



UNIVERSIDAD DE GRANADA

Ingeniería de Servidores

Memoria de la práctica 2

Javier León Palomares

24 de noviembre de 2016

Índice

1. Cuestión 1: a) Liste los argumentos de yum necesarios para instalar, buscar y eliminar paquetes. 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) c) ¿Cómo añadimos un nuevo repositorio? 6
2. Cuestión 2: a) Liste los argumentos de apt necesarios para instalar, buscar y eliminar paquetes. 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) c) ¿Cómo añadimos un nuevo repositorio? 6
3. Cuestión 3: a) ¿Con qué comando puede abrir/cerrar un puerto usando ufw? Muestre un ejemplo de cómo lo ha hecho. b) ¿Con qué comando puede abrir/cerrar un puerto usando firewall-cmd en CentOS? Muestre un ejemplo de cómo lo ha hecho. c) Utilice el comando nmap para ver que, efectivamente, los puertos están accesibles. 8
4. Cuestión 4: ¿Qué diferencia hay entre telnet y ssh? 9
5. Cuestión 5: a) ¿Para qué sirve la opción -X? 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? 9
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). 11
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. 12
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. 13
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. 13
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. 15

- 11.Cuestión 11: Muestre un ejemplo de uso del comando. 20
- 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. 21
- 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. 23
- 14.Cuestión 14: Visite 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: a) Ejecute los ejemplos de find, grep. b) Escriba el script que haga uso de sed para cambiar la configuración de ssh y reiniciar el servicio. c) Muestre un ejemplo de uso para awk. 26
- 16.Cuestión 16: Escriba el script para cambiar el acceso a ssh usando PHP o Python. 27
- 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. 28

Índice de figuras

1. Ejecución de <code>sudo apt update</code> que no llega a terminar debido a que no puede acceder al exterior.	7
2. Ejecución de <code>sudo apt update</code> tras modificar el archivo de configuración. Comprobamos que acaba normalmente.	7
3. Mediante <code>sudo ufw allow 22</code> he abierto en la máquina virtual (192.168.56.101) el puerto por defecto de ssh, el 22. Con <code>nmap -Pn -p 22 192.168.56.101</code> compruebo desde el host que dicho puerto está abierto.	8
4. Mediante <code>sudo ufw deny 22</code> he cerrado en la máquina virtual el puerto por defecto de ssh, el 22. Con <code>nmap -Pn -p 22 192.168.56.101</code> compruebo que está filtrado por el firewall.	8
5. Usando los comandos anteriores, abrimos el puerto 22 y recargamos la configuración de firewall-cmd.	9
6. Ejecutando <code>nmap</code> desde el host, observamos que el puerto 22 de CentOS (en la dirección 192.168.56.102) está, en efecto, abierto.	9
7. En esta primera captura se puede ver cómo no es posible ejecutar gedit sin la opción <code>-X</code> debido a lo explicado anteriormente.	10
8. Al frente se muestra una terminal del host de VirtualBox. En ella se puede ver cómo está conectada a mi usuario (jlp) de la máquina virtual de Ubuntu Server, cuya IP es 192.168.56.101, y ejecutando gedit remotamente (de fondo en la captura).	10
9. <code>ssh-keygen</code> genera un par de claves pública y privada, y guarda ambas en archivos. Aquí podemos ver un ejemplo en mi máquina local.	11
10. En esta captura vemos la ejecución de <code>ssh-copy-id</code> , que por defecto busca la clave pública más reciente y la copia en el servidor al que pertenece el usuario al que queremos acceder. Podemos comprobar que la conexión posterior a este proceso se completa sin introducir la contraseña.	11
11. En el primer intento de login remoto, el parámetro <code>PermitRootLogin</code> tiene el valor <code>yes</code> , lo cual hace que el usuario root pueda acceder. Cambiando la configuración a <code>PermitRootLogin no</code> , comprobamos que obtiene un error de permiso denegado.	12
12. Tras cambiar su valor, en la imagen se puede ver que la línea correspondiente al puerto de ssh contiene el valor 22022. Para probar el funcionamiento, reiniciamos el servicio ssh y abrimos el nuevo puerto.	12
13. Tras las acciones de la captura anterior, comprobamos que el acceso en el puerto 22022 es correcto.	13
14. Tras los pasos anteriores, observamos que Apache y MySQL están ejecutándose, y que el intérprete de Python funciona.	14
15. Creando el archivo <code>/var/www/html/info.php</code> , cuyo contenido llama a <code>phpinfo()</code> , comprobamos que PHP funciona. Se muestra una captura de pantalla al acceder a él desde el host de Virtualbox en <code>http://192.168.56.101/info.php</code>	14
16. De manera similar a Ubuntu Server, comprobamos que Apache, MariaDB y Python funcionan.	15

17.	Aquí debemos seleccionar entre instalar características del servidor o herramientas de escritorio remoto. Elegimos la primera opción.	15
18.	En esta pantalla seleccionamos el servidor. Como sólo hay uno, la elección está clara.	16
19.	Buscamos la opción de servidor web (IIS) y la marcamos.	16
20.	Aquí podemos leer una pequeña explicación sobre la función de IIS antes de instalarlo.	17
21.	En esta sección debemos asegurarnos de tener marcadas las siguientes casillas: <i>Estado y Diagnóstico</i> , <i>Herramientas de registro</i> , <i>Seguimiento</i> , <i>Scripts y herramientas de administración de IIS</i> y <i>Servicio de administración</i> . . .	17
22.	El paso final antes de instalar es confirmar todo lo que hemos seleccionado en etapas anteriores.	18
23.	Pantalla final tras instalar con éxito.	18
24.	Le asignamos la IP 192.168.56.103	19
25.	Acceso desde el navegador a la IP 192.168.56.103 correspondiente a Windows Server. Se observa cómo IIS funciona.	19
26.	Mediante la orden diff obtengo las diferencias entre ambas versiones del programa. Si ejecuto el comando patch sobre la versión antigua, se comprueba que el contenido cambia en concordancia con la versión nueva. . .	20
27.	Instalación de dependencias de Webmin.	21
28.	Pantalla de información del sistema de Webmin.	21
29.	En la captura se aprecia cómo el puerto se cambia al 22022. Tras esto, se guardan los cambios.	22
30.	Como podemos observar, entramos con éxito por el puerto 22022, y vemos que en <code>/etc/ssh/sshd_config</code> aparece la modificación.	22
31.	Selección del servidor que vamos a administrar con phpMyAdmin.	23
32.	Elección entre configurar automáticamente la base de datos o no.	23
33.	Creación de contraseña.	24
34.	Pantalla de usuarios mostrada tras entrar como usuario root a phpMyAdmin. .	24
35.	Mediante grep filtro las líneas correspondientes a los parámetros modificados para mostrar los nuevos valores.	24
36.	En esta captura se observa la creación de un nuevo usuario, rellenando distintos campos. Posteriormente, se utilizará para definir un nuevo sitio web perteneciente a él.	25
37.	Aquí podemos ver la creación de un sitio web asociado al usuario definido anteriormente. Podemos asignar, entre otras cosas, su dominio, su cuota de disco o su tráfico máximo.	25
38.	El ejemplo de grep sólo devuelve el propio proceso encontrándose a sí mismo buscando firefox, ya que éste no está siquiera instalado (mucho menos ejecutándose). El ejemplo de find (ligeramente adaptado) no encuentra archivos PDF que copiar, por lo que la carpeta destino está vacía.	26
39.	Al principio, el acceso por contraseña está desactivado. Lanzo el script y detengo su proceso para comprobar que ha cambiado el valor. Finalmente, lo reanudo y veo que lo ha devuelto a su valor original al terminar.	26
40.	Ejecución del ejemplo del comando awk	27

- 41. Mostramos el valor de `PasswordAuthentication` antes de ejecutar el script, justo cuando se espera que lo haya cambiado, y después de terminar. Vemos que los resultados concuerdan. 28
- 42. Abrimos *notepad* desde la consola. 28
- 43. Usando la orden `stop-process -Name notepad` lo cerramos y vemos que ya no aparece en la barra de tareas. 29

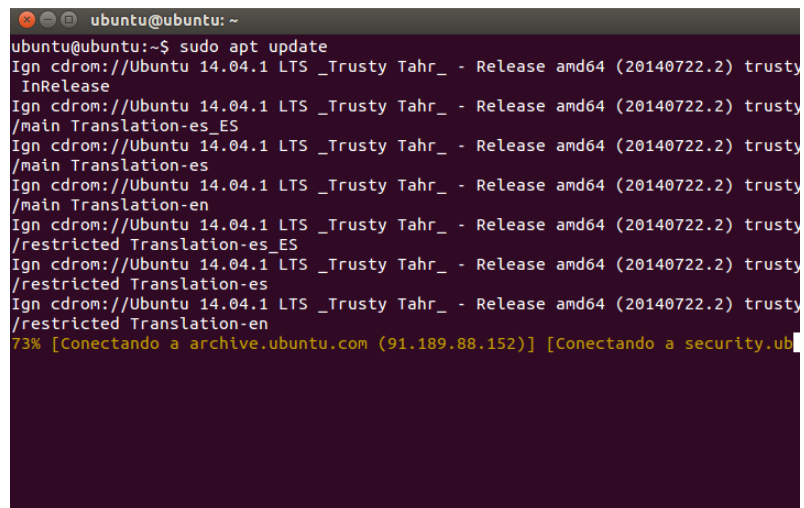
1. Cuestión 1: a) Liste los argumentos de yum necesarios para instalar, buscar y eliminar paquetes. b) ¿Qué ha de hacer para que yum pueda tener acceso a Internet en el PC del aula? (Pistas: archivo de configuración en /etc, proxy: stargate.ugr.es:3128) c) ¿Cómo añadimos un nuevo repositorio?

- a) Siempre como superusuario:
 - Instalar: `yum install <paquetes>`. [1]
 - Eliminar: `yum remove <paquetes>`. [1]
 - Buscar: `yum search <término a buscar>`. [1]
- b) Debemos editar el archivo `/etc/yum.conf` [2] y añadir:
`proxy=http://stargate.ugr.es:3128`
- c) Según la documentación oficial de CentOS [3], hay que copiar el archivo de definición del repositorio en `/etc/yum.repos.d/`. Esto se puede hacer mediante la orden `su -c 'cp example.repo /etc/yum.repos.d/'` siempre que tengamos la contraseña del administrador.

2. Cuestión 2: a) Liste los argumentos de apt necesarios para instalar, buscar y eliminar paquetes. 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) c) ¿Cómo añadimos un nuevo repositorio?

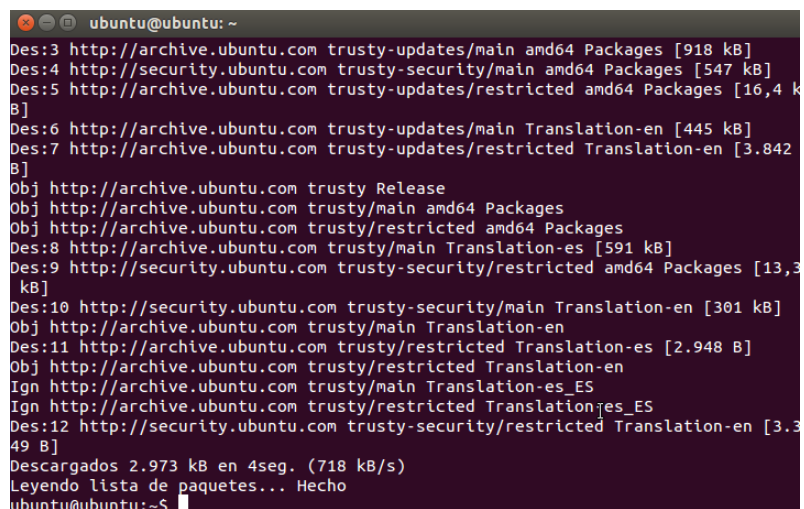
- a) Siempre como superusuario:
 - Instalar: `apt-get install <paquetes>`. [4]
 - Eliminar: `apt-get remove <paquetes>` o `apt-get purge` para eliminar también los archivos de configuración asociados. [4]
 - Buscar: `apt-cache search <término a buscar>`. [5]
- b) Hemos de editar el archivo `/etc/apt/apt.conf` [6] y añadir la siguiente línea:
`Acquire::http::Proxy "http://stargate.ugr.es:3128";`.

A continuación se observa el funcionamiento previo y posterior a esta acción:



```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ sudo apt update  
Ign cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release amd64 (20140722.2) trusty  
InRelease  
Ign cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release amd64 (20140722.2) trusty  
/main Translation-es_ES  
Ign cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release amd64 (20140722.2) trusty  
/main Translation-es  
Ign cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release amd64 (20140722.2) trusty  
/main Translation-en  
Ign cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release amd64 (20140722.2) trusty  
/restricted Translation-es_ES  
Ign cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release amd64 (20140722.2) trusty  
/restricted Translation-es  
Ign cdrom://Ubuntu 14.04.1 LTS _Trusty Tahr_ - Release amd64 (20140722.2) trusty  
/restricted Translation-en  
73% [Conectando a archive.ubuntu.com (91.189.88.152)] [Conectando a security.ub
```

Figura 1: Ejecución de `sudo apt update` que no llega a terminar debido a que no puede acceder al exterior.



```
ubuntu@ubuntu: ~  
Des:3 http://archive.ubuntu.com trusty-updates/main amd64 Packages [918 kB]  
Des:4 http://security.ubuntu.com trusty-security/main amd64 Packages [547 kB]  
Des:5 http://archive.ubuntu.com trusty-updates/restricted amd64 Packages [16,4 k  
B]  
Des:6 http://archive.ubuntu.com trusty-updates/main Translation-en [445 kB]  
Des:7 http://archive.ubuntu.com trusty-updates/restricted Translation-en [3.842  
B]  
Obj http://archive.ubuntu.com trusty Release  
Obj http://archive.ubuntu.com trusty/main amd64 Packages  
Obj http://archive.ubuntu.com trusty/restricted amd64 Packages  
Des:8 http://archive.ubuntu.com trusty/main Translation-es [591 kB]  
Des:9 http://security.ubuntu.com trusty-security/restricted amd64 Packages [13,3  
kB]  
Des:10 http://security.ubuntu.com trusty-security/main Translation-en [301 kB]  
Obj http://archive.ubuntu.com trusty/main Translation-en  
Des:11 http://archive.ubuntu.com trusty/restricted Translation-es [2.948 B]  
Obj http://archive.ubuntu.com trusty/restricted Translation-en  
Ign http://archive.ubuntu.com trusty/main Translation-es_ES  
Ign http://archive.ubuntu.com trusty/restricted Translation-es_ES  
Des:12 http://security.ubuntu.com trusty-security/restricted Translation-en [3.3  
49 B]  
Descargados 2.973 kB en 4seg. (718 kB/s)  
Leyendo lista de paquetes... Hecho  
ubuntu@ubuntu:~$
```

Figura 2: Ejecución de `sudo apt update` tras modificar el archivo de configuración. Comprobamos que acaba normalmente.

- c) Se puede hacer de varias maneras [7] (con privilegios de superusuario):
 - Editando el archivo `/etc/apt/sources.list`, que enumera fuentes (repositorios).
 - Añadiendo o editando un archivo con extensión `.list` bajo el directorio `/etc/apt/sources.list.d/`.
 - Utilizando `add-apt-repository <repositorio>`.

3. Cuestión 3: a) ¿Con qué comando puede abrir/cerrar un puerto usando ufw? Muestre un ejemplo de cómo lo ha hecho. b) ¿Con qué comando puede abrir/cerrar un puerto usando firewall-cmd en CentOS? Muestre un ejemplo de cómo lo ha hecho. c) Utilice el comando nmap para ver que, efectivamente, los puertos están accesibles.

- a) y c) Con el comando `sudo ufw allow/deny <puerto>` (según man ufw). Para deshacer la acción habría que introducir `sudo ufw delete allow/deny <puerto>`.

```
javi@javi-X555LJ:~$ nmap -Pn -p 22 192.168.56.101

Starting Nmap 6.40 ( http://nmap.org ) at 2016-11-11 17:58 CET
Nmap scan report for 192.168.56.101
Host is up (0.00021s latency).
PORT      STATE SERVICE
22/tcp    open  ssh

Nmap done: 1 IP address (1 host up) scanned in 0.03 seconds
javi@javi-X555LJ:~$
```

Figura 3: Mediante `sudo ufw allow 22` he abierto en la máquina virtual (192.168.56.101) el puerto por defecto de ssh, el 22. Con `nmap -Pn -p 22 192.168.56.101` compruebo desde el host que dicho puerto está abierto.

```
javi@javi-X555LJ:~$ nmap -Pn -p 22 192.168.56.101

Starting Nmap 6.40 ( http://nmap.org ) at 2016-11-11 17:53 CET
Nmap scan report for 192.168.56.101
Host is up.
PORT      STATE SERVICE
22/tcp    filtered ssh

Nmap done: 1 IP address (1 host up) scanned in 2.03 seconds
javi@javi-X555LJ:~$
```

Figura 4: Mediante `sudo ufw deny 22` he cerrado en la máquina virtual el puerto por defecto de ssh, el 22. Con `nmap -Pn -p 22 192.168.56.101` compruebo que está filtrado por el firewall.

- b) y c) Según `man firewall-cmd`, con los comandos `sudo firewall-cmd --zone=public --add-port=22/tcp --permanent` y `sudo firewall-cmd --reload` (para recargar la configuración) abriríamos el puerto 22 (por ejemplo). Para deshacer esta acción, se ejecutaría el mismo comando pero sustituyendo `add` por `remove`.

```
jlp@localhost mar nov 22:~$ sudo firewall-cmd --zone=public --add-port=22/tcp --permanent
[sudo] password for jlp:
success
jlp@localhost mar nov 22:~$ sudo firewall-cmd --reload
success
jlp@localhost mar nov 22:~$ _
```

Figura 5: Usando los comandos anteriores, abrimos el puerto 22 y recargamos la configuración de `firewall-cmd`.

```
javi@javi-X555LJ:~$ nmap -Pn -p 22 192.168.56.102

Starting Nmap 6.40 ( http://nmap.org ) at 2016-11-22 19:40 CET
Nmap scan report for 192.168.56.102
Host is up (0.00025s latency).
PORT      STATE SERVICE
22/tcp    open  ssh

Nmap done: 1 IP address (1 host up) scanned in 0.14 seconds
javi@javi-X555LJ:~$
```

Figura 6: Ejecutando `nmap` desde el host, observamos que el puerto 22 de CentOS (en la dirección 192.168.56.102) está, en efecto, abierto.

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

La diferencia principal es la seguridad. Telnet no está recomendado porque, entre otras vulnerabilidades, envía los datos en texto plano, siendo susceptibles de robo y posterior uso malicioso; esto es especialmente peligroso en el caso de las contraseñas. En cambio, ssh proporciona cifrado de la información y métodos de autenticación. [8, p. 376-377]

5. Cuestión 5: a) ¿Para qué sirve la opción `-X`? 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?

- a) En esencia sirve para ejecutar en la máquina local la interfaz gráfica del programa que ejecutamos remotamente en el servidor (mediante X11). [9]
- b) En primer lugar, hay que instalar `gedit`, ya que no viene instalado por defecto. Tras esto, si no añadimos la opción `-X`, ocurre un error al no poder ejecutar el servidor la interfaz gráfica de `gedit` ya que no tiene instalada ninguna.

```
javi@javi-X555LJ:~$ ssh jlp@192.168.56.101
jlp@192.168.56.101's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Pueden actualizarse 54 paquetes.
0 actualizaciones son de seguridad.

Last login: Thu Nov  3 21:26:36 2016 from 192.168.56.1
jlp@UbuntuServerISEjue nov 03:~$ gedit
Failed to connect to Mir: Failed to connect to server socket: No such file or directory
Unable to init server: Could not connect: Connection refused

(gedit:1999): Gtk-WARNING **: cannot open display:
jlp@UbuntuServerISEjue nov 03:~$
```

Figura 7: En esta primera captura se puede ver cómo no es posible ejecutar gedit sin la opción `-X` debido a lo explicado anteriormente.

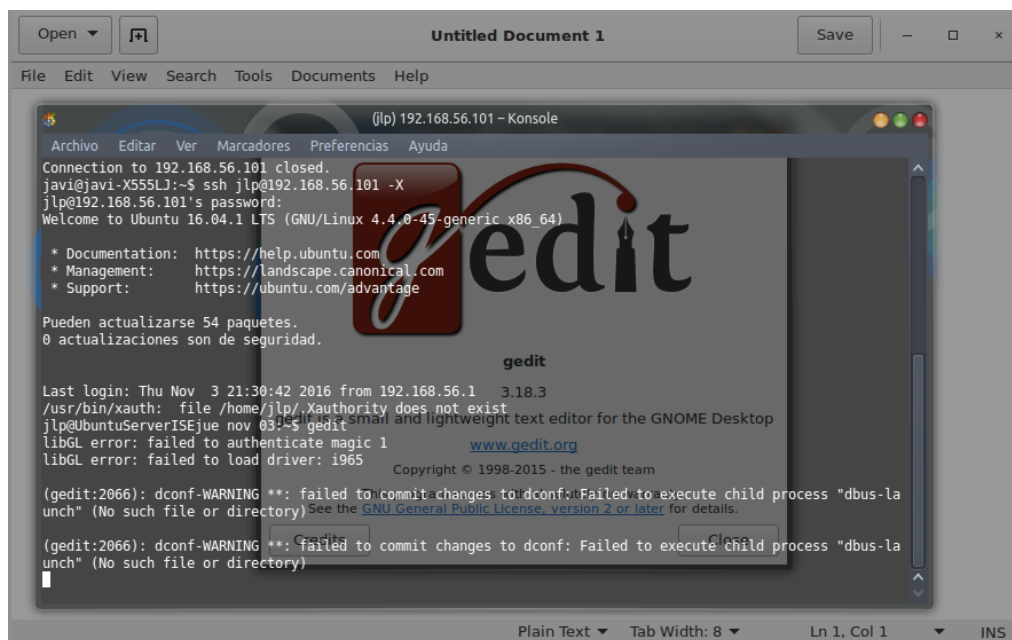


Figura 8: Al frente se muestra una terminal del host de VirtualBox. En ella se puede ver cómo está conectada a mi usuario (jlp) de la máquina virtual de Ubuntu Server, cuya IP es 192.168.56.101, y ejecutando gedit remotamente (de fondo en la captura).

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).

Para poder acceder remotamente sin introducir la contraseña, debemos generar dos claves: una privada que guardamos y una pública que enviamos a cada máquina a la que queramos conectarnos.

```
javi@javi-X555LJ:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/javi/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/javi/.ssh/id_rsa.
Your public key has been saved in /home/javi/.ssh/id_rsa.pub.
The key fingerprint is:
c0:44:cd:9f:f9:b6:41:6a:16:0b:14:02:2e:55:d9:60 javi@javi-X555LJ
The key's randomart image is:
+--[ RSA 2048 ]-----+
|      o+E*..          |
|      O +..=         |
|     . . O . . O     |
|      . . . = .      |
|       S. *          |
|       = +          |
|      O . O          |
|      .              |
+-----+
javi@javi-X555LJ:~$
```

Figura 9: ssh-keygen genera un par de claves pública y privada, y guarda ambas en archivos. Aquí podemos ver un ejemplo en mi máquina local.

```
javi@javi-X555LJ:~$ ssh-copy-id jlp@192.168.56.101 -p 22
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already in
stalled
/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
jlp@192.168.56.101's password:
Number of key(s) added: 1

Now try logging into the machine, with: "ssh -p '22' 'jlp@192.168.56.101'"
and check to make sure that only the key(s) you wanted were added.

javi@javi-X555LJ:~$ ssh -p '22' 'jlp@192.168.56.101'
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Pueden actualizarse 54 paquetes.
0 actualizaciones son de seguridad.

Last login: Fri Nov  4 17:06:51 2016 from 192.168.56.1
jlp@UbuntuServerISEvie nov 04:~$
```

Figura 10: En esta captura vemos la ejecución de ssh-copy-id, que por defecto busca la clave pública más reciente y la copia en el servidor al que pertenece el usuario al que queremos acceder. Podemos comprobar que la conexión posterior a este proceso se completa sin introducir la contraseña.

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 indica el manual para el servidor ssh (`man sshd_config`), el archivo que contiene la configuración global de ssh es `/etc/ssh/sshd_config`. Para evitar que el usuario root acceda debemos darle al parámetro `PermitRootLogin` el valor `no`. A continuación se muestra el resultado del login para los valores `yes` y `no`:

```
javi@javi-X555LJ:~$ ssh root@192.168.56.101
root@192.168.56.101's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Pueden actualizarse 54 paquetes.
0 actualizaciones son de seguridad.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@UbuntuServerISE:~# exit
logout
Connection to 192.168.56.101 closed.
javi@javi-X555LJ:~$ ssh root@192.168.56.101
root@192.168.56.101's password:
Permission denied, please try again.
root@192.168.56.101's password:
```

Figura 11: En el primer intento de login remoto, el parámetro `PermitRootLogin` tiene el valor `yes`, lo cual hace que el usuario `root` pueda acceder. Cambiando la configuración a `PermitRootLogin no`, comprobamos que obtiene un error de permiso denegado.

Para cambiar el puerto por defecto, modificamos el parámetro `Port`.

```
j1p@UbuntuServerISEdom nov 06:~$ cat /etc/ssh/sshd_config | grep Port
Port 22022
j1p@UbuntuServerISEdom nov 06:~$ sudo service ssh restart
j1p@UbuntuServerISEdom nov 06:~$ sudo ufw allow 22022
Rule added
Rule added (v6)
```

Figura 12: Tras cambiar su valor, en la imagen se puede ver que la línea correspondiente al puerto de ssh contiene el valor 22022. Para probar el funcionamiento, reiniciamos el servicio ssh y abrimos el nuevo puerto.

```
javi@javi-X555LJ:~$ ssh jlp@192.168.56.101 -p 22022
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Pueden actualizarse 54 paquetes.
0 actualizaciones son de seguridad.

Last login: Sun Nov  6 12:43:03 2016 from 192.168.56.1
jlp@UbuntuServerISEdom nov 06:~$
```

Figura 13: Tras las acciones de la captura anterior, comprobamos que el acceso en el puerto 22022 es correcto.

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.

La experiencia en el ejercicio anterior y la ayuda de Ubuntu [10] nos indican que debemos reiniciar el servicio (o al menos hacer que vuelva a cargar la configuración) para que los cambios tengan efecto.

Para Ubuntu, un ejemplo de reinicio de servicio lo tenemos en la figura 12 con la orden `sudo service ssh restart`.

Para CentOS se puede ejecutar lo mismo, aunque con un matiz: el comando `service` ya no existe en CentOS 7, por lo que redirige automáticamente a `systemctl [11] restart <servicio>.service`. En el caso particular de `ssh` el servicio se llama `sshd`.

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.

Para instalar *LAMP* en Ubuntu Server he utilizado los siguientes comandos (después de ejecutar `sudo apt-get update`):

- Apache: `sudo apt-get install apache2`.
- MySQL: `sudo apt-get install mysql-server`.
- Python: he elegido Python en lugar de PHP, y ya estaba instalado por defecto. De todas formas, para la cuestión 13 he de instalar PHP, lo cual hago mediante `sudo apt-get install php libapache2-mod-php php-mcrypt php-mysql`.

Además, hay que iniciar Apache y MySQL usando `service [apache2/mysql-server] start`.

```

jlp@UbuntuServerISEdom nov 06:~$ ps -Af | grep apache2
root      5934      1  0 13:41 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  5937    5934  0 13:41 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  5938    5934  0 13:41 ?        00:00:00 /usr/sbin/apache2 -k start
jlp       6008    1560  0 13:42 tty1    00:00:00 grep --color=auto apache2
jlp@UbuntuServerISEdom nov 06:~$ ps -Af | grep mysql
mysql     5732      1  0 13:40 ?        00:00:00 /usr/sbin/mysqld
jlp       6010    1560  0 13:42 tty1    00:00:00 grep --color=auto mysql
jlp@UbuntuServerISEdom nov 06:~$ python
Python 2.7.12 (default, Jul 1 2016, 15:12:24)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> _

```

Figura 14: Tras los pasos anteriores, observamos que Apache y MySQL están ejecutándose, y que el intérprete de Python funciona.


PHP Version 7.0.8-0ubuntu0.16.04.3	
	
System	Linux UbuntuServerISE 4.4.0-45-generic #66-Ubuntu SMP Wed Oct 19 14:12:37 UTC 2016 x86_64
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.0/apache2
Loaded Configuration File	/etc/php/7.0/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.0/apache2/conf.d
Additional .ini files parsed	/etc/php/7.0/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.0/apache2/conf.d/10-opcache.ini, /etc/php/7.0/apache2/conf.d/10-pdo.ini, /etc/php/7.0/apache2/conf.d/15-xml.ini, /etc/php/7.0/apache2/conf.d/20-calendar.ini, /etc/php/7.0/apache2/conf.d/20-ctype.ini, /etc/php/7.0/apache2/conf.d/20-dom.ini, /etc/php/7.0/apache2/conf.d/20-exif.ini, /etc/php/7.0/apache2/conf.d/20-fileinfo.ini, /etc/php/7.0/apache2/conf.d/20-ftp.ini, /etc/php/7.0/apache2/conf.d/20-gd.ini, /etc/php/7.0/apache2/conf.d/20-gettext.ini, /etc/php/7.0/apache2/conf.d/20-iconv.ini, /etc/php/7.0/apache2/conf.d/20-json.ini, /etc/php/7.0/apache2/conf.d/20-mbstring.ini, /etc/php/7.0/apache2/conf.d/20-mcrypt.ini, /etc/php/7.0/apache2/conf.d/20-mysql.ini, /etc/php/7.0/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.0/apache2/conf.d/20-ghar.ini, /etc/php/7.0/apache2/conf.d/20-posix.ini, /etc/php/7.0/apache2/conf.d/20-readline.ini, /etc/php/7.0/apache2/conf.d/20-shmop.ini, /etc/php/7.0/apache2/conf.d/20-simplexml.ini, /etc/php/7.0/apache2/conf.d/20-sockets.ini, /etc/php/7.0/apache2/conf.d/20-sysmsg.ini, /etc/php/7.0/apache2/conf.d/20-syssem.ini, /etc/php/7.0/apache2/conf.d/20-sysvshm.ini, /etc/php/7.0/apache2/conf.d/20-tokenizer.ini, /etc/php/7.0/apache2/conf.d/20-wddx.ini, /etc/php/7.0/apache2/conf.d/20-xmlreader.ini, /etc/php/7.0/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.0/apache2/conf.d/20-xsl.ini
PHP API	20151012
PHP Extension	20151012
Zend Extension	320151012
Zend Extension Build	API320151012.NTS
PHP Extension Build	API20151012.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled

Figura 15: Creando el archivo `/var/www/html/info.php`, cuyo contenido llama a `phpinfo()`, comprobamos que PHP funciona. Se muestra una captura de pantalla al acceder a él desde el host de Virtualbox en `http://192.168.56.101/info.php`.

Por otra parte, para instalar *LAMP* en CentOS:

- Apache: `yum install httpd`
- MariaDB: `yum install mariadb-server mariadb`
- Python: de la misma forma que Ubuntu Server, ya está instalado en CentOS. Igualmente, aunque ya tenemos Python, podríamos instalar PHP para cuestiones posteriores. El comando es `yum install php`. Además, por si acaso, es posible instalar también un paquete para trabajar con MySQL (MariaDB): `php-mysql`.

Para iniciar Apache y MariaDB, usamos:

```
systemctl start [httpd.service/mariadb.service].
```

```

jlp@localhost mar nov 22:~$ ps -af | grep apache
apache 3212 3211 0 20:50 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache 3213 3211 0 20:50 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache 3214 3211 0 20:50 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache 3215 3211 0 20:50 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache 3216 3211 0 20:50 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
jlp     3524 2408 0 20:51 tty1     00:00:00 grep --color=auto apache
jlp@localhost mar nov 22:~$ ps -af | grep mysql
mysql   3331 1 0 20:51 ?        00:00:00 /bin/sh /usr/bin/mysqld_safe --b
asedir=/usr
mysql   3488 3331 0 20:51 ?        00:00:00 /usr/libexec/mysqld --basedir=/u
sr --datadir=/var/lib/mysql --plugin-dir=/usr/lib64/mysql/plugin --log-error=/va
r/log/mariadb/mariadb.log --pid-file=/var/run/mariadb/mariadb.pid --socket=/var/
lib/mysql/mysql.sock
jlp     3526 2408 0 20:51 tty1     00:00:00 grep --color=auto mysql
jlp@localhost mar nov 22:~$ python
Python 2.7.5 (default, Nov 20 2015, 02:00:19)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-4)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> _

```

Figura 16: De manera similar a Ubuntu Server, comprobamos que Apache, MariaDB y Python funcionan.

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.

Ya que dispongo de GUI, realizaré la instalación a través de ella. En primer lugar, elegimos el tipo de instalación que queremos realizar (figura 17). Tras esto, hemos de seleccionar en qué servidor vamos a instalar características (figura 18). Después, es el momento de seleccionar el servidor IIS (figura 19). En la pantalla de características no tenemos que hacer nada, así que la ignoramos. A continuación, tenemos una explicación acerca de IIS (figura 20) y, por último, la selección de características según lo indicado en el guión de la práctica (figura 21) y la confirmación del proceso (figura 22). Si todo ha salido bien, debe aparecernos algo similar a la figura 23.



Figura 17: Aquí debemos seleccionar entre instalar características del servidor o herramientas de escritorio remoto. Elegimos la primera opción.

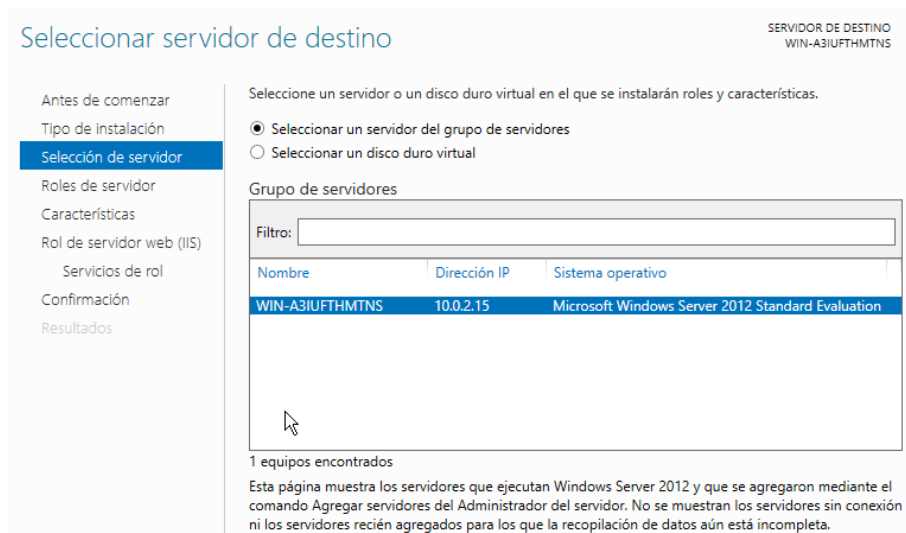


Figura 18: En esta pantalla seleccionamos el servidor. Como sólo hay uno, la elección está clara.

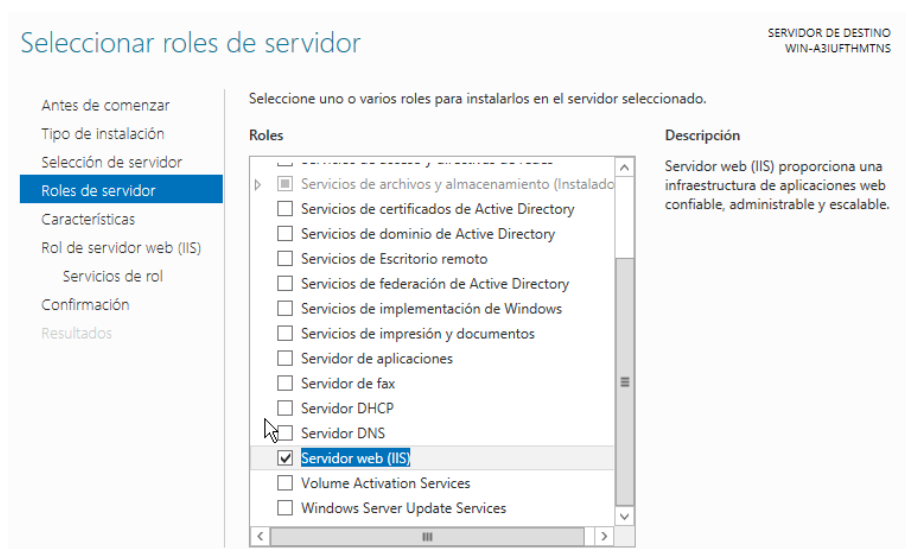


Figura 19: Buscamos la opción de servidor web (IIS) y la marcamos.

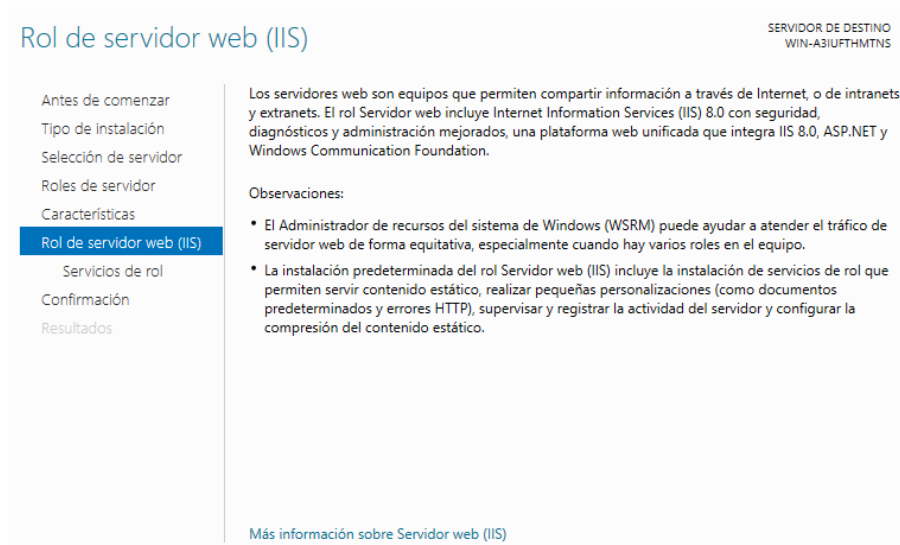


Figura 20: Aquí podemos leer una pequeña explicación sobre la función de IIS antes de instalarlo.

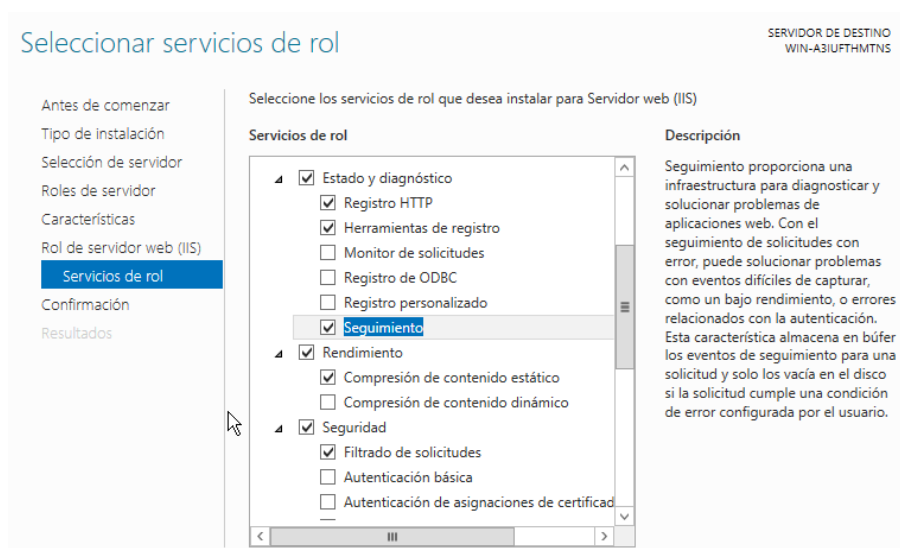


Figura 21: En esta sección debemos asegurarnos de tener marcadas las siguientes casillas: *Estado y Diagnóstico*, *Herramientas de registro*, *Seguimiento*, *Scripts y herramientas de administración de IIS* y *Servicio de administración*.

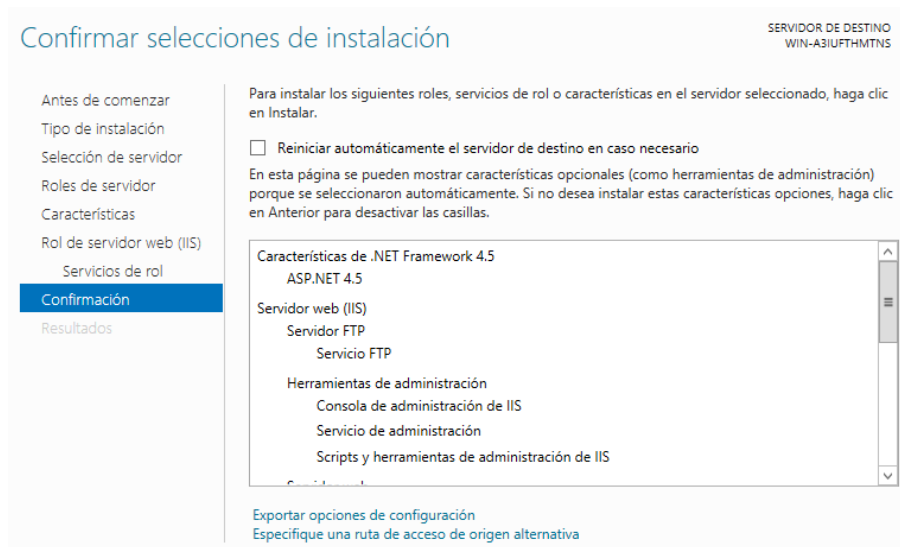


Figura 22: El paso final antes de instalar es confirmar todo lo que hemos seleccionado en etapas anteriores.

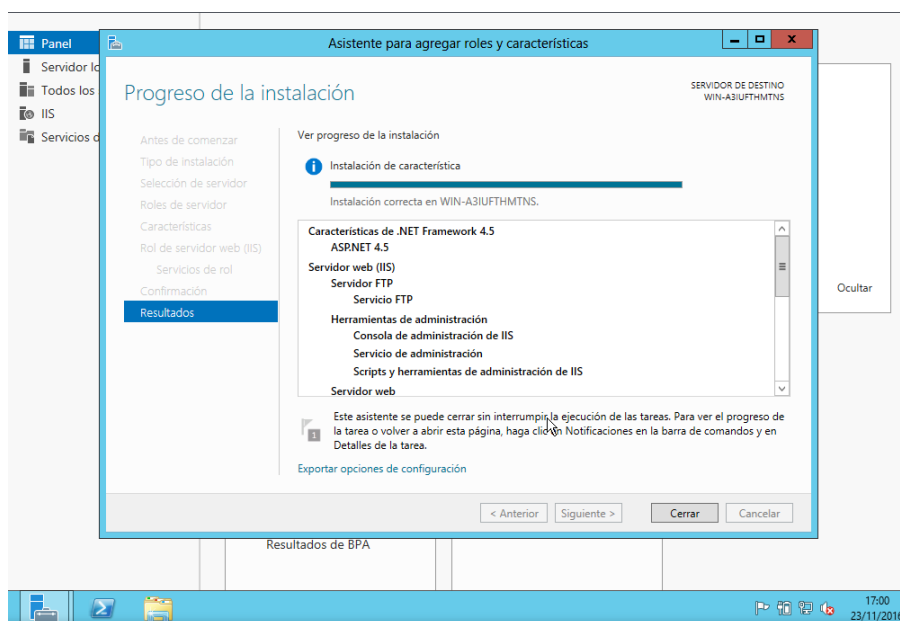


Figura 23: Pantalla final tras instalar con éxito.

Ya instalado IIS hemos de comprobar su funcionamiento. Para ello, utilizamos una red *Host-only* en VirtualBox, y comprobamos que Windows Server la reconoce. Asignamos una IP estática:

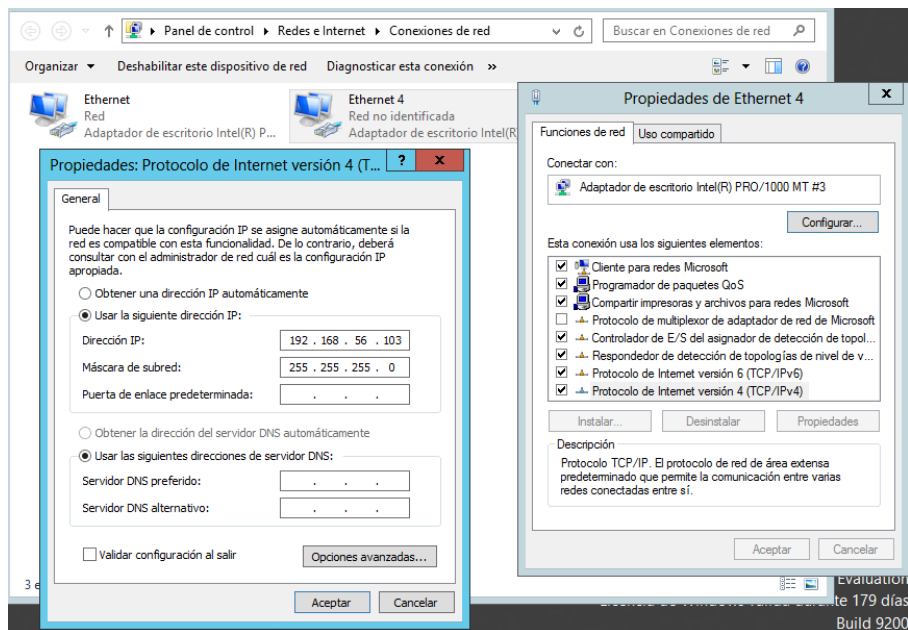


Figura 24: Le asignamos la IP 192.168.56.103 .

Finalmente, accediendo desde el host de VirtualBox a la IP definida, vemos que IIS está activo:



Figura 25: Acceso desde el navegador a la IP 192.168.56.103 correspondiente a Windows Server. Se observa cómo IIS funciona.

11. Cuestión 11: Muestre un ejemplo de uso del comando.

Voy a seguir las instrucciones de [esta página](#), incluida también como referencia ([12]).

En primer lugar, voy a crear dos pequeños programas similares; uno servirá de versión antigua y otro de versión actualizada:

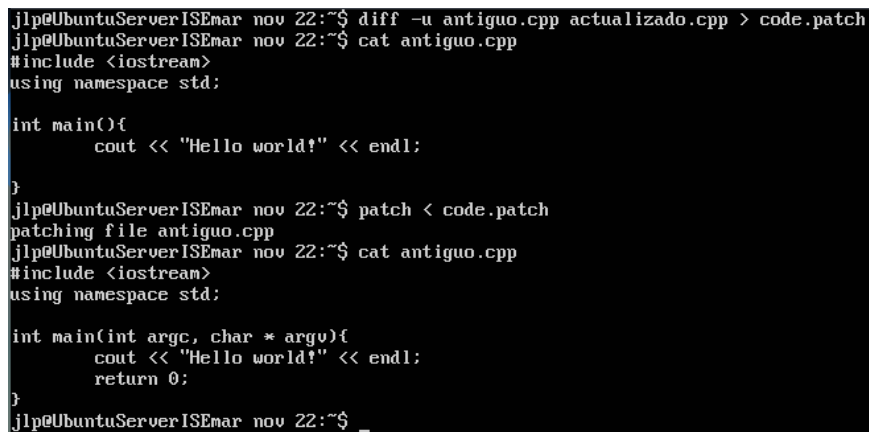
```
// Archivo antiguo.
#include <iostream>
using namespace std;

int main(){
    cout << "Hello world!" << endl;
}

// Archivo nuevo.
#include <iostream>
using namespace std;

int main(int argc, char * argv){
    cout << "Hello world!" << endl;
    return 0;
}
```

Para aplicar un parche, primero se crea con `diff` y luego se aplica con `patch`, como se ve en la siguiente captura:



```
jlpe@UbuntuServerISEmar nov 22:~$ diff -u antiguo.cpp actualizado.cpp > code.patch
jlpe@UbuntuServerISEmar nov 22:~$ cat antiguo.cpp
#include <iostream>
using namespace std;

int main(){
    cout << "Hello world!" << endl;
}
jlpe@UbuntuServerISEmar nov 22:~$ patch < code.patch
patching file antiguo.cpp
jlpe@UbuntuServerISEmar nov 22:~$ cat antiguo.cpp
#include <iostream>
using namespace std;

int main(int argc, char * argv){
    cout << "Hello world!" << endl;
    return 0;
}
jlpe@UbuntuServerISEmar nov 22:~$ _
```

Figura 26: Mediante la orden `diff` obtengo las diferencias entre ambas versiones del programa. Si ejecuto el comando `patch` sobre la versión antigua, se comprueba que el contenido cambia en concordancia con la versión nueva.

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.

Para realizar la instalación, he seguido las instrucciones de la página oficial de Webmin [13]. Como anticipa la referencia, he necesitado instalar paquetes extra (el resto del proceso es exactamente igual):

```
jlope@UbuntuServer:~$ sudo apt-get install libnet-ssleay-perl libauthen-pam-perl libio-pty-perl apt-show-versions libapt-pkg-perl
```

Figura 27: Instalación de dependencias de Webmin.

Tras la instalación, el acceso desde el host muestra la siguiente pantalla:

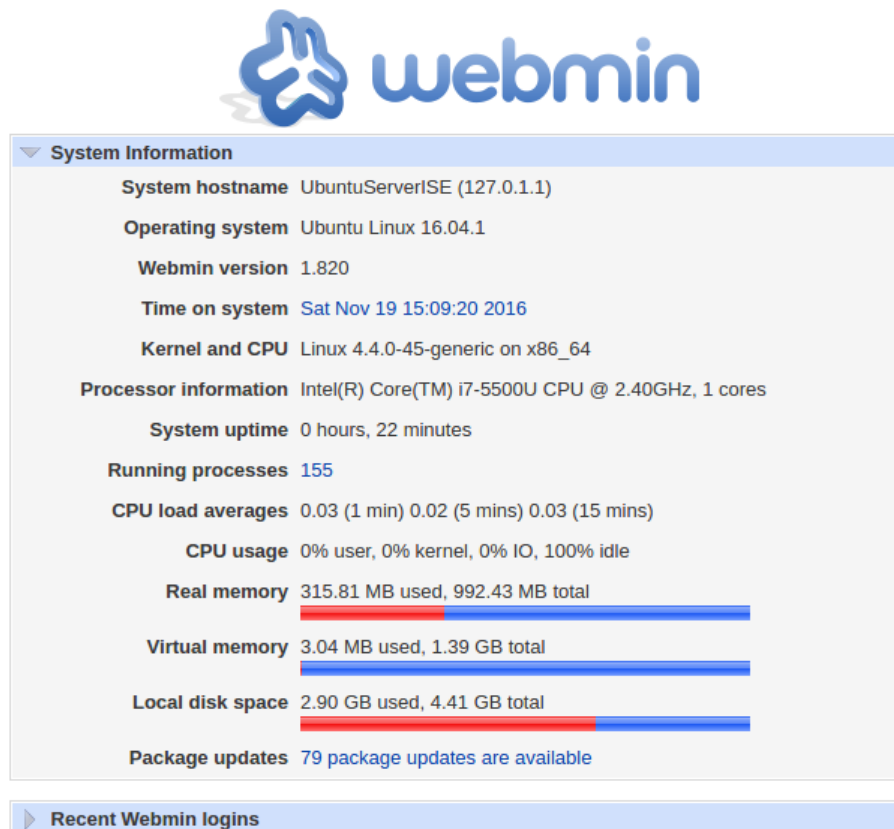
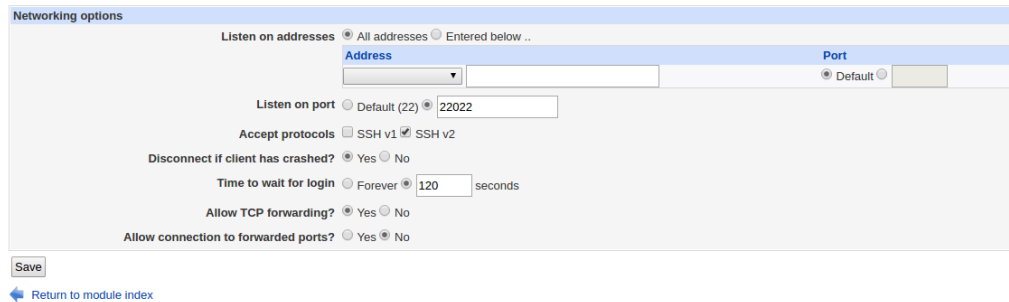


Figura 28: Pantalla de información del sistema de Webmin.

Probamos a cambiar el puerto en el que escucha el servicio ssh de 22 a 22022:



Networking options

Listen on addresses ☒ All addresses ☐ Entered below ..

Address	Port
	<input checked="" type="radio"/> Default <input type="radio"/>

Listen on port ☐ Default (22) ☒ 22022

Accept protocols ☐ SSH v1 ☒ SSH v2

Disconnect if client has crashed? ☒ Yes ☐ No

Time to wait for login ☐ Forever ☒ 120 seconds

Allow TCP forwarding? ☒ Yes ☐ No

Allow connection to forwarded ports? ☐ Yes ☒ No

[Return to module index](#)

Figura 29: En la captura se aprecia cómo el puerto se cambia al 22022. Tras esto, se guardan los cambios.

Sólo queda probar si se ha realizado con éxito:

```
javi@javi-X555LJ:~$ ssh jlp@192.168.56.101 -p 22022
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Pueden actualizarse 79 paquetes.
9 actualizaciones son de seguridad.

Last login: Sat Nov 19 15:32:34 2016 from 192.168.56.1
jlp@UbuntuServerISEsáb nov 19:~$ cat /etc/ssh/sshd_config | grep "Port "
Port 22022
jlp@UbuntuServerISEsáb nov 19:~$
```

Figura 30: Como podemos observar, entramos con éxito por el puerto 22022, y vemos que en `/etc/ssh/sshd_config` aparece la modificación.

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, en primer lugar debemos ejecutar `sudo apt-get install phpmyadmin`. Tras descargar los archivos necesarios, nos pide que elijamos el servidor que será configurado para funcionar con phpMyAdmin (figura 31). Después, nos pregunta si queremos que configure automáticamente la base de datos (figura 32) y, por último, nos solicita una contraseña (figura 33). Adicionalmente, debemos abrir el puerto por defecto (80). Para comprobar que todo funciona, accedemos con el usuario root (figura 34).

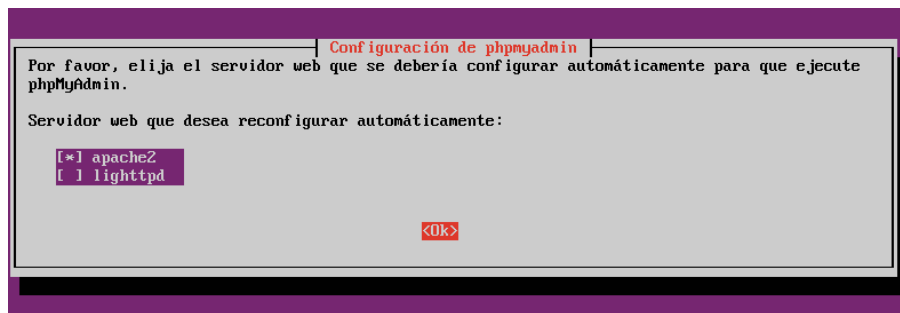


Figura 31: Selección del servidor que vamos a administrar con phpMyAdmin.

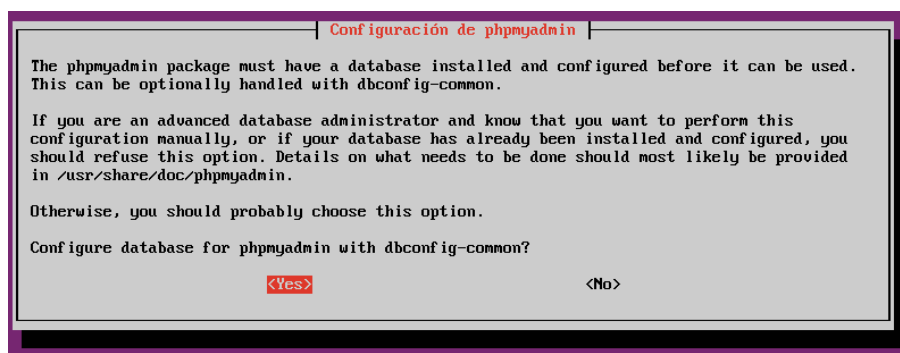


Figura 32: Elección entre configurar automáticamente la base de datos o no.

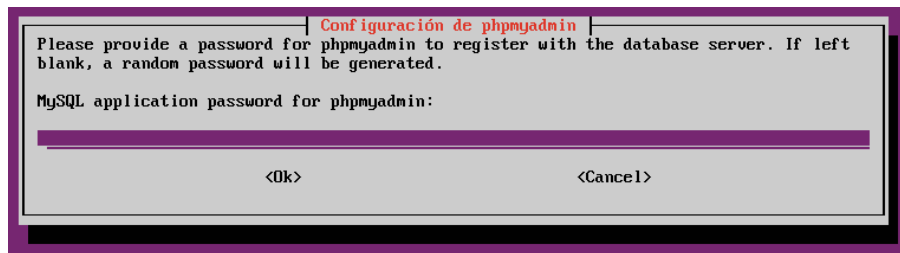


Figura 33: Creación de contraseña.



Figura 34: Pantalla de usuarios mostrada tras entrar como usuario root a phpMyAdmin.

Para poder importar bases de datos de hasta 25 MiB, debemos modificar el archivo `/etc/php/7.0/apache2/php.ini` (en la figura 15 se observa en el apartado **Loaded Configuration File**) y recargar el servidor Apache.

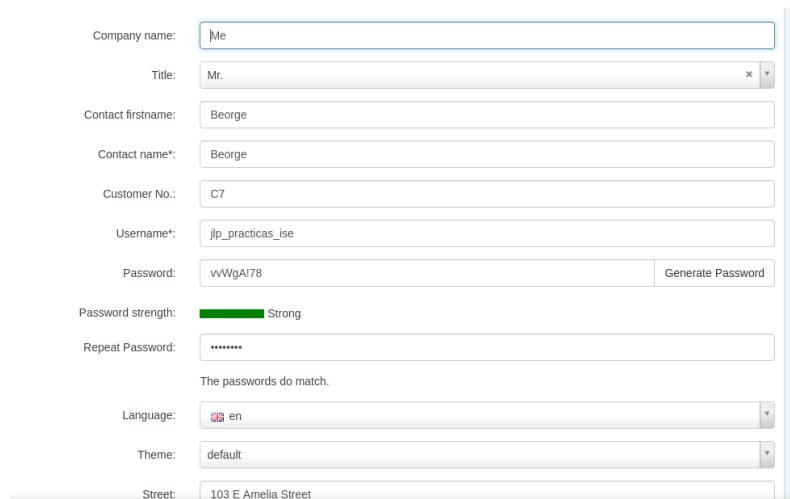
El parámetro principal que hay que cambiar es `post_max_size` [14], de 8M a 25M. Además, según indica la referencia, `memory_limit` debería ser mayor que dicho parámetro, así que habría que aumentarlo si fuera necesario (no es nuestro caso, ya que vale 128M). A continuación, se ve el resultado de explorar el archivo en busca de ambos parámetros tras cambiarlos.

```
j1p@UbuntuServerISEmar nov 22:~$ cat /etc/php/7.0/apache2/php.ini | grep post_max_size
post_max_size = 25M
j1p@UbuntuServerISEmar nov 22:~$ cat /etc/php/7.0/apache2/php.ini | grep memory_limit
memory_limit = 128M
j1p@UbuntuServerISEmar nov 22:~$
```

Figura 35: Mediante `grep` filtro las líneas correspondientes a los parámetros modificados para mostrar los nuevos valores.

14. Cuestión 14: Visite 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.

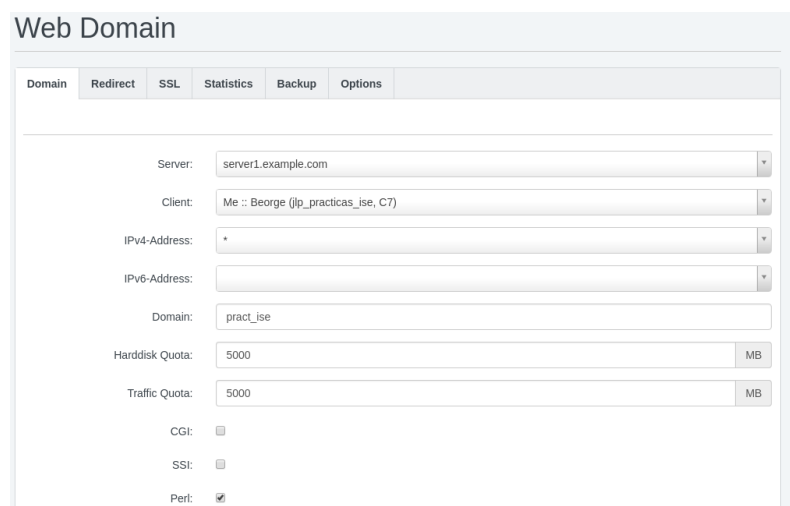
Para responder a esta cuestión he visitado [ISPConfig](#). He probado a crear un nuevo usuario y un nuevo sitio web.



The screenshot shows the 'Add User' form in ISPConfig. The fields are filled with the following data:

- Company name: Me
- Title: Mr.
- Contact firstname: George
- Contact name*: George
- Customer No.: C7
- Username*: jlp_practicas_ise
- Password: vWw9AI78 (with a 'Generate Password' button)
- Password strength: Strong (indicated by a green bar)
- Repeat Password: (masked with dots)
- The passwords do match.
- Language: en
- Theme: default
- Street: 103 E Amelia Street

Figura 36: En esta captura se observa la creación de un nuevo usuario, rellenando distintos campos. Posteriormente, se utilizará para definir un nuevo sitio web perteneciente a él.



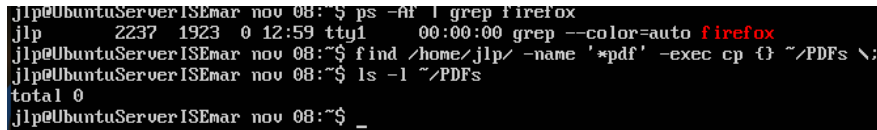
The screenshot shows the 'Web Domain' configuration page in ISPConfig. The 'Domain' tab is selected. The configuration is as follows:

- Server: server1.example.com
- Client: Me :: George (jlp_practicas_ise, C7)
- IPv4-Address: *
- IPv6-Address: (empty)
- Domain: pract_ise
- Harddisk Quota: 5000 MB
- Traffic Quota: 5000 MB
- CGI: ☐
- SSI: ☐
- Perl: ☒

Figura 37: Aquí podemos ver la creación de un sitio web asociado al usuario definido anteriormente. Podemos asignar, entre otras cosas, su dominio, su cuota de disco o su tráfico máximo.

15. Cuestión 15: a) Ejecute los ejemplos de find, grep. b) Escriba el script que haga uso de sed para cambiar la configuración de ssh y reiniciar el servicio. c) Muestre un ejemplo de uso para awk.

■ a)



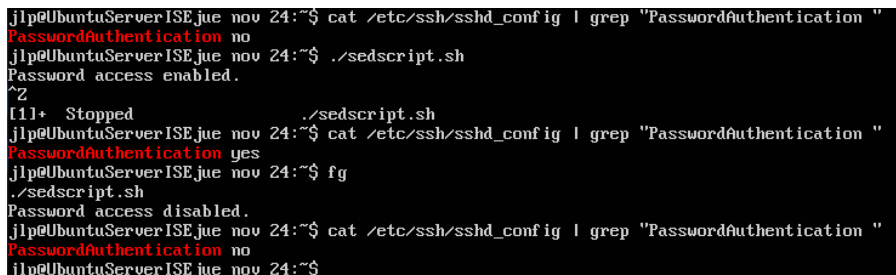
```
jlp@UbuntuServerISEmar nov 08:~$ ps -A | grep firefox
jlp      2237  1923  0 12:59 tty1    00:00:00 grep --color=auto firefox
jlp@UbuntuServerISEmar nov 08:~$ find /home/jlp/ -name '*.pdf' -exec cp {} ~/PDFs \;
jlp@UbuntuServerISEmar nov 08:~$ ls -l ~/PDFs
total 0
jlp@UbuntuServerISEmar nov 08:~$ _
```

Figura 38: El ejemplo de `grep` sólo devuelve el propio proceso encontrándose a sí mismo buscando firefox, ya que éste no está siquiera instalado (mucho menos ejecutándose). El ejemplo de `find` (ligeramente adaptado) no encuentra archivos PDF que copiar, por lo que la carpeta destino está vacía.

■ b) Asumiendo, por supuesto, que previamente no se permitiese el acceso ssh por contraseña:

```
#!/bin/bash/
sudo sed -i s/"PasswordAuthentication no"/"PasswordAuthentication yes"/
/etc/ssh/sshd_config
sudo service ssh restart
echo "Password access enabled."
sleep [tiempo decidido por el administrador]
sudo sed -i s/"PasswordAuthentication yes"/"PasswordAuthentication no"/
/etc/ssh/sshd_config
sudo service ssh restart
echo "Password access disabled."
```

En la siguiente captura de pantalla se observa cómo cambia correctamente el parámetro de configuración:




```
jlp@UbuntuServerISE.jue nov 24:~$ cat /etc/ssh/sshd_config | grep "PasswordAuthentication "
PasswordAuthentication no
jlp@UbuntuServerISE.jue nov 24:~$ ./sedscript.sh
Password access enabled.
^Z
[1]+  Stopped                  ./sedscript.sh
jlp@UbuntuServerISE.jue nov 24:~$ cat /etc/ssh/sshd_config | grep "PasswordAuthentication "
PasswordAuthentication yes
jlp@UbuntuServerISE.jue nov 24:~$ fg
./sedscript.sh
Password access disabled.
jlp@UbuntuServerISE.jue nov 24:~$ cat /etc/ssh/sshd_config | grep "PasswordAuthentication "
PasswordAuthentication no
jlp@UbuntuServerISE.jue nov 24:~$ _
```

Figura 39: Al principio, el acceso por contraseña está desactivado. Lanzo el script y detengo su proceso para comprobar que ha cambiado el valor. Finalmente, lo reanudo y veo que lo ha devuelto a su valor original al terminar.

- c) La línea

```
awk '$1 !~ /#/ && $2 ~ /yes/ {print $1 "\t" $2}' /etc/ssh/sshd_config
```

significa: “en este archivo, encuentra todas las líneas cuya primera columna contenga el carácter # y cuya segunda columna contenga el valor 'yes', e imprime ambas”. En esencia, devuelve todos los parámetros activos cuyo valor es 'yes'. El resultado de ejecutarlo en mi máquina virtual de Ubuntu Server se ve en la figura 40.



```
jlp@UbuntuServerISE:~$ awk '$1 !~ /#/ && $2 ~ /yes/ {print $1 "\t" $2}' /etc/ssh/sshd_config
UsePrivilegeSeparation yes
StrictModes yes
RSAAuthentication yes
PubkeyAuthentication yes
IgnoreRhosts yes
PasswordAuthentication yes
X11Forwarding yes
PrintLastLog yes
TCPKeepAlive yes
UsePAM yes
jlp@UbuntuServerISE:~$
```

Figura 40: Ejecución del ejemplo del comando awk.

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

El script en Python, ejecutado con `sudo python <script.py>`, es el siguiente:

```
import fileinput, sys, subprocess
inputfile = fileinput.FileInput("/etc/ssh/sshd_config", inplace = True)

for line in inputfile:

    sys.stdout.write(line.replace("PasswordAuthentication no",
                                  "PasswordAuthentication yes"))

inputfile.close()
subprocess.call(["sudo", "service", "ssh", "restart"])
print "Password access enabled"
subprocess.call(["sleep", "10"])
inputfile = fileinput.FileInput("/etc/ssh/sshd_config", inplace = True)

for line in inputfile:

    sys.stdout.write(line.replace("PasswordAuthentication yes",
                                  "PasswordAuthentication no"))

inputfile.close()
subprocess.call(["sudo", "service", "ssh", "restart"])
print "Password access disabled"
```

Para comprobar que este script también cambia con éxito el parámetro de acceso por contraseña, seguimos el mismo procedimiento que en la cuestión anterior:

```
jlp@UbuntuServerISE.jue nov 24:~$ cat /etc/ssh/sshd_config | grep "PasswordAuthentication "
```

PasswordAuthentication no

```
jlp@UbuntuServerISE.jue nov 24:~$ sudo python pythonscript.py
```

Password access enabled

^Z

[1]+ Stopped sudo python pythonscript.py

```
jlp@UbuntuServerISE.jue nov 24:~$ cat /etc/ssh/sshd_config | grep "PasswordAuthentication "
```

PasswordAuthentication yes

```
jlp@UbuntuServerISE.jue nov 24:~$ fg
```

sudo python pythonscript.py

Password access disabled

```
jlp@UbuntuServerISE.jue nov 24:~$ cat /etc/ssh/sshd_config | grep "PasswordAuthentication "
```

PasswordAuthentication no

```
jlp@UbuntuServerISE.jue nov 24:~$ _
```

Figura 41: Mostramos el valor de PasswordAuthentication antes de ejecutar el script, justo cuando se espera que lo haya cambiado, y después de terminar. Vemos que los resultados concuerdan.

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.

Vamos a parar un programa en ejecución, concretamente el bloc de notas (*notepad*). Para ello, primero lo abrimos:

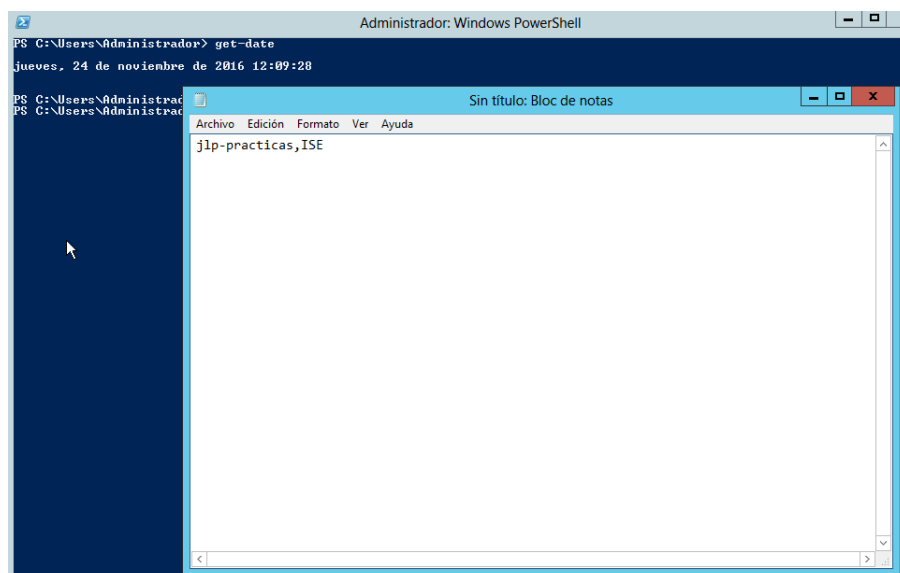


Figura 42: Abrimos *notepad* desde la consola.

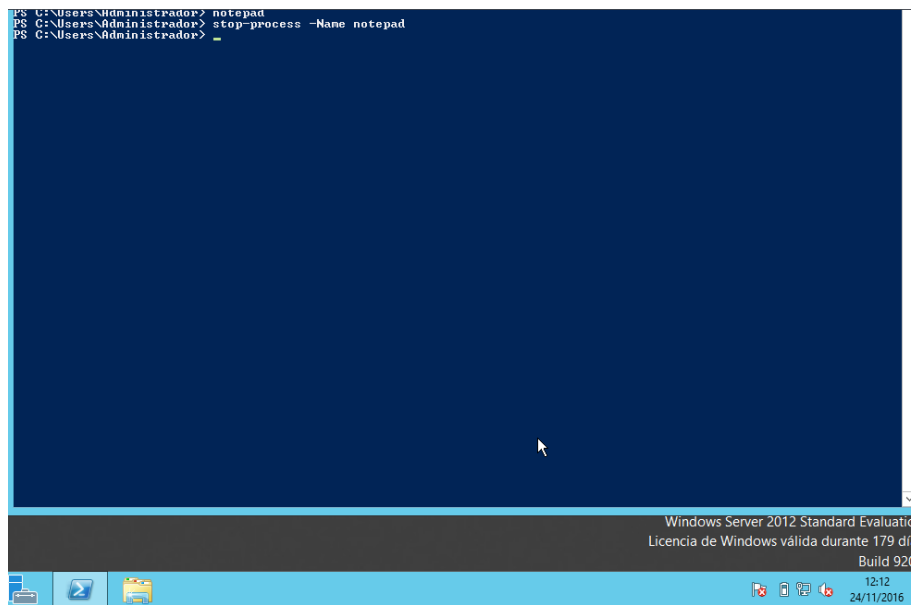


Figura 43: Usando la orden `stop-process -Name notepad` lo cerramos y vemos que ya no aparece en la barra de tareas.

Referencias

- [1] “Documentación oficial de RHEL. https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/s1-yum-useful-commands.html,” consultado el 30 de Octubre de 2016.
- [2] “Documentación oficial de Oracle. https://docs.oracle.com/cd/E37670_01/E37355/html/ol_proxy_config.html,” consultado el 13 de Noviembre de 2016.
- [3] “Documentación oficial de CentOS. <https://www.centos.org/docs/5/html/yum/sn-using-repositories.html>,” consultado el 1 de Noviembre de 2016.
- [4] “Página del manual de apt-get. <http://manpages.ubuntu.com/manpages/wily/man8/apt-get.8.html>,” consultado el 30 de Octubre de 2016.
- [5] “Página del manual de apt-cache. <http://manpages.ubuntu.com/manpages/wily/man8/apt-cache.8.html>,” consultado el 30 de Octubre de 2016.
- [6] “Ayuda de Ubuntu. <https://help.ubuntu.com/community/AptGet/Howto>,” consultado el 13 de Noviembre de 2016.
- [7] “Ayuda de Ubuntu. <https://help.ubuntu.com/community/Repositories/CommandLine>,” consultado el 1 de Noviembre de 2016.
- [8] P. García Teodoro, J. Díaz Verdejo, and J. M. López Soler, *Transmisión de Datos y Redes de Computadores*. Pearson, 2 ed., 2014.
- [9] “Documentación oficial de OpenBSD: SSH. <http://man.openbsd.org/ssh.1>,” consultado el 1 de Noviembre de 2016.
- [10] “Ayuda de Ubuntu: configurar SSH. <https://help.ubuntu.com/community/SSH/OpenSSH/Configuring>,” consultado el 6 de Noviembre de 2016.
- [11] “Documentación de systemctl. <https://www.freedesktop.org/software/systemd/man/systemctl.html>,” consultado el 7 de Noviembre de 2016.
- [12] “TheGeekStuff.com. <http://www.thegeekstuff.com/2014/12/patch-command-examples/>,” consultado el 22 de Noviembre de 2016.
- [13] “Página web oficial de Webmin. <http://www.webmin.com/deb.html>,” consultado el 19 de Noviembre de 2016.
- [14] “Manual de PHP. <http://php.net/manual/en/ini.core.php>,” consultado el 22 de Noviembre de 2016.