

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

Memoria Cuestiones Opcionales

Ignacio Romero Cabrerizo

18 de enero de 2015

Índice

1. Cuestión Opcional 1: Muestre (con capturas de pantalla) cómo ha comprobado que el RAID1 funciona.	4
2. Cuestión Opcional 2. ¿Qué relación hay entre los atajos de teclado de emacs y los de la consola bash? ¿y entre los de vi y las páginas del manual?	5
3. Cuestión Opcional 3. ¿Qué gestores utiliza OpenSuse?	5
4. Cuestión Opcional 4. Instale el servicio (fail2ban) y pruebe su funcionamiento.	5
5. Cuestión opcional 5: Muestre un ejemplo de uso para awk	7
6. Cuestión opcional 6: ZABBIX: 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.	8
7. Cuestión opcional 7: instale Nagios en su sistema (el que prefiera) documentando el proceso y muestre el resultado de la monitorización de su sistema comentando qué aparece	11
8. Cuestión Opcional 8. Haga lo mismo con Munin	13
9. Cuestión Opcional 9. Instale el monitor (AWSTATS) y muestre y comente algunas capturas de pantalla	13
10.Cuestión Opcional 10. Seleccione, instale y ejecute uno, comente los resultados.	15
11.Cuestión Opcional 11: Lea el artículo y elabore un breve resumen.	16
12.Cuestión Opcional 12: Seleccione un benchmark entre SisoftSandra y Aida. Ejecútelo y muestre capturas de pantalla comentando los resultados.	17
13.Cuestión opcional 13: Realice lo mismo que en la cuestión 8 pero para otro servicio.	20

Índice de figuras

1.1.	Check RAID mediante cat	4
1.2.	Check RAID mediante mdadm	4
4.1.	Configuración de ssh en Fail2ban	6
4.2.	Cambio de IP en Fail2ban	6
4.3.	IP baneada tras 4 intentos fallidos en ssh por Fail2ban	6
5.1.	Ejemplo de ejecución awk	7
5.2.	Ejemplo 2 de ejecución awk	7
6.1.	Obtención de paquetes y dpkg del fichero deb	8
6.2.	Configuración Zabbix-mysql	8
6.3.	Configuración contraseña Zabbix	9
6.4.	Configuración Zabbix	9
6.5.	Zabbix: Pre-requisitos	10
6.6.	Zabbix: Base datos y usuario	10
6.7.	Zabbix: Página final con los datos del servidor	11
7.1.	Monitorización con Nagios	12
8.1.	Monitorización con Ganglia	13
9.1.	Archivo awstats.conf	14
9.2.	Configuración de awstats	14
9.3.	Archivo Default de apache	14
9.4.	Funcionamiento Awstats	15
10.1.	Instalación Benchmark para Memoria RAM	15
10.2.	Prueba del benchmark para Memoria RAM	16
10.3.	Resultados benchmark	16
12.1.	AIDA Software	18
12.2.	Benchmark de Memoria y CPU	18
12.3.	CPUID benchmark	19
12.4.	AIDA: Estabilidad del sistema	20
13.1.	Desfragmentación del disco en Windows Server	21

CUESTIONES OPCIONALES

1. Cuestión Opcional 1: Muestre (con capturas de pantalla) cómo ha comprobado que el RAID1 funciona.

Para comprobar el correcto funcionamiento del RAID1 configurado en UbuntuServer podemos hacerlo mirando con:

```
cat /proc/mdstat
```

```
machorc@ubuntuS:~$ cat /proc/mdstat
Personalities : [raid1] [linear] [multipath] [raid0] [raid6] [raid5] [raid4] [raid10]
md0 : active raid1 sdb1[1] sda1[0]
      10475392 blocks super 1.2 [2/2] [UU]

unused devices: <none>
```

Figura 1.1: Check RAID mediante cat

También podemos verificar el estado mediante varios comandos siendo root o superusuario(ambos muestran la misma configuración):

```
mdadm --detail /dev/md0
```

```
mdadm -D /dev/md0
```

```
root@ubuntuS:/dev# mdadm --detail /dev/md0
/dev/md0:
      Version : 1.2
      Creation Time : Wed Oct  8 16:53:51 2014
      Raid Level : raid1
      Array Size : 10475392 (9.99 GiB 10.73 GB)
      Used Dev Size : 10475392 (9.99 GiB 10.73 GB)
      Raid Devices : 2
      Total Devices : 2
      Persistence : Superblock is persistent

      Update Time : Sun Jan 18 15:30:14 2015
      State : clean
      Active Devices : 2
      Working Devices : 2
      Failed Devices : 0
      Spare Devices : 0

      Name : ubuntuS:0 (local to host ubuntuS)
      UUID : 36aae815:3e504b96:7268c3a0:72f36e18
      Events : 41

      Number  Major  Minor  RaidDevice State
            0      8        1          0    active sync   /dev/sda1
            1      8       17          1    active sync   /dev/sdb1
```

Figura 1.2: Check RAID mediante mdadm

2. Cuestión Opcional 2. ¿Qué relación hay entre los atajos de teclado de emacs y los de la consola bash? ¿y entre los de vi y las páginas del manual?

Emacs¹ es un editor de texto perteneciente al proyecto GNU, por lo que tanto bash como emacs comparten atajos de teclas en su uso.

Vi es un editor de texto en consola para sistemas basados en UNIX cuya herramienta para editar el contenido es mediante el modo comandos o inserción de texto. Tiene los mismos atajos de teclado que el manual exceptuando los comandos para la modificación.

3. Cuestión Opcional 3. ¿Qué gestores utiliza OpenSuse?

OpenSuse utiliza para gestionar los paquetes 2 herramientas dentro de RPM: Con gestor gráfico disponemos de **YaST** y por línea de comandos en el terminal **Zypper**.

Ejemplos de uso Zypper:²

```
# zypper search nombre_del_paquete  
# zypper install nombre_del_paquete
```

install se puede sustituir por «in»:

```
#zypper in nombre.rpm
```

4. Cuestión Opcional 4. Instale el servicio (fail2ban) y pruebe su funcionamiento.

Instalamos el servicio:

```
sudo apt-get install fail2ban
```

Pasamos a configurar varios de sus parámetros:

```
sudo nano /etc/fail2ban/jail.conf
```

En este caso, configuramos Fail2ban³ para vetar el acceso durante 60 segundos a una IP tras 3 intentos fallidos de contraseña en ssh:

¹<http://www.cs.colostate.edu/helpdocs/emacs.html>

²https://es.opensuse.org/Zypper#Manejo_de_paquetes

³http://www.fail2ban.org/wiki/index.php/MANUAL_0_8#Jail_Options

```

#
# in /etc/fail2ban/jail.local.
#
# Optionally you may override any other parameter (e.g. banaction,
# action, port, logpath, etc) in that section within jail.local

[ssh]

enabled  = true
port      = ssh
filter    = sshd
logpath   = /var/log/auth.log
maxretry  = 3
findtime  = 600
bantime  = 60

```

Figura 4.1: Configuración de ssh en Fail2ban

Podemos añadir también IPs a Fail2ban para evitar conexiones o que notifique al email de un intento de conexión no autorizado:

```

GNU nano 2.2.6          Archivo: /etc/fail2ban/jail.conf

[DEFAULT]
# "ignoreip" can be an IP address, a CIDR mask or a DNS host. Fail2ban will not
# ban a host which matches an address in this list. Several addresses can be
# defined using space separator.
ignoreip = 127.0.0.1/8 ←

# "bantime" is the number of seconds that a host is banned.
bantime = 600

# A host is banned if it has generated "maxretry" during the last "findtime"
# seconds.
findtime = 600
maxretry = 3

# Destination email address used solely for the interpolations in
# jail.{conf,local} configuration files.
destemail = nachorc@correo.ugr.es

```

(a) Cambio de IP en Fail2ban

(b) Configuración aviso a email

Figura 4.2: Cambio de IP en Fail2ban

Comprobamos el ban tras los intentos fallidos y que tras 60 segundos podemos de nuevo realizar el ssh correctamente:

```

nachorc@nachorc-Parallels-Virtual-Platform:~$ ssh nachorc@10.211.55.25
ssh: connect to host 10.211.55.25 port 22: Connection refused
nachorc@nachorc-Parallels-Virtual-Platform:~$ ssh nachorc@10.211.55.25
nachorc@10.211.55.25's password:
Welcome to Ubuntu 12.04.5 LTS (GNU/Linux 3.13.0-39-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

New release '14.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2017.

Last login: Sun Nov  9 12:54:41 2014 from nachorc-parallels-virtual-platform.local
nachorc@nachorc-Parallels-Virtual-Platform:~$ 

```

Figura 4.3: IP baneada tras 4 intentos fallidos en ssh por Fail2ban

5. Cuestión opcional 5: Muestre un ejemplo de uso para awk

El lenguaje awk en linux permite procesar (secuencialmente) uno o varios archivos basados en texto. También se considera a **awk** como un programa de UNIX que interpreta programas escritos en AWK.

Un ejemplo de uso de awk es el siguiente:

```
awk '{ print }' /etc/passwd
```

```
root@ubuntuS:~# awk '{ print }' /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
libuuid:x:100:101::/var/lib/libuuid:
syslog:x:101:104::/home/syslog:/bin/false
messagebus:x:102:106::/var/run/dbus:/bin/false
landscape:x:103:109::/var/lib/landscape:/bin/false
nachorc:x:1000:1000:nacho,,,,:/home/nachorc:/bin/bash
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
```

Figura 5.1: Ejemplo de ejecución awk

En el ejemplo anterior se muestra el contenido de cada línea de passwd en orden, salida idéntica al comando **cat /etc/passwd**.

Otro ejemplo de uso puede ser el siguiente:

```
awk '{print $1}' fichero
```

```
root@ubuntuS:~# awk '{ print }' fichero
Hola
Documento de Prueba
Línea 3
root@ubuntuS:~# awk '{ print $1}' fichero
Hola
Documento
Línea
```

Figura 5.2: Ejemplo 2 de ejecución awk

En este ejemplo se muestra en orden el contenido del archivo «fichero» en una primera instancia, y añadiendo «\$1» se muestra solamente la primera columna del archivo de texto.

6. Cuestión opcional 6: ZABBIX: 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.

Preparamos el sistema para instalar zabbix y los componentes necesarios para su funcionamiento:

```
sudo wget http://repo.zabbix.com/zabbix/2.2/ubuntu/pool/main/z/zabbix-release/zabbix-release_2.2-1+trusty_all.deb  
sudo dpkg -i zabbix-release_2.2-1+trusty_all.deb  
sudo apt-get update
```

Configuramos Zabbix en su proceso de instalación:

```
nachorc@ubuntu:~$ sudo wget http://repo.zabbix.com/zabbix/2.2/ubuntu/pool/main/z/zabbix-release/zab  
bix-release_2.2-1+trusty_all.deb  
--2015-01-18 17:06:49-- http://repo.zabbix.com/zabbix/2.2/ubuntu/pool/main/z/zabbix-release/zabbix-  
release_2.2-1+trusty_all.deb  
Resolviendo repo.zabbix.com (repo.zabbix.com)... 87.110.183.174  
Conectando con repo.zabbix.com (repo.zabbix.com)[87.110.183.174]:80... conectado.  
Petición HTTP enviada, esperando respuesta... 200 OK  
Longitud: 2656 (2,6K) [application/x-debian-package]  
Grabando a: "zabbix-release_2.2-1+trusty_all.deb"  
  
100%[=====] 2.656 --.-K/s en 0s  
  
2015-01-18 17:06:49 (71,5 MB/s) - "zabbix-release_2.2-1+trusty_all.deb" guardado [2656/2656]  
  
nachorc@ubuntu:~$ sudo dpkg -i zabbix-release_2.2-1+trusty_all.deb  
Seleccionando el paquete zabbix-release previamente no seleccionado.  
(Leyendo la base de datos ... 58908 ficheros o directorios instalados actualmente.)  
Preparing to unpack zabbix-release_2.2-1+trusty_all.deb ...  
Unpacking zabbix-release (2.2-1+trusty) ...  
Configurando zabbix-release (2.2-1+trusty) ...
```

Figura 6.1: Obtención de paquetes y dpkg del fichero deb

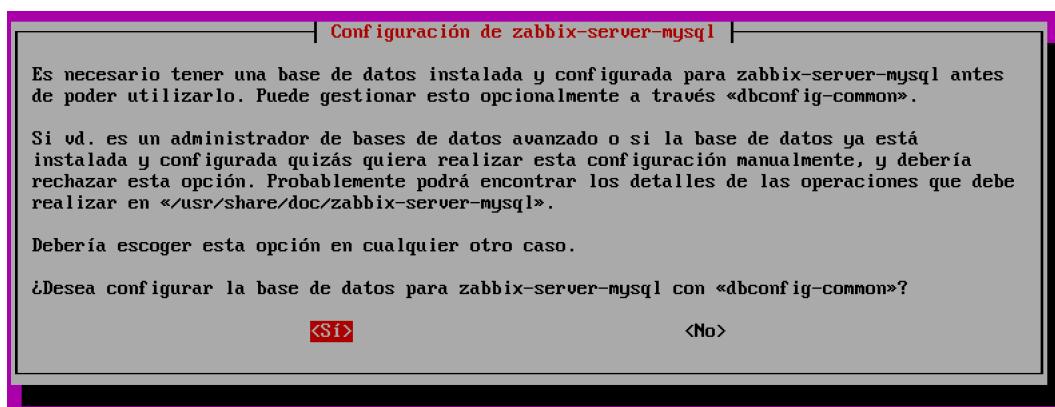


Figura 6.2: Configuración Zabbix-mysql



Figura 6.3: Configuración contraseña Zabbix

Copiamos la configuración de apache de zabbix al directorio apache2, editamos el archivo zabbix.conf cambiando la zona horaria y reiniciamos apache:

```
sudo cp /etc/zabbix/apache.conf /etc/apache2/sites-enabled/zabbix.conf
sudo vi /etc/apache2/sites-enabled/zabbix.conf
sudo service apache2 restart
```

```
GNU nano 2.2.6          Archivo: /etc/apache2/sites-enabled/zabbix.conf

# Define /zabbix alias, this is the default
<IfModule mod_alias.c>
    Alias /zabbix /usr/share/zabbix
</IfModule>

<Directory "/usr/share/zabbix">
    Options FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all

    php_value max_execution_time 300
    php_value memory_limit 128M
    php_value post_max_size 16M
    php_value upload_max_filesize 2M
    php_value max_input_time 300
    php_value date.timezone Europe/Madrid
</Directory>
```

Figura 6.4: Configuración Zabbix

Configuramos Zabbix entrando en el navegador con nuestra IP/zabbix:

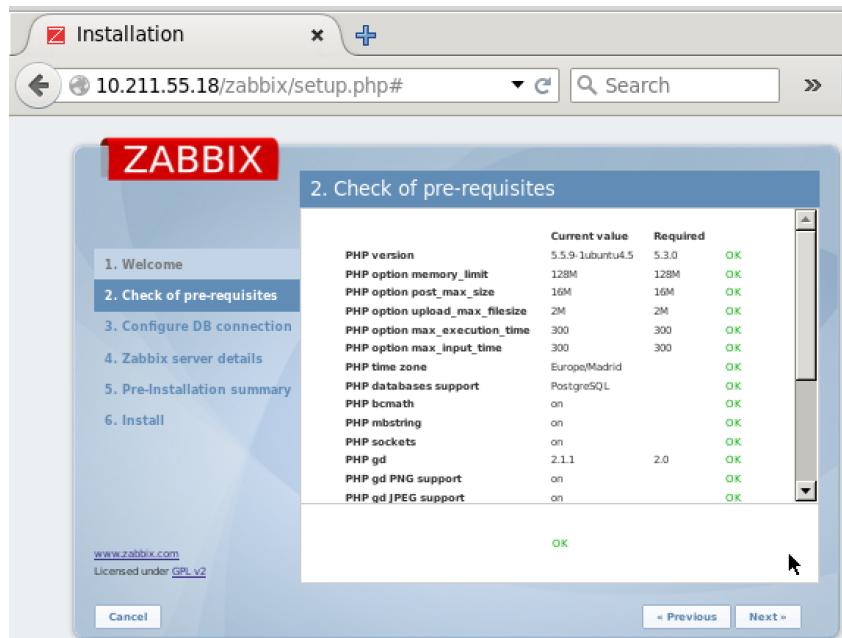


Figura 6.5: Zabbix: Pre-requisitos

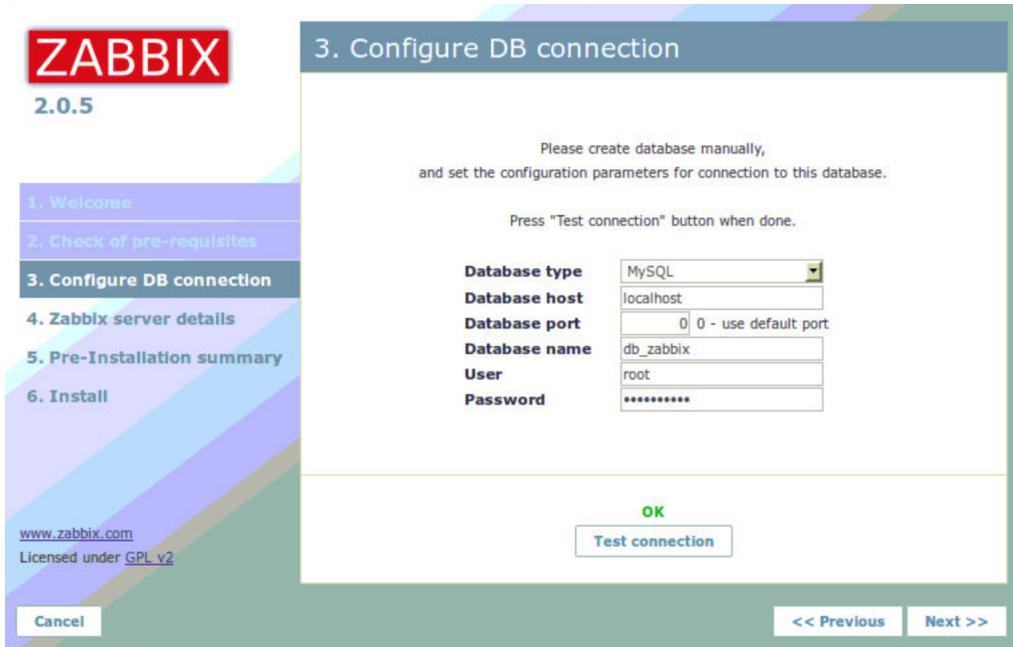


Figura 6.6: Zabbix: Base datos y usuario

The screenshot shows the Zabbix personal dashboard. At the top, there's a navigation bar with links like Monitoring, Inventory, Reports, Configuration, Administration, Dashboard, Overview, Web, Latest data, Triggers, Events, Graphs, Screens, Maps, Discovery, and IT services. Below the navigation is a breadcrumb trail: History: Dashboard > Overview. The main area is titled 'PERSONAL DASHBOARD' and contains three sections: 'Favourite graphs' (No graphs added), 'Favourite screens' (No screens added), and 'Favourite maps' (No maps added). To the right, there's a large table titled 'Status of Zabbix' with columns for Parameter, Value, and Details. The table includes rows for the Zabbix server being running (Value: Yes, Details: 10.211.55.18), number of hosts (39, 0 / 1 / 38), number of items (0, 0 / 0 / 0), number of triggers (0, 0 / 0 [0 / 0]), number of users (2, 2), and required server performance (0, -).

Figura 6.7: Zabbix: Página final con los datos del servidor

7. Cuestión opcional 7: instale Nagios en su sistema (el que prefiera) documentando el proceso y muestre el resultado de la monitorización de su sistema comentando qué aparece

Para instalar y configurar Nagios seguimos los siguientes pasos:

```
sudo -s
apt-get install wget build-essential apache2 php5 openssl perl make php5-gd
libgd2-xpm libgd2-xpm-dev
```

```
./configure --with-command-group=nagcmd
make all
make install
make install-init
make install-config
make install-commandmode
make install-webconf
```

Creamos en el sistema un usuario y grupo específico para Nagios:

```
useradd nagios
groupadd nagcmd
usermod -a -G nagcmd nagios
htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

Instalamos los plugins descargados de su web necesarios para que Nagios funcione correctamente:

```
./configure --with-nagios-user=nagios --with-nagios-group=nagios
make
make install
```

Reiniciamos el servicio apache e iniciamos nagios:

```
/etc/init.d/apache2 restart  
service nagios start
```

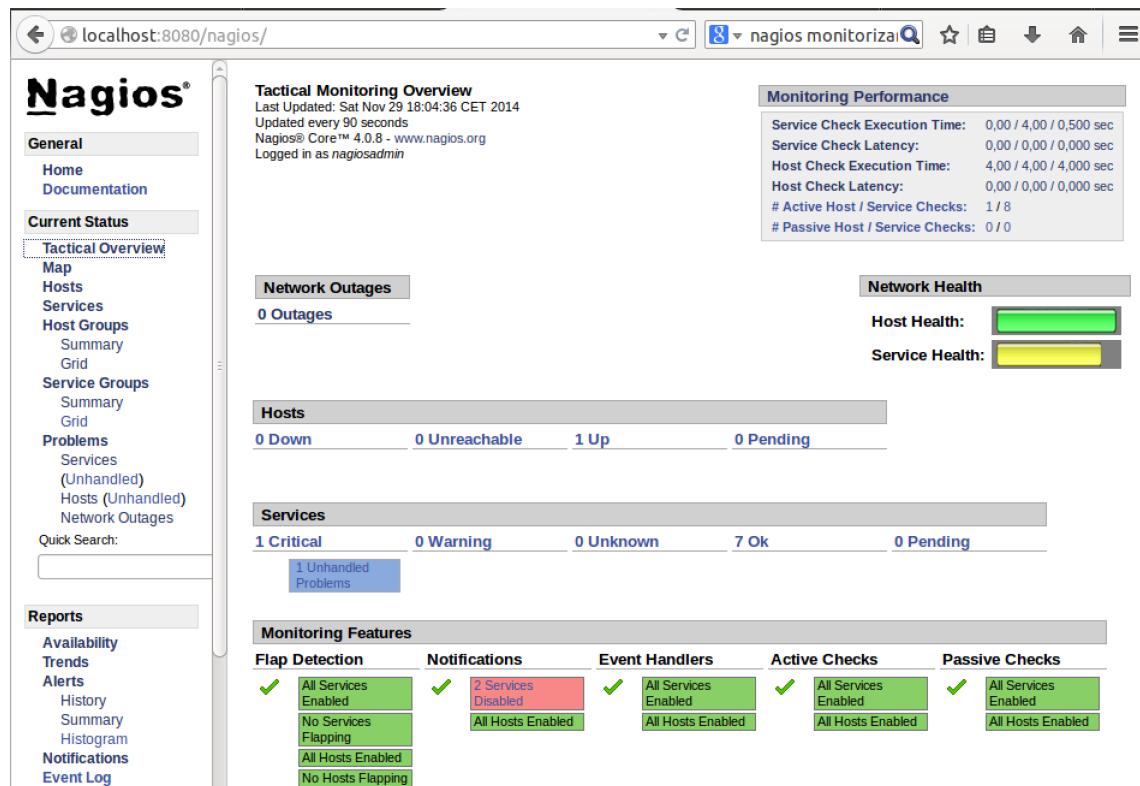


Figura 7.1: Monitorización con Nagios

Tras realizar el login en Nagios obtenemos toda la información del sistema, de los servicios, los Hosts y las redes actuales.

En la parte derecha se indica la Vida de Hosts y Servicios así como los tiempos de ejecución de cada uno.

8. Cuestión Opcional 8. Haga lo mismo con Munin

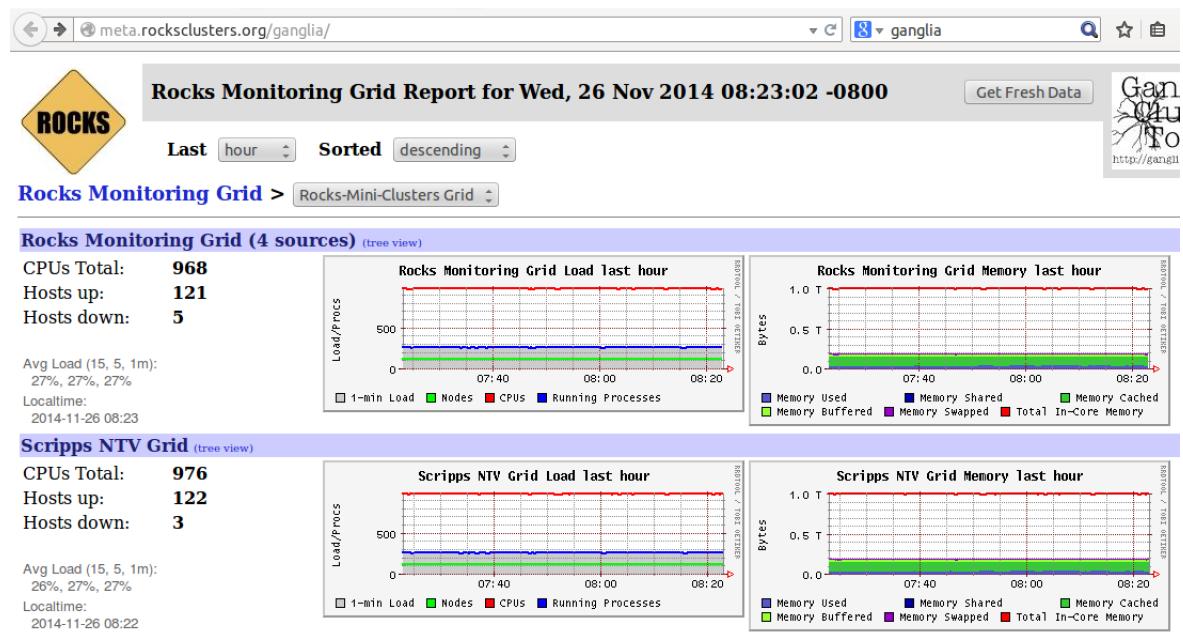


Figura 8.1: Monitorización con Ganglia

En los gráficos podemos observar una demo de los tiempos de ejecución de los procesos y Bytes consumidos por los Nodos, CPU y Memoria usada y compartida en la última hora sobre Rocks Grid Clusters.

9. Cuestión Opcional 9. Instale el monitor (AWSTATS) y muestre y comente algunas capturas de pantalla

```
sudo apt-get install awstats
```

```
sudo apt-get install awstats
```

Realizamos una copia seguridad del archivo de configuración:

```
sudo cp /etc/awstats/awstats.conf /etc/awstats/awstats.bk.com.conf
```

Configuramos el parámetro **SiteDomain** en el archivo de configuración anterior:

```
SiteDomain="localhost"
```

```
sudo /usr/lib/cgi-bin/awstats.pl -config=domain.com -update
```

```

GNU nano 2.2.6                               Archivo: awstats.conf

# server name, used to reach the web site.
# If you share the same log file for several virtual web servers, this
# parameter is used to tell AWStats to filter record that contains records for
# this virtual host name only (So check that this virtual hostname can be
# found in your log file and use a personalized log format that include the
# %virtualname tag).
# But for multi hosting a better solution is to have one log file for each
# virtual web server. In this case, this parameter is only used to generate
# full URL's links when ShowLinksOnUrl option is set to 1.
# If analyzing mail log, enter here the domain name of mail server.
# Example: "myintranetserver"
# Example: "www.domain.com"
# Example: "ftp.domain.com"
# Example: "domain.com"
#
SiteDomain="localhost"

```

Figura 9.1: Archivo awstats.conf

```

nachorc@nachorc-Parallels-Virtual-Platform:~$ sudo /usr/lib/cgi-bin/awstats.pl -config=domain.com -update
Create/Update database for config "/etc/awstats/awstats.conf" by AWStats version 7.0 (build 1.971)
From data in log file "/var/log/apache2/access.log"...
Phase 1 : First bypass old records, searching new record...
Searching new records from beginning of log file...
Jumped lines in file: 0
Parsed lines in file: 0
Found 0 dropped records,
Found 0 comments,
Found 0 blank records,
Found 0 corrupted records,
Found 0 old records,
Found 0 new qualified records.

```

Figura 9.2: Configuración de awstats

Editamos el archivo /etc/apache2/sites-available/default (o 000-default) y añadimos lo siguiente:

```

GNU nano 2.2.6                               Archivo: default

        CustomLog ${APACHE_LOG_DIR}/access.log combined

        Alias /doc/ "/usr/share/doc/"
        <Directory "/usr/share/doc/">
                Options Indexes MultiViews FollowSymLinks
                AllowOverride None
                Order deny,allow
                Deny from all
                Allow from 127.0.0.0/255.0.0.0 ::1/128
        </Directory>

        Alias /awstatsclasses "/usr/share/awstats/lib/"
        Alias /awstats-icon "/usr/share/awstats/icon/"
        Alias /awstatscss "/usr/share/doc/awstats/examples/css"
        ScriptAlias /awstats/ /usr/lib/cgi-bin/
        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch

</VirtualHost>

```

Figura 9.3: Archivo Default de apache

```
service apache2 restart
```

Entramos a Awstats (en la dirección y puerto configurados en apache), en nuestro caso:
<http://localhost:8080/awstats/awstats.pl>

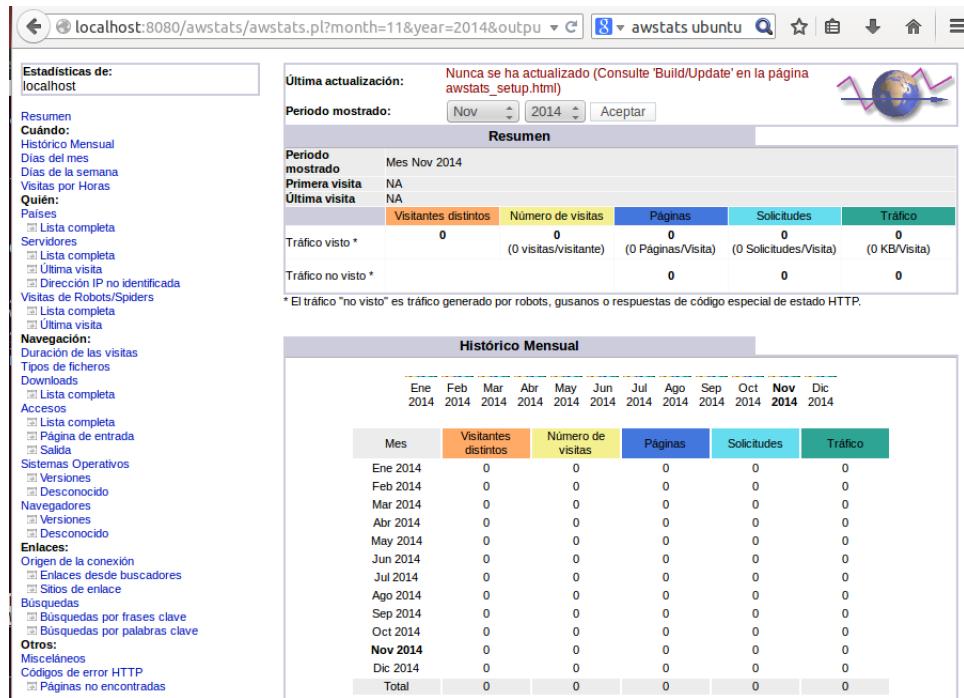


Figura 9.4: Funcionamiento Awstats

10. Cuestión Opcional 10. Seleccione, instale y ejecute uno, comente los resultados.

Podemos instalar cualquiera de los paquetes listados mediante el comando

```
phoronix-test-suite install "nombre test"
```

```
Phoronix Test Suite v5.2.1
To Install: pts/ramspeed-1.4.0

Determining File Requirements .....
Searching Download Caches .....

1 Test To Install
 1 File To Download [0.08MB]
 1MB Of Disk Space Is Needed

pts/ramspeed-1.4.0:
  Test Installation 1 of 1
  1 File Needed [0.08 MB]
  Downloading: ramsp-3.5.0.tar.gz
  Downloading .....
  Installation Size: 0.72 MB
  Installing Test @ 13:06:19 [0.08MB]
```

Figura 10.1: Instalación Benchmark para Memoria RAM

Para ejecutar el test seleccionado podemos hacerlo de la siguiente forma:

```
phoronix-test-suite benchmark "nombre test"
```

```
nachorc@nachorc-Parallels-Virtual-Platform:~$ phoronix-test-suite benchmark pts/ramspeed
Phoronix Test Suite v5.2.1

Installed: pts/ramspeed-1.4.0

RAMspeed SMP 3.5.0:
pts/ramspeed-1.4.0
Memory Test Configuration
 1: Copy
 2: Scale
 3: Add
 4: Triad
 5: Average
 6: Test All Options
Type: 5

 1: Integer
 2: Floating Point
 3: Test All Options
Benchmark: 1
```

Figura 10.2: Prueba del benchmark para Memoria RAM

Los resultados tras ejecutar el test o benchmark son los siguientes:

```
Software:
OS: Ubuntu 12.04, Kernel: 3.13.0-40-generic (x86_64), Desktop: Unity 5,
OpenGL: 2.1, Compiler: GCC 4.6.3, File-System: ext4, Screen Resolution:
ualization

Would you like to save these test results (Y/n): n

RAMspeed SMP 3.5.0:
pts/ramspeed-1.4.0 [Type: Average - Benchmark: Integer]
Test 1 of 1
Estimated Trial Run Count:    1
Estimated Time To Completion: 5 Minutes
Started Run 1 @ 13:10:55

Test Results:
10250.66

Average: 10250.66 MB/s
```

Figura 10.3: Resultados benchmark

11. Cuestión Opcional 11: Lea el artículo y elabore un breve resumen.

JMeter y Gatling son dos de las herramientas de código abierto más utilizadas para realizar benchmarks de carga en sistemas. Para la comparación de ambos software Flood.io ha realizado las pruebas sobre **nginx**, un servidor HTTP muy rápido, en el que se han tenido en consideración los GET y POST al servir contenido estático y dinámico simulando planes de uso con una gran cantidad de usuarios conectados.

Los resultados obtenidos al realizar la prueba para 10.000 usuarios simulados ha sido muy similar en ambas aplicaciones (1788 +/- 362 ms) y (1698 +/- 31 ms).

Como diferencias destacables, Gatling no registra tamaño de respuesta en bytes y JMeter por su parte, tiene una carga de recursos superior en JVM y utilización de la CPU, lo que puede afectar más si la complejidad de la prueba aumenta o recibe concurrencia en el JVM.

En cualquier caso, ambos demuestran tiempos de respuesta cercanos en las características y transacciones medidas con poca variación. Ambos mantienen un rendimiento promedio dentro de las 30.000 peticiones por minuto sin desviaciones, así como un comportamiento de caché correcto. Por lo tanto, en términos de concurrencia y rendimiento, JMeter y Gatling son similares, con limitaciones de Gatling al registrar con precisión la carga útil de respuesta en bytes y JMeter por el mayor uso de recursos en términos de CPU.

12. Cuestión Opcional 12: Seleccione un benchmark entre SisoftSandra y Aida. Ejecútelo y muestre capturas de pantalla comentando los resultados.

Probando la ejecución del software AIDA:

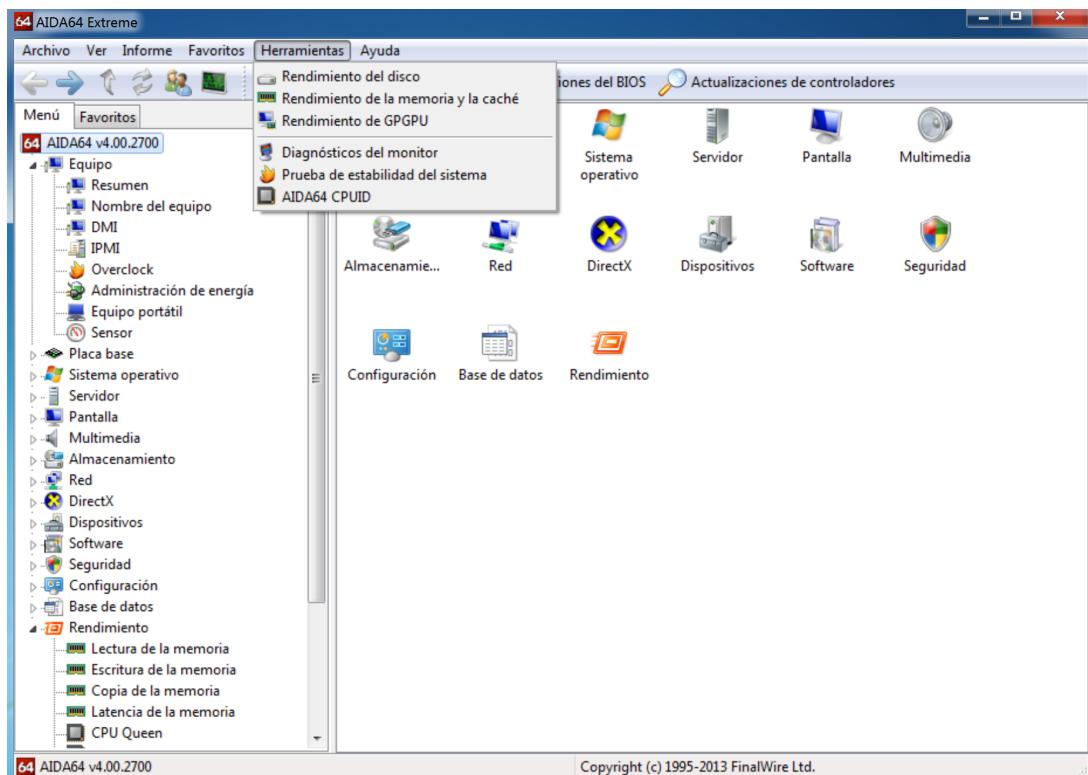
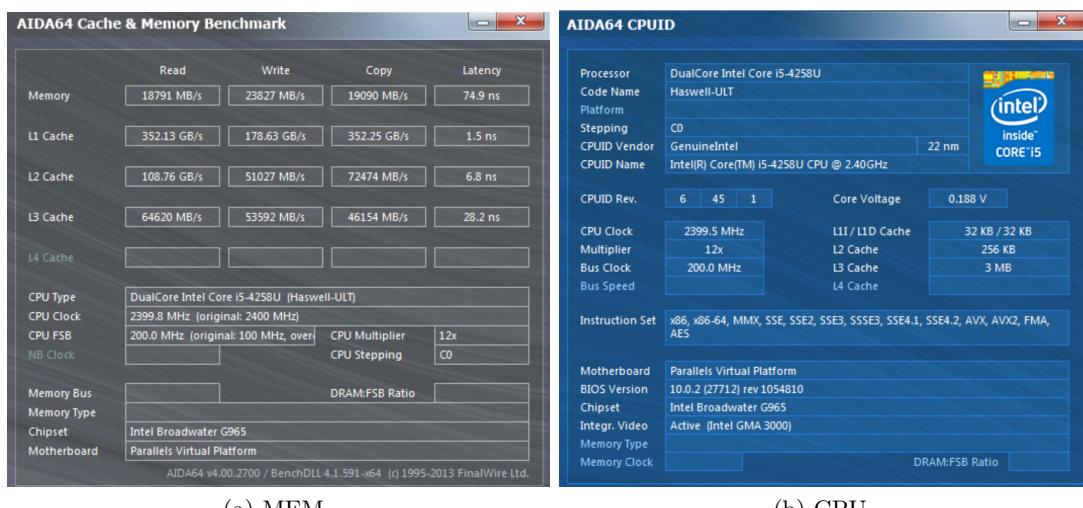


Figura 12.1: AIDA Software

Realizamos varios test tal y como se muestra en las siguientes capturas:



(a) MEM

(b) CPU

Figura 12.2: Benchmark de Memoria y CPU

Como vemos, en el caso del Benchmark de Memoria, AIDA tiene en consideración las 4 principales características de las memorias RAM y mide éstas según los diferentes niveles de caché que poseen:

- Velocidad de Lectura
- Velocidad de Escritura
- Velocidad de Copia
- Latencia

Para la CPU nos muestra toda la información detallada de la CPU de nuestro PC como la velocidad de reloj y caché, así como el voltaje y el tipo de chip.

Otra herramienta de AIDA para medir las características de la CPU:

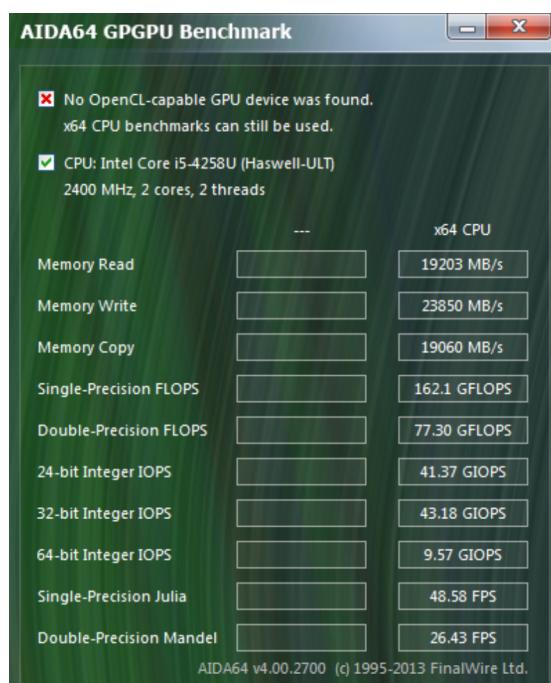


Figura 12.3: CPUID benchmark

En el Benchmark CPU/GPU nos muestra valores obtenidos como:

- Memoria Leída
- Memoria Escrita
- Memoria Copiada
- FLOPS con precisión simple y doble

Así mismo, AIDA nos permite hacer un benchmark para conocer la estabilidad del sistema aumentando el uso de la CPU:

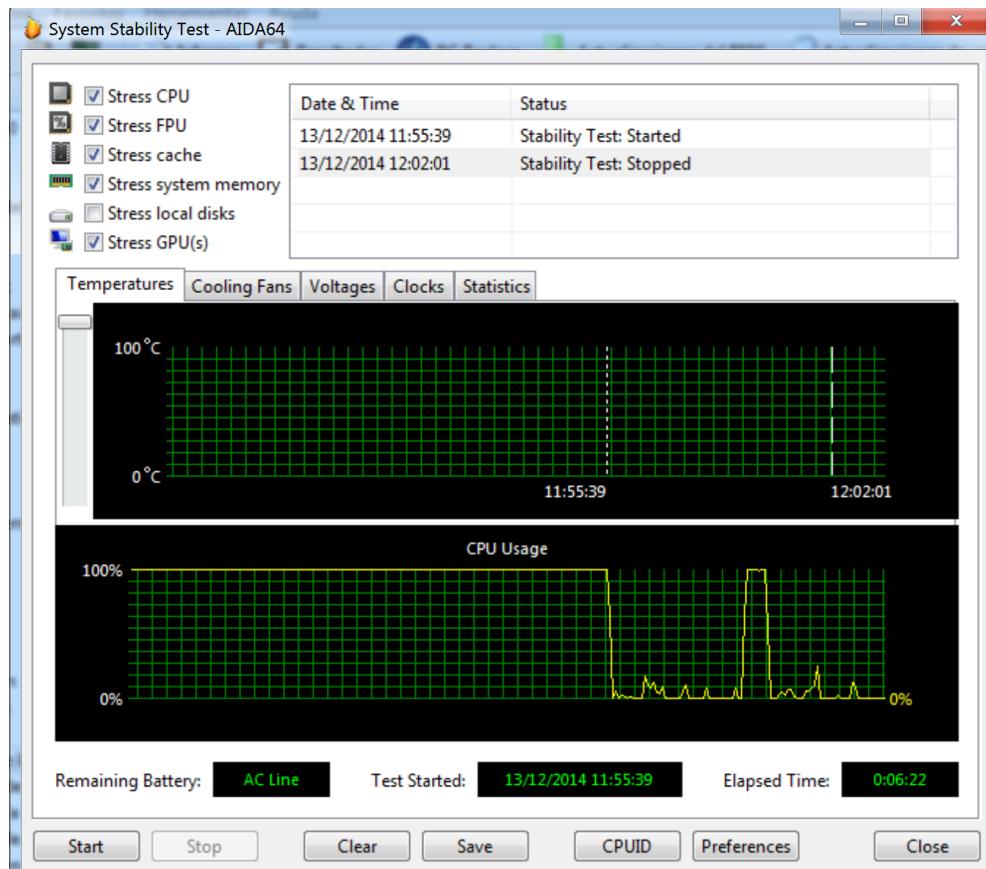


Figura 12.4: AIDA: Estabilidad del sistema

AIDA nos muestra el rendimiento y estabilidad del sistema del PC aumentando el uso a valores altos y monitorizando dichos valores en un gráfico de uso.

13. Cuestión opcional 13: Realice lo mismo que en la cuestión 8 pero para otro servicio.

Otra mejora del SO del servidor consiste en programar una desfragmentación del disco semanal para evitar clústers vacíos en el RAID montado en el servidor y espacio libre sin usar en el mismo:

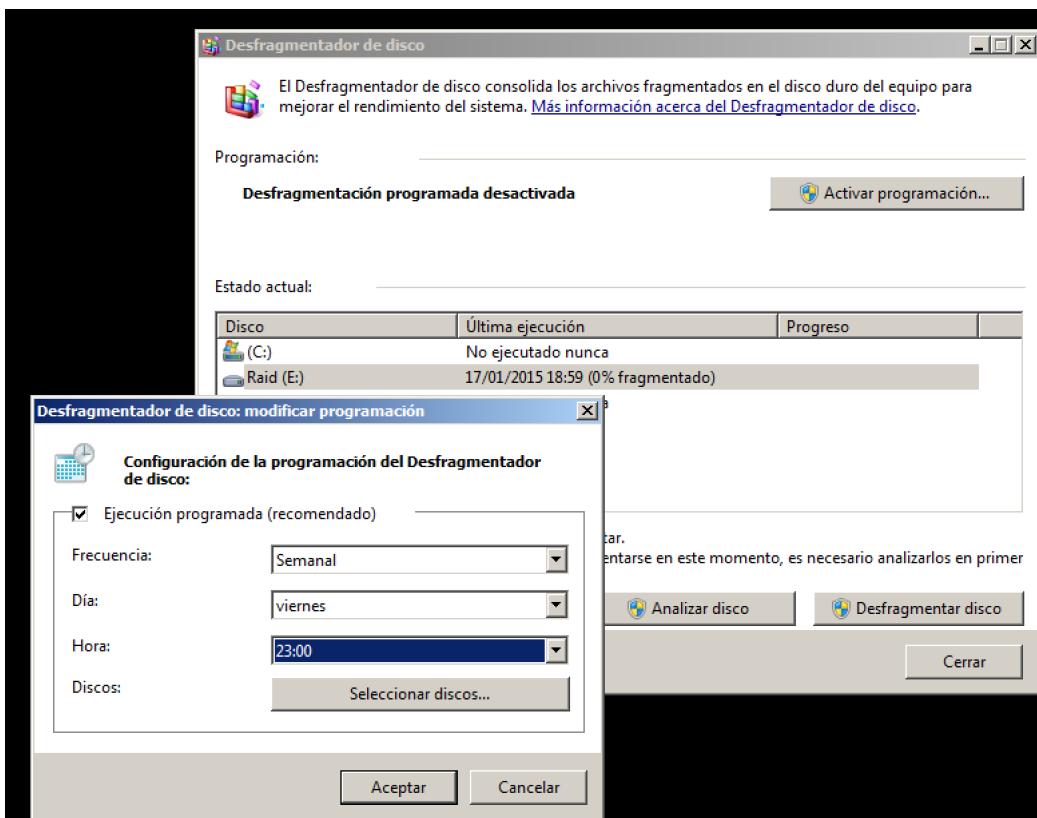


Figura 13.1: Desfragmentación del disco en Windows Server