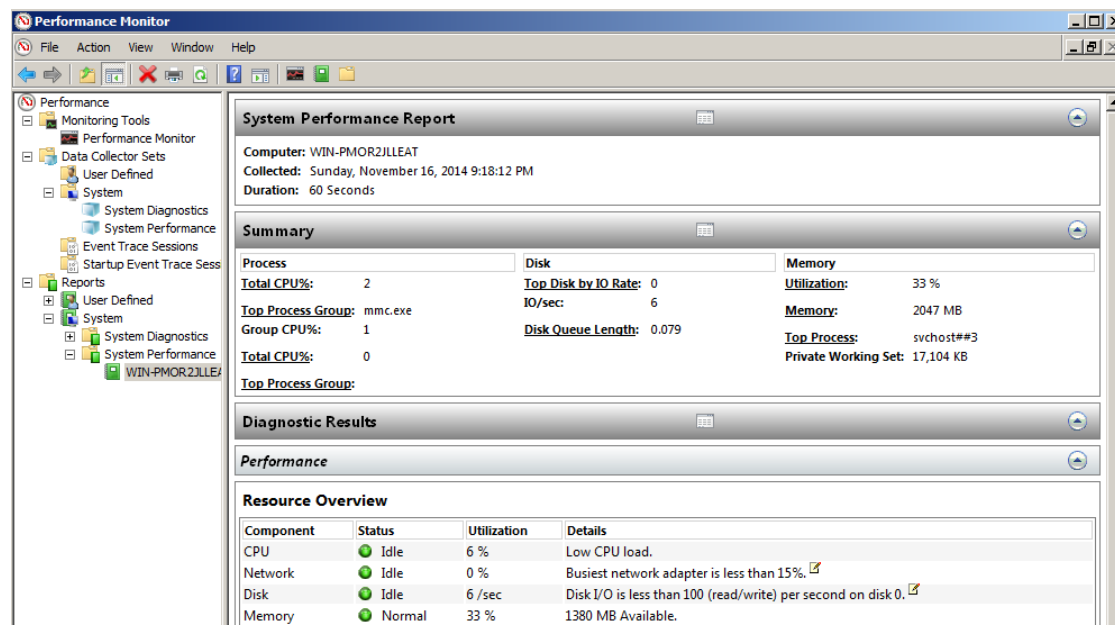


Figura 5.2: Resumen del contenido de datos regidos.

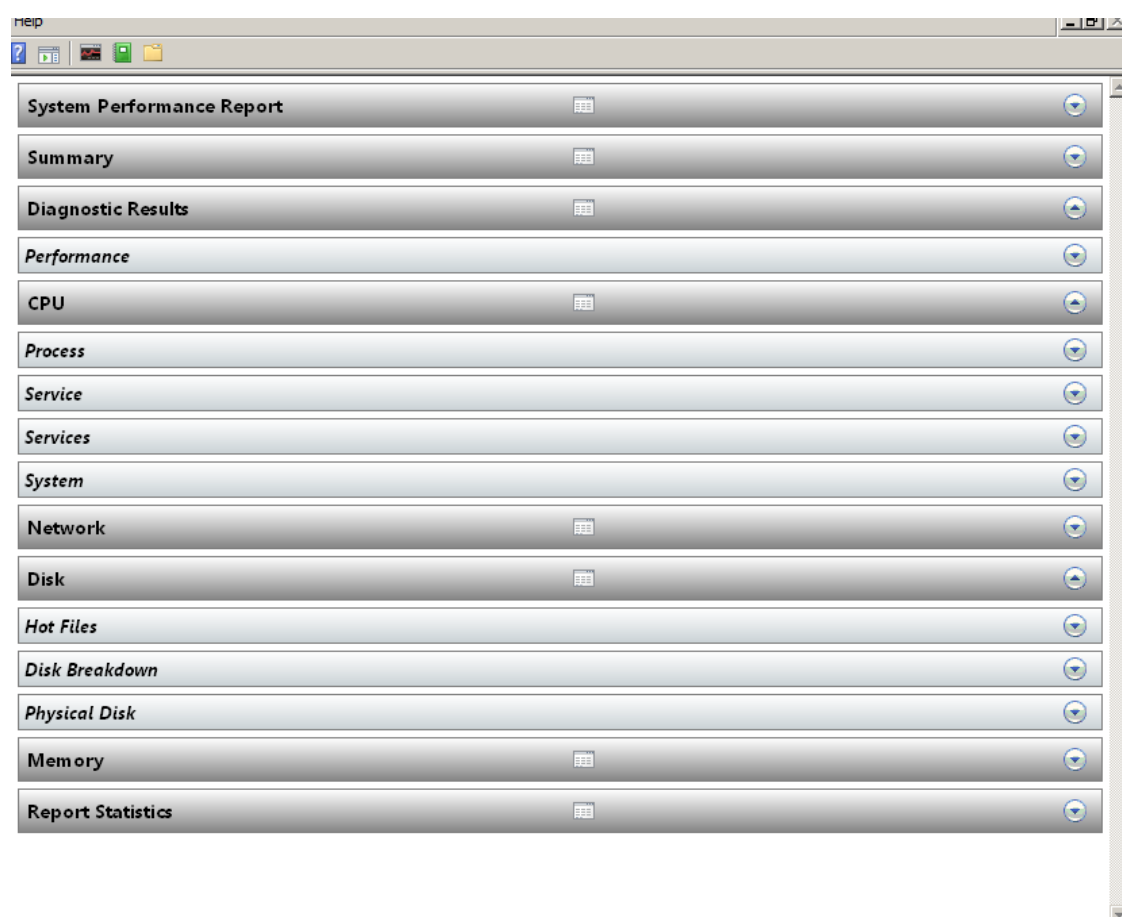


Figura 5.3: Menú donde se recogen todos los datos almacenados por categorías.

6. 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.

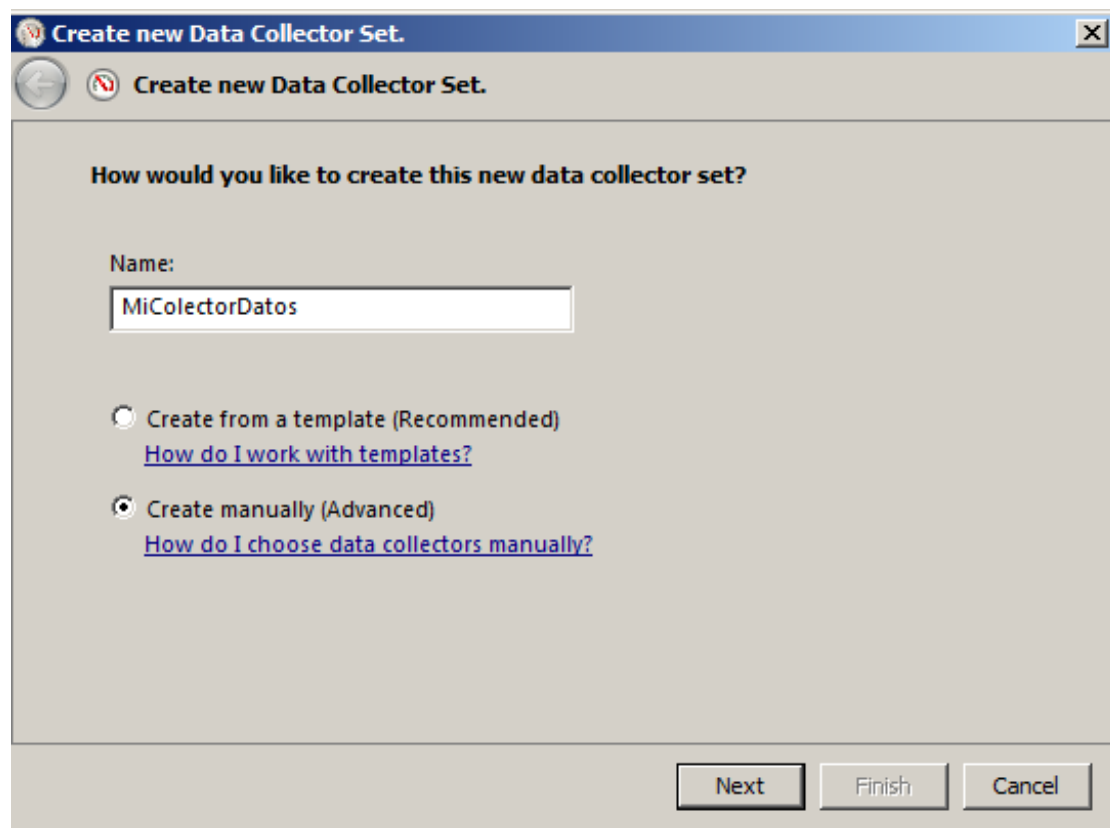


Figura 6.1: Iniciando un recopilador personal.

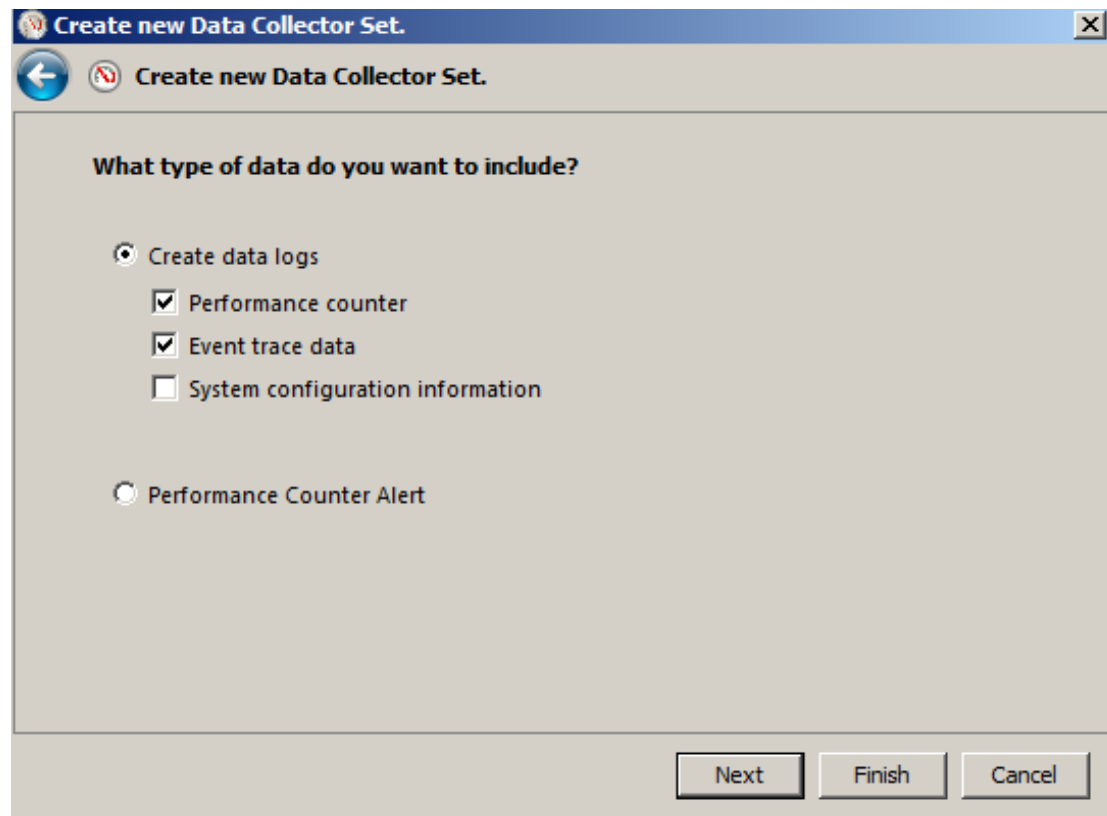


Figura 6.2: Tipo de datos que vamos a incluir en el recopilador.

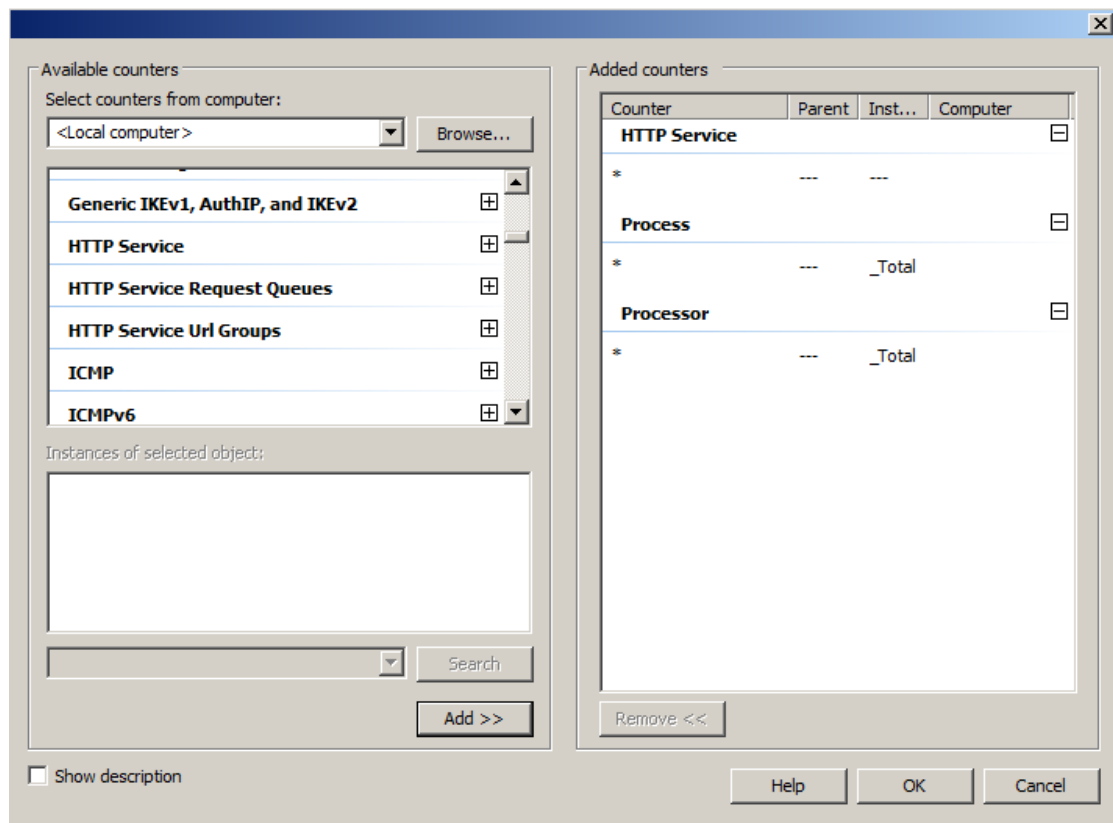


Figura 6.3: Datos que vamos a incluir en el recopilador.

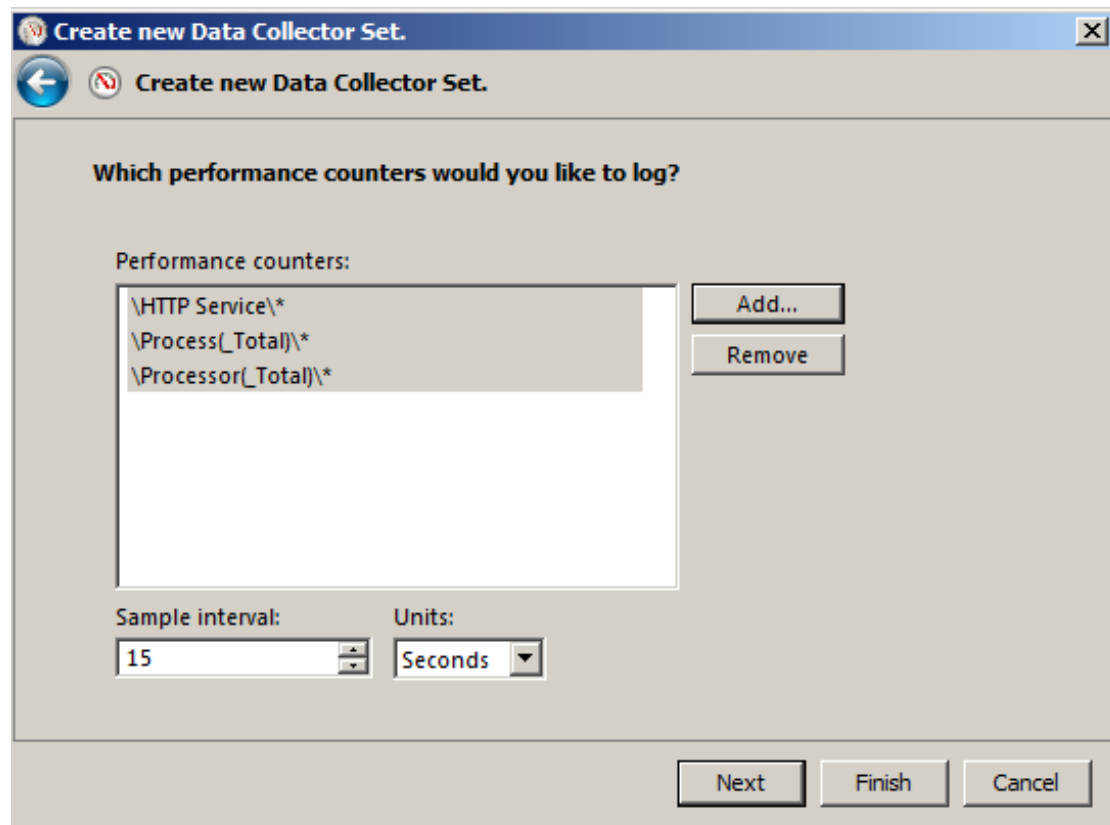


Figura 6.4: Intervalo y confirmación de datos a incluir.

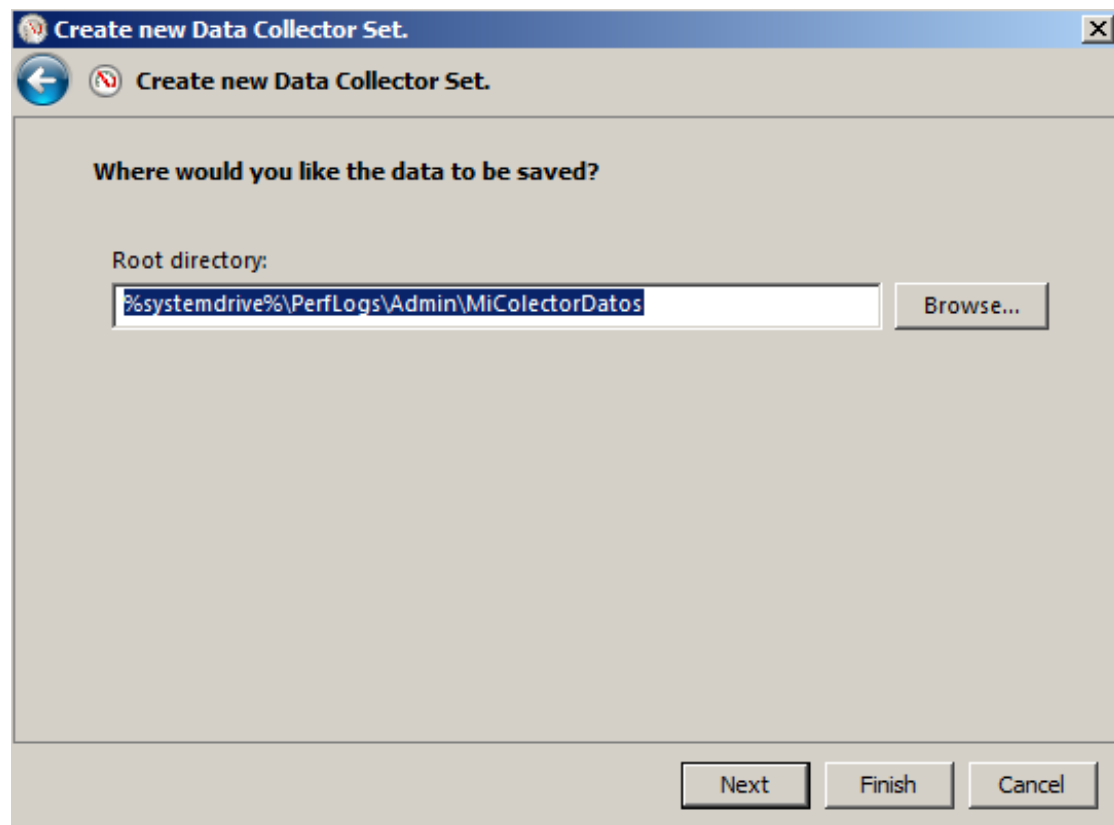


Figura 6.5: Fichero de salida de datos.

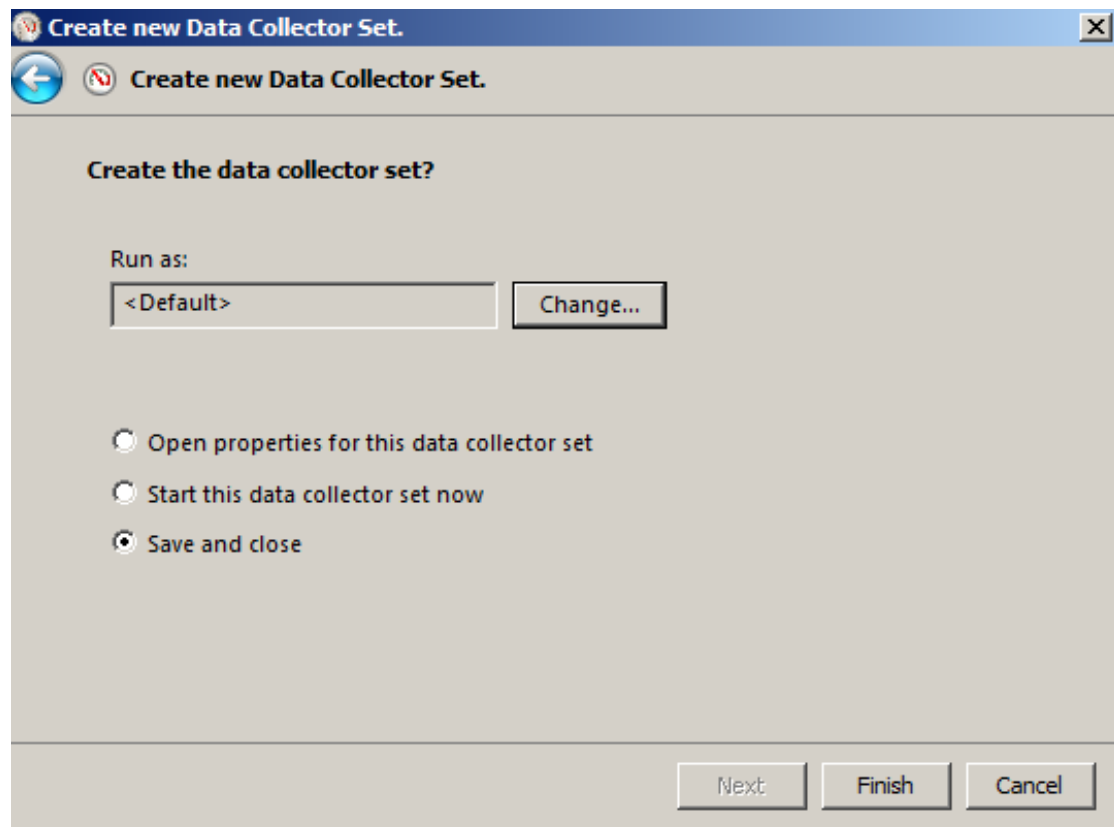
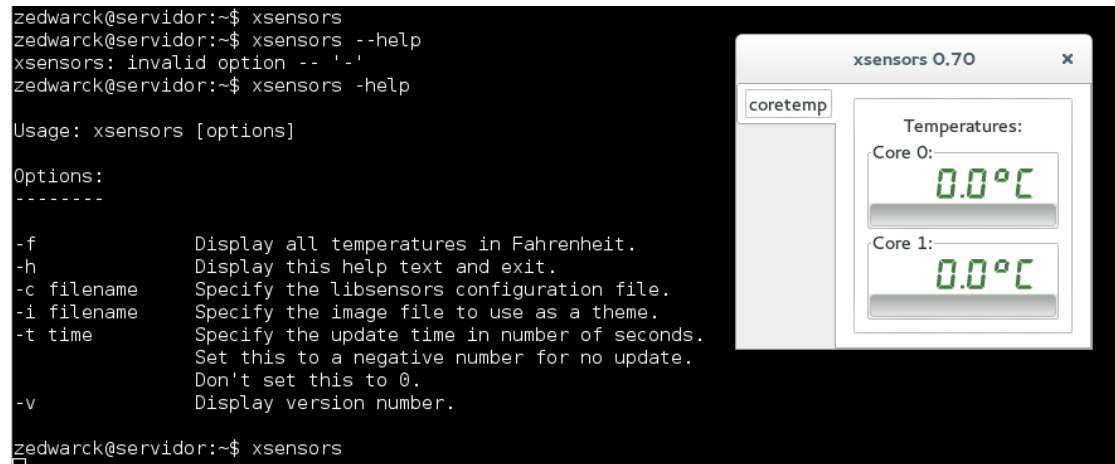


Figura 6.6: Finalización de la configuración manual del recopilador de datos.



**7. Instale alguno de los monitores comentados arriba en su máquina y pruebe a ejecutarlos (tenga en cuenta que si lo hace en la máquina virtual, los resultados pueden no ser realistas). Alternativamente, busque otros monitores para hardware comerciales o de código abierto para Windows y Linux.**

He elegido para probar el xsensors para linux:



```
zedwarck@servidor:~$ xsensors
zedwarck@servidor:~$ xsensors --help
xsensors: invalid option -- '.'
zedwarck@servidor:~$ xsensors -help

Usage: xsensors [options]

Options:
-----
-f          Display all temperatures in Fahrenheit.
-h          Display this help text and exit.
-c filename Specify the libsensors configuration file.
-i filename Specify the image file to use as a theme.
-t time     Specify the update time in number of seconds.
            Set this to a negative number for no update.
            Don't set this to 0.
-v          Display version number.

zedwarck@servidor:~$ xsensors
```

Figura 7.1: xsensors ejecutándose junto con los comandos de opciones disponibles.

En la figura podemos ver que no marca temperatura es debido a que estamos corriendo el programa bajo una maquina virtual.

Para linux hay otros incluso bajo consola bastante buenos como es el nmon:

```

zedwarck@servidor: /
File Edit View Search Terminal Help
nmon-14g [H for help] Hostname=servidor Refresh= 2secs 19:35.46

-----
# # # # ##### # #
## # ## ## # # #
# # # ## # # # #
# # # # # # # #
# ## # # # # #
# # # # ##### # #
-----

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

To start the same way every time
set the NMON ksh variable

Use these keys to toggle statistics on/off:
c = CPU          l = CPU Long-term    - = Faster screen updates
m = Memory       j = Filesystems      + = Slower screen updates
d = Disks        n = Network          V = Virtual Memory
r = Resource     N = NFS              v = Verbose hints
k = kernel       t = Top-processes    . = only busy disks/procs
h = more options q = Quit

```

Figura 7.2: nmon ejecutándose

Podemos ver como tiene muchos parámetros configurables y poder monitorizar bastantes cosas como por ejemplo la memoria:

```

zedwarck@servidor: /
File Edit View Search Terminal Help
nmon-14g Hostname=servidor Refresh= 2secs 19:36.28
Memory Stats
RAM      High      Low      Swap      Page Size=4 KB
Total MB 1994.8    -0.0     -0.0     1458.0
Free MB  290.4    -0.0     -0.0     1458.0
Free Percent 14.6%  100.0%  100.0%  100.0%

MB                               MB                               MB
          Cached= 741.4      Active= 899.0
Buffers= 144.6 Swapcached= 0.0 Inactive = 545.1
Dirty  = 0.1 Writeback = 0.0 Mapped = 158.2
Slab   = 178.5 Commit_AS = 2782.5 PageTables= 24.7

```

Figura 7.3: Monitorización de memoria con nmon.

Para Windows tenemos un amplio repertorio, como por ejemplo:

- Open Hardware Monitor.
- HWMonitor.

- Visite la web del proyecto y acceda a la demo que proporcionan (<http://demo.munin-monitoring.org/>) donde se muestra cómo monitorizan un servidor. Monitoree varios parámetros y haga capturas de pantalla de lo que está mostrando comentando qué observa.

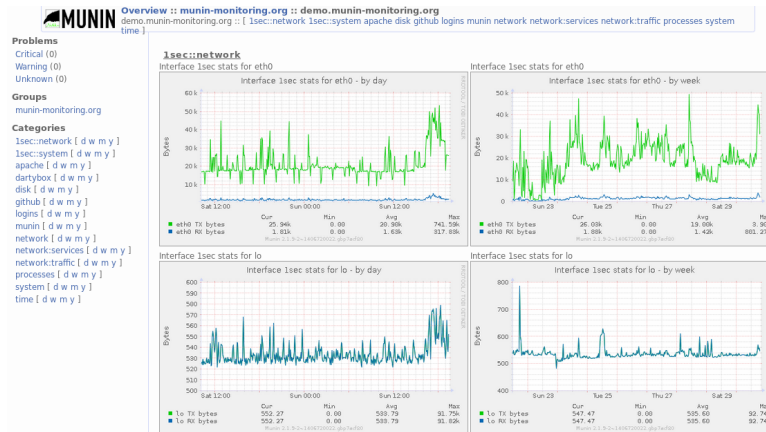


Figura 8.1: Monitorización de las interfaces de red desde la webdemo

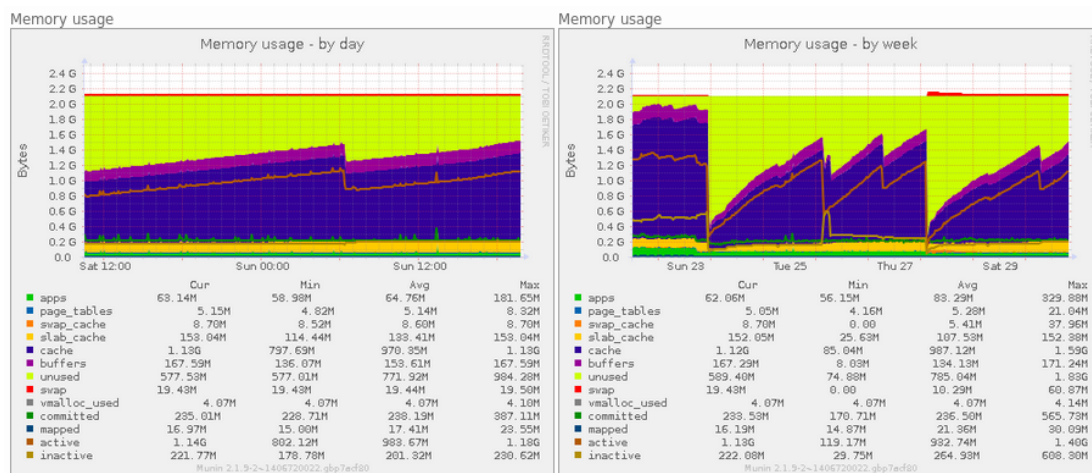


Figura 8.2: Monitorización del uso de memoria desde la webdemo

En ambos casos podemos ver 2 gráficas, la primera muestra la monitorización que hemos elegido a lo largo del día y la segunda a lo largo de la semana.

En la segunda gráfica de memoria podemos ver caídas de memoria seguramente originadas por reinicios del servidor o caídas del sistema.

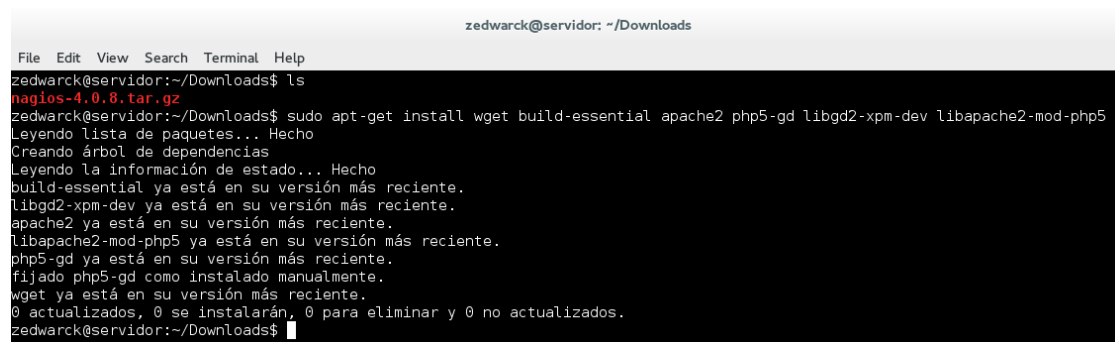
## 9. Instale Nagios en su sistema (el que prefiera) documentando el proceso y muestre el resultado de la monitorización de su sistema comentando qué aparece. [3]

Para instalar Nagios en Ubuntu primero preinstalamos algunos requisitos. según el manual habría que poner:

```
sudo apt-get install wget build-essential apache2 php5-gd libgd2-xpm libgd2-xpm-dev libapache2-modphp5
```

Pero vamos a dejar la línea de la siguiente manera para Ubuntu 14.04:

```
sudo apt-get install wget build-essential apache2 php5-gd libgd2-xpm-dev apache2-utils
```



```
zedwarck@servidor: ~/Downloads
File Edit View Search Terminal Help
zedwarck@servidor:~/Downloads$ ls
nagios-4.0.8.tar.gz
zedwarck@servidor:~/Downloads$ sudo apt-get install wget build-essential apache2 php5-gd libgd2-xpm-dev libapache2-mod-php5
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
build-essential ya está en su versión más reciente.
libgd2-xpm-dev ya está en su versión más reciente.
apache2 ya está en su versión más reciente.
libapache2-mod-php5 ya está en su versión más reciente.
php5-gd ya está en su versión más reciente.
fijado php5-gd como instalado manualmente.
wget ya está en su versión más reciente.
0 actualizados, 0 se instalarán, 0 para eliminar y 0 no actualizados.
zedwarck@servidor:~/Downloads$
```

Figura 9.1: Prerequisitos de nagios con apt-get.

Luego nos descargamos nagios:

```
cd /tmp
```

```
wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.4.tar.gz
```

```
wget http://nagios-plugins.org/download/nagios-plugins-2.0.tar.gz
```

Nos creamos un grupo y un usuario en él que se necesita para su correcto funcionamiento:

```
useradd nagios
```

```
groupadd nagcmd
```

```
usermod -a -G nagcmd nagios
```

```
root@servidor: /home/zedwarck/Downloads
File Edit View Search Terminal Help
zedwarck@servidor:~/Downloads$ sudo su
root@servidor:/home/zedwarck/Downloads# useradd nagios
useradd: user 'nagios' already exists
root@servidor:/home/zedwarck/Downloads# groupadd nagcmd
groupadd: group 'nagcmd' already exists
root@servidor:/home/zedwarck/Downloads# usermod -a -G nagcmd nagios
root@servidor:/home/zedwarck/Downloads#
```

Figura 9.2: Creacion de usuario y grupo Nagios.

Descomprimimos:

tar zxvf nagios-4.0.4.tar.gz

tar zxvf nagios-plugins-2.0.tar.gz

```
root@servidor:/home/zedwarck/Downloads# ls
nagios-4.0.8  nagios-4.0.8.tar.gz  nagios-plugins-2.0.3  nagios-plugins-2.0.3.tar.gz
root@servidor:/home/zedwarck/Downloads#
```

Figura 9.3: Archivos de instalación de Nagios descomprimidos.

Accedemos al directorio de instalación y configuramos el instalador:

`./configure --with-nagios-group=nagios --with-command-group=nagcmd --with-mail=/usr/bin/sendmail`

```

config.status: creating lib/iobroker.h
Creating sample config files in sample-config/ ...

*** Configuration summary for nagios 4.0.8 08-12-2014 ***:

General Options:
-----
    Nagios executable:  nagios
    Nagios user/group:  nagios,nagios
    Command user/group: nagios,nagcmd
    Event Broker:       yes
    Install ${prefix}:  /usr/local/nagios
    Install ${includedir}: /usr/local/nagios/include/nagios
    Lock file:          ${prefix}/var/nagios.lock
    Check result directory: ${prefix}/var/spool/checkresults
    Init directory:     /etc/init.d
    Apache conf.d directory: /etc/httpd/conf.d
    Mail program:       /usr/bin/sendmail
    Host OS:            linux-gnu
    IOBroker Method:    epoll

Web Interface Options:
-----
    HTML URL:  http://localhost/nagios/
    CGI URL:   http://localhost/nagios/cgi-bin/
Traceroute (used by WAP):

Review the options above for accuracy.  If they look okay,
type 'make all' to compile the main program and CGIs.

root@servidor:/home/zedwarck/Downloads/nagios-4.0.8# █

```

Figura 9.4: Preconfiguración de Nagios completada.

Luego hacemos los makes:

```

make all
make install
make install-init
make install-config
make install-commandmode
make install-webconf

```

```

Website: http://www.nagios.org
Reading configuration data...
  Read main config file okay...
  Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
  Checked 8 services.
  Checked 1 hosts.
  Checked 1 host groups.
  Checked 0 service groups.
  Checked 1 contacts.
  Checked 1 contact groups.
  Checked 24 commands.
  Checked 5 time periods.
  Checked 0 host escalations.
  Checked 0 service escalations.
Checking for circular paths...
  Checked 1 hosts
  Checked 0 service dependencies
  Checked 0 host dependencies
  Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors: 0

Things look okay - No serious problems were detected during the pre-flight check
root@servidor:/home/zedwarck/Downloads/nagios-4.0.8# /etc/init.d/nagios start
Starting nagios: done.
root@servidor:/home/zedwarck/Downloads/nagios-4.0.8# █

```

Figura 9.5: Instalacion de Nagios completada y servicio corriendo.

El último puede que nos de error, si es así nos creamos el directorio `/etc/httpd/conf.d` y volvemos a ejecutar el ultimo make. Después de eso nos vamos a ese directorio y copiamos el archivo `nagios.conf` al directorio de configuraciones de apache: `/etc/apache2/conf-enabled/`

Finalmente ponemos:

```

cp -R contrib/eventhandlers/ /usr/local/nagios/libexec/
chown -R nagios:nagios /usr/local/nagios/libexec/eventhandlers
/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

```

Iniciamos el servicio naglios: `service nagios restart`

Por ultimo reiniciamos apache: `service apache2 restart`

Para poder acceder nos hace falta configurar un usuario:

```
htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

Y para poder visualizar gráficas nos hace falta el plugin:

```
cd /tmp/nagios-plugins-2.0
./configure --with-nagios-user=nagios --with-nagios-group=nagios
make
make install
```

Con esto ya casi lo tendríamos todo, solo nos faltaría configurar nuestro apache para que acepte scripts CGI.

Para acceder, desde un navegador ponemos la IP o localhost(si esta activo desde el archivo .conf)/nagios:

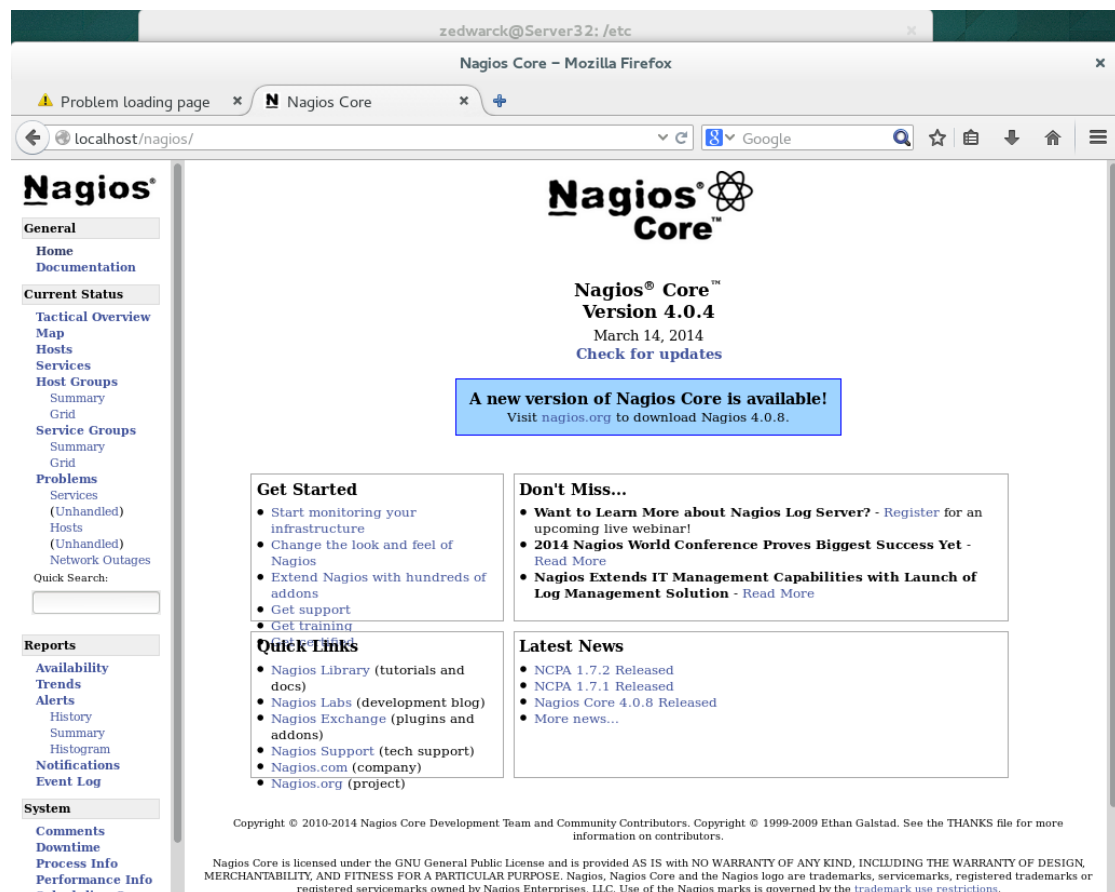


Figura 9.6: Home Web de Nagios funcionando.



## 10. Con Ganglia haga lo mismo que con Munin.

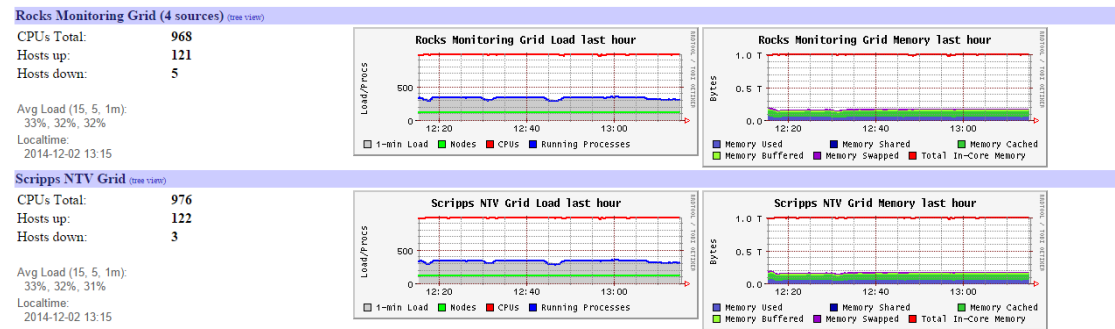


Figura 10.1: Monitorización de servidor de pruebas con Munin

## 11. Prueba a instalar este monitor es alguno de sus tres sistemas. Realice capturas de pantalla del proceso de instalación y comente capturas de pantalla del programa en ejecución.[4]

Instalamos y configuramos nuestro servidor con zabbix siguiendo los pasos indicados en la cita(exceptuando el usuario de la base de datos que en nuestro caso era root).

Una vez instalado podemos acceder con el navegador:



Figura 11.1: Pantalla de login de Zabbix.

Introducimos usuario: admin y clave: zabbix, para acceder y ya tendremos la pantalla de inicio:

The screenshot displays the Zabbix web interface. At the top, there is a navigation bar with tabs for Monitoring, Inventory, Reports, Configuration, and Administration. Below this is a breadcrumb trail: Dashboard > Overview > Web > Latest data > Triggers > Events > Graphs > Screens > Maps > Discovery > IT services. A 'History' section shows the current path: Configuration of hosts > Configuration of items > Configuration of actions > Configuration of maintenance periods > Configuration of host groups. The main area is titled 'PERSONAL DASHBOARD' and contains several widgets:

- Favourite graphs:** A section with a 'No graphs added.' message and a 'Graphs »' link.
- Favourite screens:** A section with a 'No screens added.' message and a 'Screens »' link.
- Favourite maps:** A section with a 'No maps added.' message and a 'Maps »' link.
- Status of Zabbix:** A table showing system parameters and their values.
- System status:** A table showing the status of various host groups.
- Host status:** A table showing the status of individual hosts.
- Last 20 issues:** A table showing the most recent issues.
- Web monitoring:** A table showing the status of web scenarios.

Parameter	Value	Details
Zabbix server is running	No	localhost:10051
Number of hosts (monitored/not monitored/templates)	39	1 / 0 / 38
Number of items (monitored/disabled/not supported)	63	63 / 0 / 0
Number of triggers (enabled/disabled) [problem/ok]	42	42 / 0 [0 / 42]
Number of users (online)	2	2
Required server performance, new values per second	0.94	-

Host group	Disaster	High	Average	Warning	Information	Not classified
Zabbix servers	0	0	0	0	0	0

Host group	Without problems	With problems	Total
Zabbix servers	1	0	1

Host	Issue	Last change	Age	Info	Ack	Actions
No events found.						

Host group	Ok	Failed	Unknown
No web scenarios found.			

Figura 11.2: Pantalla Principal de Zabbix.

## 12. Pruebe a instalar este monitor en alguno de sus tres sistemas. Realice capturas de pantalla del proceso de instalación y comente las capturas de pantalla del programa de ejecución

Necesitamos tener apache, php5 y mysql como prerequisite, para ello ponemos:

```
sudo apt-get install apache2 php5 mysql-server phpmyadmin
```

Una vez tengamos todo los prerequisites, instalamos cacti:

```
sudo apt-get install cacti cacti-spine
```

Después de hacer toda la instalación ya ponemos ir a un navegador y poner como dirección: localhost/cacti

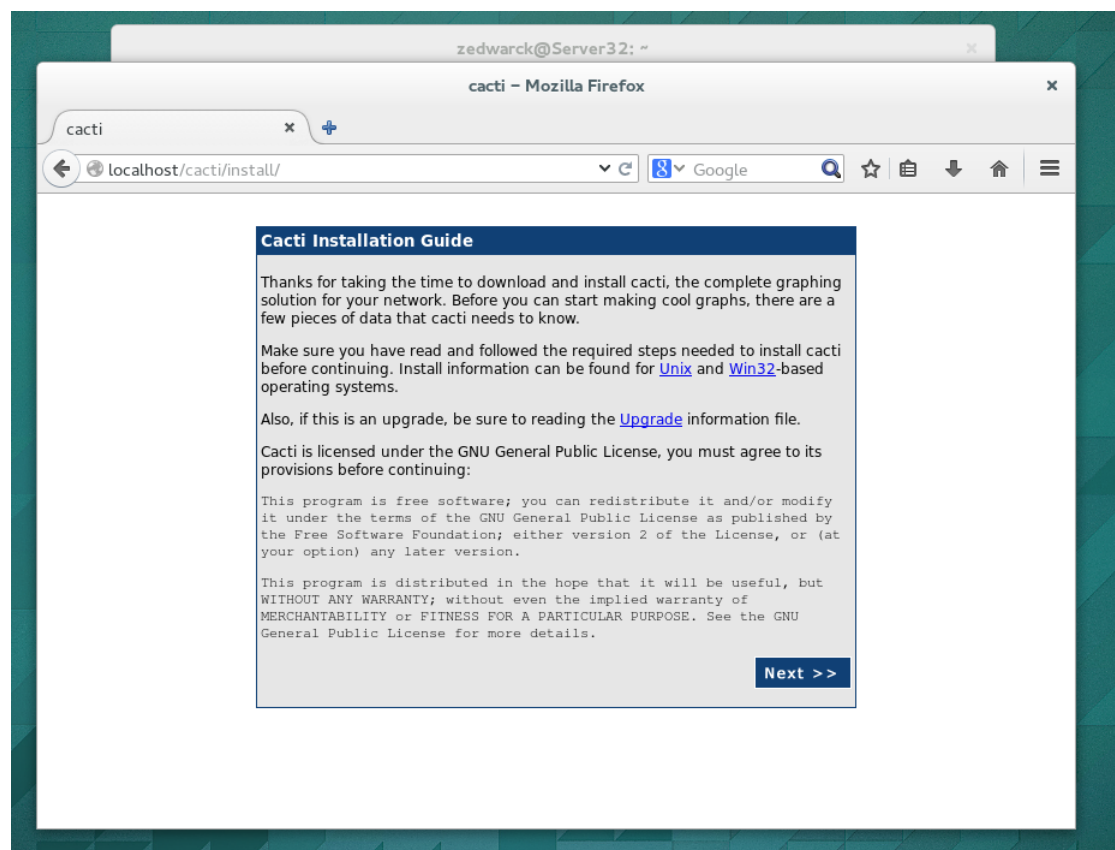


Figura 12.1: Pantalla inicial de preinstalación de Cacti.

Pulsamos a siguiente y le decimos que queremos una nueva instalación:

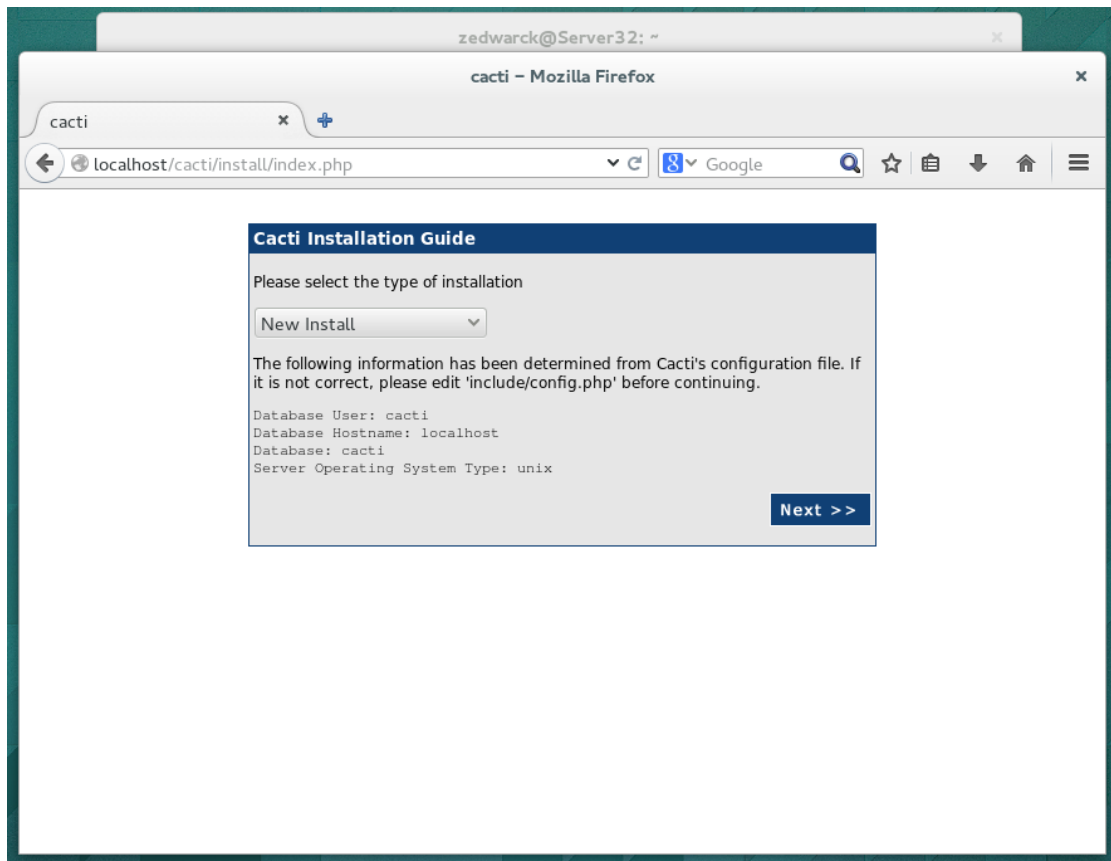


Figura 12.2: Opción de nueva instalación o actualización.

Luego el sistema comprueba que están los archivos que necesita en sus rutas adecuadas. Pulsamos terminar para completar el proceso:

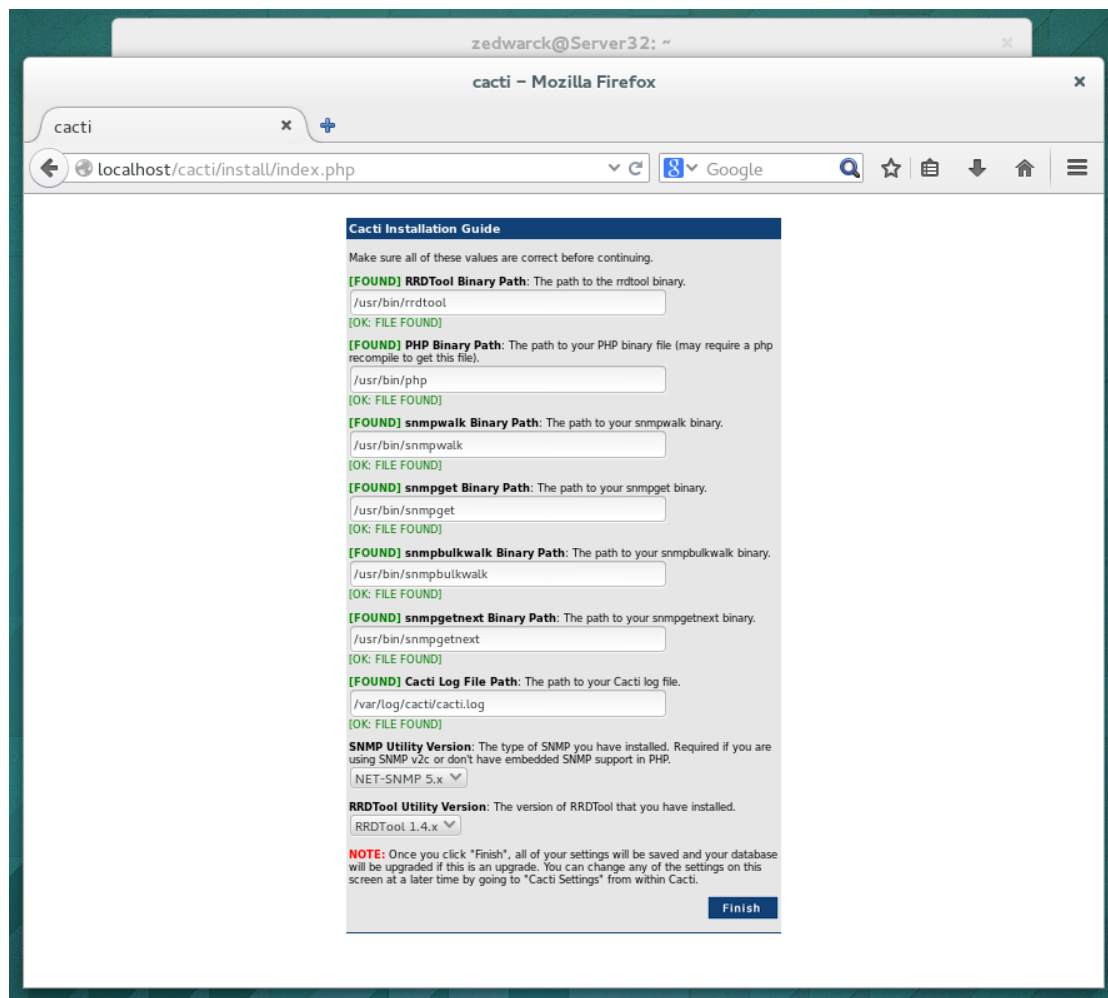


Figura 12.3: Comprobación de rutas en la instalación de cacti.

Luego nos pide un usuario y un password que le ponemos "admin admin":

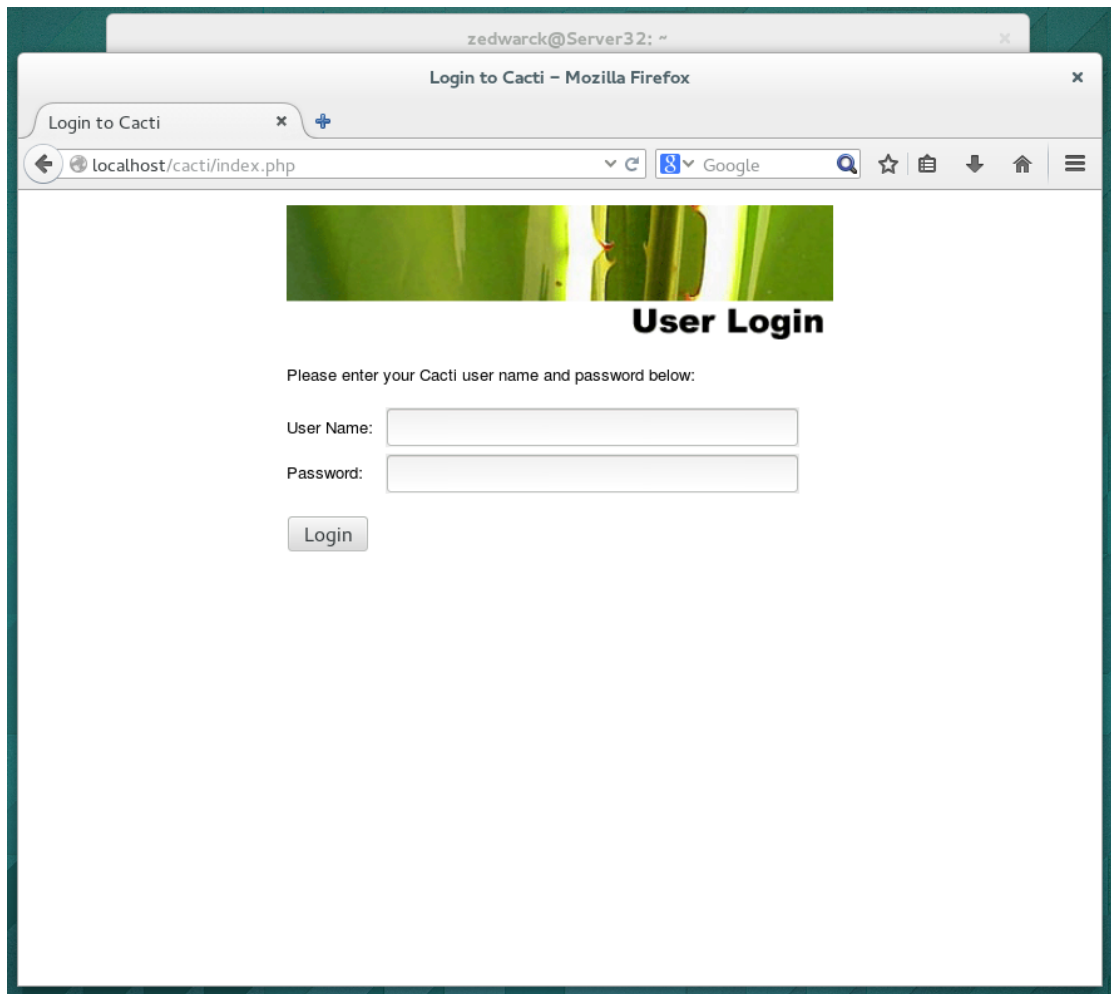


Figura 12.4: Login en cacti.

Luego nos obliga a cambiar de password:

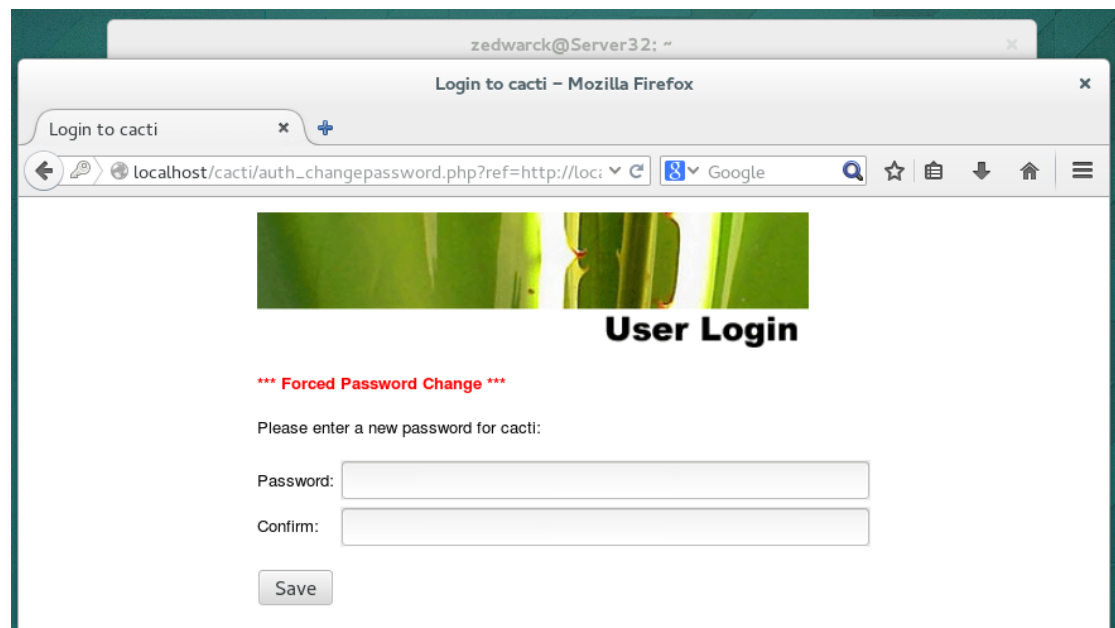


Figura 12.5: Cambio de password para admin en cacti.

Finalmente accedemos al entorno web de cacti:

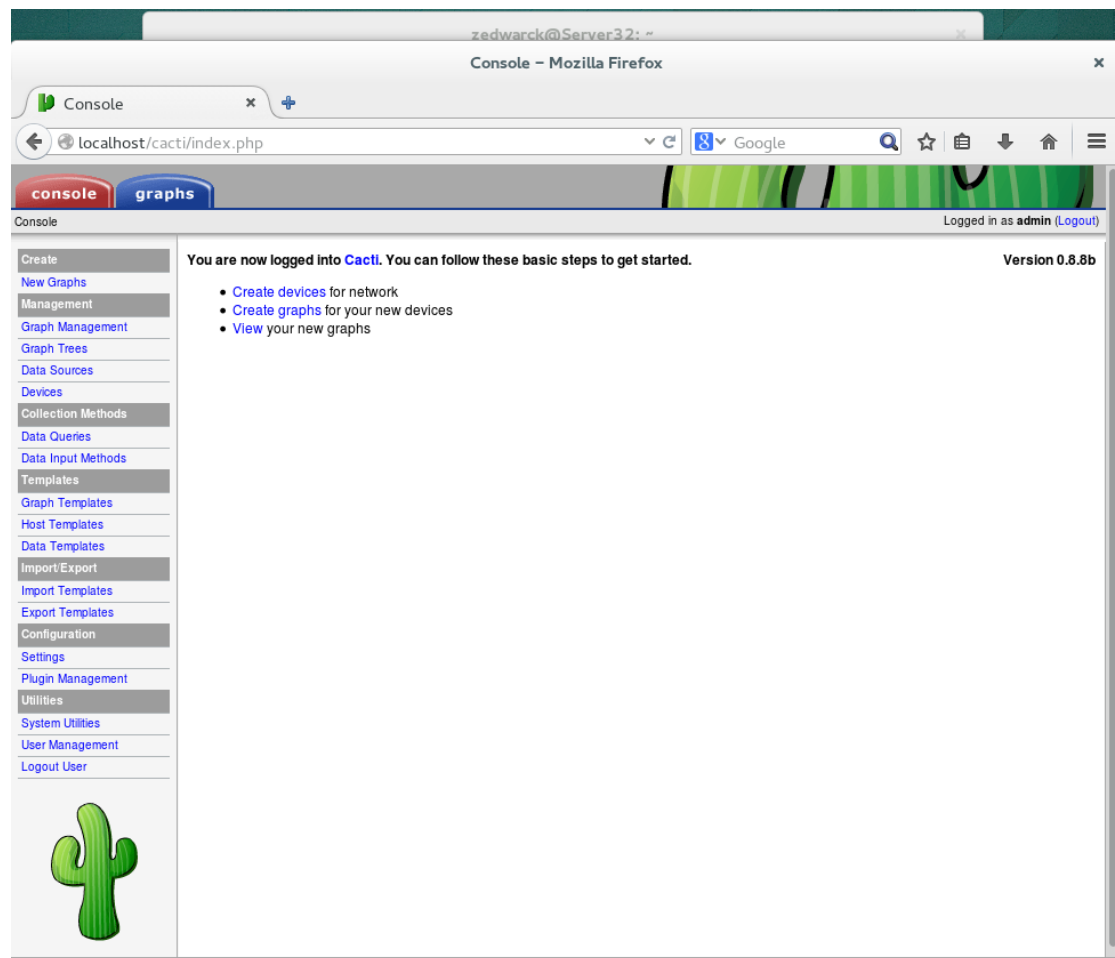
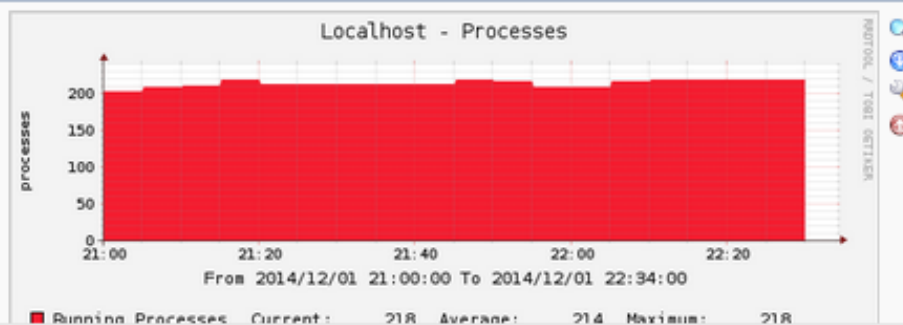
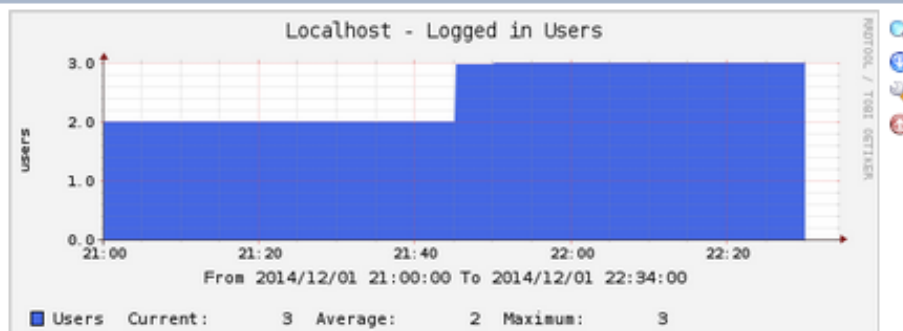
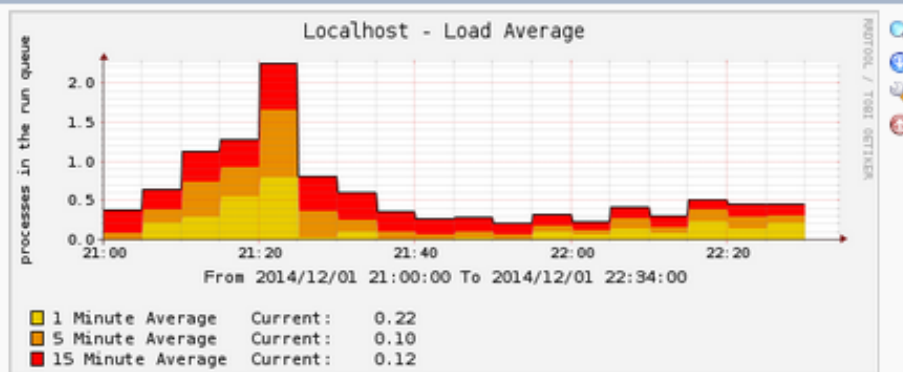
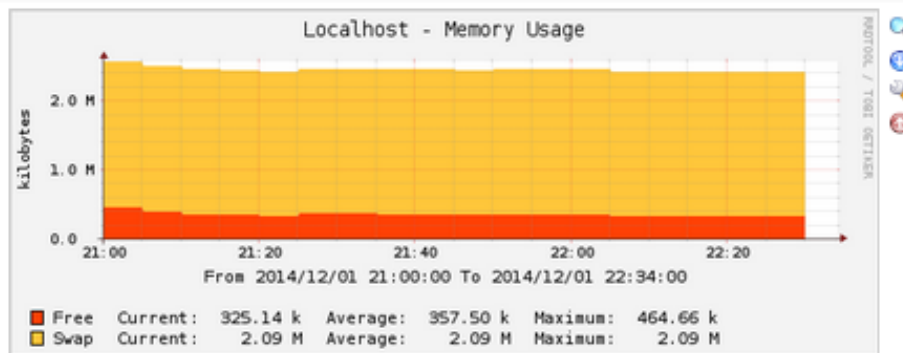


Figura 12.6: Home de cacti.

Podemos configurar parámetros de rango de tiempo o que es lo que queremos muestrear en gráficas, en nuestro caso el uso de memoria, la carga media del servidor, usuario logueados y procesos activos:





+Alt.

Figura 12.7: Gráficas de muestreo en cacti.

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

El comando strace es una utilidad que permite seguir la ejecución de un programa, su uso en principio es sencillo solo poner strace y el nombre del programa a trazar. En el caso de que la ejecución a estudiar tenga hilos podemos igualmente seguir los hijos del ejecutable mediante el modificador -fF.

Una vez visto las trazas de los hijos podemos ver más concretamente uno de ellos mediante su identificador con el modificador -p <id>

strace acepta como argumentos expresiones generadas con egrep o pgrep para poder seguir por ejemplo a un conjunto de procesos.

### 14. Desarrolle una página en C o C++ y analice su comportamiento usando valgrind. Visite <http://www.cs.tut.fi/~jkorpela/forms/cgic.html> para ver un ejemplo sencillo de una página web generada por un programa escrito en C.

Mi código:

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <time.h>
5
6  int main(int argc, char **argv){
7
8      if (argc>6 || argc<5){
9          printf("\nERROR. Numero de parametros incorrecto. USO
          : %s <fila_A> <columna_A> <fila_B> <columna_B> [-v
          ].\n", argv[0]);
10         exit(EXIT_FAILURE);
11     }
12
13     int f1 = atoi(argv[1]);
14     int c1 = atoi(argv[2]);
15     int f2 = atoi(argv[3]);
16     int c2 = atoi(argv[4]);
17
18     if(c1!=f2){
```

```

19         printf("\nERROR. Las matrices no son multiplicables.
20             Tamano incorrecto.\n");
21         exit(EXIT_FAILURE);
22     }
23
24     int i, j, k;
25     int suma;
26     srand(time(NULL));
27
28     //Se crean las matrices y se reserva la memoria
29     int **A;
30     int **B;
31     int **C;
32
33     double tiempo;
34     struct timespec tiempoI, tiempoF;
35
36     A = (int **)malloc(f1*sizeof(int *));
37     for(i=0;i<f1;i++)
38         A[i]=(int *)malloc(c2*sizeof(int));
39
40     B = (int **)malloc(f1*sizeof(int *));
41     for(i=0;i<f1;i++)
42         B[i]=(int *)malloc(c1*sizeof(int));
43
44     C = (int **)malloc(f2*sizeof(int *));
45     for(i=0;i<f2;i++)
46         C[i]=(int *)malloc(c2*sizeof(int));
47
48     //Se rellenan con enteros aleatorios del 0 al 9
49     for (i=0; i<f1; i++)
50         for (j=0; j<c1; j++)
51             B[i][j] = rand() %10;
52
53     for (i=0; i<f2; i++)
54         for (j=0; j<c2; j++)
55             C[i][j] = rand() %10;
56
57
58     //Se multiplica
59     clock_gettime(CLOCK_REALTIME, &tiempoI);
60     for (i=0; i<f1; i++)
61         for (j=0; j<c2; j++){
62             suma = 0;
63             for (k=0; k<c1; k++)
64                 suma += B[i][k] * C[k][j];
65             A[i][j] = suma;
66         }

```

```

67     clock_gettime(CLOCK_REALTIME, &tiempoF);
68     //Se muestra por pantalla la Solucion Completa
69     if (argc==6){
70         if (strcmp("-v",argv[5])==0){
71             for (i=0; i<f1; i++){
72                 printf("\n");
73                 for (j=0; j<c2; j++){
74                     if (A[i][j]>=1000)
75                         printf(" %d",A[i][j]);
76                     else if (A[i][j]>=100)
77                         printf(" %d",A[i][j])
78                         ;
79                     else if (A[i][j]>=10)
80                         printf(" %d",A[i][j]
81                             );
82                     else
83                         printf(" %d",A[i][j]
84                             );
85                 }
86             }
87             printf("\n");
88             }
89         else{
90             tiempo = (double) (tiempoF.tv_sec - tiempoI.tv_sec) +
91                 (double) ((tiempoF.tv_nsec - tiempoI.tv_nsec) /
92                     (1.e+9));
93
94             printf("Tiempo: %8.6f\n",tiempo);
95             printf("Componente(0,0): %d\n",A[0][0]);
96             printf("Componente(N-1,N-1): %d\n",A[f1-1][c2-1]);
97         }
98     }
99     //Se libera la memoria
100     for(i=0;i<f1;i++)
101         free(A[i]);
102     free(A);
103
104     for(i=0;i<f1;i++)
105         free(B[i]);
106     free(B);
107
108     for(i=0;i<f2;i++)
109         free(C[i]);
110     free(C);

```

```

111
112     return (EXIT_SUCCESS);
113 }

```

y compilamos con:

```
$ gcc -o matrices matrices.c
```

Y finalmente ejecutamos para ver el resultado de valgrind con los parametros adecuados:

```

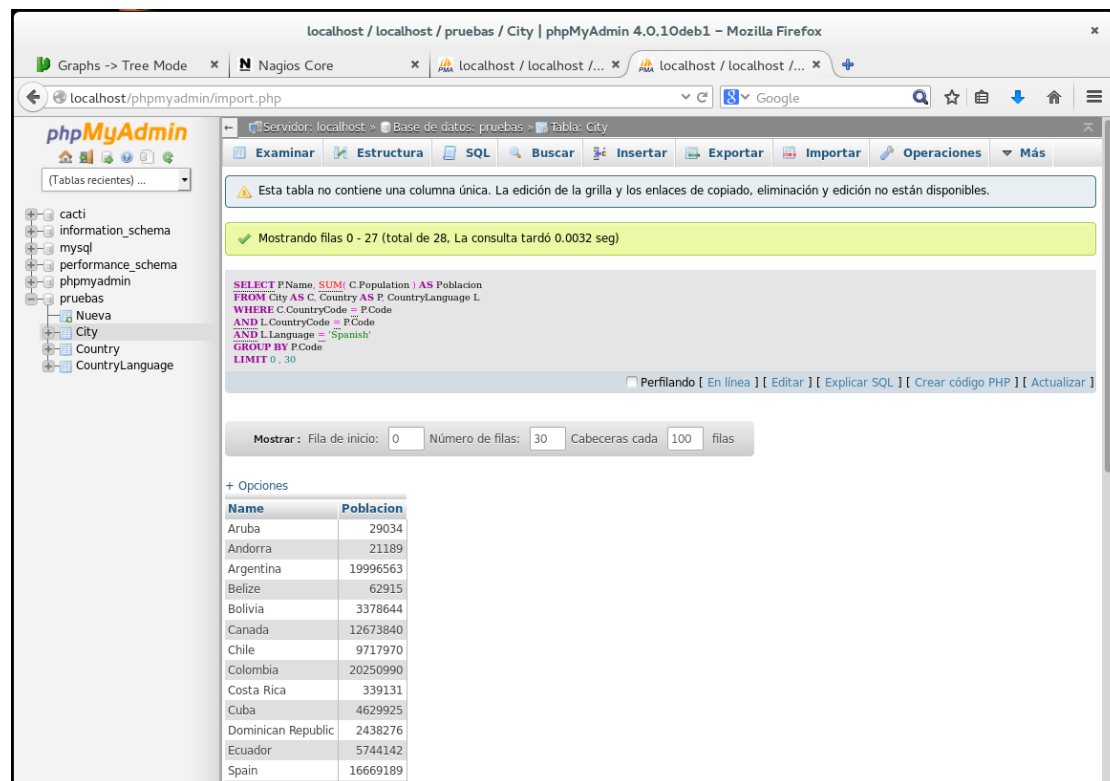
root@Server32: ~
File Edit View Search Terminal Help
root@Server32:~# valgrind --leak-check=yes ./matrices 10 10 10 10 -v
==11679== Memcheck, a memory error detector
==11679== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==11679== Using Valgrind-3.10.0.SVN and LibVEX; rerun with -h for copyright info
==11679== Command: ./matrices 10 10 10 10 -v
==11679==
278 317 249 214 243 275 260 259 213 247
304 297 268 209 274 264 194 239 255 237
125 129 176 92 138 186 81 99 108 101
249 282 228 178 222 194 154 173 218 216
232 154 220 179 155 217 162 231 218 106
253 269 201 162 199 171 177 211 211 213
185 92 153 164 117 136 131 169 169 70
123 126 93 81 94 99 67 59 114 106
182 209 184 165 174 228 163 183 174 135
235 235 177 253 211 246 268 281 192 161
==11679==
==11679== HEAP SUMMARY:
==11679==    in use at exit: 0 bytes in 0 blocks
==11679==   total heap usage: 33 allocs, 33 frees, 1,320 bytes allocated
==11679==
==11679== All heap blocks were freed -- no leaks are possible
==11679==
==11679== For counts of detected and suppressed errors, rerun with: -v
==11679== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
root@Server32:~#

```

Figura 14.1: Resultado en consola de valgrind sobre un programa de multiplicación de matrices.

## 15. 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).

A partir de una base de datos de prueba de poblaciones, países y lenguas en el mundo he creado una consulta la cual esperaba que tardara algo más de lo normal para consultas sencillas. En este caso he sumado la población que habla español en el mundo agrupados por países y he sacado una gráfica. Podemos ver que nos muestra cuanto a tardado, en nuestro caso 0.0032 segundos en realizar toda la consulta.



The screenshot shows the phpMyAdmin interface in a Mozilla Firefox browser. The browser address bar shows 'localhost/phpmyadmin/import.php'. The phpMyAdmin interface is in 'Tree Mode' and shows a database named 'pruebas'. The 'City' table is selected. The SQL query is displayed in the 'SQL' tab, and the result is shown in a table format. The query execution time is 0.0032 seconds.

Mostrando filas 0 - 27 (total de 28. La consulta tardó 0.0032 seg)

```
SELECT P.Name, SUM(C.Population) AS Poblacion
FROM City AS C, Country AS P, CountryLanguage AS L
WHERE C.CountryCode = P.Code
AND L.CountryCode = P.Code
AND L.Language = 'Spanish'
GROUP BY P.Code
LIMIT 0, 30
```

Mostrar : Fila de inicio: 0 Número de filas: 30 Cabeceras cada 100 filas

Name	Poblacion
Aruba	29034
Andorra	21189
Argentina	19996563
Belize	62915
Bolivia	3378644
Canada	12673840
Chile	9717970
Colombia	20250990
Costa Rica	339131
Cuba	4629925
Dominican Republic	2438276
Ecuador	5744142
Spain	16669189

Figura 15.1: Resultado de la consulta en tabla y el tiempo que a tardado en realizarla.

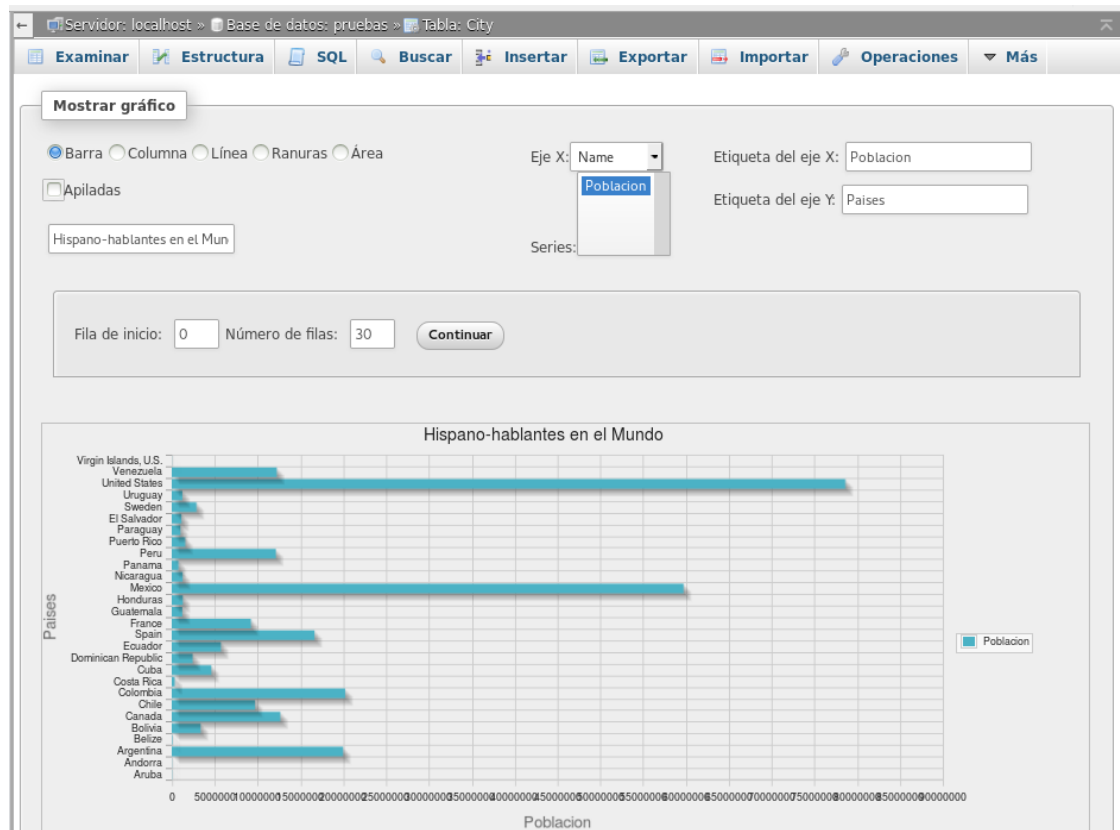


Figura 15.2: Resultado de la consulta en formato gráfico.

## Referencias

- [1] HowToForge. [http://www.howtoforge.com/replacing\\_hard\\_disks\\_in\\_a\\_raid1\\_array](http://www.howtoforge.com/replacing_hard_disks_in_a_raid1_array).
- [2] Debian. <http://debian-handbook.info/browse/es-ES/stable/sect.task-scheduling-cron-atd.html>.
- [3] Nagios. [http://assets.nagios.com/downloads/nagioscore/docs/Installing\\_Nagios\\_Core\\_From\\_Source.pdf](http://assets.nagios.com/downloads/nagioscore/docs/Installing_Nagios_Core_From_Source.pdf).
- [4] thedutchlab.com. <https://thedutchlab.com/en/news/installing-zabbix-on-ubuntu-14-04>.