

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

Memoria Práctica 1

José Álvaro Garrido López

25 de noviembre de 2016

Índice

1 Cuestión 1.	5
1.1 a) Liste los argumentos de yum necesarios para instalar, buscar y eliminar paquetes.	5
1.2 b) ¿Qué ha de hacer para que yum pueda tener acceso a Internet en el PC del aula?(Pistas: archivo de configuración en /etc, proxy: stargate.ugr.es:3128)	5
1.3 c) ¿Cómo añadimos un nuevo repositorio?	6
2 Cuestión 2.	7
2.1 a) Liste los argumentos de apt necesarios para instalar, buscar y eliminar paquetes.	7
2.2 b) ¿Qué ha de hacer para que apt pueda tener acceso a Internet en el PC del aula?(Pistas: archivo de configuración en /etc, proxy: stargate.ugr.es:3128)	7
2.3 c) ¿Cómo añadimos un nuevo repositorio?	8
3 Cuestión 3.	9
3.1 a) ¿Con qué comando puede abrir/cerrar un puerto usando ufw? Muestre un ejemplo de cómo lo ha hecho	9
3.2 b) ¿Con qué comando puede abrir/cerrar un puerto usando firewall-cmd en CentOS? Muestre un ejemplo de cómo lo ha hecho	10
3.3 c) Utilice el comando nmap para ver que, efectivamente, los puertos están accesibles	10
4 Cuestión 4. ¿Qué diferencia hay entre telnet y ssh?	12
5 Cuestión 5.	12
5.1 a) ¿Para qué sirve la opción -X?	12
5.2 b) Ejecute remotamente, es decir, desde la máquina anfitriona (si tiene Linux) o desde la otra máquina virtual, el comando gedit en una sesión abierta con ssh. ¿Qué ocurre?	12
6 Cuestión 6. Muestre la secuencia de comandos y las modificaciones a los archivos correspondientes para permitir acceder a la consola remota sin introducir la contraseña. Pruebe que funciona. (Pistas: ssh-keygen, ssh-copy-id)	13
7 Cuestión 7. ¿Qué archivo es el que contiene la configuración del servicio ssh? ¿Qué parámetro hay que modificar para evitar que el usuario root acceda? Cambie el puerto por defecto y compruebe que puede acceder.	15
8 Cuestión 8. Indique si es necesario reiniciar el servicio ¿Cómo se reinicia un servicio en Ubuntu? ¿Y en CentOS? Muestre la secuencia de comandos para hacerlo.	16

9 Cuestión 9. Muestre los comandos que ha utilizado en Ubuntu Server y en CentOS (aunque en este último puede utilizar la GUI, en tal caso, realice capturas de pantalla). Compruebe que la instalación ha sido correcta.	17
10 Cuestión 10. Realice la instalación usando GUI o PowerShell y compruebe que el servicio está funcionando accediendo a la MV a través de la anfitriona	18
11 Cuestión 11. Muestre un ejemplo de uso del comando (p.ej. http://fedoraproject.org/wiki/VMWare)	
12 Cuestión 12. Realice la instalación de esta aplicación y pruebe a modificar algún parámetro de algún servicio. Muestre las capturas de pantalla pertinentes así como el proceso de instalación.	22
13 Cuestión 13. Instale phpMyAdmin, indique cómo lo ha realizado y muestre algunas capturas de pantalla. Configure PHP para poder importar BDs de hasta 25MiB (en vez de los 8 MiB de límite por defecto). Indique cómo ha realizado el proceso y muestre capturas de pantalla.	24
14 Cuestión 14. Viste al menos una de las webs de los software mencionados y pruebe las demos que ofrecen realizando capturas de pantalla y comentando qué está realizando	25
15 Cuestión 15.	28
15.1 a) Ejecute los ejemplos de find, grep	28
15.2 b) Escriba el script que haga uso de sed para cambiar la configuración de ssh y reiniciar el servicio	29
15.3 c) Muestre un ejemplo de uso para awk	31
16 Cuestión 16. Escriba el script para cambiar el acceso a ssh usando PHP o Python	31
17 Cuestión 17. Abra una consola de PowerShell y pruebe a parar un programa en ejecución (p.ej), realice capturas de pantalla y comente lo que muestra	33
18 Opinión sobre la práctica	35

Índice de figuras

1.1. Edición del fichero de configuración de <code>/etc/yum.conf</code> con nano	5
1.2. Adición de la sentencia <code>proxy = http://stargate.ugr.es:3128</code> en <code>/etc/yum.conf</code>	6
2.1. Edición del fichero de configuración de <code>/etc/apt/apt.conf</code> con nano	8
3.1. Admisión de la entrada de paquetes tcp en puerto 80	9
3.2. Rechazo de la entrada de paquetes tcp en puerto 80	9
3.3. Habilitación de la entrada de paquetes tcp en puerto 8080 y reinicio del servicio firewalld	10

3.4.	Habilitación de entrada de datos tcp en puerto 10000	11
5.1.	Utilización de <i>ssh</i> con redireccionamiento de X11	13
6.1.	Uso de <i>ssh-keygen</i> y <i>ssh-copy-id</i> para acceder remotamente por SSH sin contraseña	14
6.2.	Uso de <i>ssh-copy-id</i> y acceso sin contraseña a máquina remota por SSH . .	14
7.1.	Fichero de configuración <i>/etc/ssh/sshd_config</i> modificado	15
7.2.	Fichero de configuración <i>/etc/ssh/sshd_config</i> modificado	16
8.1.	Reinicio de un servicio en Ubuntu	16
8.2.	Reinicio de un servicio en CentOS	17
9.1.	Comprobación de servidor apache2 corriendo en CentOS	17
9.2.	Comprobación de servidor apache2 corriendo en Ubuntu	18
9.3.	Comprobación de que PHP y MySQL están instalados	18
10.1.	Ejecución de IIS	19
10.2.	Salida ipconfig máquina virtual	19
10.3.	Acceso desde máquina anfitriona a IIS	20
11.1.	Archivos sum.cpp y sum2.cpp (sum.cpp corregido)	21
11.2.	Creación y aplicación del parche	21
11.3.	Resultado final del archivo	22
12.1.	Instalación de webmin con interfaz gráfica	22
12.2.	Interfaz de webmin	23
12.3.	Interfaz de webmin	23
12.4.	Creación de una nueva tarea de cron	24
13.1.	Instalación de phpmyadmin	24
13.2.	Edición de <i>/etc/php5/apache2/php.ini</i>	25
14.1.	Pantalla de bienvenida al administrador de Webmin	26
14.2.	Nuevo usuario para bases de datos	26
14.3.	Nueva base de datos	27
14.4.	Herramientas interesantes	28
15.1.	Ejecución del comando <i>ps -Af grep firefox</i>	28
15.2.	Archivos en el directorio antes de la ejecución	28
15.3.	Ejecución del comando	29
15.4.	Resultado de la ejecución	29
15.5.	Script de sed en bash	30
15.6.	Resultado de la ejecución del script y comprobación de su funcionamiento	30
15.7.	Fichero inicial	31
15.8.	Resultado de la ejecución del comando	31
16.1.	Script en Python para modificar la configuración de ssh	32
16.2.	Resultado de la ejecución del script y comprobación de su funcionamiento	33
17.1.	Programa Notepad abierto	34
17.2.	Procesos en ejecución	35
17.3.	Ejecución de <i>Stop-Process -Name notepad</i>	35

1. Cuestión 1.

1.1. a) Liste los argumentos de yum necesarios para instalar, buscar y eliminar paquetes.

Según la documentación de CentOS, en [21], se nos explica la sintaxis de yum.

- Para instalar paquetes:

```
yum install <nombre del paquete o paquetes>
```

Instala la última versión del paquete o del grupo de paquetes especificado como argumento. Para instalar paquetes locales existe su versión

```
yum localinstall <ruta absoluto del paquete o paquetes>
```

- Para buscar paquetes:

```
yum search <palabra clave>
```

. De esta forma encontraremos los paquetes que contienen la palabra clave especificada como argumento, un resumen de su uso y nombre del paquete en los repositorios de RPM.

- Para eliminar paquetes:

```
yum remove <nombre del paquete o paquetes>
```

1.2. b) ¿Qué ha de hacer para que yum pueda tener acceso a Internet en el PC del aula?(Pistas: archivo de configuración en /etc, proxy: stargate.ugr.es:3128)

[6] Como podemos observar en 1.1 y en 1.2, tenemos que editar el fichero de configuración de yum localizado en */etc/yum.conf* y añadir

```
proxy = http://stargate.ugr.es:3128
```

```
alvarogl nov 2016 > sudo nano /etc/yum.conf
```

Figura 1.1: Edición del fichero de configuración de */etc/yum.conf* con nano

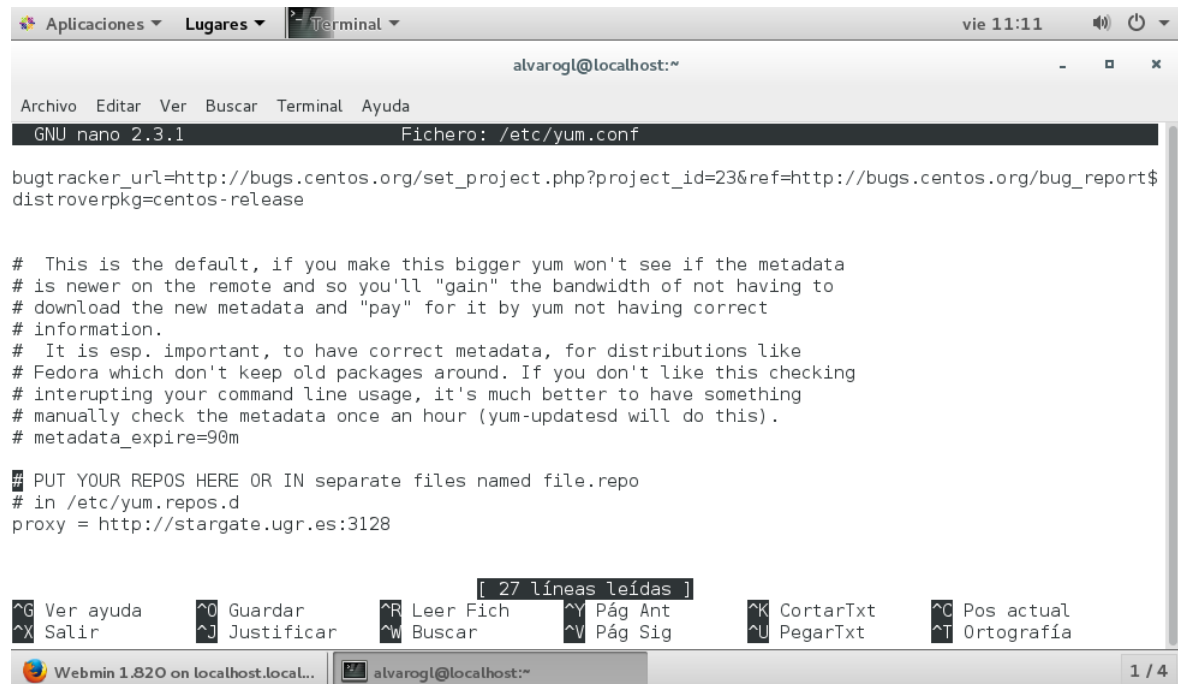


Figura 1.2: Adición de la sentencia `proxy = http://stargate.ugr.es:3128` en `/etc/yum.conf`

1.3. c) ¿Cómo añadimos un nuevo repositorio?

Según el manual online de Ubuntu, en [20], `yum-config-manager` es una orden para administrar y configurar algunas opciones de yum. Entre ellas se encuentra la de añadir un repositorio de una dirección URL o de un archivo especificado, además, el repositorio será habilitado. La sintaxis sería la siguiente:

```
yum-config-manager --add-repo=ADDREPO <url o fichero>
```

2. Cuestión 2.

2.1. a) Liste los argumentos de apt necesarios para instalar, buscar y eliminar paquetes.

Según podemos comprobar en [19], apt es un comando que se utiliza para el manejo de paquetes.

- Para instalar paquetes:

```
apt install <nombre del paquete o paquetes>
```

- Para buscar paquetes:

```
apt search <términos de búsqueda>
```

Busca los paquetes que coincidan con los términos de búsqueda.

- Para eliminar paquetes:

```
apt remove <nombre del paquete o paquetes>
```

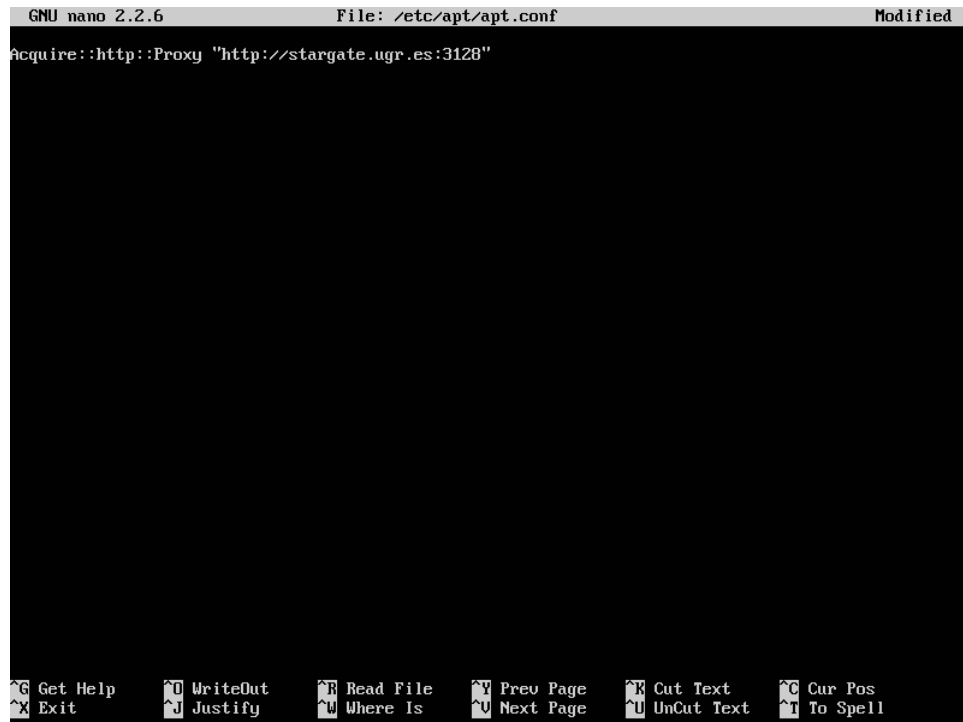
Elimina el paquete o paquetes sin borrar sus ficheros de configuración del sistema. Para eliminar estos ficheros de configuración debemos utilizar *purge*, su sintaxis sería la siguiente: [18]

```
apt purge <nombre del paquete o paquetes>
```

2.2. b) ¿Qué ha de hacer para que apt pueda tener acceso a Internet en el PC del aula?(Pistas: archivo de configuración en /etc, proxy: stargate.ugr.es:3128)

Para ello debemos editar, en este caso, el fichero de configuración de apt en */etc/apt/apt.conf* como se muestra en 2.1 [2] y escribir en él la línea:

```
Acquire::http::Proxy "http://stargate.ugr.es:3128"
```



```
GNU nano 2.2.6      File: /etc/apt/apt.conf      Modified
Acquire::http::Proxy "http://stargate.ugr.es:3128"

^G Get Help      ^O WriteOut      ^R Read File      ^V Prev Page      ^X Cut Text      ^C Cur Pos
^X Exit          ^J Justify       ^W Where Is       ^U Next Page      ^U UnCut Text    ^I To Spell
```

Figura 2.1: Edición del fichero de configuración de */etc/apt/apt.conf* con nano

2.3. c) ¿Cómo añadimos un nuevo repositorio?

Los repositorios en Ubuntu se almacenan en */etc/apt/sources.list* o en */etc/apt/sources.list.d*. Podemos añadir un nuevo repositorio con el comando *add-apt-repository*. La sintaxis sería:

```
add-apt-repository <"repositorio">
```


3. Cuestión 3.

3.1. a) ¿Con qué comando puede abrir/cerrar un puerto usando ufw? Muestre un ejemplo de cómo lo ha hecho

Según [4] y [3], para abrir un puerto, es decir, para permitir la entrada de paquetes en dicho puerto (cada vez que se diga abrir puertos nos referiremos a esto a lo largo de la memoria), primero debemos habilitar *ufw* de la siguiente forma:

```
sudo ufw enable
```

La sintaxis para habilitar la entrada de paquetes es la siguiente:

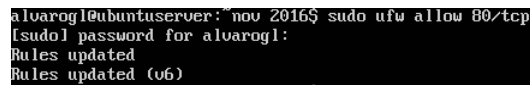
```
sudo ufw allow <port>/<option: protocol>
```

Entonces, para habilitar la entrada de paquetes tcp y udp en el puerto 80 escribiríamos en terminal:

```
sudo ufw allow 80
```

Para habilitar entrada de paquetes solo tcp en el puerto 80, como se observa en 3.1:

```
sudo ufw allow 80/tcp
```



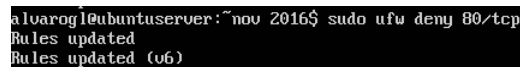
```
alvarog1@ubuntu-server:~$ sudo ufw allow 80/tcp
[sudo] password for alvarog1:
Rules updated
Rules updated (v6)
```

Figura 3.1: Admisión de la entrada de paquetes tcp en puerto 80

Se procedería de forma análoga para los paquetes udp. Para rechazar la entrada de paquetes la sintaxis es:

```
sudo ufw deny <port>/<optional: protocol>
```

En 3.2



```
alvarog1@ubuntu-server:~$ sudo ufw deny 80/tcp
Rules updated
Rules updated (v6)
```

Figura 3.2: Rechazo de la entrada de paquetes tcp en puerto 80

Su uso es similar al de *ufw allow*

3.2. b) ¿Con qué comando puede abrir/cerrar un puerto usando firewall-cmd en CentOS? Muestre un ejemplo de cómo lo ha hecho

Según [8] y [28], la sintaxis para abrir puertos en CentOS hay que iniciar *firewall-cmd*, para ello [26]:

```
systemctl start firewalld.service
```

Como se nos explica en [26], *systemctl* es el comando principal para controlar e inspeccionar *systemd*, una suite de gestión y mantenimiento de servicios.

Para abrir puertos en CentOS, la sintaxis es:

```
firewall-cmd --zone=<set zone> --add-port=<port>/<optional: protocol>
```

Se puede añadir la opción *-permanent* para que el puerto permanezca abierto a pesar de reinicios del sistema.

Por, ejemplo, en 3.3 se habilita la entrada de paquetes tcp en el puerto 8080.

```
[alvarogl@localhost ~]$ sudo firewall-cmd --zone=public --add-port=8080/tcp
success
[alvarogl@localhost ~]$ sudo firewall-cmd --reload
success
```

Figura 3.3: Habilitación de la entrada de paquetes tcp en puerto 8080 y reinicio del servicio firewalld

Después de configurar los puertos, es posible que sea necesario reiniciar el servicio:

```
firewall-cmd --reload
```

De forma análoga, para cerrar puertos:

```
firewall-cmd --zone<set zone> --remove-port=<port>/<optional: protocol>
```

3.3. c) Utilice el comando nmap para ver que, efectivamente, los puertos están accesibles

En [22] se nos explica cómo se utiliza el comando. Una versión resumida de su sintaxis sería:

```
nmap <ip> <optional: -p <port>>
```

Como se aprecia en 3.4, he abierto el puerto 10000 para zona pública desde la máquina virtual de la izquierda, a partir de ahora CentOS 3, para escanear el mapa de puertos desde la máquina virtual de la derecha, a partir de ahora CentOS 4.

En CentOS 4, el puerto 10000 aparecía filtrado, pues el servicio estaba en marcha, si bien es cierto que en CentOS 3, el puerto aparecía abierto, pero esta es la máquina local,

donde siempre aparecerán abiertos aquellos puertos que utilicen servicios que se estén ejecutando.

Después de habilitar la entrada de datos tcp en CentOS 3 en el puerto 10000, hice un escaneo desde CentOS 4 y el puerto 10000 aparecía abierto.

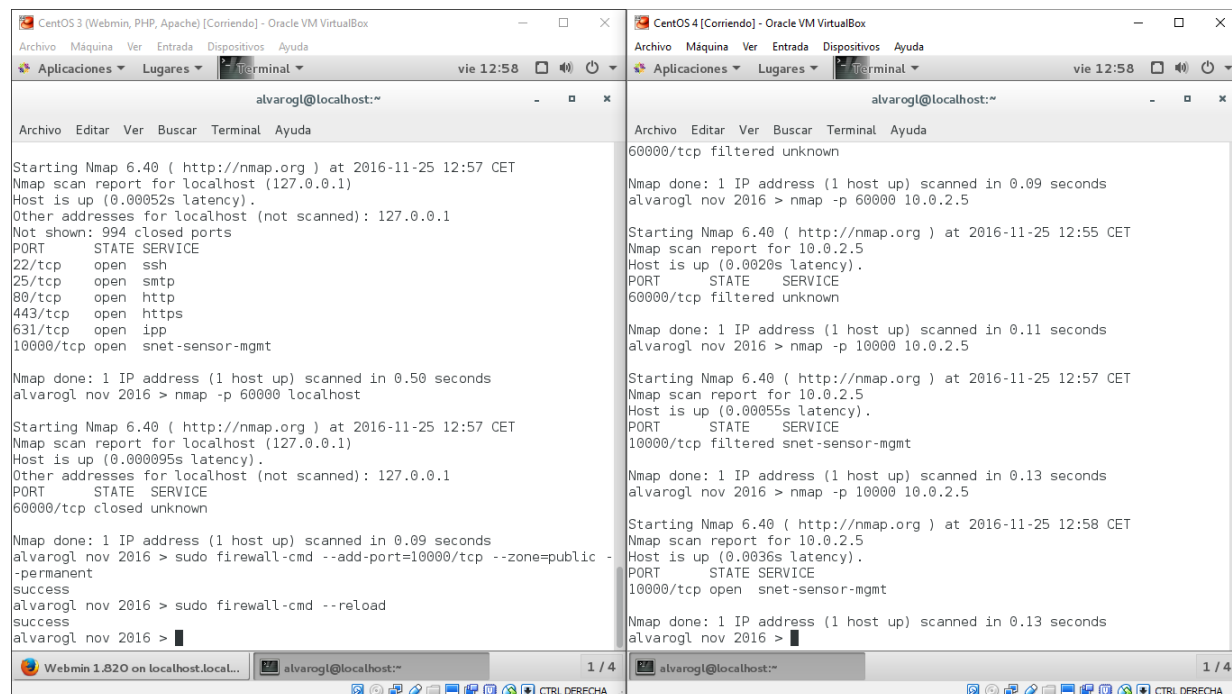


Figura 3.4: Habilitación de entrada de datos tcp en puerto 10000

4. Cuestión 4. ¿Qué diferencia hay entre telnet y ssh?

La diferencia principal entre ambos servicios reside en la seguridad. Según [5], un servidor SSH asegura una conexión segura y encriptada con un cliente SSH. Además, una encriptación fuerte para la autenticación.

En cuanto al protocolo Telnet, habilita una conexión TCP/IP a un servidor. No realiza ningún tipo de encriptación, incluso para la autenticación, pues la contraseña de un usuario se envía en texto plano sin cifrado.

5. Cuestión 5.

5.1. a) ¿Para qué sirve la opción -X?

Según [17] y [29], la opción -X sirve para habilitar el redireccionamiento de X11. Esto consiste en que el usuario pueda ejecutar aplicaciones que requieran de entorno gráfico a través de SSH. De esta manera, la aplicación sería ejecutada desde la máquina remota, pero utilizando el servicio de ventanas X11 del usuario local.

5.2. b) Ejecute remotamente, es decir, desde la máquina anfitriona (si tiene Linux) o desde la otra máquina virtual, el comando gedit en una sesión abierta con ssh. ¿Qué ocurre?

Al utilizar la opción -X, como podemos ver en 5.1, se utiliza el servicio de ventanas de la máquina local para ejecutar una aplicación desde la máquina remota y editar un archivo en ella con gedit

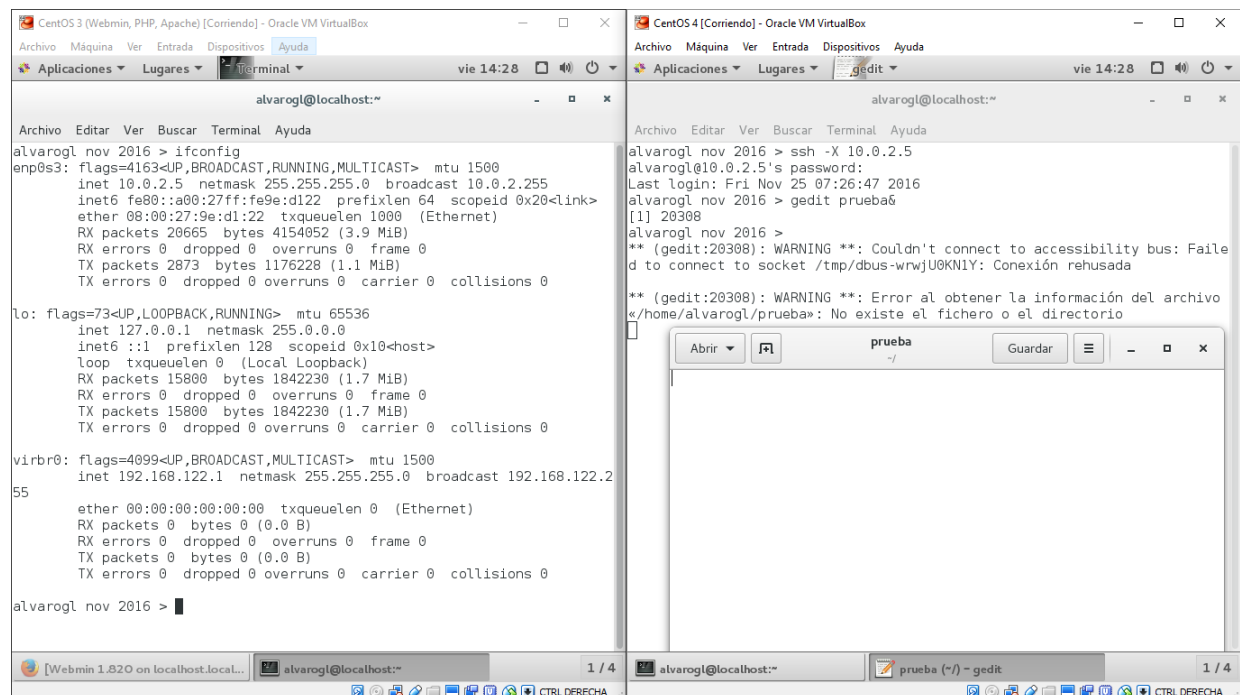


Figura 5.1: Utilización de `ssh` con redireccionamiento de X11

6. Cuestión 6. Muestre la secuencia de comandos y las modificaciones a los archivos correspondientes para permitir acceder a la consola remota sin introducir la contraseña. Pruebe que funciona. (Pistas: `ssh-keygen`, `ssh-copy-id`)

Como se nos explica en [25], gracias a `ssh-keygen`, podemos generar una claves de SSH. Dichas claves se generan por pares, es decir, se genera una clave privada y una clave pública. La clave pública se guarda en la ruta `/home/<user>/.ssh/id_rsa.pub` al ejecutar `ssh-keygen`

como se muestra en 6.1.

```

alvarogl nov 2016 > ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/alvarogl/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/alvarogl/.ssh/id_rsa.
Your public key has been saved in /home/alvarogl/.ssh/id_rsa.pub.
The key fingerprint is:
88:42:a8:7f:c1:2e:23:b3:83:b0:fc:17:9b:2a:df:58 alvarogl@localhost.localdo
main
The key's randomart image is:
+--[ RSA 2048]-----+
|
| .
|..
|...
|..+..S
|..o o
|*..+ oE+
|*+ ++
|..+=O.
+-----+

```

Figura 6.1: Uso de *ssh-keygen* y *ssh-copy-id* para acceder remotamente por SSH sin contraseña

Después de generar el par de claves, tendremos que copiar la clave pública al servidor remoto para que este pueda conectarse por SSH con autenticación por claves (que no contraseña de caracteres ordinaria). Para ello utilizamos *ssh-copy-id*:

ssh-copy-id

Tras realizar la copia, podemos acceder a la máquina remota sin que se nos pida la contraseña de acceso. Ver 6.2

```

alvarogl nov 2016 > ssh-copy-id 10.0.2.5
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to f
ilter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are
prompted now it is to install the new keys
alvarogl@10.0.2.5's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh '10.0.2.5'"
and check to make sure that only the key(s) you wanted were added.

alvarogl nov 2016 > ssh 10.0.2.5
Last login: Fri Nov 25 14:27:45 2016 from 10.0.2.6

```

Figura 6.2: Uso de *ssh-copy-id* y acceso sin contraseña a máquina remota por SSH

7. Cuestión 7. ¿Qué archivo es el que contiene la configuración del servicio ssh? ¿Qué parámetro hay que modificar para evitar que el usuario root acceda? Cambie el puerto por defecto y compruebe que puede acceder.

Como podemos comprobar en [27], para evitar que el usuario root acceda debemos editar la entrada de *PermitRootLogin*, y para cambiar el puerto por defecto tenemos que cambiar la entrada *Port 22*. Podemos escoger el puerto que queramos mientras no lo ocupe otro servicio, en mi caso escogí el 2345. Las entradas a cambiar quedarían así:

```
PermitRootLogin no
Port 2345
```

El fichero de configuración cambiado se muestra en 7.1.

```
# What ports, IPs and protocols we listen for
Port 2345
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
#Privilege Separation is turned on for security
UsePrivilegeSeparation yes

# Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600
ServerKeyBits 1024

# Logging
SyslogFacility AUTH
LogLevel INFO

# Authentication:
LoginGraceTime 120
PermitRootLogin no
StrictModes yes
```

Figura 7.1: Fichero de configuración */etc/ssh/sshd_config* modificado

En 7.2 se comprueba el acceso mediante el nuevo puerto indicado en *sshd_config* después de reiniciar el servicio *ssh* utilizando

```
sudo service ssh restart
```

en Ubuntu Server.

```

alvarogl@ubuntu:~$ ssh -p 2345 alvarogl@10.0.2.15
The authenticity of host '10.0.2.15:2345 (10.0.2.15:2345)' can't be established.
ECDSA key fingerprint is 62:b6:84:bd:50:2b:99:dd:08:b9:76:d6:81:7e:10:9b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.15:2345' (ECDSA) to the list of known hosts.
alvarogl@10.0.2.15's password:
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information as of Fri Nov 25 16:06:34 CET 2016

System load:   0.01           Memory usage: 6%      Processes:    138
Usage of /home: 0.7% of 451MB Swap usage:   0%      Users logged in: 0

Graph this data and manage this system at:
https://landscape.canonical.com/

0 packages can be updated.
0 updates are security updates.

New release '16.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Fri Nov 25 16:06:33 2016

```

Figura 7.2: Fichero de configuración `/etc/ssh/sshd_config` modificado

8. Cuestión 8. Indique si es necesario reiniciar el servicio ¿Cómo se reinicia un servicio en Ubuntu? ¿Y en CentOS? Muestre la secuencia de comandos para hacerlo.

Sí. Es necesario en este caso. Pues si no reiniciamos el servicio, `ssh` seguirá ejecutándose en el puerto 22, que es el que tenía por defecto. Para que el servicio se ejecute en el puerto 2345 según la última modificación del fichero de configuración, el servicio debe ser reiniciado.

Para reiniciar servicios en Ubuntu, según [11] debemos ejecutar:

```
sudo service <service-name> restart
```

Ver 8.1

```

alvarogl nov 2016 > sudo systemctl restart sshd.service
[sudo] password for alvarogl:
alvarogl nov 2016 >

```

Figura 8.1: Reinicio de un servicio en Ubuntu

En CentOS, el equivalente, según [26] es:

```
sudo systemctl restart <service-name>
```

Ver 8.2


```
alvarogl@ubuntu-server:~$ sudo service ssh restart
[sudo] password for alvarogl:
ssh stop/waiting
ssh start/running, process 4504
```

Figura 8.2: Reinicio de un servicio en CentOS

9. Cuestión 9. Muestre los comandos que ha utilizado en Ubuntu Server y en CentOS (aunque en este último puede utilizar la GUI, en tal caso, realice capturas de pantalla). Compruebe que la instalación ha sido correcta.

Según [12], podemos utilizar el gestor de paquetes taskel para instalar LAMP si no lo hemos hecho durante la instalación de Ubuntu Server. Esta interfaz nos permite instalar todos los componentes que se nos piden en el ejercicio de una vez:

```
sudo tasksel
```

En CentOS se deben ejecutar las siguientes sentencias según [16], [14] y [15]:

```
sudo yum install httpd
sudo yum install mysql-server
sudo yum install php php-mysql
```

Para comprobar el correcto funcionamiento de nuestro servidor corriendo con Apache podemos crear un archivo de texto con un mensaje de prueba en `/var/www/html/index.html` y acceder a localhost desde un navegador o `http://<IP>` desde una máquina remota. En 9.1 se muestra en buen funcionamiento el servidor de apache2.

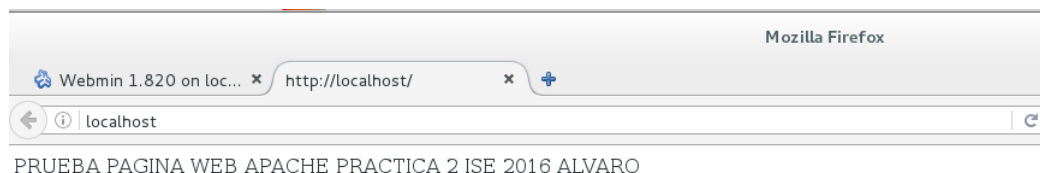


Figura 9.1: Comprobación de servidor apache2 corriendo en CentOS

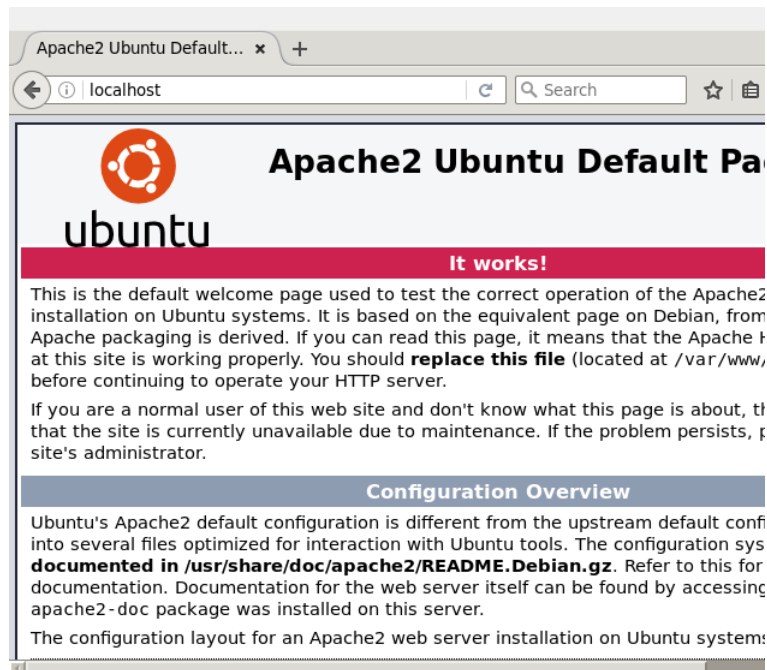


Figura 9.2: Comprobación de servidor apache2 corriendo en Ubuntu

Para comprobar que se han instalado php y mysql basta con ejecutar:

```
php --version
mysql --version
```

```
alvarogl@ubuntu-server:~$ php --version
PHP 5.5.9-1ubuntu4.20 (cli) (built: Oct  3 2016 13:00:37)
Copyright (c) 1997-2014 The PHP Group
Zend Engine v2.5.0, Copyright (c) 1998-2014 Zend Technologies
    with Zend OPcache v7.0.3, Copyright (c) 1999-2014, by Zend Technologies
alvarogl@ubuntu-server:~$ mysql --version
mysql Ver 14.14 Distrib 5.5.53, for debian-linux-gnu (x86_64) using readline 6.3
```

Figura 9.3: Comprobación de que PHP y MySQL están instalados

10. Cuestión 10. Realice la instalación usando GUI o PowerShell y compruebe que el servicio está funcionando accediendo a la MV a través de la anfitriona

Para instalar IIS seguimos los pasos indicados en el guión de prácticas. Buscamos la aplicación IIS y la iniciamos, como se observa en 10.1 [13]

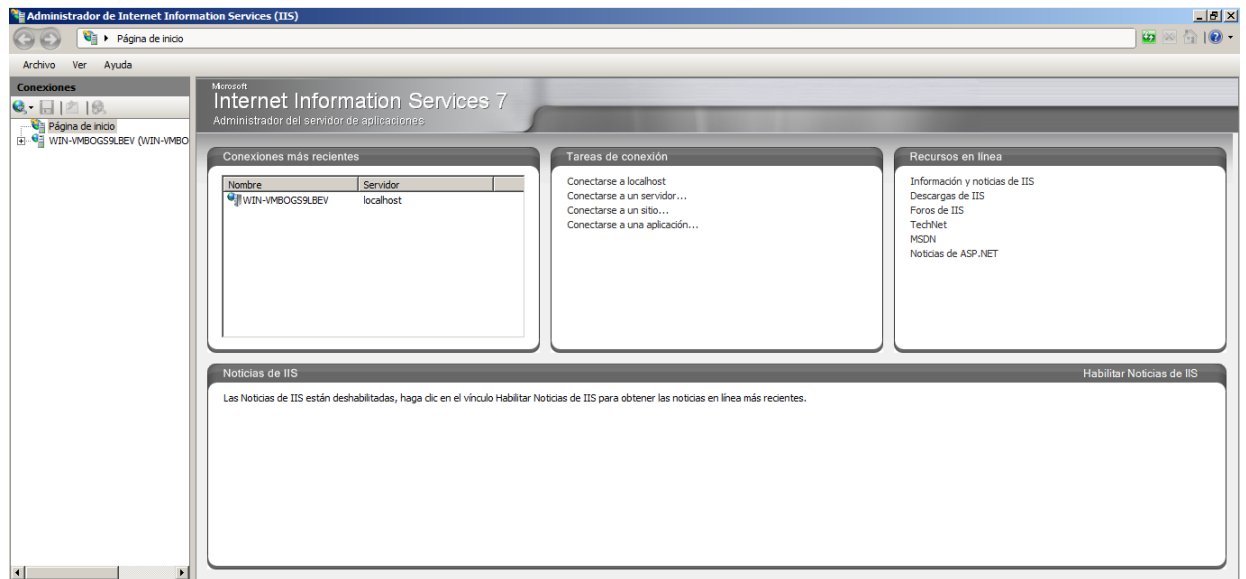


Figura 10.1: Ejecución de IIS

En 10.2 se muestra la IP de la máquina virtual

```

Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) 2009 Microsoft Corporation. Reservados todos los derechos.

PS C:\Users\Administrador>
PS C:\Users\Administrador> ipconfig

Configuración IP de Windows

Adaptador de Ethernet Conexión de área local:

    Sufijo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . : fe80::49bf:bfed:bf52:6e53%11
    Dirección IPv4. . . . . : 192.168.56.101
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . :

Adaptador de túnel isatap.{553BA89B-299E-44AE-A835-393583A3F11D}:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :

Adaptador de túnel Conexión de área local* 2:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :
PS C:\Users\Administrador>

```

Figura 10.2: Salida ipconfig máquina virtual

En 10.3 se muestra el acceso exitoso desde la máquina anfitriona al servidor IIS.

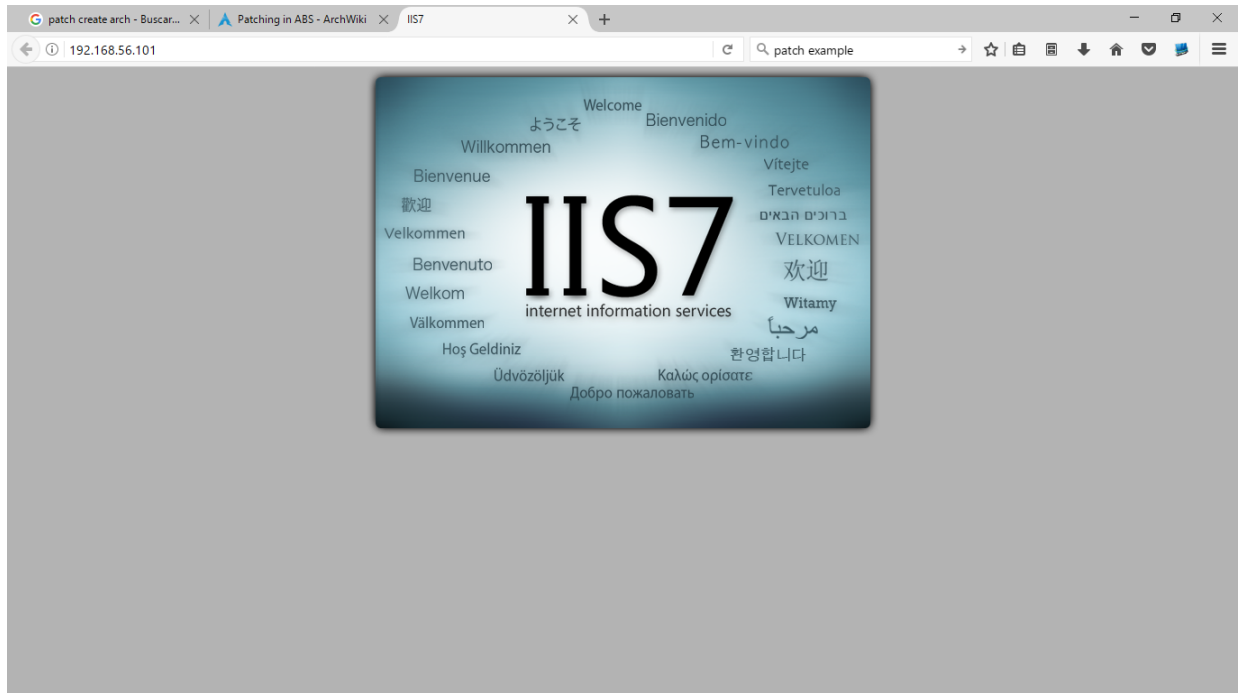


Figura 10.3: Acceso desde máquina anfitriona a IIS

Para que el acceso se produzca satisfactoriamente la máquina virtual debe estar configurada con una conexión host-only.

11. Cuestión 11. Muestre un ejemplo de uso del comando (p.ej. <http://fedoraproject.org/wiki/VMWare>)

En [1] se nos enseña a utilizar este útil comando de Linux.

Para mostrar un ejemplo de su uso, creamos un archivo `sum.cpp`. Supongamos que este tiene un fallo, como se puede apreciar en 11.1.

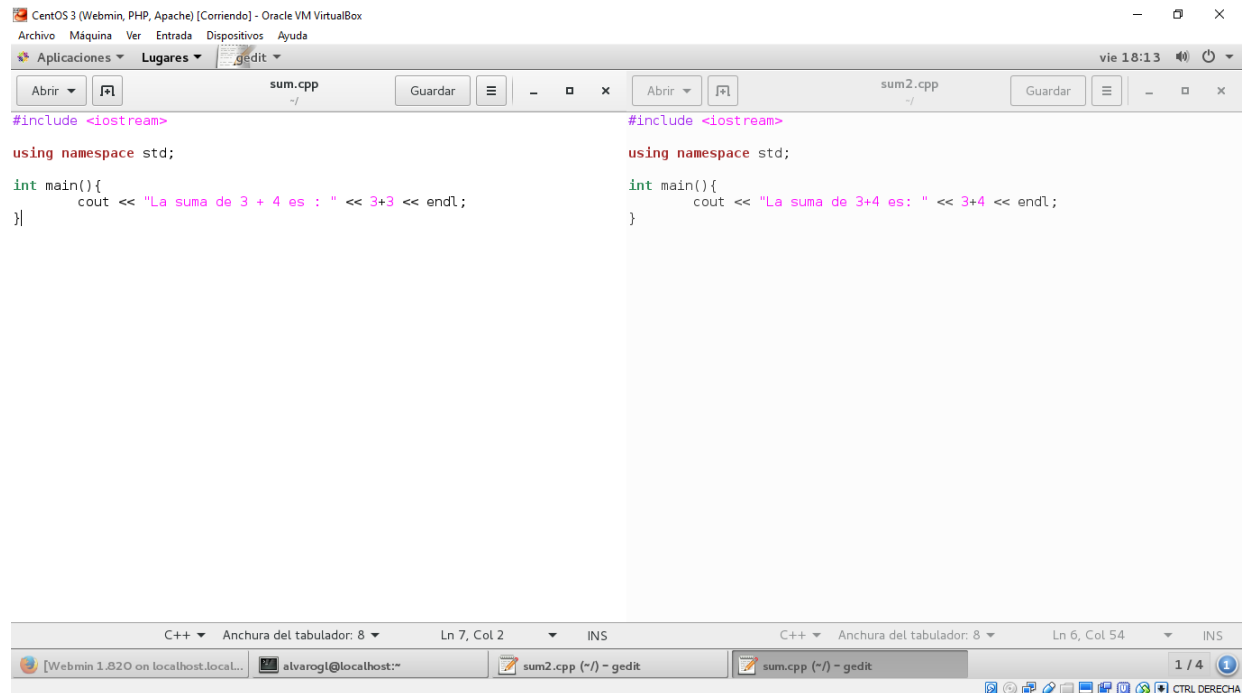


Figura 11.1: Archivos sum.cpp y sum2.cpp (sum.cpp corregido)

Creamos el parche y lo aplicamos:

```
diff sum.cpp sum2.cpp > p.patch
patch sum.cpp p.patch
```

```
alvarogl nov 2016 > diff sum.cpp sum2.cpp
6c6
<      cout << "La suma de 3 + 4 es : " << 3+3 << endl;
---
>      cout << "La suma de 3+4 es: " << 3+4 << endl;
alvarogl nov 2016 > diff sum.cpp sum2.cpp > p.patch
alvarogl nov 2016 > ls
centos4 Descargas Documentos Escritorio Imágenes Música Plantillas p.patch Público sum2.cpp sum.cpp Vídeos
alvarogl nov 2016 > patch sum.cpp p.patch
patching file sum.cpp
```

Figura 11.2: Creación y aplicación del parche

En 11.3 se muestra el resultado final de sum.cpp, ya habiendo aplicado el parche sobre él, de forma que hemos conseguido corregirlo.

```

GNU nano 2.3.1                                Fichero: sum.cpp
#include <iostream>

using namespace std;

int main(){
    cout << "La suma de 3+4 es: " << 3+4 << endl;
}

```

Figura 11.3: Resultado final del archivo

12. Cuestión 12. Realice la instalación de esta aplicación y pruebe a modificar algún parámetro de algún servicio. Muestre las capturas de pantalla pertinentes así como el proceso de instalación.

Webmin está disponible en [24]

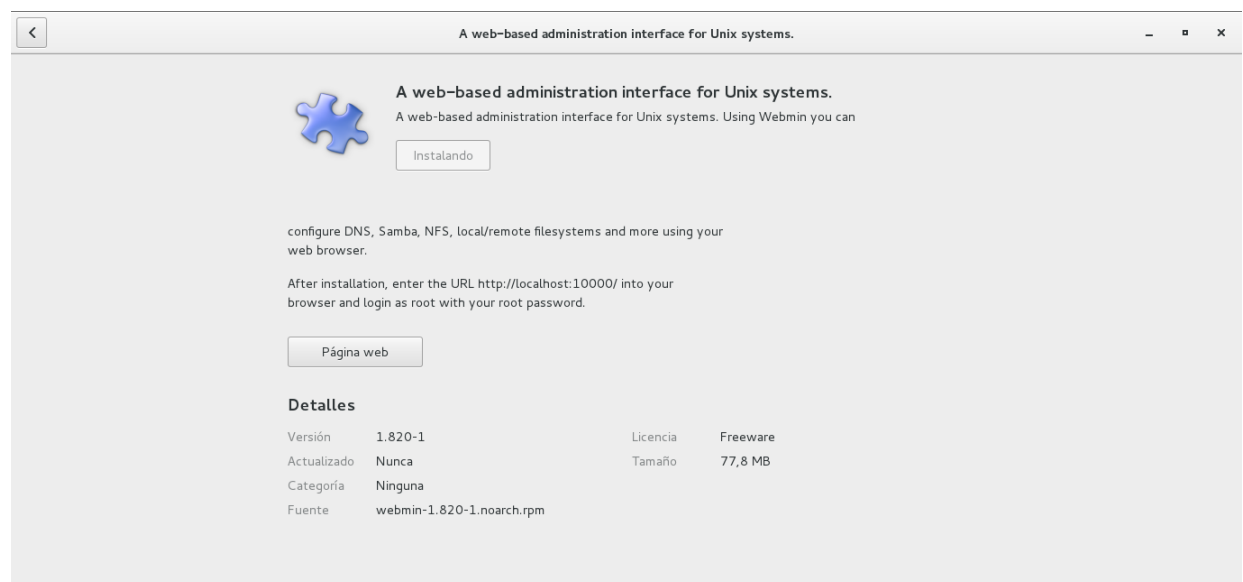


Figura 12.1: Instalación de webmin con interfaz gráfica

Accedemos a webmin a través de localhost:10000 en un navegador 12.2.

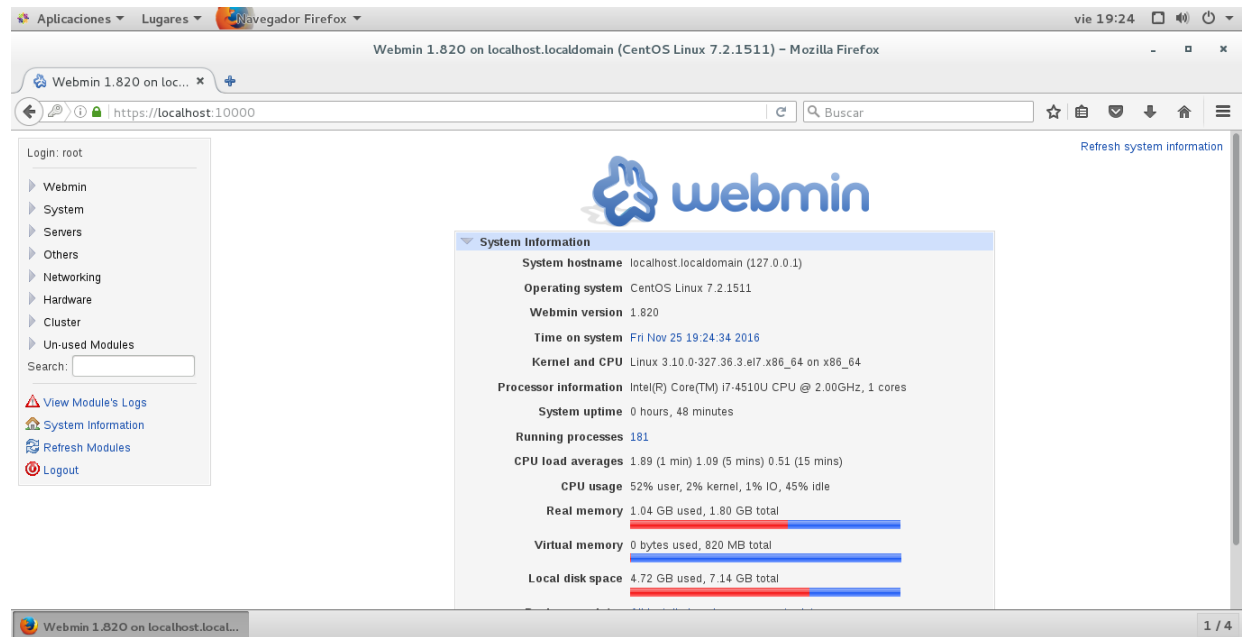


Figura 12.2: Interfaz de webmin

En mi caso, he probado el *shell* de comandos, como se puede ver en 12.3 para ver información sobre la CPU.

Module Config

Command Shell

```
> cat /proc/cpuinfo
processor       : 0
vendor_id     : GenuineIntel
cpu family    : 6
model        : 69
model name    : Intel(R) Core(TM) i7-4510U CPU @ 2.00GHz
stepping     : 1
cpu MHz      : 2593.992
cache size   : 4096 KB
physical id  : 0
siblings    : 1
core id     : 0
cpu cores   : 1
apicid      : 0
initial apicid : 0
fpu        : yes
fpu_exception : yes
cpuid level : 13
wp         : yes
flags      : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 syscall nx rdtscp lm cc
bogomips   : 5187.98
clflush size : 64
cache_alignment : 64
address sizes : 39 bits physical, 48 bits virtual
```

Figura 12.3: Interfaz de webmin

También he creado una nueva tarea para que el demonio cron la ejecute de manera semanal para comprobar el espacio disponible en las particiones, y que esta información se almacene en */var/log/weeklyscan.log* (ver 12.4)

Module Index

Create Cron Job

Job Details

Execute cron job as:

Active? ☒ Yes ☐ No

Command:

Input to command:

Description:

When to execute

☒ Simple schedule .. ☐ Weekly (on Sunday) ☐ Times and dates selected below ..

Figura 12.4: Creación de una nueva tarea de cron

13. Cuestión 13. Instale phpMyAdmin, indique cómo lo ha realizado y muestre algunas capturas de pantalla. Configure PHP para poder importar BDs de hasta 25MiB (en vez de los 8 MiB de límite por defecto). Indique cómo ha realizado el proceso y muestre capturas de pantalla.

Para instalar phpmyadmin ejecutamos:

```
sudo apt install phpmyadmin
```

En 13.1 aparece la interfaz de instalación de phpmyadmin.

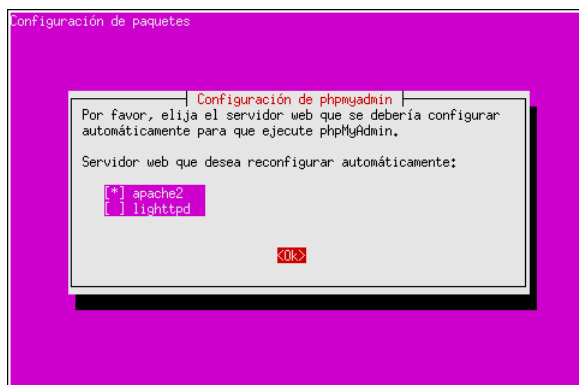
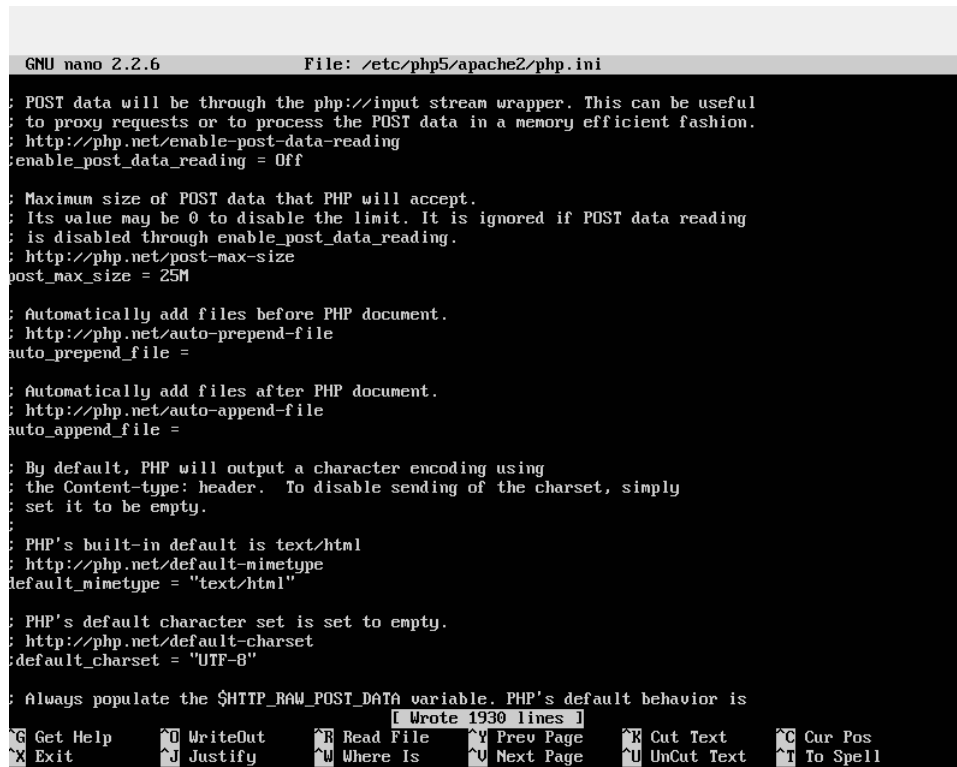


Figura 13.1: Instalación de phpmyadmin

Según [23], donde se nos describen todas las entradas del fichero `php.ini` de `/etc/php5/apache2/php.ini` y cómo afectan las modificaciones de dicho archivo de configuración al servidor, podemos cambiar el tamaño máximo de datos de POST permitidos. Debemos cambiar el valor antiguo de `post_max_size` por `'25M'` como se muestra en 13.2 dicho valor debe

ser mayor que *upload_max_filesize*. *memory_limit* debe ser mayor que *post_max_size*. Comprobamos que se cumplen estas dos restricciones y salvamos el archivo.



```
GNU nano 2.2.6 File: /etc/php5/apache2/php.ini
; POST data will be through the php://input stream wrapper. This can be useful
; to proxy requests or to process the POST data in a memory efficient fashion.
; http://php.net/enable-post-data-reading
enable_post_data_reading = Off

; Maximum size of POST data that PHP will accept.
; Its value may be 0 to disable the limit. It is ignored if POST data reading
; is disabled through enable_post_data_reading.
; http://php.net/post-max-size
post_max_size = 25M

; Automatically add files before PHP document.
; http://php.net/auto-prepend-file
auto_prepend_file =

; Automatically add files after PHP document.
; http://php.net/auto-append-file
auto_append_file =

; By default, PHP will output a character encoding using
; the Content-type: header. To disable sending of the charset, simply
; set it to be empty.
;
; PHP's built-in default is text/html
; http://php.net/default-mimetype
default_mimetype = "text/html"

; PHP's default character set is set to empty.
; http://php.net/default-charset
default_charset = "UTF-8"

; Always populate the $HTTP_RAW_POST_DATA variable. PHP's default behavior is
; to not populate the variable.
[Wrote 1930 lines]
^G Get Help      ^O WriteOut     ^R Read File    ^V Prev Page    ^K Cut Text      ^C Cur Pos
^X Exit          ^J Justify      ^W Where Is     ^U Next Page    ^U UnCut Text   ^T To Spell
```

Figura 13.2: Edición de */etc/php5/apache2/php.ini*

14. Cuestión 14. Viste al menos una de las webs de los software mencionados y pruebe las demos que ofrecen realizando capturas de pantalla y comentando qué está realizando

En mi caso, he visitado la web de ISPCConfig, la interfaz se muestra en 14.1. He accedido con las credenciales de la demo para probar el software.

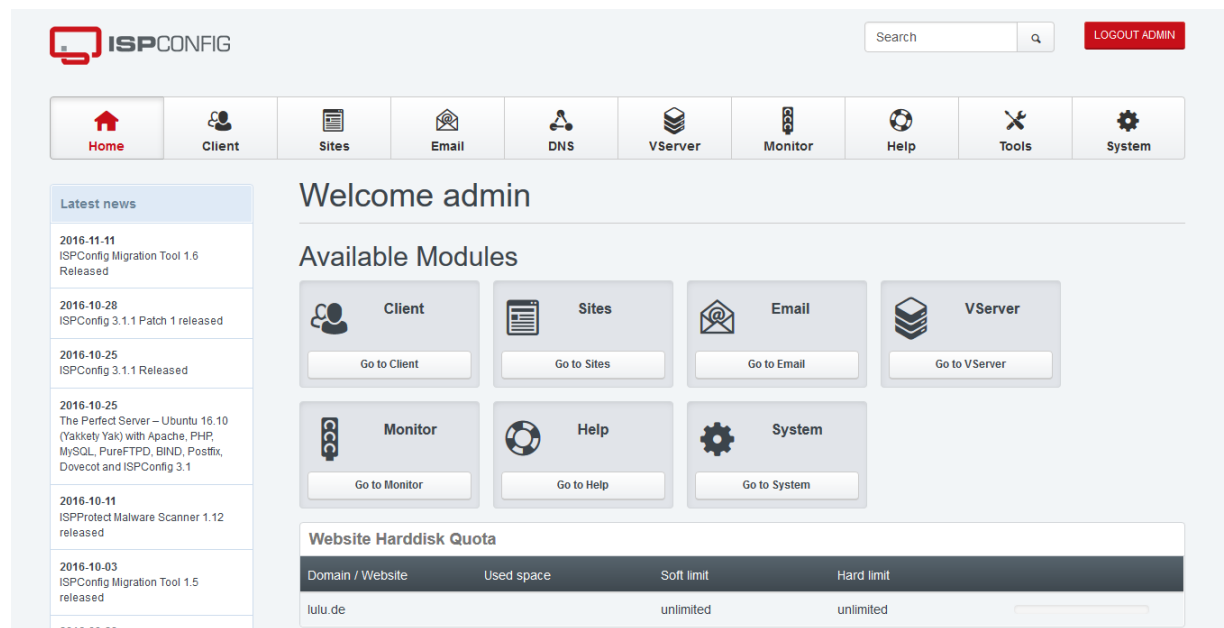


Figura 14.1: Pantalla de bienvenida al administrador de Webmin

He creado un nuevo usuario para bases de datos (Ver 14.2) para después crear una base de datos nueva (Ver 14.3).

Database User

Database User

Client:

Database user:

Database password:

Password strength: Very strong

Repeat Password:

The passwords do match.

Figura 14.2: Nuevo usuario para bases de datos

Database

Server: server1.example.com

Site: - Select Site -

Type: MySQL

Database name: c[CLIENTID]

Database quota: -1 MB

Database user:

Read-only database user:

optional

Database charset: DB-Default

Remote Access: ☒

Remote Access IPs (separate by , and leave blank for any):

Active: ☒

Save Cancel

Figura 14.3: Nueva base de datos

Algunas herramientas interesantes de administración del servidor se pueden apreciar en 14.4, entre las que destaco las de mostrar la carga del servidor, la utilización del disco, el uso de memoria y los servicios activos.

Hardware Information
Show CPU Info
Server State
Show Overview
Show Update State
Show RAID State
Show Server Load
Show Disk Usage
Show MySQL Database size
Show Memory Usage
Show Services
Show Monit
Show OpenVz VE BeanCounter
Show Munin

Figura 14.4: Herramientas interesantes

Me hubiera gustado probar más características de esta herramienta, pero la versión demo limita mucho la funcionalidad de la misma.

15. Cuestión 15.

15.1. a) Ejecute los ejemplos de *find*, *grep*

En 15.1 se aprecia el ejemplo de *grep* del guión de prácticas ejecutado:

```
ps -Af | grep firefox
```

```
alvarog1@ubuntu:~$ ps -Af | grep firefox
alvarog1 17097  4479  0 20:24 pts/0    00:00:00 grep --color=auto firefox
```

Figura 15.1: Ejecución del comando *ps -Af | grep firefox*

Para el ejemplo de *find*, ejecutamos en el directorio que aparece en 15.2 el comando(Ver 15.3):

```
find -name 'test*' -exec cp {} ~/tests \
```

```
alvarog1@ubuntu:~$ ls
1          14211          2      3          537821  853217  test4  test8
13285      153153287532181532  2135   318401342571532  58      Desktop test5  test9
134        15321542753218      23175  32175      583215  test2   test6  test_raid
1352713258 17853215327321      25318  53217      813275  test3   test7  tests
```

Figura 15.2: Archivos en el directorio antes de la ejecución

```

alvarogl@ubuntu-server:~$ find -name 'test*' -exec cp {} ~/tests \;
cp: omitting directory './tests'
cp: './tests/test2' and '/home/alvarogl/tests/test2' are the same file
find: './.config': Permission denied
find: './.cache/mozilla': Permission denied
find: './.cache/dconf': Permission denied
find: './.mozilla': Permission denied

```

Figura 15.3: Ejecución del comando

Los errores que se aprecian en 15.3 salen porque hay archivos que el sistema no permite copiar sin permisos de root.

El resultado de la ejecución puede verse en 15.4. Todos los archivos que comienzan por test han sido transferidos al directorio indicado.

```

alvarogl@ubuntu-server:~$ ls tests
test2  test9          test-phish-simple.cache  test-trackwhite-simple.cache
test3  test-block-simple.cache  test-phish-simple.pset   test-trackwhite-simple.pset
test4  test-block-simple.pset   test-phish-simple.sbstore test-trackwhite-simple.sbstore
test5  test-block-simple.sbstore test RAID                 test-unwanted-simple.cache
test6  test-malware-simple.cache test-track-simple.cache   test-unwanted-simple.pset
test7  test-malware-simple.pset test-track-simple.pset    test-unwanted-simple.sbstore
test8  test-malware-simple.sbstore test-track-simple.sbstore

```

Figura 15.4: Resultado de la ejecución

15.2. b) Escriba el script que haga uso de sed para cambiar la configuración de ssh y reiniciar el servicio

Vamos a realizar el script que se pide en el guión de prácticas, cuya ejecución brindará acceso a usuarios por autenticación con contraseña al servidor por unos segundos.

Según 15.5, podemos modificar un archivo *'in-place'* con la opción *-i* de sed. Con el comando *'s'* los caracteres entre */* son reemplazados por los que se encuentran detrás de la segunda */*. Con el comando *'g'* los espacios son tenidos en cuenta.

El script 15.5 primero modifica con sed el archivo de configuración de ssh para proporcionar el acceso con contraseña, después espera 30 segundos, vuelve a dejar el fichero de configuración en su estado original con sed, y finalmente reinicia el servicio [10].

```

script-sed.sh x
#!/bin/bash

#Proporcionar acceso mediante password
sudo sed -i 's/^#PasswordAuthentication yes/PasswordAuthentication yes/g' /etc/ssh/sshd_config;

#Reiniciar servicio ssh
sudo service ssh restart;

echo "Acceso habilitado por 30 segundos";

#Esperar unos segundos para que los usuarios puedan acceder al servidor
sleep 30;

#Dejar todo como estaba
sudo sed -i 's/PasswordAuthentication yes/#PasswordAuthentication yes/g' /etc/ssh/sshd_config

echo "Acceso cerrado";

#Reiniciar servicio ssh
sudo service ssh restart;
|

```

Figura 15.5: Script de sed en bash

Probamos el funcionamiento del script, comprobando antes, durante, y después de la ejecución del mismo el estado del fichero de configuración de *ssh* como se puede ver en 15.6.

```

alvarogl nov 2016 > cat /etc/ssh/sshd_config | grep PasswordAuth
PasswordAuthentication yes
# PasswordAuthentication. Depending on your PAM configuration,
# PAM authentication, then enable this but set PasswordAuthentication
alvarogl nov 2016 > ./script-sed.sh&
[3] 4581
alvarogl nov 2016 > ssh stop/waiting
ssh start/running, process 4595
Acceso habilitado por 30 segundos

alvarogl nov 2016 > cat /etc/ssh/sshd_config | grep PasswordAuth
PasswordAuthentication yes
# PasswordAuthentication. Depending on your PAM configuration,
# PAM authentication, then enable this but set PasswordAuthentication
alvarogl nov 2016 > Acceso cerrado
ssh stop/waiting
ssh start/running, process 4626

[3] Hecho
alvarogl nov 2016 > cat /etc/ssh/sshd_config | grep PasswordAuth
#PasswordAuthentication yes
# PasswordAuthentication. Depending on your PAM configuration,
# PAM authentication, then enable this but set PasswordAuthentication

```

Figura 15.6: Resultado de la ejecución del script y comprobación de su funcionamiento

El script será añadido al archivo comprimido de la entrega de la práctica.

15.3. c) Muestre un ejemplo de uso para awk

En [9] se nos explica el uso del comando *awk*. Es muy útil para la manipulación de textos. En el siguiente ejemplo, *awk* toma los valores de las columnas que le indicamos y las imprime. Supongamos que tenemos un fichero con las notas de ISE de cada práctica como en 15.7 se muestra.

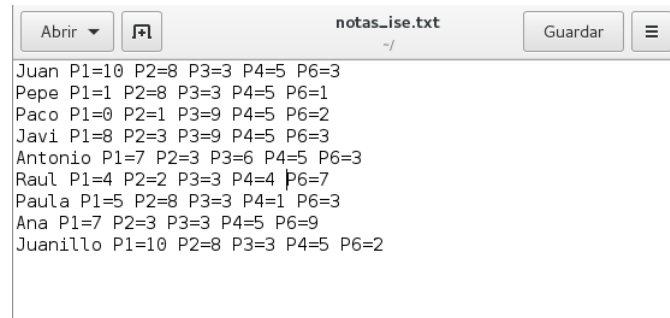


Figura 15.7: Fichero inicial

Resulta que solo queremos saber las notas de las prácticas 1 y 2. Entonces ejecutamos el siguiente comando:

```
awk '/a {$1 "\t" $2 "\t" $3 notas_ise.txt}}
```

El resultado de la ejecución se muestra en 15.8.

```
alvarogl nov 2016 > awk '/a/ {print $1 "\t" $2 "\t" $3}' notas_ise.txt
Juan      P1=10    P2=8
Paco      P1=0     P2=1
Javi      P1=8     P2=3
Raul      P1=4     P2=2
Paula     P1=5     P2=8
Ana       P1=7     P2=3
Juanillo  P1=10    P2=8
```

Figura 15.8: Resultado de la ejecución del comando

16. Cuestión 16. Escriba el script para cambiar el acceso a ssh usando PHP o Python

En [7] se explica cómo abrir, leer y escribir ficheros en Python, una guía útil para confeccionar el script que se pide, como se puede ver en 15.5.

```

1 #coding=utf-8
2 import subprocess
3 import time
4
5 aux = []
6
7 with open("/etc/ssh/sshd_config", "r") as file:
8     for line in file:
9         if ("#PasswordAuthentication yes") or ("PasswordAuthentication no") in line:
10             line = "PasswordAuthentication yes\n"
11             aux.append(line)
12
13 with open("/etc/ssh/sshd_config", "w") as file:
14     for line in aux:
15         file.write(line)
16
17 command = ['service', 'ssh', 'restart'];
18 subprocess.call(command, shell=False)
19
20 print("Acceso por contraseña habilitado")
21 time.sleep(30)
22
23 aux2 = []
24
25 with open("/etc/ssh/sshd_config", "r") as file:
26     for line in file:
27         if "PasswordAuthentication yes" in line:
28             line = "#PasswordAuthentication yes\n"
29             aux2.append(line)
30
31 with open("/etc/ssh/sshd_config", "w") as file:
32     for line in aux2:
33         file.write(line)
34
35 subprocess.call(command, shell=False)
36
37 print("Acceso cerrado")

```

Figura 16.1: Script en Python para modificar la configuración de ssh

Probamos el funcionamiento del script, comprobando antes, durante, y después de la ejecución del mismo el estado del fichero de configuración de *ssh* como se puede ver en 16.2.


```
alvarogl@alvarogl-SATELLITE-S50-B: ~/Escritorio
[5] Hecho sudo gedit /etc/ssh/sshd_config
alvarogl nov 2016 > cat /etc/ssh/sshd_config | grep PasswordAuth
#PasswordAuthentification yes
alvarogl nov 2016 > sudo python script-sed.py &
[5] 5270
alvarogl nov 2016 > ssh stop/waiting
ssh start/running, process 5282
Acceso por contraseña habilitado

alvarogl nov 2016 > cat /etc/ssh/sshd_config | grep PasswordAuth
PasswordAuthentification yes
alvarogl nov 2016 > cat /etc/ssh/sshd_config | grep PasswordAuthssh stop/waiting
ssh start/running, process 5296
Acceso cerrado

#PasswordAuthentification yes
[5] Hecho sudo python script-sed.py
alvarogl nov 2016 > cat /etc/ssh/sshd_config | grep PasswordAuth
#PasswordAuthentification yes
alvarogl nov 2016 > █
```

Figura 16.2: Resultado de la ejecución del script y comprobación de su funcionamiento

El script será añadido al archivo comprimido de la entrega de la práctica.

17. Cuestión 17. Abra una consola de PowerShell y pruebe a parar un programa en ejecución (p.ej), realice capturas de pantalla y comente lo que muestra

Abrimos un programa cualquiera, por ejemplo el Bloc de Notas, como se muestra en 17.1.

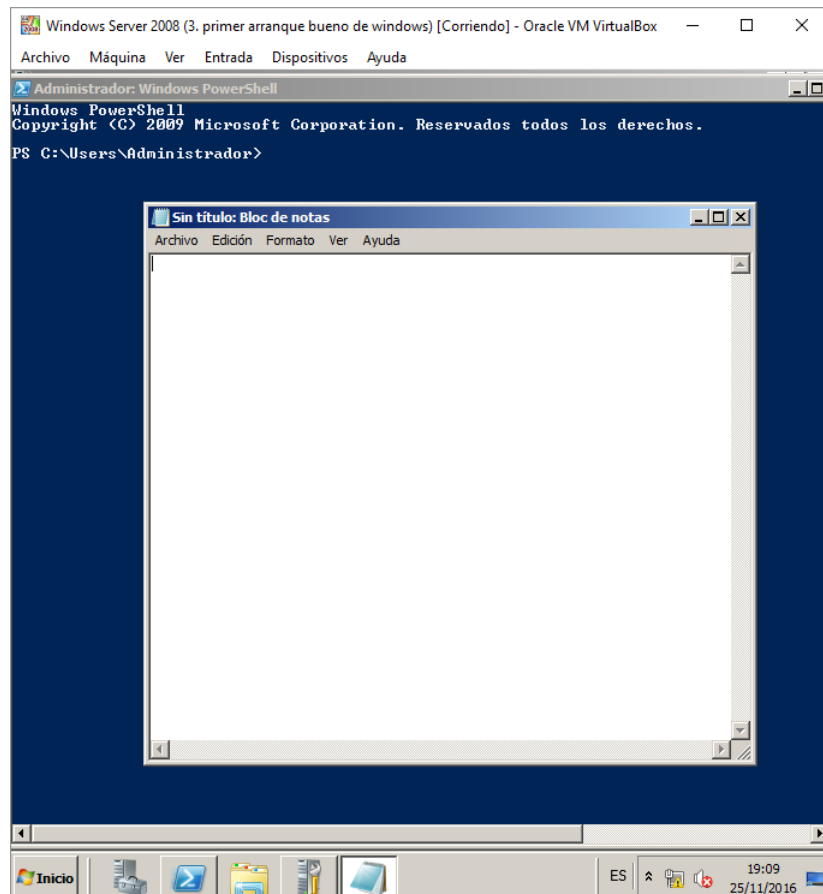


Figura 17.1: Programa Notepad abierto

Mostramos los procesos en ejecución como podemos ver en 17.2 con:

`Get-Process`

Handles	NPM(K)	PM(K)	WS(K)	UM(M)	CPU(s)	Id	ProcessName
38	5	1812	4140	43	0.00	360	conhost
347	10	1740	3552	42	0.06	316	csrss
218	12	1980	5264	44	0.20	356	csrss
69	7	1376	4132	50	0.03	1940	dsm
596	39	22164	38204	216	0.72	1960	explorer
0	0	0	24	0	0.00	0	idle
522	18	2956	8628	37	0.30	460	lsass
155	7	2140	3784	18	0.05	468	lsm
86	9	1680	4716	34	0.05	2236	msccorsvw
77	8	1836	5024	33	0.03	2268	msccorsvw
151	18	3344	7464	61	0.03	2376	msdtc
53	6	1236	4120	61	0.00	2148	notepad
273	40	36872	39124	586	1.11	1816	oobe
382	23	54044	51484	564	0.42	1160	powershell
220	13	4028	7696	33	0.45	452	services
29	2	352	1008	5	0.05	248	smss
274	18	6100	10404	73	0.06	772	spoolsv
175	9	5624	11600	40	0.61	1576	appsvcs
324	35	9692	11776	53	0.13	260	svchost
348	14	3420	8452	41	0.19	556	svchost
238	15	2644	6384	30	0.06	680	svchost
291	15	9000	11652	43	0.14	768	svchost
860	44	20216	27180	156	1.00	808	svchost
213	15	3596	7844	34	0.02	852	svchost
216	16	3668	9484	62	0.08	896	svchost
404	26	9640	13448	82	0.13	940	svchost
104	10	4484	8824	38	0.06	1056	svchost
134	13	4036	8252	39	0.03	1080	svchost
46	4	784	2564	13	0.00	1108	svchost
134	13	6320	9536	39	0.09	1136	svchost
71	6	1344	4200	29	0.02	2340	svchost
500	0	112	300	3	0.00	4	System
152	12	2796	5916	52	0.03	1880	taskhost
125	10	2036	6680	51	0.16	448	TrustedInstaller
118	9	1888	5076	48	0.06	616	UBoxService
145	10	1868	5784	72	0.00	1676	UBoxTray
80	9	1320	4204	43	0.05	364	wininit
103	7	1508	4780	27	0.06	392	winlogon

Figura 17.2: Procesos en ejecución

Por último, identificamos el proceso que queremos matar y ejecutamos (ver 17.3):

Stop-Service notepad

```
PS C:\Users\Administrador> Stop-Process -Name notepad
PS C:\Users\Administrador>
```

Figura 17.3: Ejecución de *Stop-Process -Name notepad*

18. Opinión sobre la práctica

En mi opinión, esta práctica ha sido muy extensa. Personalmente, he ido muy justo para entregarla porque requiere de mucho tiempo para hacer muchas capturas de pantalla, instalar y descargar muchos programas y me parece más compleja que la primera práctica y no me ha dado tiempo a hacer ninguna pregunta opcional. Si bien es cierto que es posible que sea mejor así, ya que las últimas prácticas coinciden con épocas en las que hay muchos exámenes y hay más carga lectiva de lo normal. Así que no me parece mal que se realice una práctica de tal densidad al principio-mitad de curso como se viene haciendo.

Sugeriría que se pudieran entregar las preguntas opcionales hasta final de curso, si no se ha contemplado ya esto.

Referencias

- [1] *Arch Linux, Patching*
https://wiki.archlinux.org/index.php/Patching_in_ABS.
- [2] *Ayuda de Ubuntu Community, configuraciones en apt*
<https://help.ubuntu.com/community/AptGet/Howto>.
- [3] *Ayuda de Ubuntu Community*
<https://help.ubuntu.com/community/UFW>.
- [4] *Ayuda de Ubuntu, ufw*
<https://help.ubuntu.com/lts/serverguide/firewall.html>.
- [5] *Cisco. Configuring Telnet and SSH*
http://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus5000/sw/configuration/guide/cli/CLIConfigurationGuide/sec_ssh.pdf.
- [6] *Documentación de Fedora, configuración de yum para un servidor proxy*
https://docs.fedoraproject.org/es-ES/Fedora_Core/4/html/Software_Management_Guide/sn-yum-proxy-server.html.
- [7] *Documentación de Python*
<https://docs.python.org/2/tutorial/inputoutput.html>.
- [8] *Documentación de Red Hat, Using Firewalls*
https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/sec-Using_Firewalls.html.
- [9] *GNU Manual, awk*
https://www.gnu.org/software/gawk/manual/html_node/Print-Examples.html.
- [10] *GNU Manual, sed*
<https://www.gnu.org/software/sed/manual/sed.txt>.
- [11] *Help Ubuntu Community, BootUp*
<https://help.ubuntu.com/community/UbuntuBootupHowto>.
- [12] *Help Ubuntu Community, Tasksel*
<https://help.ubuntu.com/community/Tasksel>.
- [13] *IIS Windows Installation*
<https://www.iis.net/learn/install/installing-iis-7/installing-iis-on-windows-vista-and-windows-7>.
- [14] *Instalación Apache*
<http://httpd.apache.org/docs/current/es/install.html>.

- [15] *Instalación MySQL*
<https://dev.mysql.com/doc/mysql-repo-excerpt/5.6/en/linux-installation-yum-repo.html>.
- [16] *Instalación y configuración de PHP*
http://doc.fedora-fr.org/wiki/Installation_et_configuration_de_PHP.
- [17] *Manual OpenBSD*
<http://man.openbsd.org/ssh>.
- [18] *Manuales de Ubuntu, apt-get*
<http://manpages.ubuntu.com/manpages/wily/man8/apt-get.8.html>.
- [19] *Manuales de Ubuntu, apt*
<http://manpages.ubuntu.com/manpages/wily/man8/apt.8.html>.
- [20] *Manuales de Ubuntu, yum-config,manager*
<http://manpages.ubuntu.com/manpages/wily/man1/yum-config-manager.1.html>.
- [21] *Página oficial de CentOS, yum commands*
https://www.centos.org/docs/5/html/5.1/Deployment_Guide/s1-yum-useful-commands.html.
- [22] *Página oficial de nmap*
<https://nmap.org/man/es/man-briefoptions.html>.
- [23] *PHP Manual*
<http://php.net/manual/es/ini.core.php>.
- [24] *Webmin*
webmin.com.
- [25] *Wiki ArchLinux, SSH keys*
https://wiki.archlinux.org/index.php/SSH_keys.
- [26] *Wiki ArchLinux, systemd*
https://wiki.archlinux.org/index.php/Systemd#Basic_systemctl_usage.
- [27] *Wiki CentOS, SSH*
<https://wiki.centos.org/es/HowTos/Network/SecuringSSH>.
- [28] *Wiki de Fedora, FirewallD/es*
<https://fedoraproject.org/wiki/FirewallD/es>.
- [29] *X Org Foundation*
<https://www.x.org/wiki/>.