

**Ingeniería de Servidores (2015-2016)**  
GRADO EN INGENIERÍA INFORMÁTICA  
UNIVERSIDAD DE GRANADA

---

## Memoria Práctica 5

---

Miguel López Campos

26 de marzo de 2016

## Índice

1. Cuestión 1: Modificación de valores del kernel permanentemente	3
2. Cuestión 2: Mostrar parámetros modificables en tiempo de ejecución y descripción de algunos	3
3. Cuestión 3: Copia de seguridad de registro y restauración del mismo	4
4. Cuestión 4: Abrir consola en Windows y modificar un registro	5
5. Cuestión 5: Tipos de los caracteres y valores numéricos del registro de Windows	6
6. Cuestión 6: Mejoras apache e IIS para Moodle	7
7. Cuestión 7: Ajuste compresión en IIS	8
8. Cuestión 8: Mejora de un servicio y monitorización del mismo	9

## Índice de figuras

2.1. Ejecución de sysctl -a . . . . .	3
2.2. Parte de los resultados de ejecutar sysctl -a . . . . .	3
3.1. Seleccionamos que queremos una configuración personal (custom) . . . . .	4
3.2. Seleccionamos que queremos hacer backup del estado del sistema . . . . .	4
3.3. Seleccionamos frecuencia con la que queremos que se haga backup . . . . .	4
3.4. Seleccionamos ubicación de la copia de seguridad . . . . .	5
3.5. Indicamos que la copia de seguridad está en otra localización . . . . .	5
3.6. Seleccionamos la fecha y hora del backup que deseamos restaurar . . . . .	5
4.1. Abrimos Run . . . . .	5
4.2. Después de ejecutar cmd ya tenemos nuestra consola de comandos . . . . .	6
4.3. Ejemplo de edición del registro . . . . .	6
7.1. Seleccionamos compresión . . . . .	8
7.2. Configuramos la compresión . . . . .	8
7.3. Abrimos las herramientas para desarrolladores en Chrome . . . . .	8
7.4. Resultado de abrir nuestro sitio web con la herramienta para desarrolladores	8
7.5. Comprobamos que funciona la compresión . . . . .	8
7.6. Comprobamos que funciona la compresión 2 . . . . .	9
8.1. Ejecución ab sobre nuestro servidor IIS sin mejoras (1) . . . . .	9
8.2. Ejecución ab sobre nuestro servidor IIS sin mejoras (2) . . . . .	9
8.3. Configuramos el output caching . . . . .	9
8.4. Configuración del output caching . . . . .	9
8.5. Ejecución ab sobre nuestro servidor IIS con las mejoras (1) . . . . .	9
8.6. Ejecución ab sobre nuestro servidor IIS con las mejoras (2) . . . . .	9

## 1. Cuestión 1: Modificación de valores del kernel permanentemente

Para que los cambios se realicen permanentemente hay que modificar el fichero `/etc/sysctl.conf`, añadiendo la línea con la variable que queremos modificar así como su valor <sup>1</sup>. Podemos hacerlo de la siguiente manera (en mi ejemplo el parámetro que modifico es el nombre del host):

```
'echo "kernel.hostname=miequipo">/etc/sysctl.conf && sysctl -p'
```

Lo que hace que de esta manera se mantengan los cambios al reiniciar el sistema es el hecho de que al iniciarse el sistema se ejecuta un script que ejecuta `sysctl` usando los valores de `sysctl.conf` (el fichero que hemos modificado).

## 2. Cuestión 2: Mostrar parámetros modificables en tiempo de ejecución y descripción de algunos

Para mostrar todos los parámetros <sup>2</sup> modificables en tiempo de ejecución usamos la opción `-a` (ver figuras 2.1 y 2.2). Lo que hacemos es hacer que el resultado de `sysctl -a` tome como salida un fichero (llamado en mi caso `holae.txt`) y después lo abro con `gedit` para ver el resultado, que son todos los parámetros modificables en tiempo de ejecución.

Figura 2.1: Ejecución de `sysctl -a`

Figura 2.2: Parte de los resultados de ejecutar `sysctl -a`

Un parámetro interesante es por ejemplo **kernel.panic** <sup>3</sup>. Este parámetro lo que hace es que cuando se produce un error interno en el sistema (un 'panic'), este se reinicie después del tiempo indicado al parámetro. Por ejemplo, si ponemos `kernel.panic = 5`, después de 5 segundos del error interno, el sistema se reiniciará. Si es 0, no hará nada.

Otro parámetro es **kernel.pid\_max** <sup>4</sup>. Este parámetro lo que hace es indicar cuál es el valor máximo para un PID. Si a un proceso le tocara un PID mayor o igual que este valor, lo que pasaría es que se le asignaría otro valor que sí esté en el rango indicado.

---

<sup>1</sup><http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/s1-proc-sysctl.html>

<sup>2</sup>Manual `sysctl` Ubuntu

<sup>3</sup><https://www.kernel.org/doc/Documentation/kernel-parameters.txt>

<sup>4</sup><https://www.kernel.org/doc/Documentation/sysctl/kernel.txt>

### 3. Cuestión 3: Copia de seguridad de registro y restauración del mismo

Para hacer la copia de seguridad y la restauración seguiré las instrucciones de [5]. Primero lo que hago es abrir Windows Server Backup para realizar la copia de seguridad. Una vez abierto, hago click sobre Action ->Backup Schedule. Nos aparecerá una ventana nueva y pulsamos siguiente. Después nos aparecerá la ventana de la figura 3.1 y marcaremos la opción Custom, para elegir de qué queremos hacer la copia de seguridad de forma manual.

Figura 3.1: Seleccionamos que queremos una configuración personal (custom)

Ahora llegaremos a una ventana en la que ya añadiremos los ficheros que queremos guardar en nuestra copia de seguridad. Hacemos click en Add Items y seleccionamos la opción System State (ver figura 3.2). A continuación podemos elegir con qué frecuencia queremos que nuestro sistema haga backups del registro. Elegimos la frecuencia deseada (ver figura 3.3).

Figura 3.2: Seleccionamos que queremos hacer backup del estado del sistema

Figura 3.3: Seleccionamos frecuencia con la que queremos que se haga backup

---

<sup>5</sup>[https://technet.microsoft.com/en-us/library/cc758453\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc758453(v=ws.10).aspx)

A continuación elegimos el destino en el que queremos que se guarde nuestra backup. Yo seleccionaré en una carpeta de uno de los discos duros. Después elegimos el dispositivo en el que queremos guardar nuestra copia de seguridad (ver figura 3.4) . Finalmente clickamos en Finalizar.

Figura 3.4: Seleccionamos ubicación de la copia de seguridad

Con estos pasos habremos creado una planificación de copias de seguridad con cierta frecuencia. Para realizar una simple copia de seguridad vamos a Action->Backup once. Elegimos que queremos realizar la copia de seguridad según las opciones que ya hemos seleccionado al crear la planificación de copia de seguridad y finalmente hacemos click sobre Backup y se nos realizará la copia de seguridad. Después de esperar un rato se habrá creado ya la copia de seguridad.

Para reestablecer un registro anterior, mediante también el programa Windows Server Backup clickamos en Action->Recover. Se nos aparecerá una ventana y nos preguntará que de dónde queremos cargar el backup. Seleccionamos la opción ‘A backup store on another location’ (ver figura 3.5). Pulsamos en siguiente y nos preguntará sobre el tipo de localización del backup (Discos locales) y cual es la localización concreta (el disco donde hemos guardado el backup). Después deberemos escoger la fecha del backup que hicimos (si solo tenemos uno nos saldrá por defecto). Ver figura 3.6. Pulsamos en siguiente y ya escogemos que sólo queremos restaurar el ‘System State’. Pulsamos en siguiente y ya podremos iniciar la recuperación y posteriormente deberemos reiniciar el sistema.

Figura 3.5: Indicamos que la copia de seguridad está en otra localización

Figura 3.6: Seleccionamos la fecha y hora del backup que deseamos restaurar

## **4. Cuestión 4: Abrir consola en Windows y modificar un registro**

Para abrir la consola en Windows Server podemos hacerlo de la siguiente manera:

Vamos al menú inicio y a All Programs->Accesories y abrimos el programa Run ( ejecutar si fuese en español). Una vez abierto tecleamos cmd y ejecutamos. Nos aparecerá ya la consola. Ver figura 4.1 y 4.2.

Figura 4.1: Abrimos Run

Figura 4.2: Después de ejecutar cmd ya tenemos nuestra consola de comandos

Para editar el registro podemos usar el comando REG. Un ejemplo es el de la figura 4.3 donde añadido una nueva clave o valor mediante ‘REG ADD’ y como vemos en regedit, ésta aparece añadida.

Figura 4.3: Ejemplo de edición del registro

## 5. Cuestión 5: Tipos de los caracteres y valores numéricos del registro de Windows

Los distintos tipos <sup>6</sup> de datos que podemos encontrar son:

- Valor binario [REG\_BINARY]: datos binarios sin formato. Información de componentes hardware.
- Valor DWORD [REG\_DWORD]: 4 bytes de longitud. Pueden representar a parámetros de controladores de dispositivos y servicios.
- Valor alfanumérico expandible [REG\_EXPAND\_SZ]: Son cadenas de datos con una longitud variable.
- Valor de cadena múltiple [REG\_MULTI\_SZ]: Cadena múltiple. Pueden contener listas o valores múltiples.
- Valor de cadena [REG\_SZ]: Son cadenas de texto con longitud fija.
- Valor binario [REG\_RESOURCE\_LIST]: Son matrices anidadas que se usan para para por ejemplo almacenar listas de recursos utilizados por el controlador de un dispositivo.
- Valor binario [REG\_RESOURCE\_REQUIREMENTS\_LIST]: También una serie de matrices anidadas que se utiliza para almacenar recursos utilizados por controladores.
- Valor binario [REG\_FULL\_RESOURCE\_DESCRIPTOR]: Misma composición y misma función que las 2 anteriores.
- Ninguna [REG\_NONE]: Datos sin ningún tipo en particular.
- Vínculo [REG\_LINK]: Cadena Unicode que da nombre a un vínculo simbólico.
- Valor QWORD [REG\_QWORD]: Datos representados por un número entero de 64 bytes.

---

<sup>6</sup><https://support.microsoft.com/es-es/kb/256986>

## 6. Cuestión 6: Mejoras apache e IIS para Moodle

Los elementos configurables <sup>7</sup> que podemos tener en cuenta para que Moodle funcione mejor son los siguientes:

En un servidor Apache:

- Ajustar el parámetro “MaxClients” en función de la memoria total disponible en nuestro equipo.
- Cargar el mínimo número posible de módulos para reducir la memoria necesaria.
- Utilizar la última versión de Apache porque reduce el uso de memoria.
- Reducir a un mínimo valor de 20-30 el parámetro “MaxRequestPerChild”, para que la bifurcación de procesos no genere una mayor sobrecarga en vez de beneficio en el rendimiento.
- Establecer el parámetro “KeepAlive” a Off o bajar el valor de “KeepAliveTimeout” a un valor de 2-5, evitando así sobrecarga del procesador en el inicio de procesos.
- En lugar de la anterior, podemos crear un servidor proxy inverso delante del servidor de Moodle para almacenar en caché los archivos HTML con imágenes.
- Si no utilizamos un archivo “.htaccess” establecer “AllowOverride” a None para no tener que buscar dichos archivos.
- Establecer correctamente “DirectoryIndex” para evitar negociación de contenido indicando el archivo de índice que debe ser cargado.
- Configurar “ExtendedStatus” a Off y desactivar “mod\_info” y “mod\_status” si no vamos a hacer trabajo de desarrollo en el servidor.
- No cambiar “HostnameLookups” de Off para reducir la latencia de DNS.
- Reducir “TimeOut” a 30-60 segundos.
- En las directivas “Options”, evitar “Options MultiViews” para reducir el uso de entrada/salida en disco.

En un servidor ISS:

- Ajustar a 2-5 el valor de “ListenBacklog”.
- Cambiar el “MemCacheSize” para ajustar la memoria que se usará como caché de archivos.
- Cambiar “MaxCachedFileSize” para ajustar el tamaño máximo de un archivo en la caché de archivos.

---

<sup>7</sup>[http://docs.moodle.org/23/en/Performance\\_recommendations](http://docs.moodle.org/23/en/Performance_recommendations)

- Crear un valor DWORD llamado “ObjectCacheTTL” para cambiar la cantidad de tiempo (en milisegundos) que los objetos de la caché se mantienen en la memoria.

## 7. Cuestión 7: Ajuste compresión en IIS

Lo primero que hacemos <sup>8</sup> es abrir IIS. Seleccionamos nuestro servidor en la ventana de la izquierda y hacemos click en Compresión (Ver figura 7.1).

Figura 7.1: Seleccionamos compresión

Para activar la compresión tenemos que marcar las dos pestañas que hay para activar tanto la compresión de páginas dinámicas como de páginas estáticas. Después configuramos cuál queremos que sea el tamaño mínimo del html a partir del cual se comprimirá. En mi caso he puesto 20 bytes para que así se pueda comprobar más fácilmente que funciona. Después seleccionaremos en qué carpeta queremos que se creen los ficheros html comprimidos de forma temporal (en mi caso la de por defecto) y opcionalmente podremos elegir el tamaño máximo que queremos que IIS use para comprimir contenidos estáticos para nuestro sitio web. Ver figura 7.2. Finalmente pulsamos en aplicar.

Figura 7.2: Configuramos la compresión

A continuación reiniciamos nuestro servidor web y seguidamente comprobaremos que efectivamente funciona la compresión. Para ello usaremos la herramienta para desarrolladores de Google Chrome desde la máquina host (ver figura 7.3). Se nos abrirá en la parte inferior de la ventana una especie de consola. A continuación introducimos en la barra de dirección el sitio web y accedemos (figura 7.4).

Figura 7.3: Abrimos las herramientas para desarrolladores en Chrome

Figura 7.4: Resultado de abrir nuestro sitio web con la herramienta para desarrolladores

Como vemos en la figura 7.5, hemos recibido el html indexa.html. Si clickamos sobre él podemos ver las cabeceras de las solicitudes y respuestas. Exploramos la cabecera de Response y como vemos en la figura 7.6 hemos recibido la ‘notificación’ de que el mensaje irá codificado en gzip y que pesará unos 363 bytes (sin comprimir pesa unos 5.75 kB).

Figura 7.5: Comprobamos que funciona la compresión

---

<sup>8</sup>[https://technet.microsoft.com/en-us/library/cc771003\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc771003(v=ws.10).aspx)



Figura 7.6: Comprobamos que funciona la compresión 2

## 8. Cuestión 8: Mejora de un servicio y monitorización del mismo

Mi mejora será sobre mi servidor web IIS 7.0 <sup>9</sup> <sup>10</sup>. Lo que haré es activar la compresión de los html y habilitar el ‘output caching’, que se trata de que cuando el servidor envía el html al cliente, se guarda en la memoria del servidor de manera que en la próxima solicitud no se tendrá que reprocesar el html. Hacemos el benchmark sin ninguna de estas dos mejoras activas. En las figuras 8.1 y 8.2 podemos ver los resultados de ejecutar apache benchmark sobre nuestro sitio web

Figura 8.1: Ejecución ab sobre nuestro servidor IIS sin mejoras (1)

Figura 8.2: Ejecución ab sobre nuestro servidor IIS sin mejoras (2)

Para activar la compresión seguimos los pasos del ejercicio 7. Para activar el ‘output caching’ tenemos que hacer click sobre nuestro servidor dentro de IIS y seleccionamos ‘output caching’ (ver figura 8.3).

Figura 8.3: Configuramos el output caching

Aquí dentro en la ventana de Actions hacemos click en ‘Edit Feature Settings...’. Se nos abrirá una ventana en la cual activaremos la cache y opcionalmente podemos modificar algunos parámetros como el tamaño máximo de respuesta que guardaremos en caché y el tamaño máximo de la caché (yo lo dejaré por defecto). Ver figura 8.4.

Figura 8.4: Configuración del output caching

Reiniciamos nuestro sitio web y volvemos a ejecutar ab sobre él. Como vemos en las figuras 8.5 y 8.6 después de las mejoras notamos cierta mejoría en el tiempo.

Figura 8.5: Ejecución ab sobre nuestro servidor IIS con las mejoras (1)

Figura 8.6: Ejecución ab sobre nuestro servidor IIS con las mejoras (2)

---

<sup>9</sup>[https://technet.microsoft.com/es-es/library/cc754957\(v=ws.10\).aspx](https://technet.microsoft.com/es-es/library/cc754957(v=ws.10).aspx)

<sup>10</sup>[https://technet.microsoft.com/en-us/library/cc770381\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc770381(v=ws.10).aspx)