

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

Memoria Práctica 5

Cristian Vélez Ruiz

14 de enero de 2017

Índice

1. Cuestión 1	3
2. Cuestión 2	4
3. Cuestión 3	4
4. Cuestión 4	7
5. Cuestión 5	7
6. Cuestión 6	9

Índice de figuras

1.1. Estado kernel panic	3
1.2. Cambiando kernel panic	3
1.3. Cambiando kernel panic	3
2.1. sysctl -a	4
3.1. Copia seguridad regedit	4
3.2. Copia seguridad regedit 1	5
3.3. Copia seguridad regedit 2	5
3.4. Copia seguridad regedit 3	5
3.5. Copia seguridad regedit 4	6
3.6. Copia seguridad regedit 5	6
3.7. Copia seguridad regedit 6	6
5.1. Archivo por defecto	7
5.2. Compresión del servidor	8
5.3. Compresión del servidor 2	8
5.4. Compresión del servidor 3	9
6.1. Optimizando Apache	10
6.2. Optimizando Apache 1	10
6.3. Configuración por defecto	10
6.4. Configuración modificada	11

Índice de tablas

1. Al modificar los valores del kernel de este modo, no logramos que persistan después de reiniciar la máquina. ¿Qué archivo hay que editar para que los cambios sean permanentes?

Para que los cambios que realicemos con `sysctl` sean persistentes necesitamos modificar el archivo `/etc/sysctl.conf`, donde se guarda la configuración del kernel y se cargará una vez arrancado nuestro sistema.

En mi caso modificare la variable para que cuando haya un kernel panic se reinicie a los 10 segundos.

- Observamos el estado actual de la variable con `sysctl -a | grep panic`:

```
(10:54:27)[CristianVelez-~]$> sysctl -a | grep panic
sysctl: permiso denegado para la clave «fs.protected_hardlinks»
sysctl: permiso denegado para la clave «fs.protected_symlinks»
sysctl: permiso denegado para la clave «kernel.cad_pid»
kernel.hardlockup_panic = 0
kernel.hung_task_panic = 0
kernel.panic = 0
kernel.panic_on_io_nmi = 0
kernel.panic_on_oops = 0
kernel.panic_on_unrecovered_nmi = 0
kernel.panic_on_warn = 0
```

Figura 1.1: Estado kernel panic

Como se puede apreciar esta configurado a 0.

- Ahora para cambiarlo a 10 debemos hacer `sudo sysctl -w kernel.panic=10`.

```
(10:58:40)[CristianVelez-~]$> sudo sysctl -w kernel.panic=10
[sudo] password for ainokila:
kernel.panic = 10
(10:58:57)[CristianVelez-~]$> █
```

Figura 1.2: Cambiando kernel panic

- Una vez modificado debemos hacerlo persistente añadiendo al archivo `/etc/sysctl.conf` con `echo "kernel.panic = 10">/etc/sysctl.conf`.
- Una vez realizado actualizamos la configuración de `sysctl` con `sysctl -p`.

```
(11:08:05)[CristianVelez-~]$> sudo sysctl -p
kernel.panic = 10
(11:08:09)[CristianVelez-~]$> █
```

Figura 1.3: Cambiando kernel panic

Como se puede apreciar se ha actualizado `kernel.panic` y ya es permanente.

2. ¿Con qué opción se muestran todos los parámetros modificables en tiempo de ejecución? Elija dos parámetros y explique, en dos líneas, qué función tienen.

Para mostrar los parámetros en tiempo de ejecución se usa `sysctl -a` [1]

```
11:22:35)[CristianVelez--]$> sysctl -a
abi.vsyscall32 = 1
debug.exception-trace = 1
debug.kprobes-optimization = 1
dev.cdrom.autoclose = 1
dev.cdrom.autoject = 0
dev.cdrom.check_media = 0
dev.cdrom.debug = 0
dev.cdrom.info = CD-ROM information, Id: cdrom.c 3.20 2003/12/17
dev.cdrom.info =
dev.cdrom.info = drive name:
dev.cdrom.info = drive speed:
dev.cdrom.info = drive # of slots:
dev.cdrom.info = Can close tray:
dev.cdrom.info = Can open tray:
dev.cdrom.info = Can lock tray:
dev.cdrom.info = Can change speed:
dev.cdrom.info = Can select disk:
dev.cdrom.info = Can read multisession:
dev.cdrom.info = Can read MCN:
```

Figura 2.1: `sysctl -a`

- **kernel.panic_on_stackoverflow** : Si se produce un desbordamiento de pila y esta configurado a 0, intentara continuar con la operacion , en cambio si esta a 1 entrara en modo panico.
- **kernel.pid_max** :Cuando el siguiente PID del kernel alcanza este valor, se ajusta a un valor mínimo de PID, ni el minimo ni el maximo se utilizan.

3. a) Realice una copia de seguridad del registro y restáurela, ilustre el proceso con capturas. b) Abra una ventana mostrando el editor del registro.

Para realizar la copia debemos ir a regedit (editor de registros):

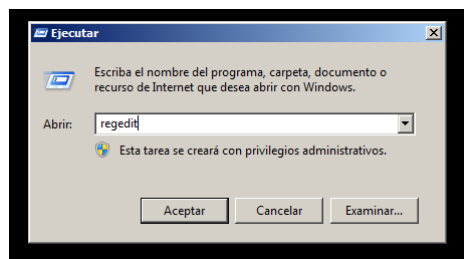


Figura 3.1: Copia seguridad regedit

Una vez abierto damos a exportar:

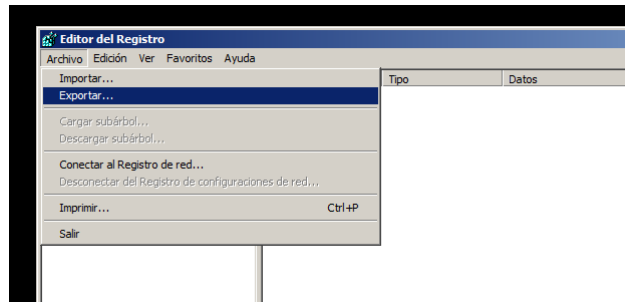


Figura 3.2: Copia seguridad regedit 1

Añadimos el nombre a la copia de seguridad y donde queremos guardarla:
Una vez abierto damos a exportar:

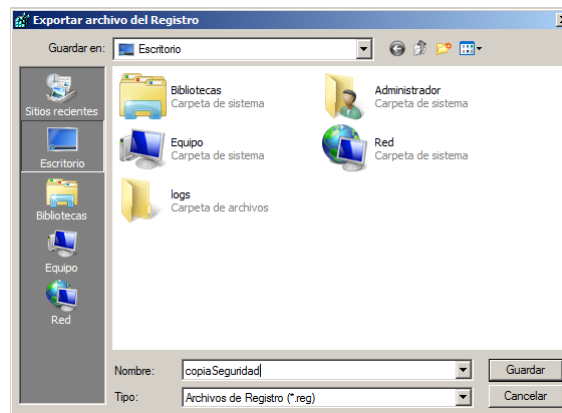


Figura 3.3: Copia seguridad regedit 2

Ya tenemos nuestra copia de seguridad realizada.

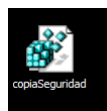


Figura 3.4: Copia seguridad regedit 3

Para hacer el paso contrario, restaurar la copia de seguridad, vamos a importar:

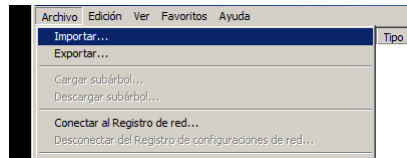


Figura 3.5: Copia seguridad regedit 4

Seleccionamos la copia de seguridad a restaurar:

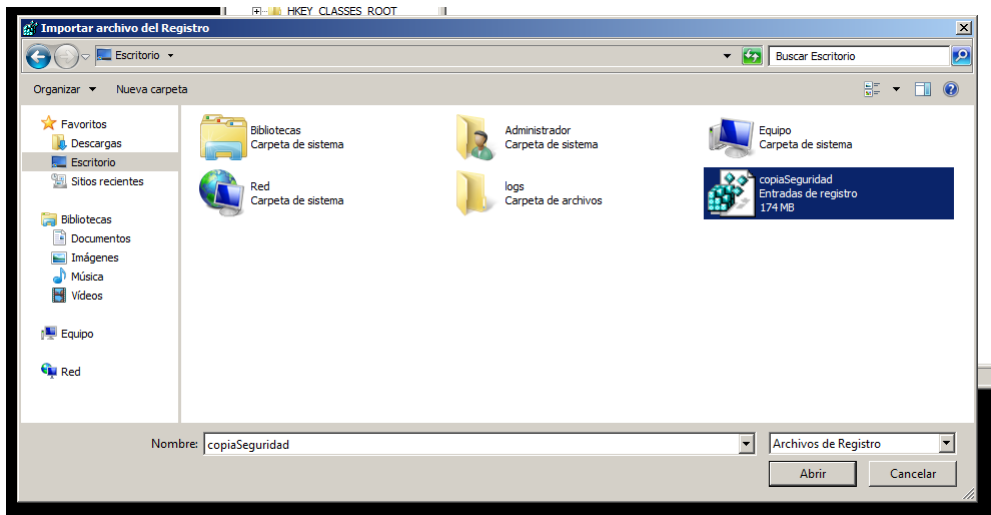


Figura 3.6: Copia seguridad regedit 5

Seleccionamos abrir y comenzará el proceso de restauracion.

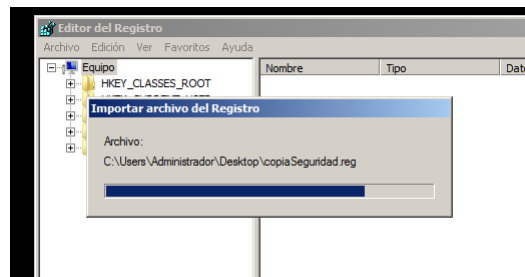


Figura 3.7: Copia seguridad regedit 6

Una vez finalice ya tenemos restaurada la copia de seguridad.

4. Enumere qué elementos se pueden configurar en Apache y en IIS para que Moodle funcione mejor.

En Apache algunos de los elementos que podemos configurar son:

- MaxClients para evitar que el sistema se quede sin memoria.
- Reducir el numero de modulos que Apache carga desde httpd.conf para reducir el uso de memoria.
- Disminuir MaxRequestsPerChild a 20-30.
- Cnfigurar un Reverse Proxy server para reducir la carga del servidor para la descarga de archivos con imagenes.
- Configurar el Directory Index DirectoryIndex correctamente para evitar la negociación de contenido.
- Desactivar HostnameLookups para reducir la latencia del dns.
- Reducir el TimeOut a 30-60.

En ISS algunos de los elementos que podemos configurar son:

- ListenBackLog es el equivalente de KeepAliveTimeout establecer entre 2-5.
- MemCacheSize tamaño de la memoria cache, por defecto esta configurado a la mitad disponible.
- ObjectCacheTTL tiempo en el que los objetos permanecen en la memoria cache.

5. Ajuste la compresión en el servidor y analice su comportamiento usando varios valores para el tamaño de archivo a partir del cual comprimir. Para comprobar que está comprimiendo puede usar el navegador o comandos como curl (see url) o lynx. Muestre capturas de pantalla de todo el proceso

Para poder comprobar la compresión del servidor voy a modificar la pagina por defecto del servidor como hicimos en la practica anterior casi 2KiB.

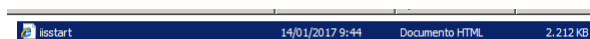


Figura 5.1: Archivo por defecto

Una vez realizado vamos a la configuración de IIS y después al apartado Compresión y habilitamos las pestañas de compresión estática, y también configuramos sus parámetros.

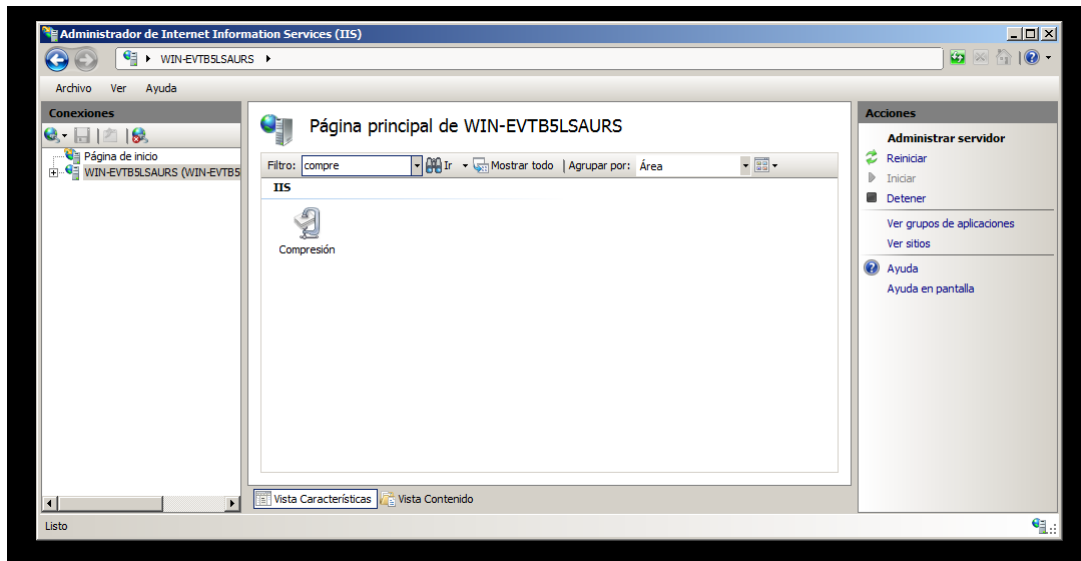


Figura 5.2: Compresión del servidor

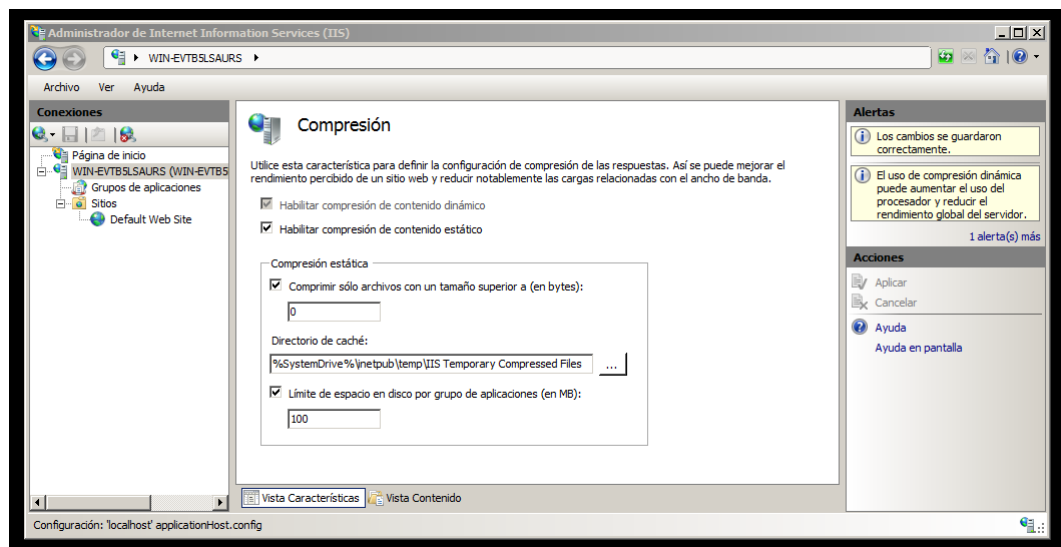


Figura 5.3: Compresión del servidor 2

Desde otra maquina lanzaré la orden `curl -I -H 'Accept-Encoding: gzip, deflate'` antes de aplicar los cambios y una vez aplicados para ver el tamaño de la respuesta a la petición.


```

(09:46:59)[CristianVelez--]$> curl 10.0.2.4 -I -H 'Accept-Encoding:gzip,deflate'
HTTP/1.1 200 OK
Content-Length: 2264167
Content-Type: text/html
Last-Modified: Sat, 14 Jan 2017 08:44:54 GMT
Accept-Ranges: bytes
ETag: "08c5b7a426ed21:0"
Server: Microsoft-IIS/7.5
Date: Sat, 14 Jan 2017 08:48:17 GMT

(09:48:17)[CristianVelez--]$> curl 10.0.2.4 -I -H 'Accept-Encoding:gzip,deflate'
HTTP/1.1 200 OK
Content-Length: 570363
Content-Type: text/html
Content-Encoding: gzip
Last-Modified: Sat, 14 Jan 2017 08:44:54 GMT
Accept-Ranges: bytes
ETag: "087f279426ed21:0"
Vary: Accept-Encoding
Server: Microsoft-IIS/7.5
Date: Sat, 14 Jan 2017 08:48:25 GMT

(09:48:25)[CristianVelez--]$>

```

Figura 5.4: Compresión del servidor 3

Como se puede apreciar la longitud de la respuesta ha pasado de 2264167 a 570363, lo que es casi un cuarto de su longitud anterior.

6. Usted parte de un SO con ciertos parámetros definidos en la instalación (Práctica 1), ya sabe instalar servicios (Práctica 2) y cómo monitorizarlos (Práctica 3) cuando los somete a cargas (Práctica 4). Al igual que ha visto cómo se puede mejorar un servidor web (Práctica 5 Sección 3.1), elija un servicio (el que usted quiera) y modifique un parámetro para mejorar su comportamiento. 6.b) Monitorice el servicio antes y después de la modificación del parámetro aplicando cargas al sistema (antes y después) mostrando los resultados de la monitorización.

(a) Modificaré el servicio apache dos parametros:

- MaxKeepAliveRequest: Número máximo de peticiones a permitir en una conexión persistente. Antes estaba: 100 , ahora: 64
- KeepAliveTimeout: Número de segundos a esperar a la siguiente petición de algún cliente. Antes estaba: 5 , ahora: 3

Antes de que sea modificado se hará la prueba del ejercicio 6.b.

Para modificar estos datos debemos ir a `/etc/apache2/apache2.conf`, hacemos una copia de seguridad, seguidamente editamos los valores y reiniciamos el servicio.

```

94 #
95 # MaxKeepAliveRequests: The maximum number of requests to allow
96 # during a persistent connection. Set to 0 to allow an unlimited amount.
97 # We recommend you leave this number high, for maximum performance.
98 #
99 MaxKeepAliveRequests 64
.00
.01 #
.02 # KeepAliveTimeout: Number of seconds to wait for the next request from the
.03 # same client on the same connection.
.04 #
.05 KeepAliveTimeout 3|
.06
.07

```

Figura 6.1: Optimizando Apache

```

(10:30:35)[CristianVelez--]$> sudo service apache2 restart
* Restarting web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain
name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this
message
[ OK ]
(10:30:47)[CristianVelez--]$> re

```

Figura 6.2: Optimizando Apache 1

- (b) Para obtener monitorizar el rendimiento usará Apache Benchmark como en la práctica anterior con `ab -n 50000 -c 50 http://direccion/index.html`, antes y después de la modificación.

- Configuración por defecto:

```

Completed 50000 requests
Finished 50000 requests

Server Software:      Apache/2.4.7
Server Hostname:      192.168.23.2
Server Port:          80

Document Path:        /index.html
Document Length:      12084 bytes

Concurrency Level:     50
Time taken for tests:  19.141 seconds
Complete requests:     50000
Failed requests:        0
Total transferred:     617850000 bytes
HTML transferred:     604200000 bytes
Requests per second:   2612.16 [#/sec] (mean)
Time per request:      19.141 [ms] (mean)
Time per request:      0.383 [ms] (mean, across all concurrent requests)
Transfer rate:         31521.99 [Kbytes/sec] received

Connection Times (ms)
  min  mean[+/-sd] median   max
Connect:    0    0  0.6    0    19
Processing:  1   19  5.5   17   72
Waiting:    0   12  6.8    9   70
Total:      2   19  5.6   17   73

Percentage of the requests served within a certain time (ms)
 50%    17
 66%    19
 75%    21
 80%    22
 90%    25
 95%    31
 98%    38
 99%    43
100%    73 (longest request)

```

Figura 6.3: Configuración por defecto

- Configuración modificada:

```

Finished 50000 requests

Server Software:      Apache/2.4.7
Server Hostname:      192.168.23.2
Server Port:          80

Document Path:        /index.html
Document Length:      12084 bytes

Concurrency Level:     50
Time taken for tests:  18.993 seconds
Complete requests:     50000
Failed requests:       0
Total transferred:     617850000 bytes
HTML transferred:     604200000 bytes
Requests per second:   2632.56 [#/sec] (mean)
Time per request:      18.993 [ms] (mean)
Time per request:      0.380 [ms] (mean, across all concurrent requests)
Transfer rate:         31768.08 [Kbytes/sec] received

Connection Times (ms)
      min    mean[+/-sd] median    max
Connect:    0       0   0.6      0      18
Processing:  1      19   5.8     17     75
Waiting:    1      11   6.8      9     74
Total:      1      19   5.9     17     77

Percentage of the requests served within a certain time (ms)
 50%    17
 60%    18
 75%    20
 80%    21
 90%    25
 95%    30
 98%    38
 99%    45
100%    77 (longest request)

Cristian@DESKTOP-RPLP927 c:\xampp
#

```

Figura 6.4: Configuración modificada

Como podemos apreciar con la configuración personalizada hemos reducido el tiempo medio por petición en 1 ms.

Referencias

- [1] Documentación variables kernel. <https://www.kernel.org/doc/Documentation/sysctl/kernel.txt>, 2016.