

Ingeniería de Servidores (2014-2015)
Grado en Ingeniería Informática
Universidad de Granada

Memoria Práctica 5

M^a Cristina Heredia Gómez

13 de enero de 2015

Índice

| | |
|---|----|
| 1. Cuestión 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? | 3 |
| 2. Cuestión 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. | 3 |
| 3. Cuestión 3 : Realice una copia de seguridad del registro y restaurela, ilustre el proceso con capturas. | 3 |
| 4. Cuestión 4 : ¿Cómo se abre una consola en Windows? ¿Qué comando hay que ejecutar para editar el registro? Muestre su ejecución con capturas de pantalla. | 15 |
| 5. Cuestión 5 : Las cadenas de caracteres y valores numéricos tienen distintos tipos. Busque en la documentación de Microsoft y liste todos los tipos de valores. Las variables relacionadas con el sistema están alojadas en HKEY_LOCAL_MACHINE (HKLM). | 17 |
| 5.1. Valores numéricos: | 17 |
| 5.2. Valores de cadena de caracteres: | 17 |
| 5.3. Valores no definidos: | 18 |
| 6. Cuestión 6 : Enumere qué elementos se pueden configurar en Apache y en IIS para que Moodle funcione mejor. | 18 |
| 6.1. En Apache: | 18 |
| 6.2. En ISS: | 19 |
| 7. Cuestión 7 : Ajuste la compresión en el servidor y analice su comportamiento usando varios valores para el tamaño a de archivo 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. | 19 |
| 8. Cuestión 8 : 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. (9.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. | 29 |

Índice de figuras

| | |
|---|----|
| 2.1. resultado de ejecutar sysctl en una terminal | 3 |
| 3.1. instalando copias de seguridad windows | 4 |
| 3.2. instalado copias de seguridad windows | 4 |
| 3.3. copias de seguridad windows | 5 |
| 3.4. haciendo una copia de seguridad windows | 6 |
| 3.5. haciendo una copia de seguridad windows | 7 |
| 3.6. haciendo una copia de seguridad windows | 8 |
| 3.7. aspecto final de los elementos seleccionados para copia | 9 |
| 3.8. seleccionando destino para copia de seguridad | 10 |
| 3.9. haciendo copia de seguridad | 11 |
| 3.10. proceso de crear copia de seguridad | 12 |
| 3.11. asistente restaurador de copia de seguridad | 13 |
| 3.12. asistente restaurador de copia de seguridad | 14 |
| 3.13. asistente restaurador de copia de seguridad | 15 |
| 4.1. comando regedit | 16 |
| 4.2. reusltado de ejecutar el comando regedit | 17 |
| 7.1. sistema y seguridad | 20 |
| 7.2. Herramientas Administrativas | 21 |
| 7.3. ISS manager | 22 |
| 7.4. ISS manager | 23 |
| 7.5. ISS manager ->Compression | 24 |
| 7.6. ejecutando curl antes de configurar el archivo applicationHost.config (no comprime) | 25 |
| 7.7. desbloqueando archivo de conf | 26 |
| 7.8. editando applicationHost.config | 27 |
| 7.9. resultado de ejecutar Curl tras habilitar compresión y modificar correctamente application- Host.config | 28 |
| 8.1. Integral.cpp | 30 |
| 8.2. Integral.cpp | 31 |
| 8.3. valores por defecto de las variables | 31 |
| 8.4. ejecución con las variables por defecto | 32 |
| 8.5. ejecución con las variables modificadas | 32 |
| 8.6. ejecución con las variables modificadas 2 | 33 |

Índice de tablas

- 1. Cuestión 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 no sean borrados al reiniciar el sistema, hay que editar las parámetros de configuración que hay en /etc/sysctl.conf O modificar los archivos del directorio /proc/sys. [sys]

- 2. Cuestión 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.**

```
cristina@cristina:~$ sysctl
Usage:
  sysctl [opciones] [variable[=valor] ...]
Options:
  -a, --all           mostrar todas las variables
  -A                 alias de -a
```

Figura 2.1: resultado de ejecutar sysctl en una terminal

por tanto, para mostrar todos los parámetros modificables en tiempo de ejecución usaremos las opciones : -a o -all.

por ejemplo, el parámetro: vm.laptop_mode tiene como función la de ahorrar batería cuando se usa el modo portátil, ya que miniza los giros que tiene que hacer el disco duro.

[wwwb] otro parámetro, por ejemplo, es: vm.swappiness, la cual puede cambiar el balance entre intercambiar páginas en caché a favor de páginas anónimas.

[uni]

- 3. Cuestión 3 : Realice una copia de seguridad del registro y restaurela, ilustre el proceso con capturas.**

Para realizar la copia de seguridad, en primer lugar abrimos el programa de copias de seguridad de windows. Nos dice como debemos instalar esta aplicación en el equipo, para ello: en Inicio->Administrador del servidor hacemos click en añadir características->copias de seguridad:

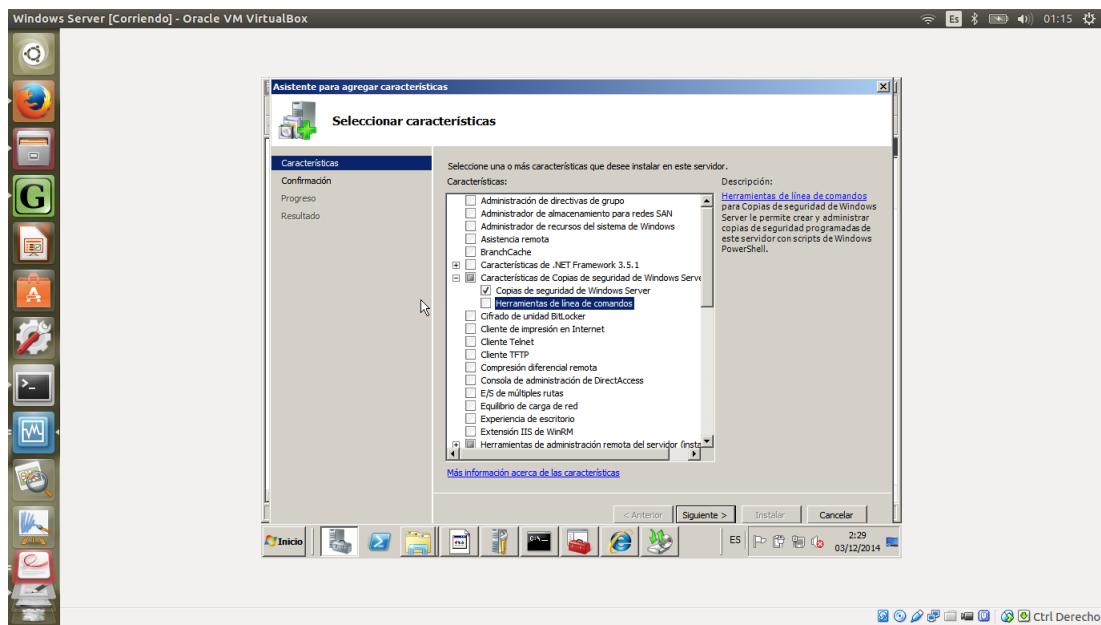


Figura 3.1: instalando copias de seguridad windows

y hacemos click en instalar. Nos sale un mensaje de confirmación de la instalación:

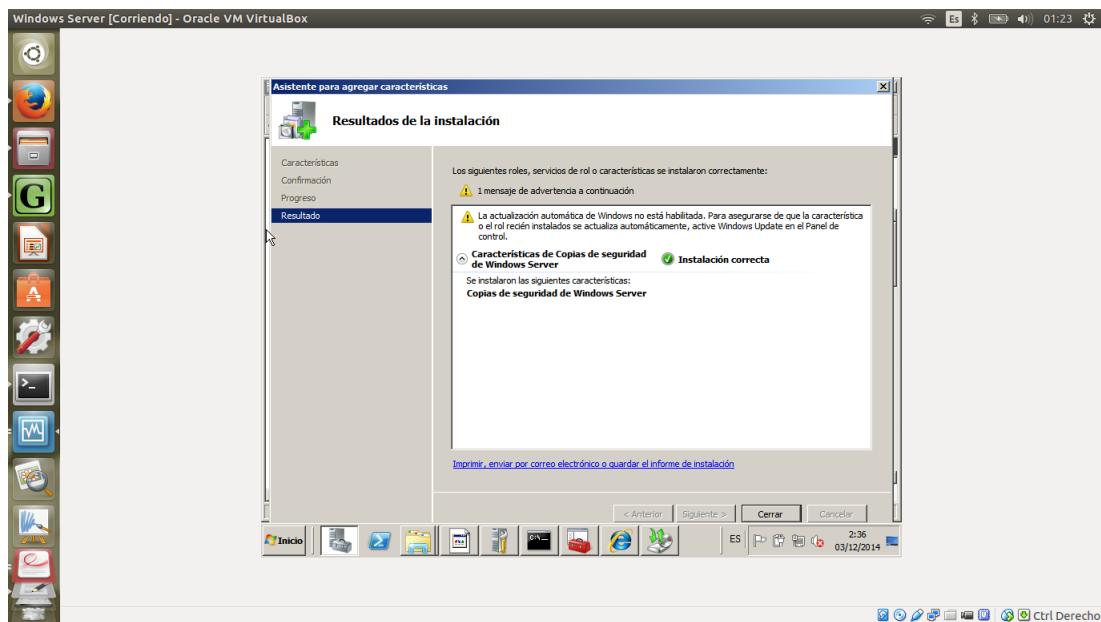


Figura 3.2: instalado copias de seguridad windows

Ahora volvemos al programa de copias de seguridad del principio, y vemos que ya no nos sale el mensaje del principio:

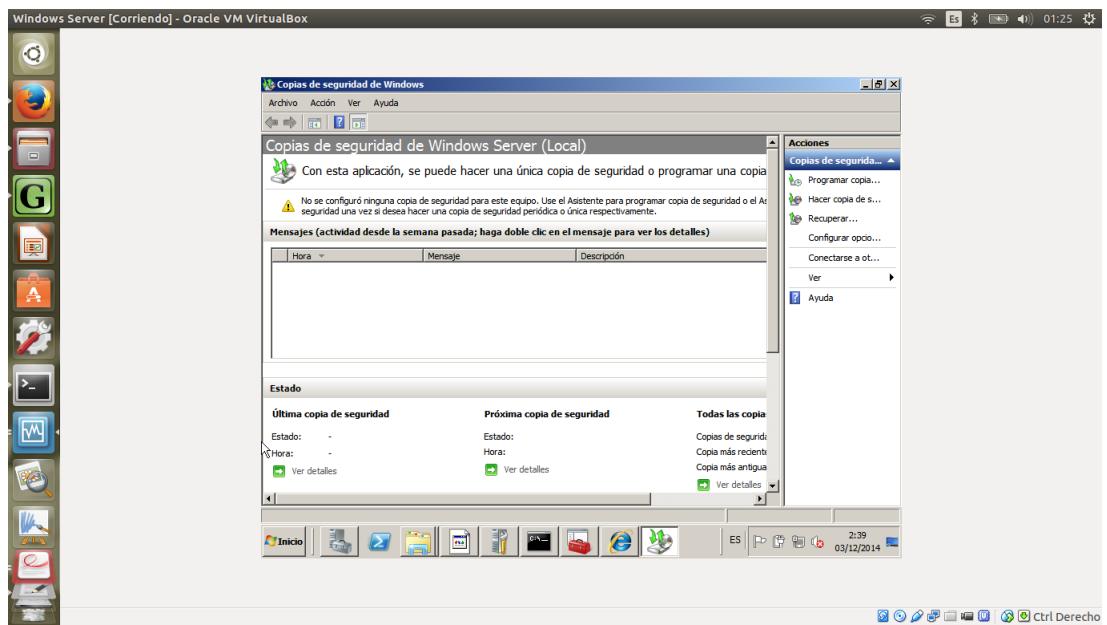


Figura 3.3: copias de seguridad windows

Seleccionamos acción->hacer copia de seguridad una vez:

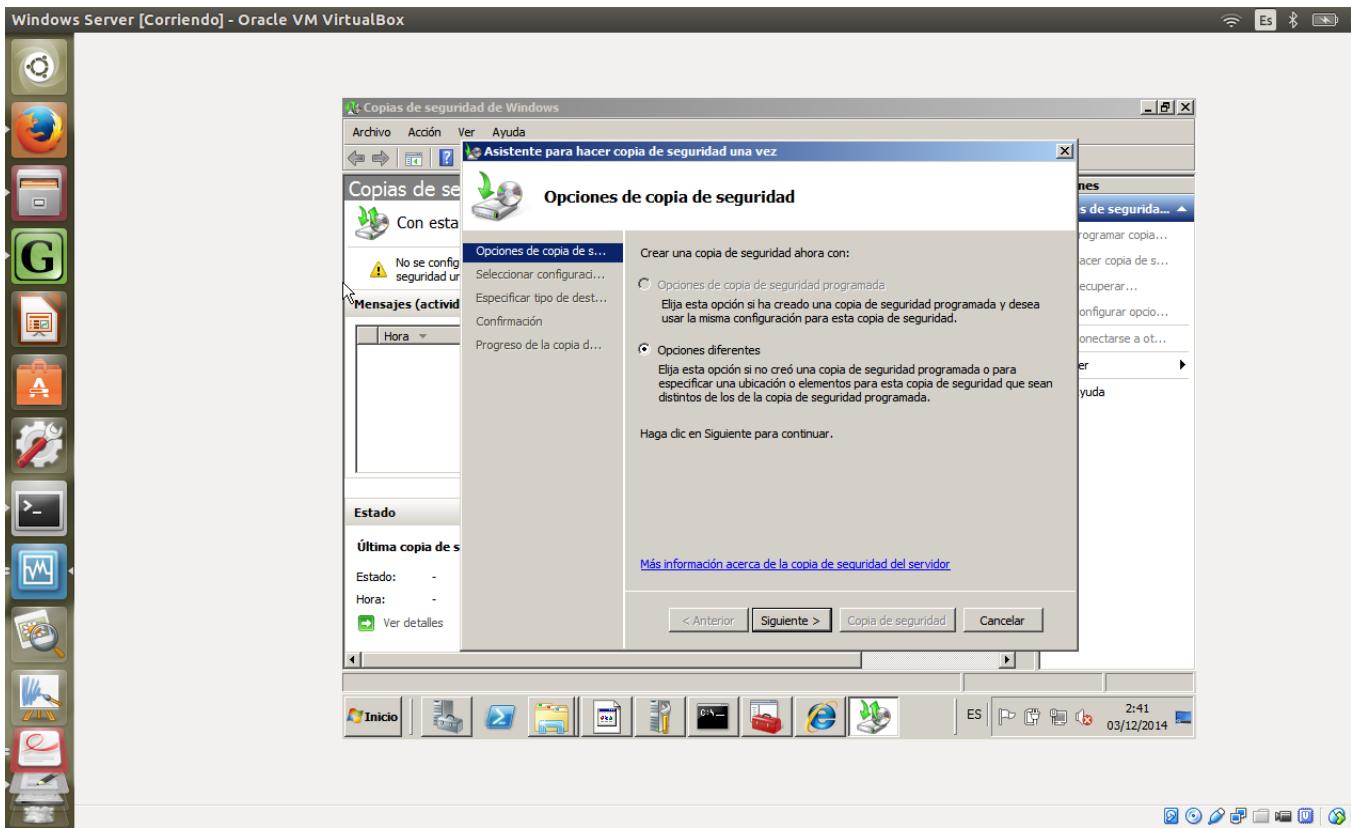


Figura 3.4: haciendo una copia de seguridad windows

y seleccionamos la opción de personalizado para poder elegir las partes de las que queremos hacer copias de seguridad: elegimos windows
system32
config y las carpetas de los usuarios que haya en el servidor:

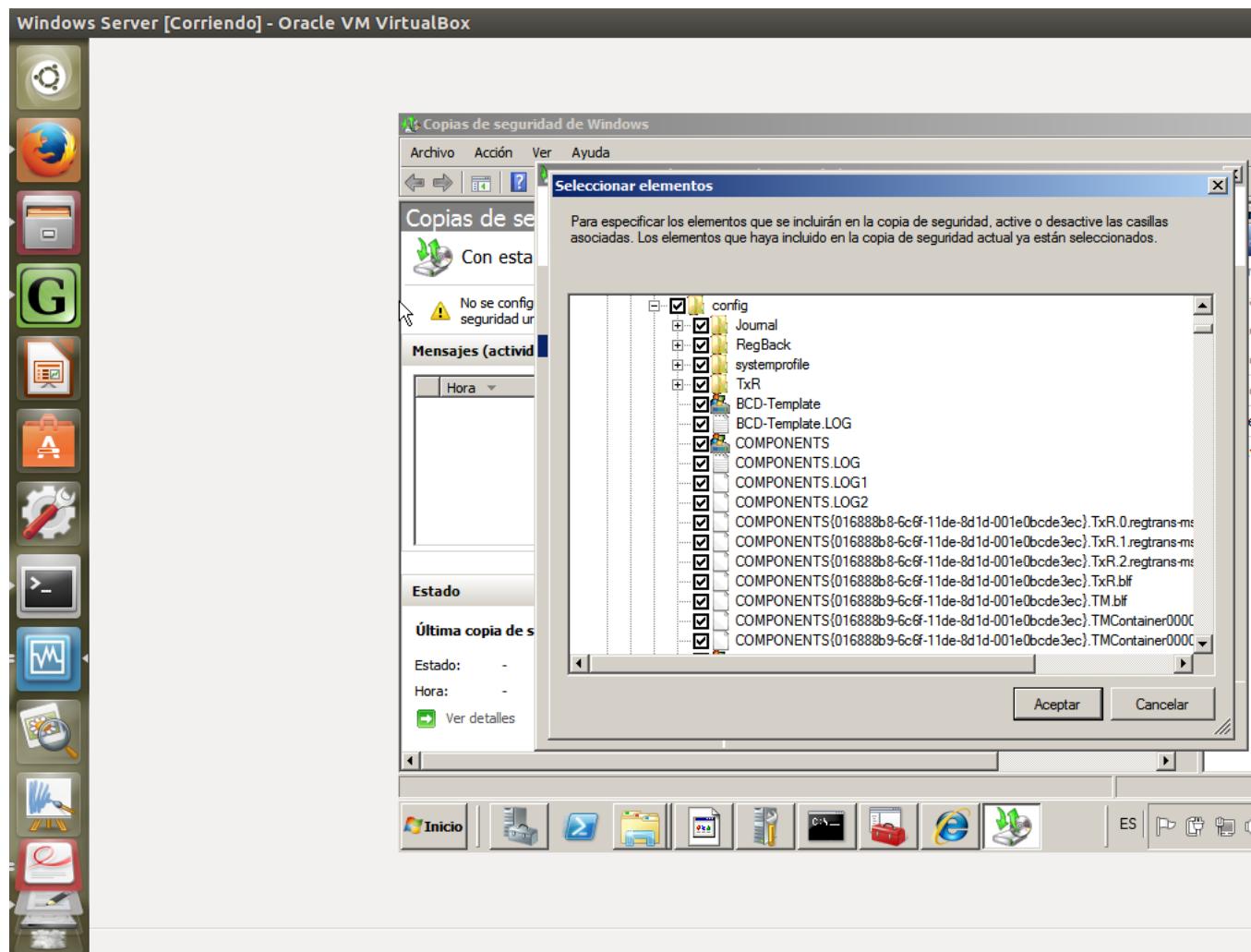


Figura 3.5: haciendo una copia de seguridad windows

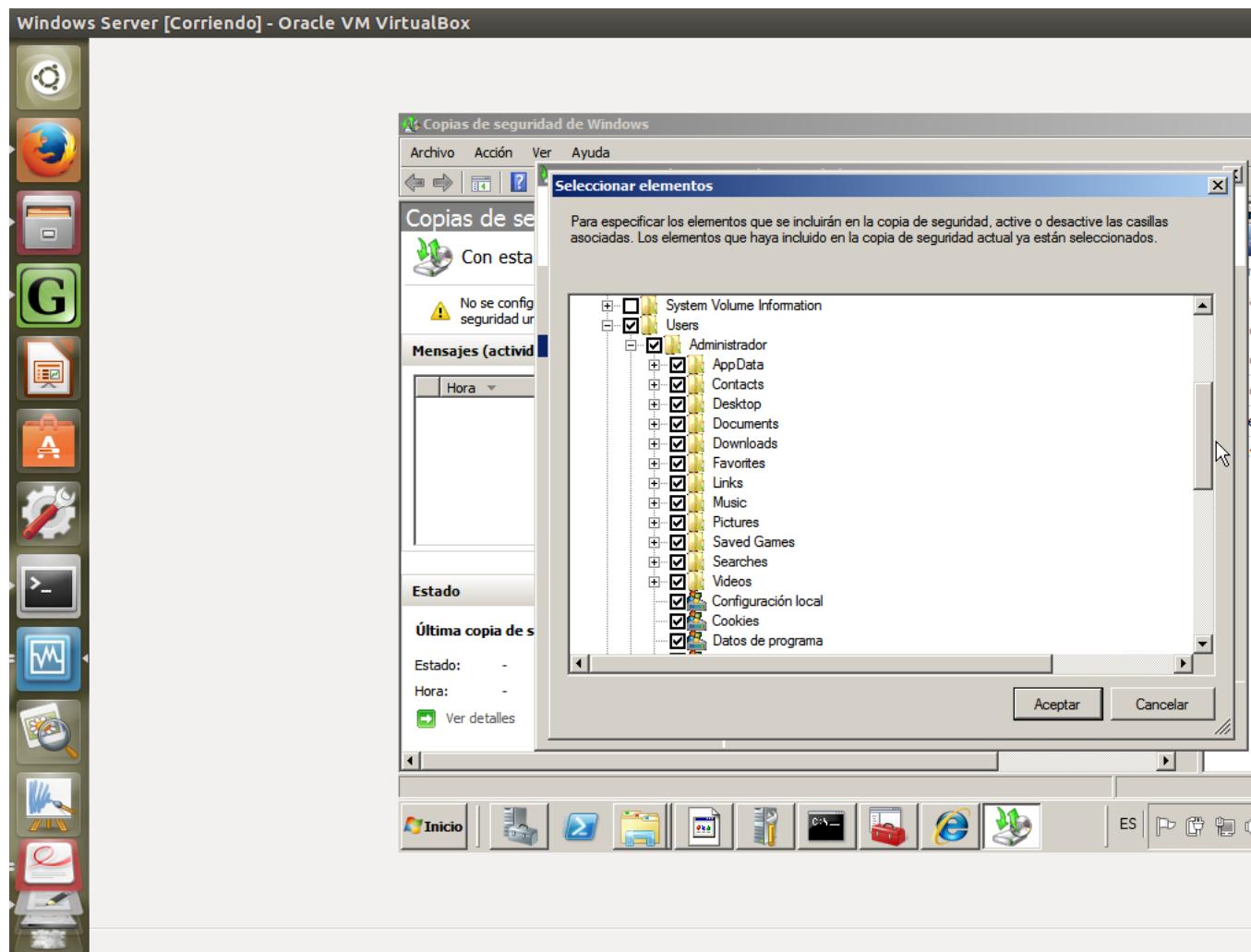


Figura 3.6: haciendo una copia de seguridad windows

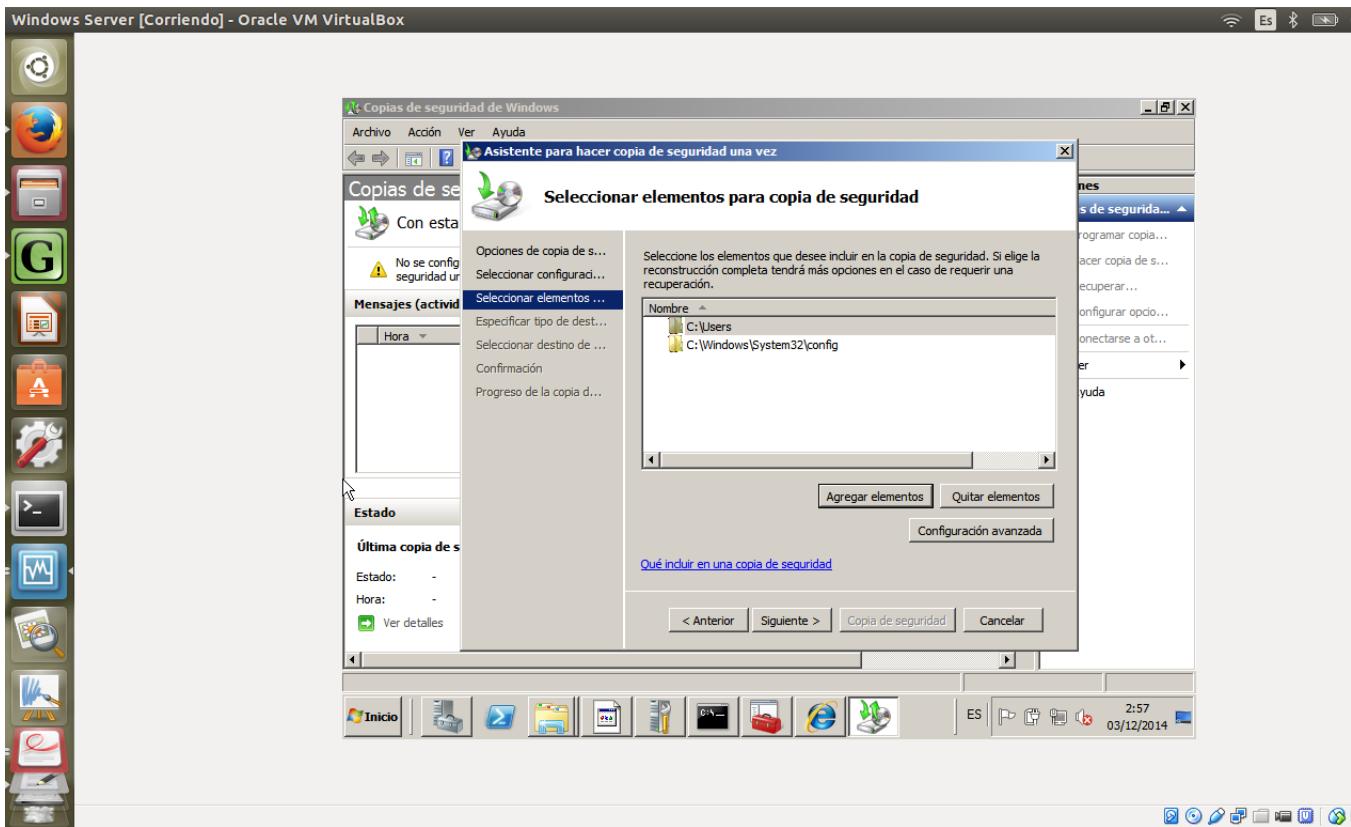


Figura 3.7: aspecto final de los elementos seleccionados para copia

ahora seleccionamos el destino de la copia de seguridad:

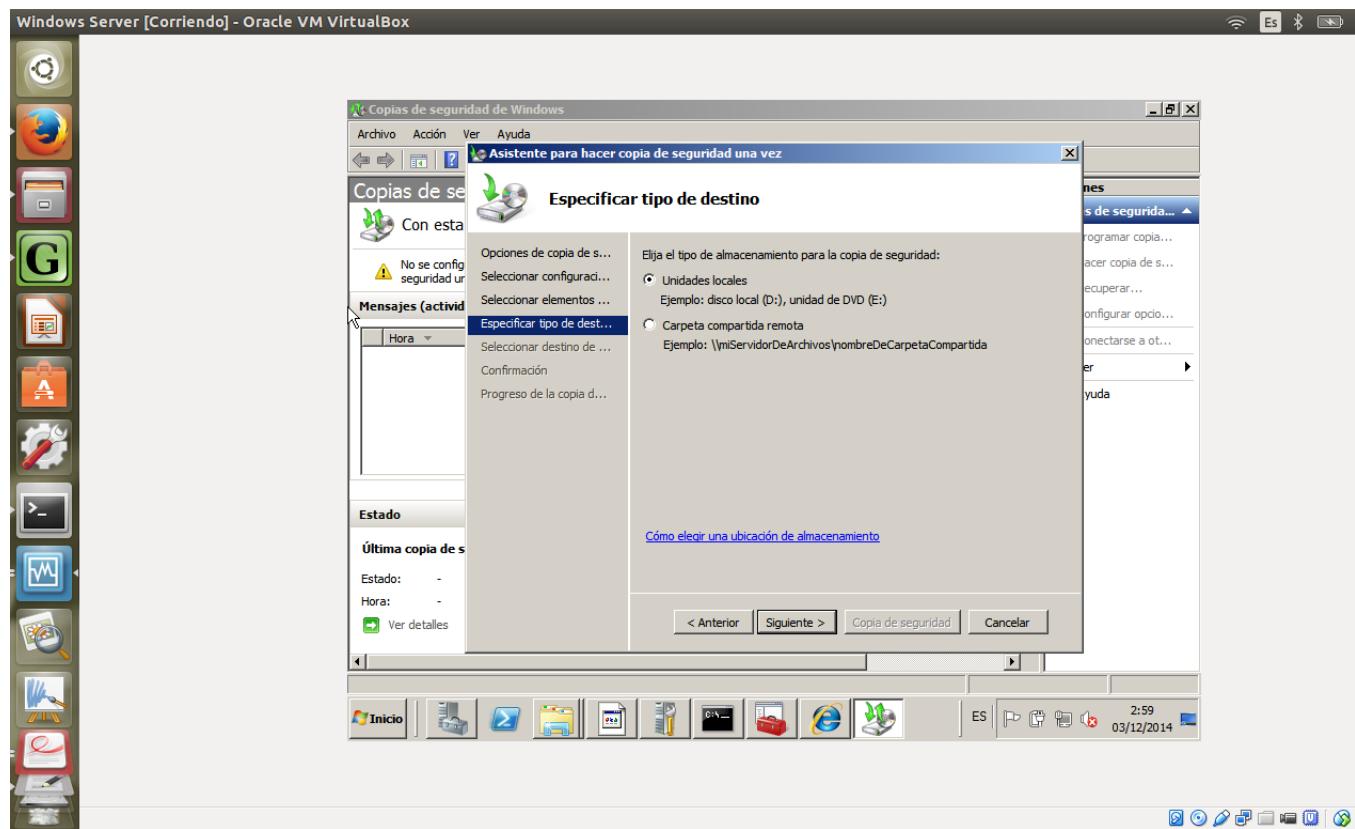


Figura 3.8: seleccionando destino para copia de seguridad

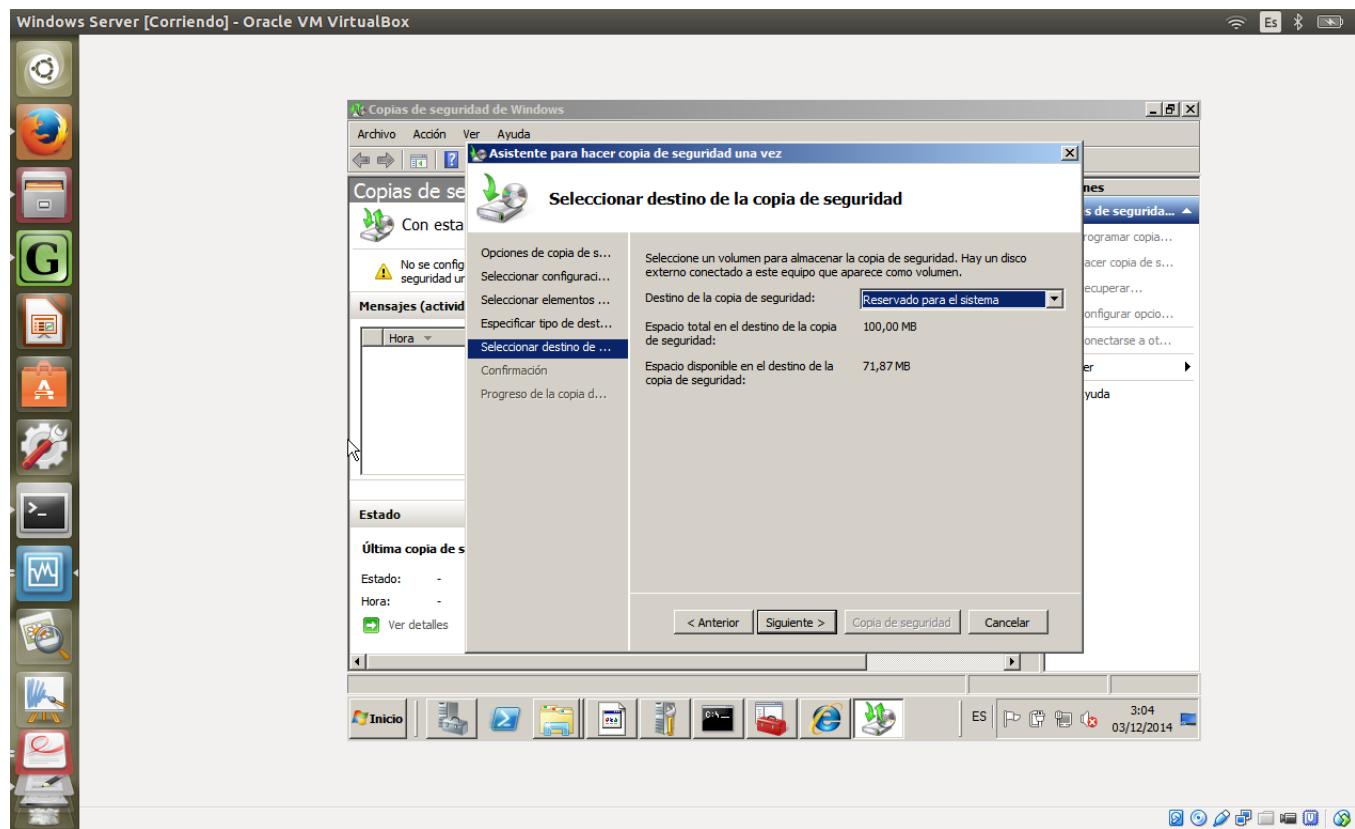


Figura 3.9: haciendo copia de seguridad

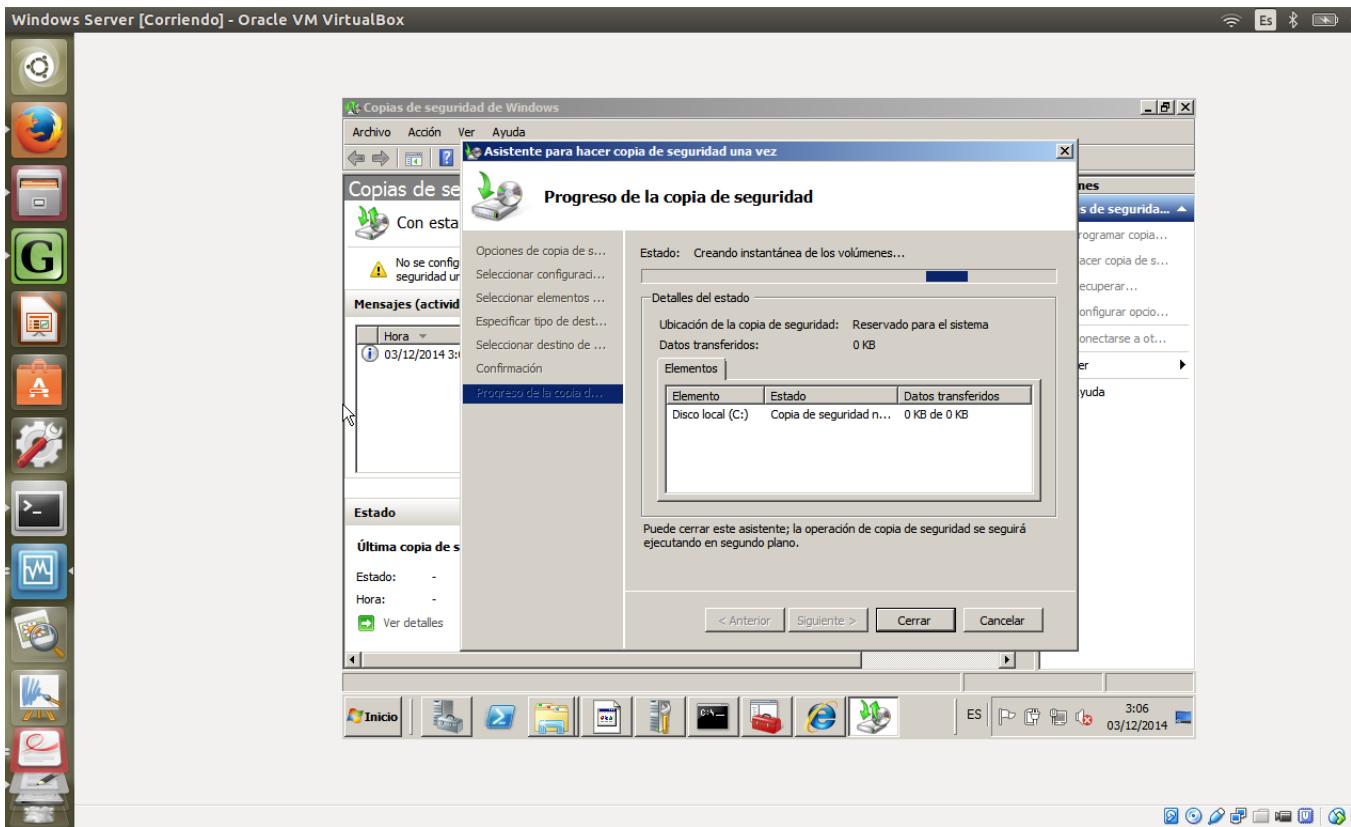


Figura 3.10: proceso de crear copia de seguridad

Para restaurar la copia, se utiliza el mismo software: en el panel de la derecha seleccionamos ->restaurar copia de seguridad y se nos abre un asistente de recuperación:

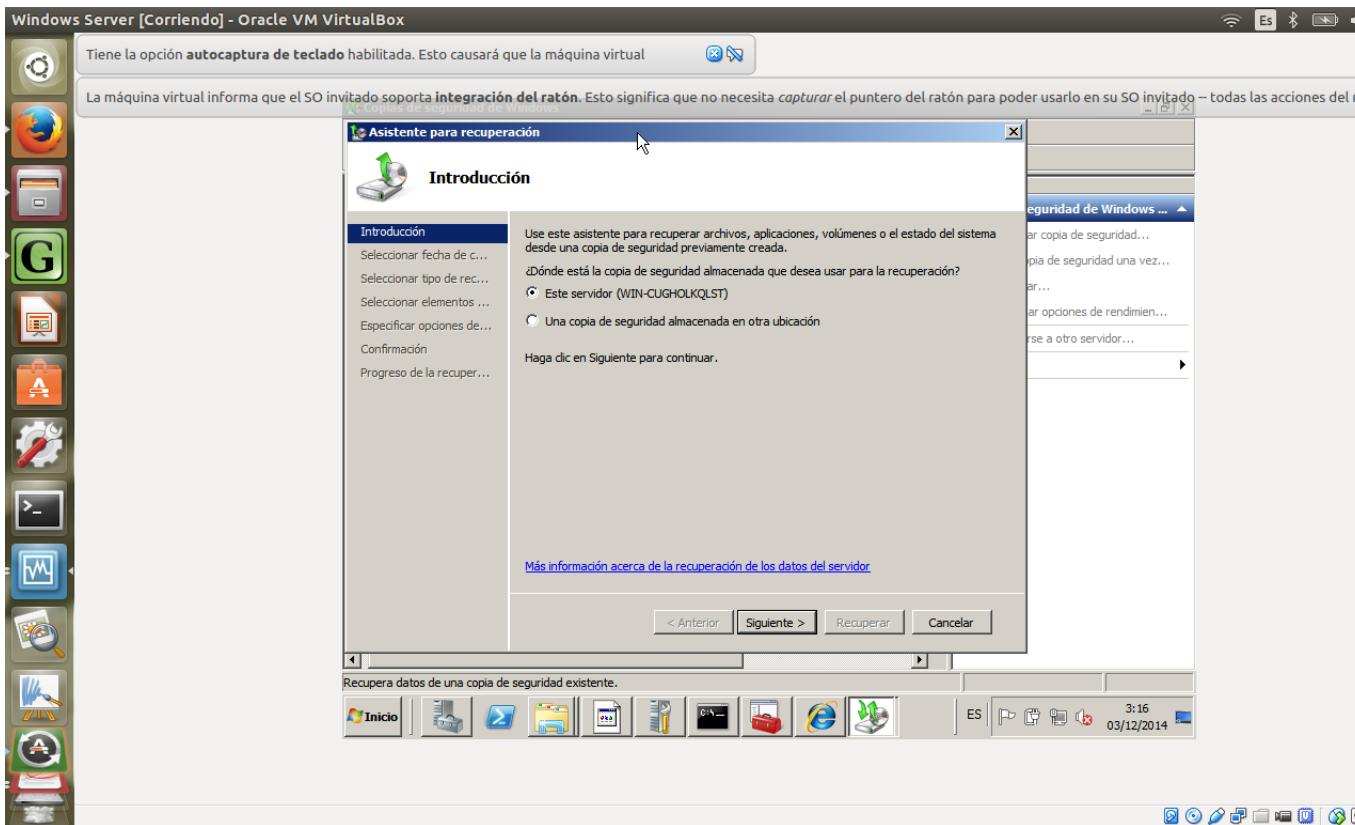


Figura 3.11: asistente restaurador de copia de seguridad

seleccionamos una copia de seguridad almacenada en otra ubicación y especificamos la unidad para recuperar la copia:

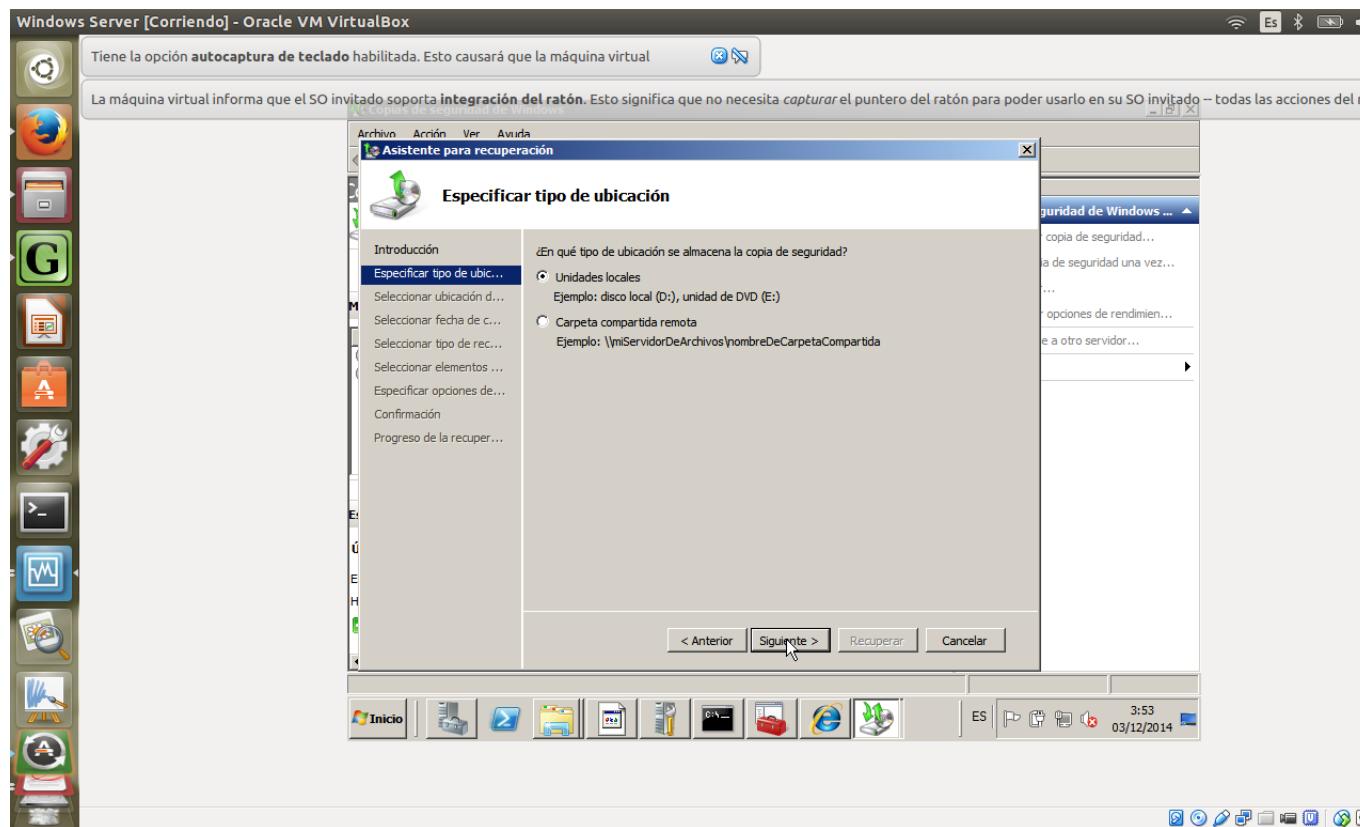


Figura 3.12: asistente restaurador de copia de seguridad

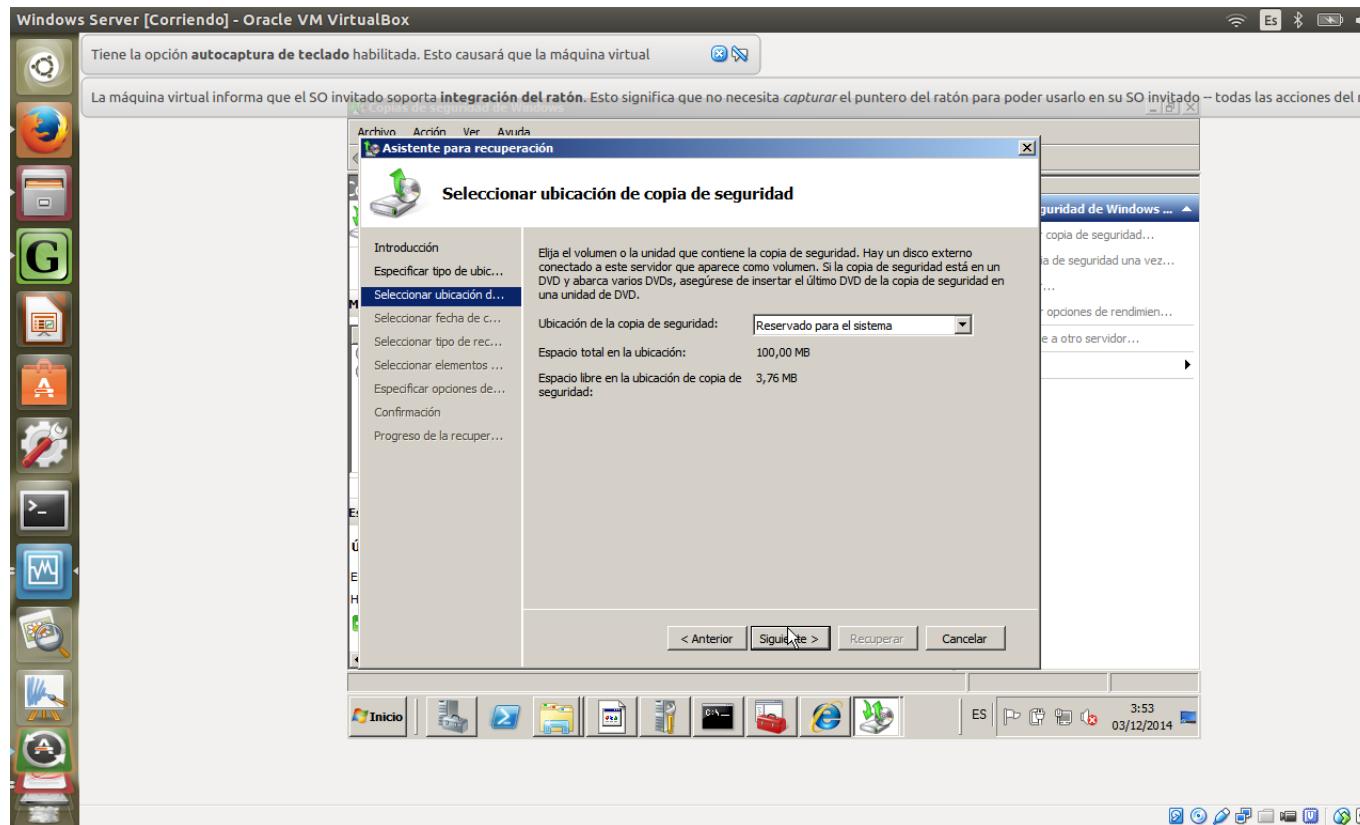


Figura 3.13: asistente restaurador de copia de seguridad

4. Cuestión 4 : ¿Cómo se abre una consola en Windows? ¿Qué comando hay que ejecutar para editar el registro? Muestre su ejecución con capturas de pantalla.

Una consola en Windows se puede abrir, por ejemplo en el botón de Inicio->símbolo del sistema. Para editar un registro, hay que ejecutar el comando regedit:

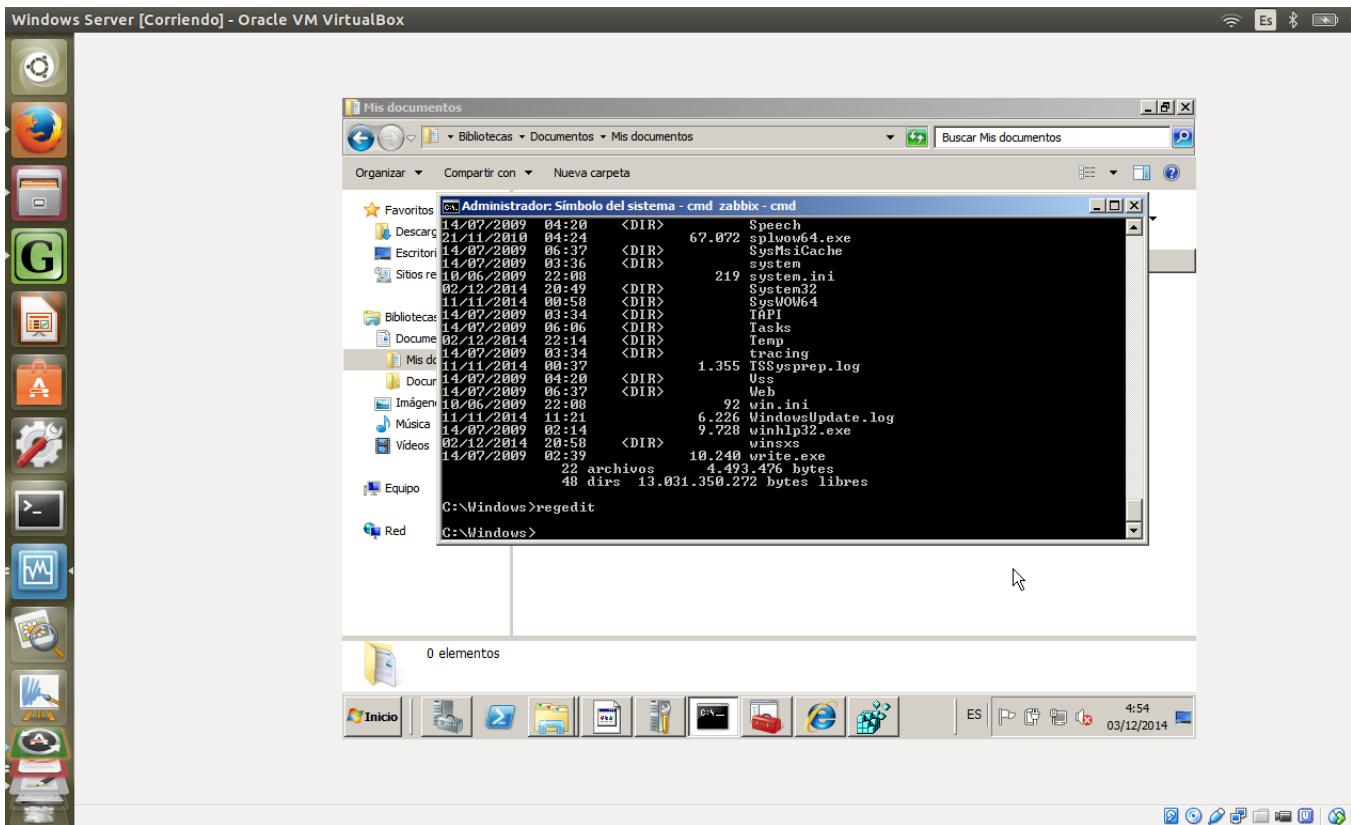


Figura 4.1: comando regedit

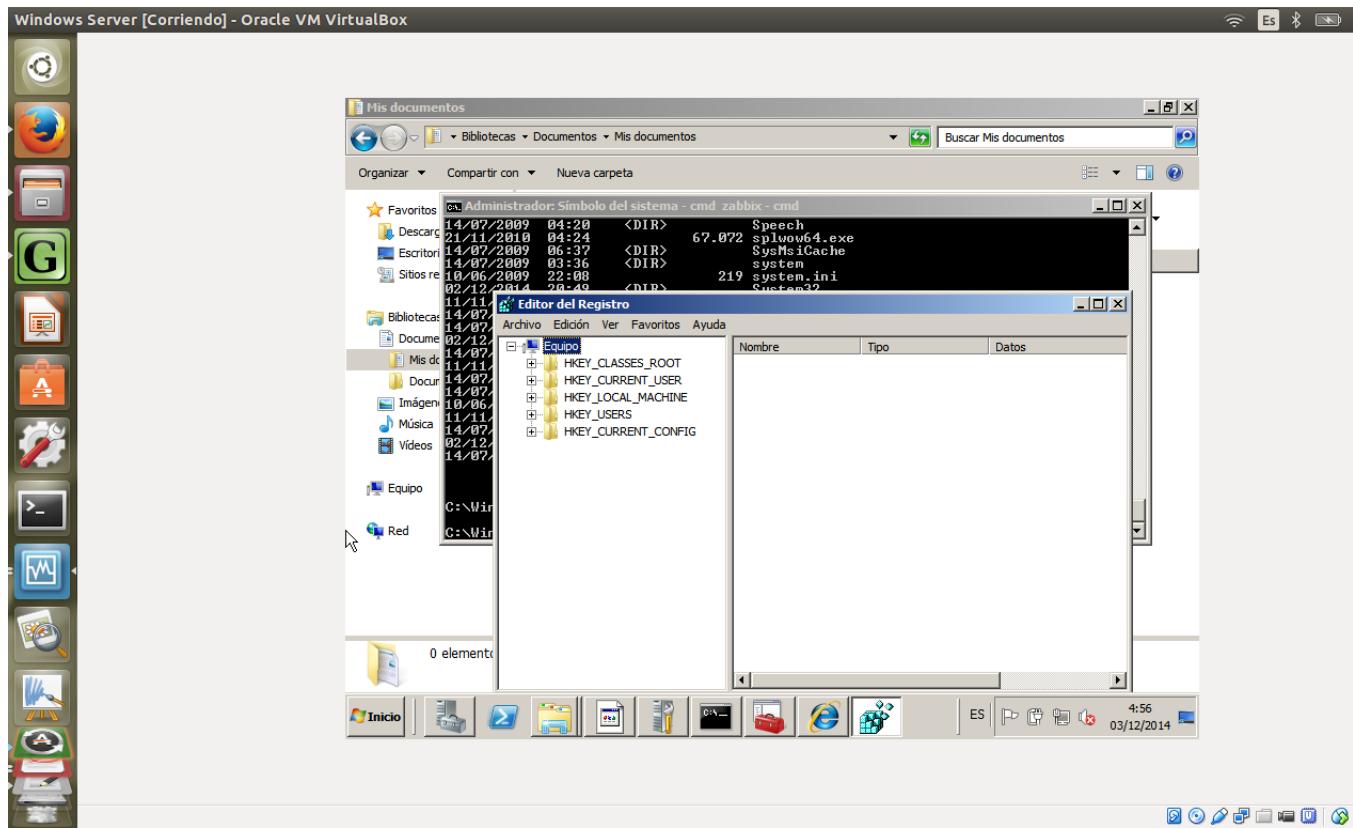


Figura 4.2: resultado de ejecutar el comando regedit

[nor]

5. Cuestión 5 : Las cadenas de caracteres y valores numéricos tienen distintos tipos. Busque en la documentación de Microsoft y liste todos los tipos de valores. Las variables relacionadas con el sistema están alojadas en HKEY_LOCAL_MACHINE (HKLM).

5.1. valores numéricos:

REG_BINARY, REG_DWORD, REG_DWORD_LITTLE_ENDIAN, REG_DWORD_BIG_ENDIAN, REG_QWORD,
REG_QWORD_LITTLE_ENDIAN

5.2. Valores de cadena de caracteres:

REG_EXPAND_SZ, REG_LINK, REG_MULTI_SZ, REG_SZ

5.3. Valores no definidos:

REG_NONE

[msd]

6. Cuestión 6 : Enumere qué elementos se pueden configurar en Apache y en IIS para que Moodle funcione mejor.

Elementos configurables:

6.1. En Apache:

1. Utilizar "Apache Lounge" dado a que tiene mejoras de estabilidad en comparación con la versión oficial de Apache.
2. Establecer el número máximo de clientes (con la variable MaxClients) de forma correcta. Para ello es aconsejable emplear una fórmula y tener en cuenta la memoria de la que se dispone.
3. reducir el número de módulos cargados por Apache al mínimo necesario.
4. Utilizar la versión más reciente de apache (que actualmente es Apache 2), dado que tiene mejoras con respecto al uso de memoria.
5. En SO basados en Linux Unix, establecer la variable MaxRequestsPerChild a un valor entre 20 y 30.
6. En servidores con mucha carga, establecer KeepAlive Off o reducir el tiempo de KeepAliveTimeOut a un valor entre 2 y 5 segs.
7. Utilizar un servidor proxy en modo cache entre los usuarios y el servidor Apache.
8. Como alternativa a KeepAlive Off, utilizar un servidor proxy en modo caché entre usuarios y servidor Apache.
9. Utilizar un archivo .htaccess o poner AllowOverride=none para evitar búsquedas descontroladas en el directorio web.
10. Establecer correctamente la variable DirectoryIndex para evitar que el servidor procese peticiones ambiguas.
11. Deshabilitar el modo desarrollador poniendo ExtendedStatus=Off.
12. Poner HostnameLookups en Off para reducir el tiempo de latencia del DNS.
13. Reducir el tiempo de la variable TimeOut a un valor entre 30 y 60 segs.
14. Evitar las Multiviews ya que realizan una exploración del directorio, para reducir el uso de disco.
15. Habilitar Caching y la compresión de archivos.

6.2. En ISS:

1. Poner ListenBackLog (equivalente a KeepAliveTimeOut) entre 2 y 5 segs.
2. Editar MemCacheSize para determinar cuánta memoria podrá utilizar ISS para su caché.
3. Editar MaxCachedFileSize , estableciendo el mayor tamaño que un archivo puede tener para ser almacenado en caché.
4. Crear un registro DWORD que se llame ObjectCacheTTL, donde se establezca el tiempo que un archivo puede estar en caché.

[doc]

7. Cuestión 7 : Ajuste la compresión en el servidor y analice su comportamiento usando varios valores para el tamaño a de archivo 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 administrar IIS vamos a Panel de Control ->Sistema y Seguridad

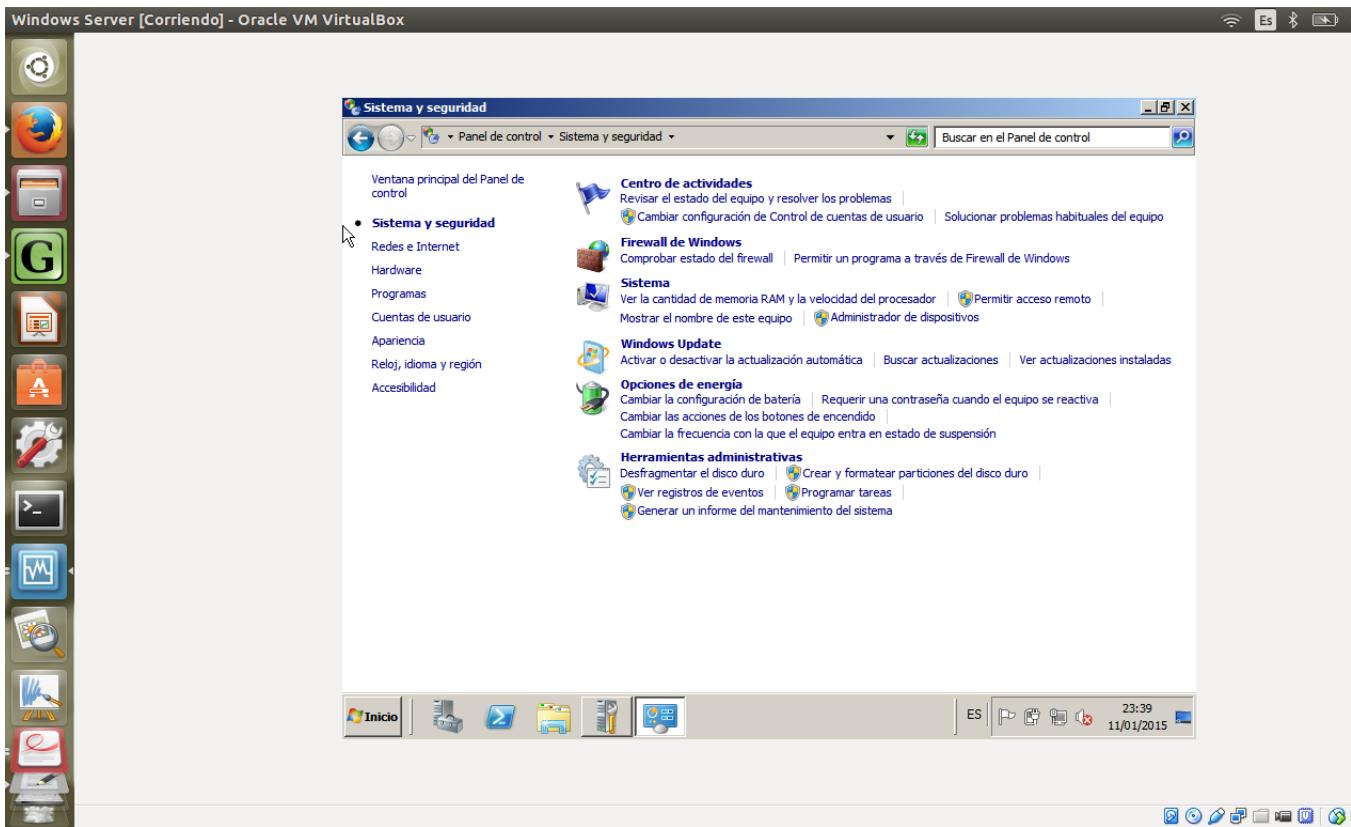


Figura 7.1: sistema y seguridad

seleccionamos “Herramientas Administrativas”... y ahora seleccionamos el ISS manager

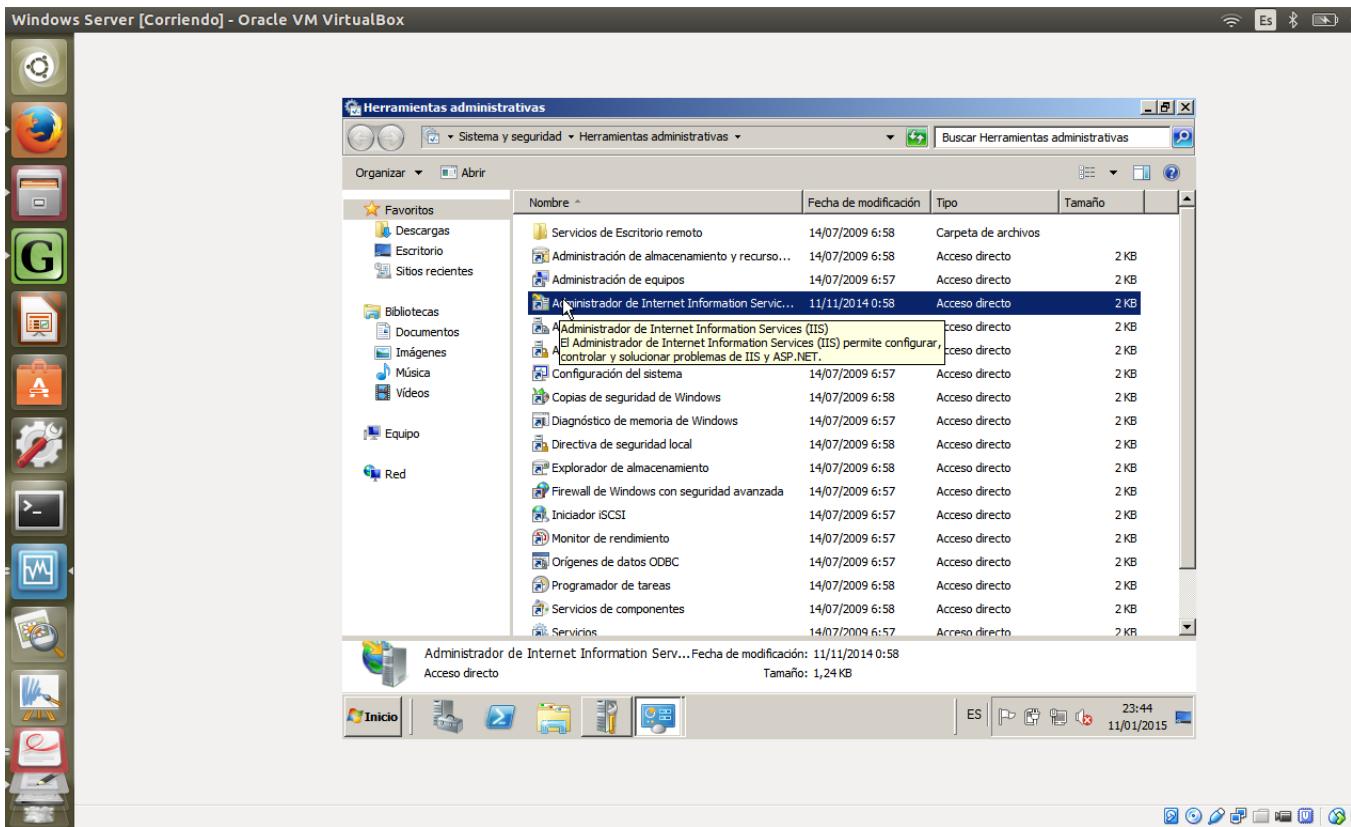


Figura 7.2: Herramientas Administrativas

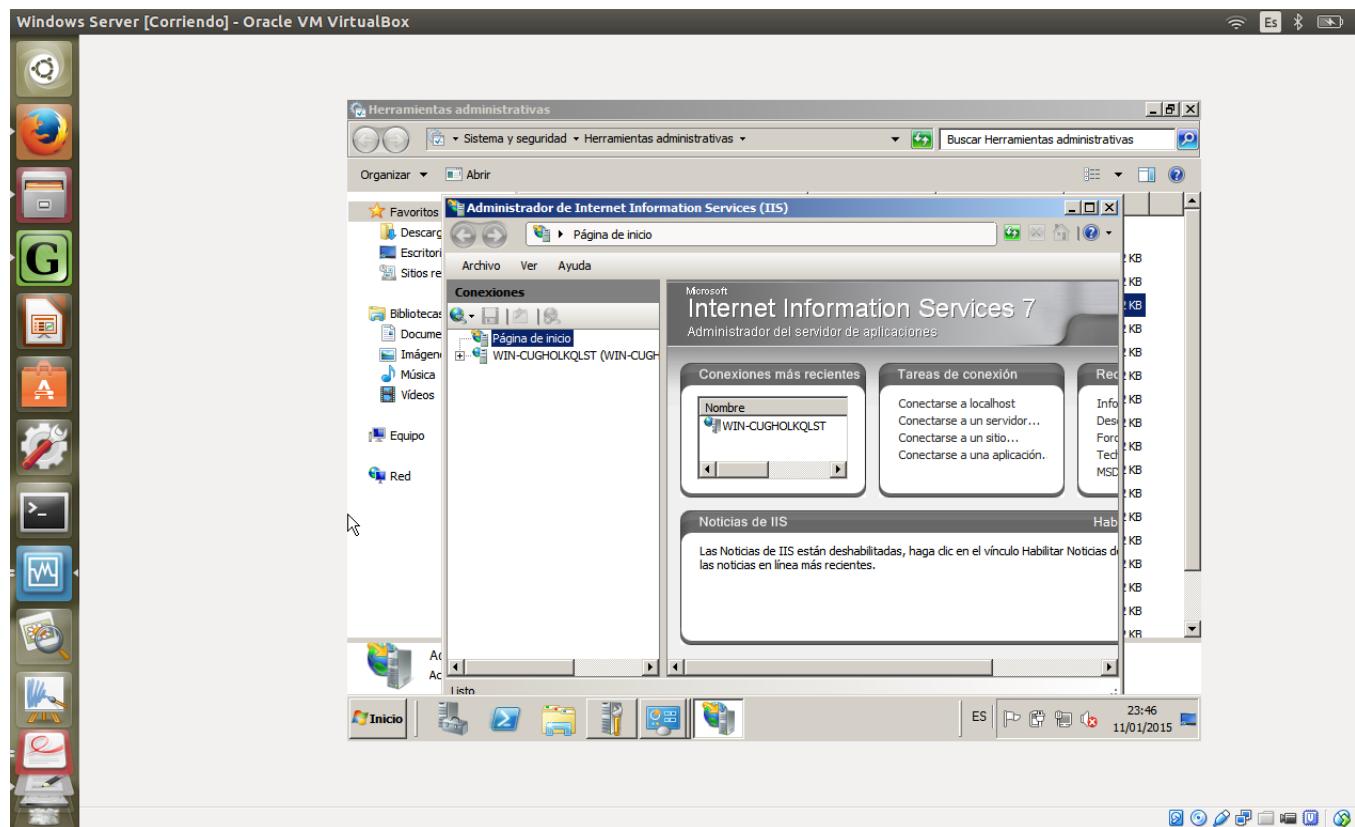


Figura 7.3: ISS manager

seleccionamos nuestro servidor, y a continuación clickeamos en Compression:

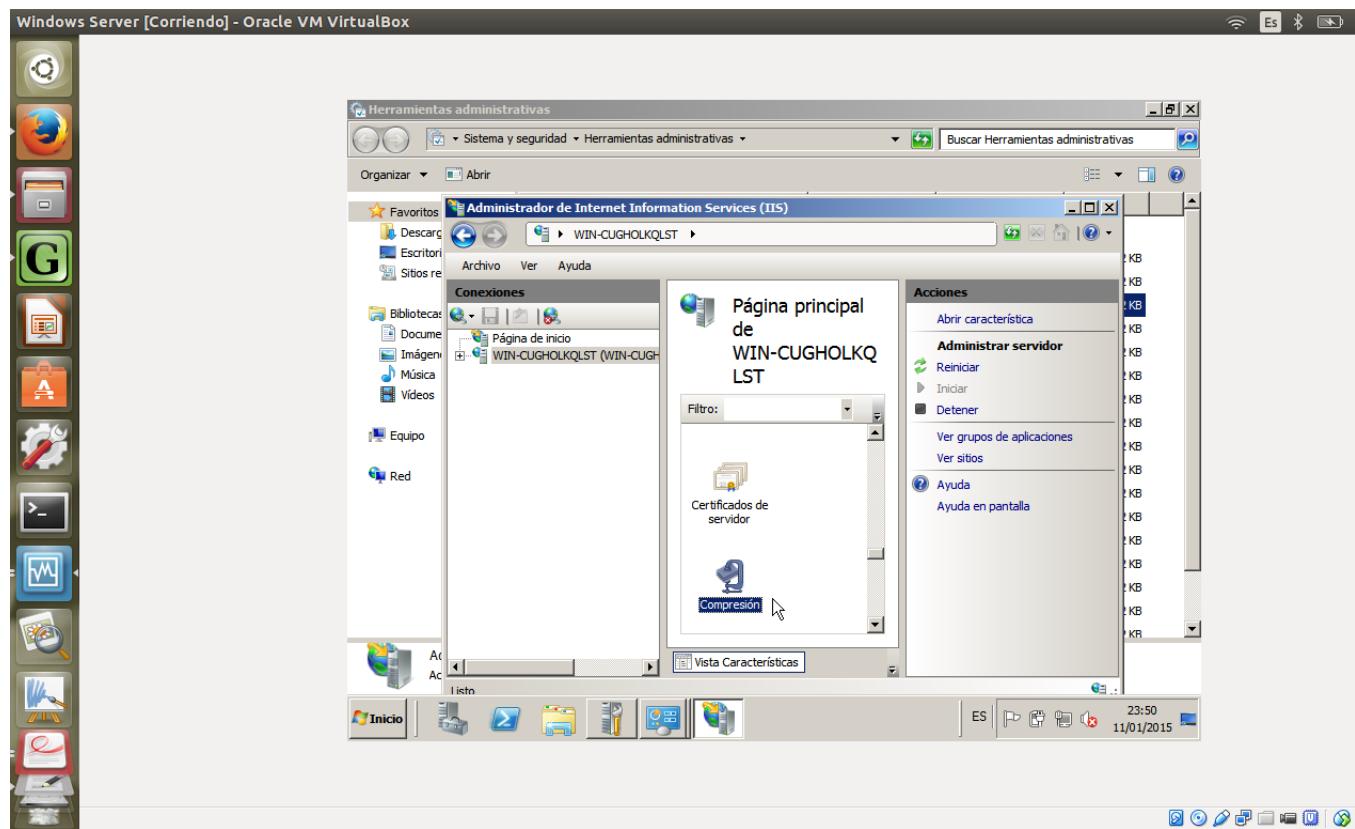


Figura 7.4: ISS manager

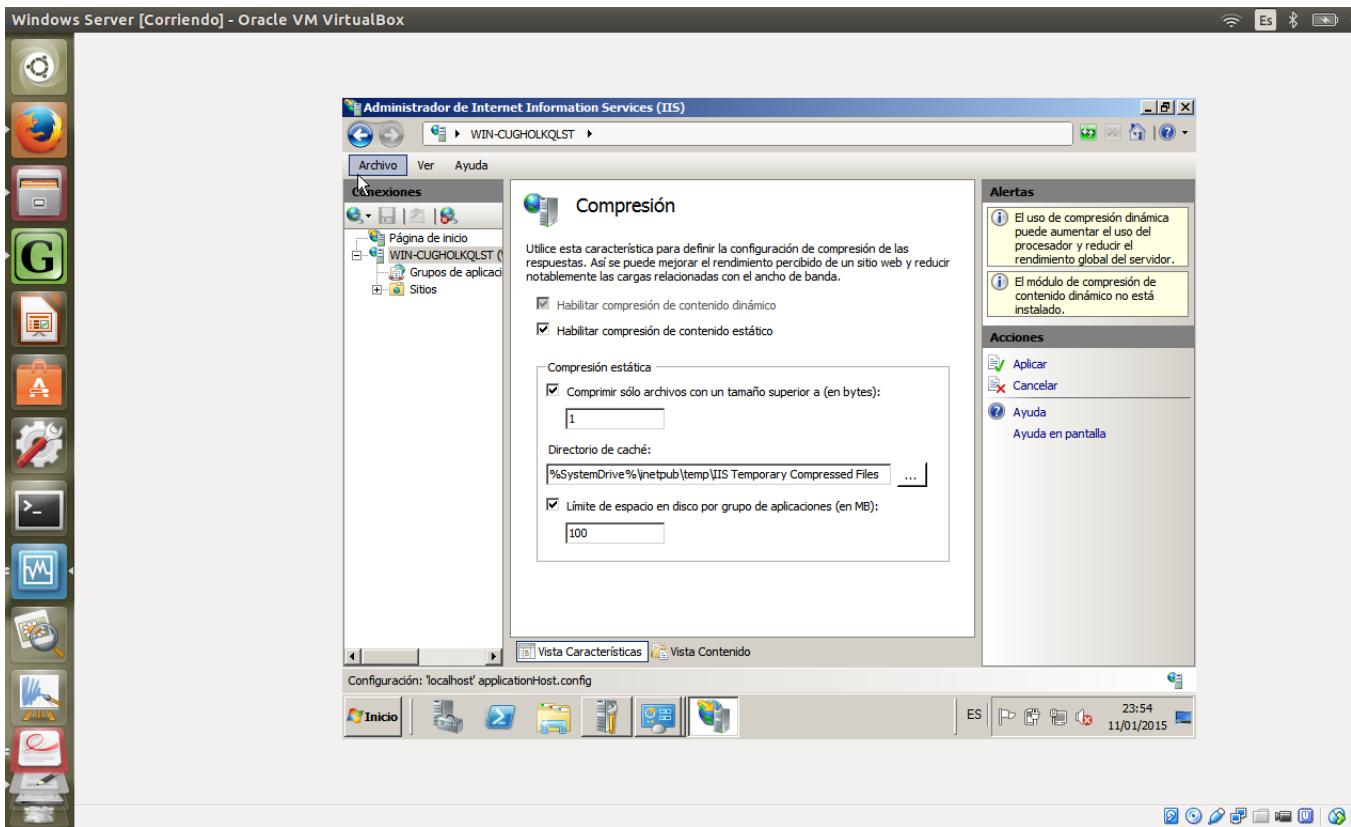


Figura 7.5: ISS manager ->Compression

Configuramos para que se compriman todos los archivos. Pero esto no es todo. Para poder configurar IIS para comprimir páginas, hay que editar el archivo de configuración de nuestra aplicación web, que se encuentra en el directorio web.config, dado que IIS sólo comprime las páginas que cumplen la restricción del tamaño y que además también cumplen que son accedidas frecuentemente. (Se considera acceder frecuentemente a 2 accesos cada 10 segundos).

Antes de modificar este archivo, observamos el resultado de ejecutar curl en la consola:

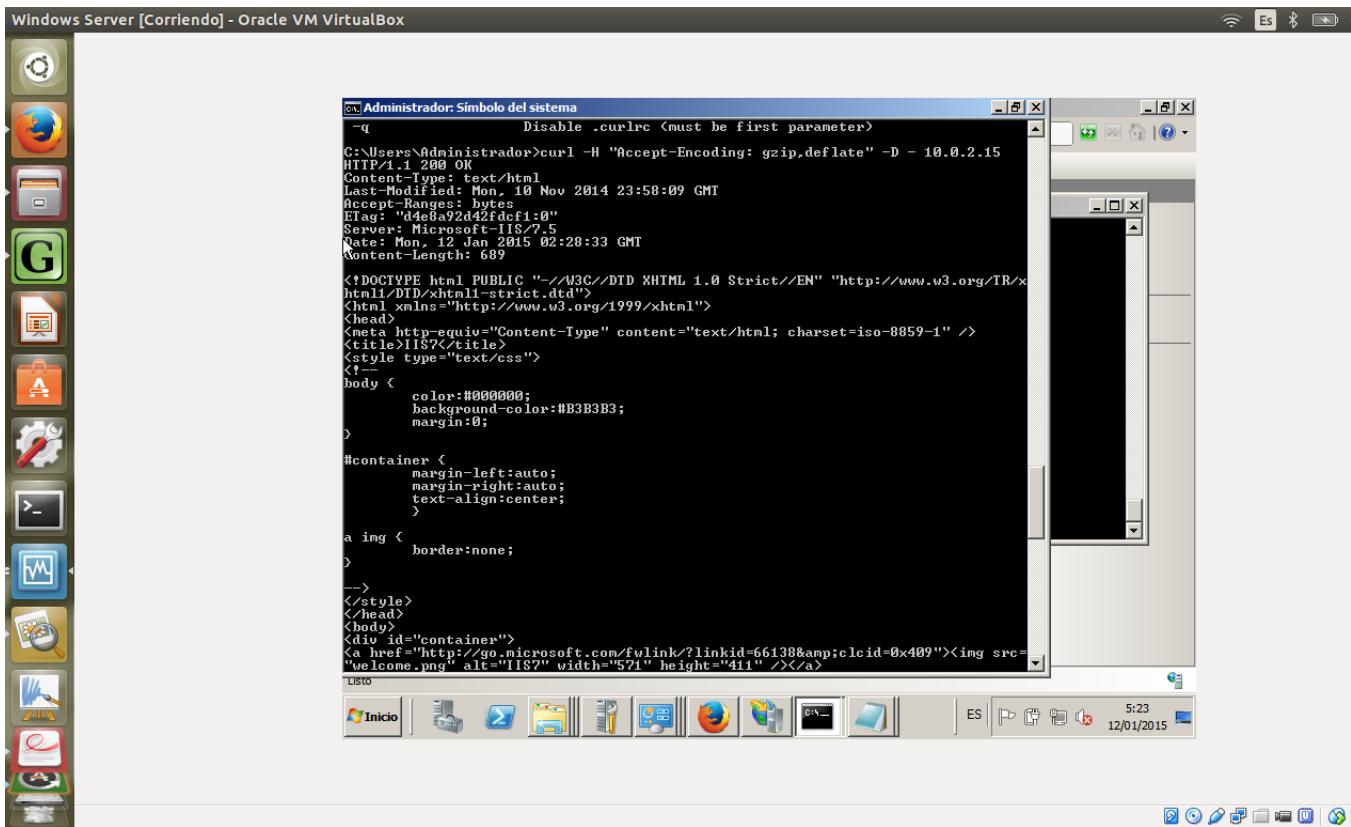


Figura 7.6: ejecutando curl antes de configurar el archivo applicationHost.config (no comprime)

y vemos que en la cabecera no hay parámetro de compresión, el archivo está en texto plano y el tamaño ahora es de 689 Bytes.

Ahora vamos a editar el archivo de configuración que necesitamos editar para que la compresión, que ya está habilitada, sea efectiva: Lo primero es desbloquear el archivo:

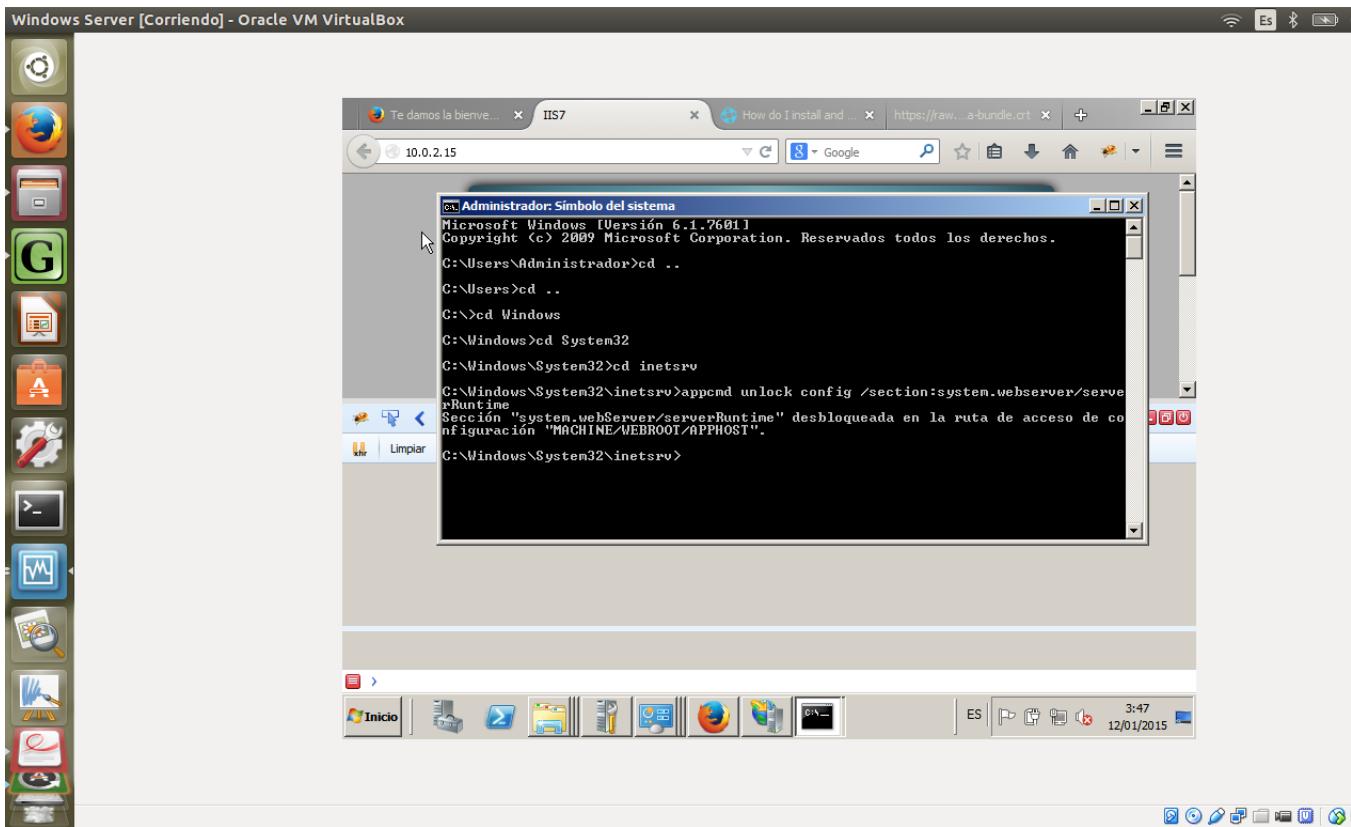


Figura 7.7: desbloqueando archivo de conf

Ahora, editamos el archivo applicationHost.config añadiendo dentro del nodo <system.webServer> la siguiente línea: <serverRuntime frequentHitThreshold="1" frequentHitTimePeriod="10:00:00"/> así estamos diciendo que se va a considerar como "acceso frecuente" 1 acceso cada 10 horas.

Encontramos en C:

Windows

System32

inetsrv

config el archivo applicationHost.config , donde buscamos la sección de <system.Server> y la editamos añadiendo la línea que hemos dicho antes:

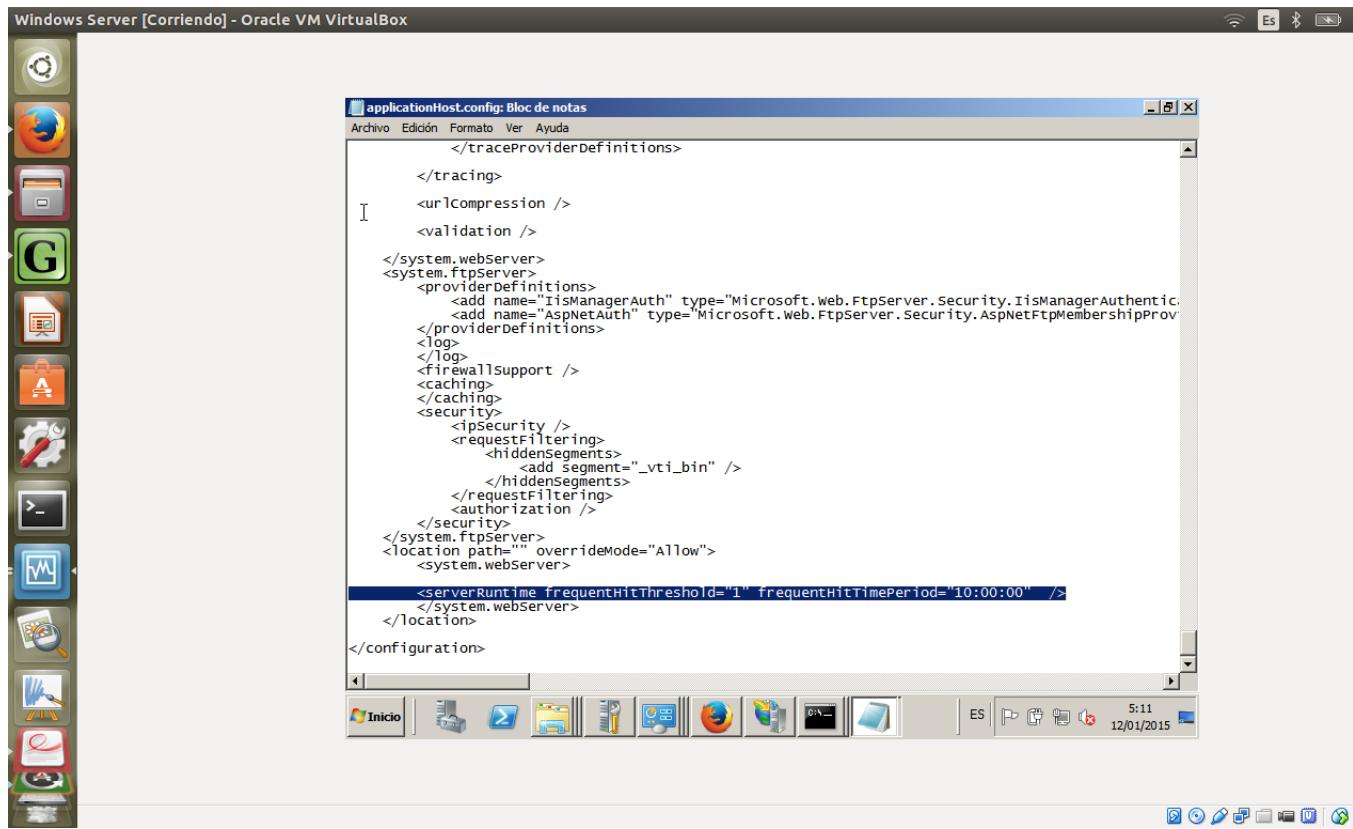


Figura 7.8: editando applicationHost.config

Por último, guardamos los cambios y reiniciamos el servidor. Al ejecutar Curl de nuevo, sobre la página de ISS, observamos que sí aparece parámetro de compresión en la cabecera : Content-Encoding: gzip y que ahora el tamaño del contenido es de 457b:

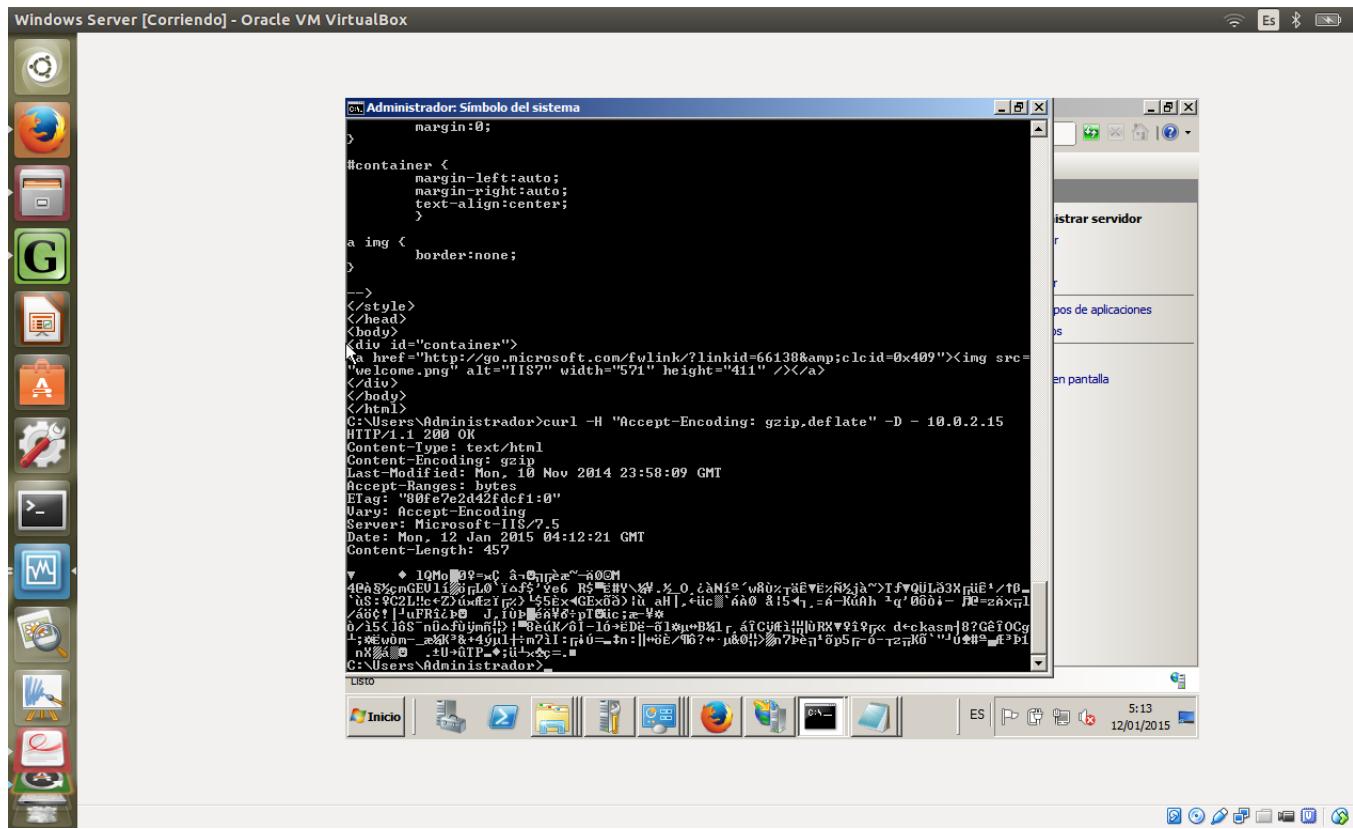


Figura 7.9: resultado de ejecutar Curl tras habilitar compresión y modificar correctamente applicationHost.config

[sta] [wwwa]

- 8. Cuestión 8 : 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.**
- (9.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.**

Vamos a modificar parámetros del kernel de Ubuntu para mejorar el tiempo de cómputo en programas de cálculo. Modificaremos pues parámetros que influyan en la planificación de la CPU, y la de las tareas. Yo voy a hacer las pruebas sobre un programa que hice el año pasado para la asignatura de SCD(para la práctica 1 en concreto), y que calcula una integral de forma concurrente, la cual da como resultado el número pi, y nos imprime por pantalla los ciclos de reloj consumidos. Usaré el comando time para medir tiempos de ejecución.

```

#include<iostream>
#include<ctime>
//#include"fun_tiempos.h"
#include<pthread.h>

using namespace std;

unsigned long m;//numero de muestras
unsigned long n;//número de hebras
double resultado_parcial[100];//vector de resultados parciales

double f(double x ){ //implementa función f:
    return 4.0/(1+ x*x); //f(x)=4~(1+x^2)
}
void*funcion_hebra(void*ih_void ){//función que ejecuta cada hebra
    unsigned long ih= (unsigned long) ih_void ;
    //unsigned long ih= (unsigned long) ih_void ; //número o indice de esta hebra
    double sumap= 0.0 ;//calcular suma parcial en "sumap";
    for ( unsigned long i=(ih * m)/n; i<((ih+1)*m)/n;i++){
        sumap=sumap + f((i+0.5)/m);
    }
    resultado_parcial[ih] =sumap/m; //guardar suma parcial en vector.
}
double calcular_integral_concurrente(){

//crear y lanzar n hebras, cada una ejecuta "funcion concurrente"
pthread_t id_hebra[n];
double sumaFinal=0;

for ( unsigned long i=0; i < n; i++){
    pthread_create(&(id_hebra[i]),NULL,funcion_hebra,(void *)i );
}
//esperar (join) a que termine cada hebra  sumar el resultado

```

Figura 8.1: Integral.cpp

```

//esperar (join) a que termine cada hebra, sumar su resultado
pthread_join(id_hebra[0],NULL);
pthread_join(id_hebra[1],NULL);
pthread_join(id_hebra[2],NULL);
pthread_join(id_hebra[3],NULL);

for (unsigned long i=0; i < n;i++){
    sumaFinal=sumaFinal + resultado_parcial[i];
}
//devolver resultado completo
return sumaFinal;
}

int main(){
    cout << "introduce un número de muestras " << endl;
    cin >> m;
    cout << "introduce un número de hebras " << endl;
    cin >> n;
    long int ti,tf;
    ti= clock();
    //struct timespec inicio=ahora();
    calcular_integral_concurrente();
    //struct timespec fin=ahora();
    tf= clock();
    cout << calcular_integral_concurrente() << endl;
    cout << "el número pi= 3.14159265358979323846...." << endl;
    cout << "tiempo consumido de CPU (clock ticks)"<< tf-ti<< endl;
    /*cout << "tiempo transcurrido == "
     << duracion(&inicio, &fin)
     <<"seg"<<endl;
    */
}

```

Figura 8.2: Integral.cpp

Para mejorar la ejecución de este programa modificaremos sched_min_granularity_ns y sched_latency_ns que por defecto traen los valores:

```

root@cristina:/home/cristina# sysctl -a | grep "sched_min_granularity"
kernel.sched_min_granularity_ns = 2250000
root@cristina:/home/cristina# sysctl -a | grep "sched_latency_ns"
kernel.sched_latency_ns = 18000000

```

Figura 8.3: valores por defecto de las variables

Ejecutamos antes de modificar las variables:

```

root@cristina:/home/cristina/Escritorio/ISE# time ./Integral
introduce un número de muestras
15
introduce un número de hebras
10
3.14196
el número pi= 3.14159265358979323846.....
tiempo consumido de CPU (clock ticks)1446

real    0m6.710s
user    0m0.007s
sys     0m0.000s

```

Figura 8.4: ejecución con las variables por defecto

Ahora, reduciremos `sched_latency_ns` a 3000000 y aumentaremos `sched_min_granularity_ns` a 10000000, por ejemplo, con el fin de disminuir la planificación y aumentar la granularidad del procesador, con lo que cada tarea podría ejecutarse durante algo más de tiempo, y así no habría tanta penalización:

```

root@cristina:/home/cristina/Escritorio/ISE# sysctl -a | grep sched_min_granularity
kernel.sched_min_granularity_ns = 2250000
root@cristina:/home/cristina/Escritorio/ISE# sysctl -w kernel.sched_min_granularity_ns=10000000
kernel.sched_min_granularity_ns = 10000000
root@cristina:/home/cristina/Escritorio/ISE# sysctl -a | grep sched_latency_ns
kernel.sched_latency_ns = 18000000
root@cristina:/home/cristina/Escritorio/ISE# sysctl -w kernel.sched_latency_ns=3000000
kernel.sched_latency_ns = 3000000
root@cristina:/home/cristina/Escritorio/ISE# time ./Integral
introduce un número de muestras
15
introduce un número de hebras
10
3.14196
el número pi= 3.14159265358979323846.....
tiempo consumido de CPU (clock ticks)1422

real    0m6.021s
user    0m0.007s
sys     0m0.000s
root@cristina:/home/cristina/Escritorio/ISE#

```

Figura 8.5: ejecución con las variables modificadas

Y observamos que el tiempo real ha pasado de 6.710s a 6.021s. Se ha reducido, aunque no gran cosa. Vamos a volver a probar: podemos, por ejemplo, reducir aún más la variable `sched_latency_ns` a 300000 (le quitamos un 0):

```

introduce un número de hebras
10
3.14196
el número pi= 3.14159265358979323846.....
tiempo consumido de CPU (clock ticks)1422

real    0m6.021s
user    0m0.007s
sys     0m0.000s
root@cristina:/home/cristina/Escritorio/ISE# sysctl -w kernel.sched_latency_ns=300000
kernel.sched_latency_ns = 300000
root@cristina:/home/cristina/Escritorio/ISE# time ./Integral
introduce un número de muestras
15
introduce un número de hebras
10
3.14196
el número pi= 3.14159265358979323846.....
tiempo consumido de CPU (clock ticks)1318

real    0m4.427s
user    0m0.000s
sys     0m0.006s

```

Figura 8.6: ejecución con las variables modificadas 2

Ahora vemos que el tiempo real que nos imprime ya no es 6.021s, ni mucho menos, el tiempo inicial que era de 6.710s. Ahora nos imprime un tiempo de 4.427s. La diferencia comienza a notarse más.

[oak]

*Bibliografía

Referencias

- [doc] docs.moodle.org. *Performance_recommendations*. https://docs.moodle.org/23/en/Performance_recommendations.
- [msd] msdn.microsoft.com. *Registry Value Types*. <http://msdn.microsoft.com/en-us/library/windows/desktop/ms724884%28v=vs.85%29.aspx>.
- [nor] norfipc.com. *claves-registro-modificar-editar*. <http://norfipc.com/registro/claves-registro-modificar-editar.html>.
- [oak] oakbytes.wordpress.com. *linux-scheduler*. <https://oakbytes.wordpress.com/2012/06/06/linux-scheduler-cfs-and-latency/>.
- [sta] stackoverflow.com. *gzip-compression-on-iis-7-5-is-not-working*. [stack overflow.com/questions/6938713/gzip-compression-on-iis-7-5-is-not-working](http://stackoverflow.com/questions/6938713/gzip-compression-on-iis-7-5-is-not-working).
- [sys] sysadmin.vazqueznanini.com. *sysctl*. <http://sysadmin.vazqueznanini.com.ar/configuraciones/etc-sysctl-conf>.
- [uni] unix.stackexchange.com. *why-is-swappiness-set-to-60-by-default*. <http://unix.stackexchange.com/questions/88693/why-is-swappiness-set-to-60-by-default>.

- [wwwa] www.iis.net. *planning-your-iis-architecture*. <http://www.iis.net/learn/get-started/planning-your-iis-architecture/editing-configuration-files>.
- [wwwb] www.kernel.org. *laptop-mode*. <https://www.kernel.org/doc/Documentation/laptops/laptop-mode.txt>.