

Ingeniería de Servidores (2014-2015)
GRADO EN INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE GRANADA

Memoria Práctica 2

Iván Sevillano García

25 de agosto de 2016

Índice

1. Instalación de servicios y configuraciones.	3
1.1. Yum. Gestor de paquetes de CentOS.	3
1.2. Apt. Gestor de paquetes de Debian.	3
1.3. Cuestión opcional: Gestor de paquetes de OpenSUSE.	4
2. Instalación del servicio de acceso seguro(SSH)	5
2.1. Seguridad en ssh. Clave pública.	8
2.2. Sshd. El demonio de ssh.	10
2.3. Utilidades: Terminator y Screen.	12
2.4. Un poco de seguridad de acceso:fail2ban.	14
3. Instalación de un servidor Web básico.	15
3.1. Servidores de Ubuntu y CentOS.	15
3.2. Windows IIS	18
4. Manteniendo los servicios actualizados	26
5. Administración web.	27
5.1. Otros servidores.	36
6. Automatización de tareas.	40

1. Instalación de servicios y configuraciones.

En esta sección se responden a las preguntas referentes a la administración y configuración de software en los distintos sistemas operativos.

1.1. Yum. Gestor de paquetes de CentOS.

Esta aplicación es el gestor predeterminado de CentOS, Red Hat, Fedora y derivados.

- **Liste los argumentos de yum necesarios para instalar, buscar y eliminar paquetes.**

Según el manual de yum[1], para instalar un paquete del que ya sabemos el nombre podremos utilizar yum seguido del comando **install** tras el que pondremos el paquete a instalar. También existe la orden **groupinstall**, que instala un grupo de paquetes de forma conjunta. En realidad funciona igual que la orden **install** junto al nombre de todos los paquetes del grupo.

Para buscar, el comando **search** seguido de alguna referencia indirecta del paquete(parte del nombre en general) nos dará como resultado los paquetes que tengan algo que ver con la clave de búsqueda.

Para eliminar paquetes, yum tiene comandos **remove** y **erase** seguidos del paquete a eliminar, lo que produce redundancia de funciones, ya que hacen exactamente el mismo trabajo. Esto es un fallo de diseño y se deberá eliminar uno de estos términos en un futuro.

- **¿Que hay que hacer para que yum tenga acceso a internet en el aula de prácticas(puesto que esta utiliza un proxy por detrás)?**

Para contestar a esta pregunta se nos ofrece dos pistas: acceder al archivo de configuración de yum en `/etc/yum/` y que el proxy a utilizar es **stargate.ugr.es:3128**. Buscamos en el manual de configuración de yum[2] y nos dice que necesitamos incluir en el archivo **yum.conf** la URL del proxy. También hay opciones para acceder al proxy con un usuario y una contraseña(proxy_username y proxy_password).

1.2. Apt. Gestor de paquetes de Debian.

Esta aplicación es el gestor predeterminado de Debian y sus derivados, como por ejemplo ubuntu.

- **Indique el comando para buscar un paquete en un repositorio y el correspondiente para instalarlo.**

Según el manual de apt[3], para buscar un paquete del que tenemos parte del

nombre o similar, debemos utilizar el comando **apt search**, y para instalarlo, el comando **apt install**.

- **Indique qué ha modificado para que apt pueda acceder a los servidores de paquetes a través del proxy. ¿Cómo añadimos un nuevo repositorio?**
Según el manual proporcionado por apt.conf[4]:

"http::Proxy define el proxy predeterminado que utilizar para direcciones HTTP URI. Utiliza el formato estándar http://[[usuario]][:contraseña]@/máquina[:puerto]/. También se puede especificar un proxy por cada máquina usando la forma http::Proxy::<máquina> con la palabra especial DIRECT que significa que no se use ningún proxy. La variable de entorno http_proxy se usará en caso de no definir ninguna de las opciones anteriores. "

También según el mismo manual, la forma de añadir repositorios es añadirlos al archivo /etc/apt/sources.list con permisos de root. Se puede utilizar el comando edit-sources, con el mismo resultado que editarlos con un editor de texto directamente, ya que este comando solo llama a un editor a escoger[3].

1.3. Cuestión opcional: Gestor de paquetes de OpenSUSE.

Por cuestión de tiempo, en esta asignatura no utilizaremos OpenSUSE, pero si vamos a responder a la pregunta de cuál es el gestor de paquetes de este Sistema Operativo.

En la página oficial de OpenSUSE[5], se dice que éste tiene dos gestores, uno con un entorno gráfico y otro para consola. El primero es YaST y el segundo, Zypper.

2. Instalación del servicio de acceso seguro(SSH)

Para el acceso remoto desde una máquina a otra hay distintos protocolos. Entre ellos, destacan telnet y ssh. Sin embargo, para las conexiones que realizaremos, utilizaremos ssh. La instalación del servidor en Ubuntu se realiza con el comando:

```
sudo apt install openssh-server
```

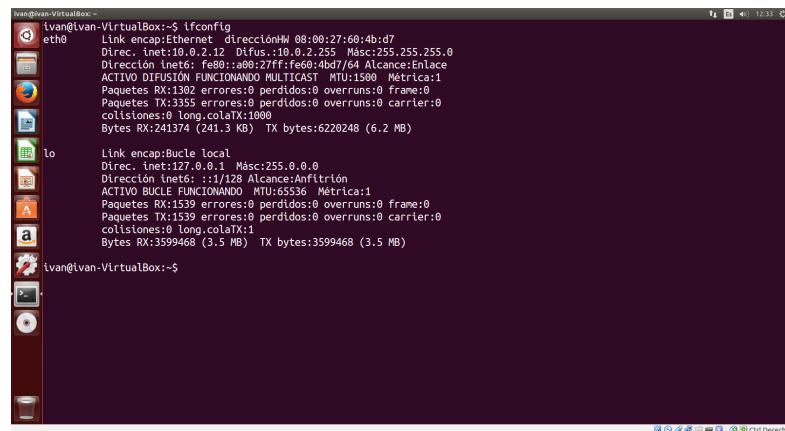
- ¿Cuál es la diferencia entre telnet y ssh?

Telnet[6] es un servicio básico de conexión entre máquinas, en el cuál sólo hay envío y recepción de mensajes. Ssh[7] (Secure shell), además de esto, los mensajes se envían de forma segura mediante distintos métodos, como pueden ser la encriptación de información. En definitiva, ssh está varios niveles por encima de telnet. Usar ssh en las prácticas es esencial ya que es mucho mas completo.

- ¿Para qué sirve la opción -X de ssh? 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?

La opción -X habilita la interfaz gráfica en las sesiones ssh[8]. Sin embargo, para que esto tenga sentido, la interfaz gráfica tiene que estar instalada en ambas máquinas, además de que el servidor tiene que aceptar en su archivo de configuración[12] la posibilidad de utilizar la misma. Este parametro es *ForwardX11Trusted*. Veamos ahora que ocurre:

Para probar esta opción, tenemos una máquina servidor y otra cliente con la interfaz gráfica de el editor de textos gedit instalada:



```
ivan@ivan-VirtualBox:~$ ifconfig
eth0      Link encap:Ethernet  direcciónHW 00:00:27:60:4b:d7
          Direc. inet:10.0.2.12  BROADCAST  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:fe60:4bd7/64  Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
          Paquetes RX:1302 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:3355 errores:0 perdidos:0 overruns:0 carrier:0
          collisions:0 long.colatX:1000
          Bytes RX:241374 (241.3 KB)  TX bytes:6220248 (6.2 MB)

lo        Link encap:Bucle Local
          Direc. inet:127.0.0.1  Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Ampliación
          ACTIVO BUCLE FUNCIONANDO MTU:65536  Métrica:1
          Paquetes RX:1539 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:1539 errores:0 perdidos:0 overruns:0 carrier:0
          collisions:0 long.colatX:1
          Bytes RX:3599468 (3.5 MB)  TX bytes:3599468 (3.5 MB)

ivan@ivan-VirtualBox:~$
```

Figura 2.1: Máquina cliente. IP:10.0.2.12

```

ivan@ivan-VirtualBox:~$ ifconfig
eth0      Link encap:Ethernet  direcciónHW 08:00:27:42:e0:68
          Direc. inet:10.0.2.11  Difus.:10.0.2.255  Másc:255.255.255.0
          Dirección inet6: fe80::a0:27ff:fe42:e068/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500  Métrica:1
          Paquetes RX:3300 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:1416 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colatx:1000
          Bytes RX:3672326 (3.6 MB)  TX bytes:312524 (312.5 KB)

lo        Link encap:Bucle local
          Direc. inet:127.0.0.1  Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Anfitrión
          ACTIVO BUCLE FUNCIONANDO MTU:65536  Métrica:1
          Paquetes RX:88 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:88 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colatx:1
          Bytes RX:7322 (7.3 KB)  TX bytes:7322 (7.3 KB)

ivan@ivan-VirtualBox:~$ 

```

Figura 2.2: Máquina servidor. IP:10.0.2.11

```

ivan@ivan-VirtualBox:~$ ssh 10.0.2.11 -X
The authenticity of host '10.0.2.11 (10.0.2.11)' can't be established.
ECDSA key fingerprint is 73:bd:2d:2b:46:9d:25:4a:c2:3c:64:9a:de:77:ff:97.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.11' (ECDSA) to the list of known hosts.
[3-ivan@10.0.2.11's password:
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

 * Documentation: https://help.ubuntu.com/
[3- New release '16.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

[3- 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.
[3- Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

[3- ivan@ivan-VirtualBox:~$ 

```

Figura 2.3: Conexión realizada con la opción -X

La conexión se lleva a cabo con normalidad. Probemos ahora a utilizar gedit:

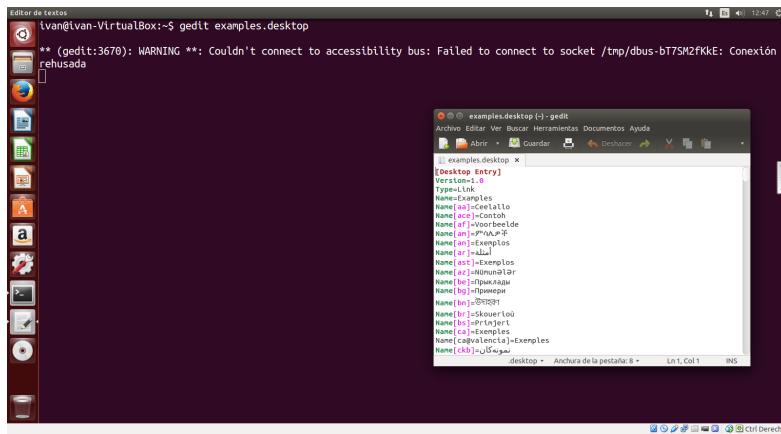


Figura 2.4: Gedit con la opción -X

Como vemos, el uso de gedit se lleva a cabo con normalidad. ¿Pero que ocurriría si utilizamos un entorno gráfico no instalado en el servidor? Cambiamos ahora a un servidor sin entorno gráfico:

```
ivan@ubuntu:~$ ifconfig
enp0s3    Link encap:Ethernet  direcciónHW 08:00:27:b1:70:00
          Direc. inet: 10.0.2.9  Difus.: 10.0.2.255  Másc: 255.255.255.0
          Dirección inet6: fe80::a00:27ff:feb1:7000/64 Alcance: Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU: 1500 Métrica: 1
          Paquetes RX: 2 errores: 0 perdidos: 0 overruns: 0 frame: 0
          Paquetes TX: 10 errores: 0 perdidos: 0 overruns: 0 carrier: 0
          colisiones: 0 long.colaTX: 1000
          Bytes RX: 1180 (1.1 KB) TX bytes: 1332 (1.3 KB)

lo        Link encap:Bucle local
          Direc. inet: 127.0.0.1  Másc: 255.0.0.0
          Dirección inet6: ::1/128 Alcance: Anfitrión
          ACTIVO BUCLE FUNCIONANDO MTU: 65536 Métrica: 1
          Paquetes RX: 512 errores: 0 perdidos: 0 overruns: 0 frame: 0
          Paquetes TX: 512 errores: 0 perdidos: 0 overruns: 0 carrier: 0
          colisiones: 0 long.colaTX: 1
          Bytes RX: 39760 (39.7 KB) TX bytes: 39760 (39.7 KB)

ivan@ubuntu:~$ _
```

Figura 2.5: Máquina servidor sin entorno gráfico. IP:10.0.2.9

La conexión la llevamos a cabo normalmente, como se ve a continuación:

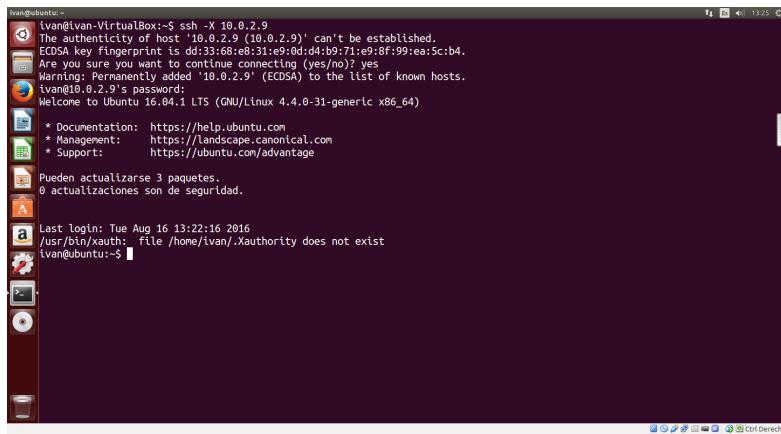


Figura 2.6: Conexión con la opción -X en un servidor sin entorno gráfico.

Ahora intentamos usar gedit:



Figura 2.7: Gedit no instalado en el servidor.

Como vemos, para poder utilizar gedit, este tiene que estar instalado en el servidor. Además, este también tiene que estar instalado en el cliente.

2.1. Seguridad en ssh. Clave pública.

Tras instalar el servicio openssh-server en la máquina, probaremos ahora a conectarnos remotamente. Sin embargo, como vemos a continuación, la máquina anfitrión nos pide una contraseña (la contraseña de la cuenta):

```

ivan@ubuntu:~$ ssh ivan@10.0.2.10
ivan@10.0.2.10's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

Pueden actualizarse 3 paquetes.
0 actualizaciones son de seguridad.

Last login: Sun Jul 31 13:34:18 2016 from 10.0.2.9
ivan@ubuntu:~$
```

Figura 2.8: Conexión ssh hecha con normalidad. Nos pide password del usuario en el servidor.

Sin embargo, esta conexión puede ser insegura. Para hacerla más segura, vamos a utilizar el algoritmo RSA. Para ello generaremos una clave pública en el cliente que enviaremos al servidor. Con esto conseguimos que cada vez que intentemos conectarnos al servidor, no nos pregunte la contraseña de la cuenta del servidor. Esta es la secuencia de operaciones:

```

ivan@ubuntu:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ivan/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ivan/.ssh/id_rsa.
Your public key has been saved in /home/ivan/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:0xLs15/nVA+lo/+OBNg0/y060pCEUr iUSBsUWKMzNE8 ivan@ubuntu
The key's randomart image is:
+---[RSA 2048]---+
| +==E. |
| ..*B.. |
| +=.. .|
| o..o. o o |
| .o.S.o o = |
| . .oo + + + |
| o =. = . .|
| o o oo+.o |
| +oo+++|
+---[SHA256]---+
ivan@ubuntu:~$
```

Figura 2.9: Comando ssh-keygen generando clave RSA

En esta imagen se muestra cuales son los pasos a seguir para generar con ssh-keygen la clave. Según el manual de ssh-keygen[9], hay muchas opciones, pero la que nos interesa a nosotros es la opción *-t*, que especifica el protocolo que usaremos. Tras esto, keygen nos pregunta que si queremos ponerle una clave, que no será la clave pública, si no una clave interna de paso del ordenador que no pasará a la red. Nos pregunta también por donde guardar la clave. Dejaremos que la guarde por defecto en */home/ivan/.ssh/id_rsa.pub*.

El siguiente paso será enviar la clave pública al servidor para que nos identifique:

```
ivan@ubuntu:~$ ssh-copy-id -i /home/ivan/.ssh/id_rsa.pub ivan@10.0.2.10
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: '/home/ivan/.ssh/id_rsa.pub'
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter 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
ivan@10.0.2.10's password:
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'ivan@10.0.2.10'"
and check to make sure that only the key(s) you wanted were added.

ivan@ubuntu:~$
```

Figura 2.10: Envío de clave pública mediante ssh-copy-id

Según el manual de ssh-copy-id[10], la forma de uso es indicando la clave pública, con la opción -i, y el servidor ssh. Las últimas líneas de información nos dicen que ya está configurado para poder acceder a la cuenta con la clave pública. Probamos ahora a conectarnos.

```
ivan@ubuntu:~$ ssh ivan@10.0.2.10
Enter passphrase for key '/home/ivan/.ssh/id_rsa':
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

Pueden actualizarse 3 paquetes.
0 actualizaciones son de seguridad.

Last login: Sun Jul 31 13:36:31 2016 from 10.0.2.9
ivan@ubuntu:~$
```

Figura 2.11: Conexión usando la clave pública como elemento de identificación.

Como puede verse, no es necesario utilizar la contraseña de la cuenta en el servidor (La contraseña que nos pide es la generada antes internamente en el ordenador cliente para desbloquear la clave pública).

2.2. Sshd. El demonio de ssh.

- ¿Qué archivo es el que contiene la configuración de sshd? ¿Qué parámetro hay que modificar para evitar que el usuario root acceda? Cambie el puerto por defecto y compruebe que puede acceder.

Según el manual de sshd[11], el directorio por defecto es */etc/sshd/sshd_config*, y cuyas opciones de configuración se detallan en [12].

Para evitar que el usuario root acceda hay que modificar el parámetro *PermitRootLogin*. Según el mismo manual, para inhabilitar al usuario root hay que cambiarlo a "no".

Para cambiar el puerto de escucha hay que modificar el comando *Port* al puerto que se desee escuchar.

Para comprobar la configuración, modificaremos el puerto de escucha al 22222 y modificaremos el parámetro dicho anteriormente:

```
ivan@ubuntu:~$ ssh ivan@10.0.2.9
ssh: connect to host 10.0.2.9 port 22: Connection refused
ivan@ubuntu:~$ ssh root@10.0.2.9
ssh: connect to host 10.0.2.9 port 22: Connection refused
ivan@ubuntu:~$ ssh ivan@10.0.2.9 -p 22222
Enter passphrase for key '/home/ivan/.ssh/id_rsa':
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

Pueden actualizarse 3 paquetes.
0 actualizaciones son de seguridad.

Last login: Mon Aug 15 11:30:54 2016
ivan@ubuntu:~$ exit
logout
Connection to 10.0.2.9 closed.
ivan@ubuntu:~$ ssh root@10.0.2.9 -p 22222
root@10.0.2.9's password:
Permission denied, please try again.
root@10.0.2.9's password:
Permission denied, please try again.
root@10.0.2.9's password:
Permission denied (publickey,password).
ivan@ubuntu:~$ _
```

Figura 2.12: Modificación de los parámetros Port y PermitRootLogin.

En esta imagen se realizan cuatro intentos de conexión ssh. En los dos primeros se pide una conexión al puerto 22 por defecto, en este caso deshabilitado. La tercera conexión se realiza con el usuario ivan y se manda al puerto 22222, por lo que nos pide la contraseña interna para enviar la clave pública y nos deja conectarnos. Muy distinto es el caso en el que intentamos acceder como root, ya que aunque pongamos correcta la contraseña, no nos deja acceder.

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

Es necesario reiniciar el servicio, ya que el archivo que modificamos es un archivo de texto que tiene que ser leido por el programa antes de inicializarse. En Ubuntu, para reiniciar un servicio se utiliza *ssudo service <servicio>start/restart*. En CentOS, *sudo systemctl restart <servicio>*.

2.3. Utilidades: Terminator y Screen.

Estas dos herramientas nos sirven para abrir varios entornos de trabajo en una misma terminal. Terminator[13] tiene distintas utilidades, como escribir en todas las terminales a la vez o asignarle distintos trabajos a las terminales dependiendo del número de ventana que sea. Para instalarlo hay que instalar el paquete terminator:

```
sudo apt install terminator
```

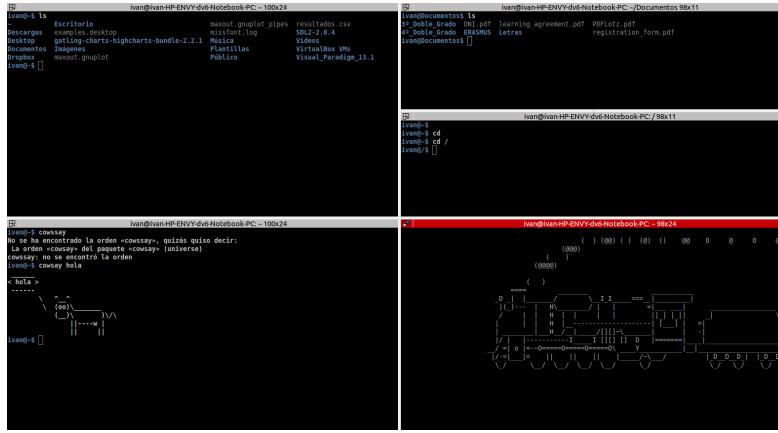


Figura 2.13: Ejemplo de división de ventanas con terminator.

La otra herramienta que se utiliza es Screen[14]. También tiene un gestor de ventanas de la terminal. Sin embargo, es más antiguo y no tiene una interfaz tan moderna como terminator. Sin embargo, puede ser usada en un servidor sin esta interfaz, ya que no le hace falta interfaz gráfica. Realizaremos ahora la conexión ssh desde screen. La dejaremos abierta para luego retomarla.

Al acceder a screen, primero nos aparece la siguiente salida:

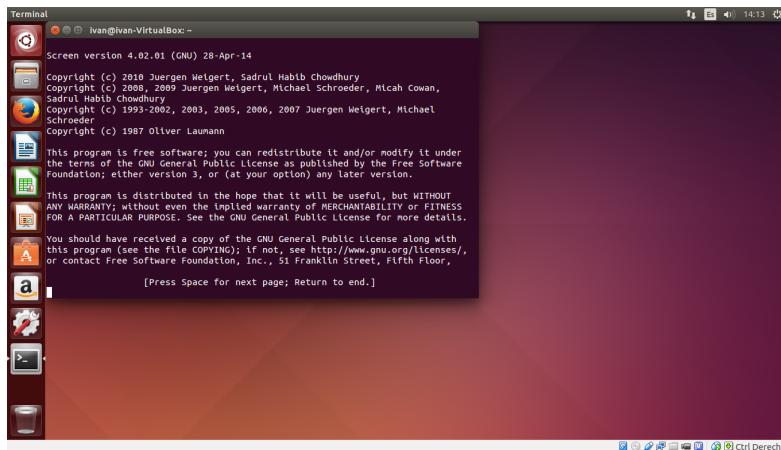


Figura 2.14: Salida al ejecutar screen

tras la cual se nos abre una ventana idéntica a la de la terminal, aunque veremos que tiene mucha más funcionalidad. Sin embargo, para ver nombrar a nuestra sesión screen y ver el proceso de forma más clara, ejecutaremos screen -S <nombre>(en nuestro caso, ivanscreen) para abrir la sesión de screen. Nos conectaremos por ssh ahora a la máquina servidora:

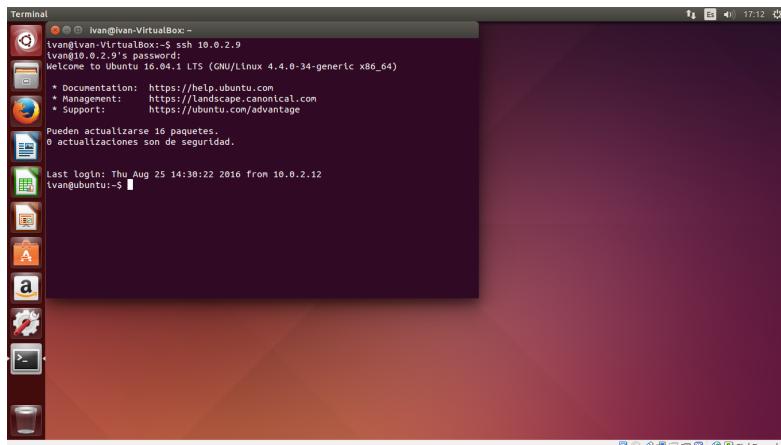
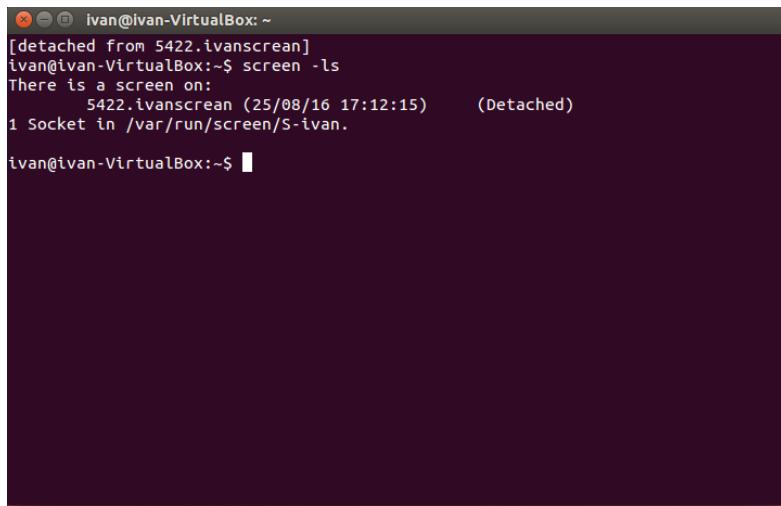


Figura 2.15: Conexión ssh con screen

y ahora vamos a dejarla para otro momento. Para ello, utilizaremos el comando de screen Ctrl-a d. Vemos entonces que nuestra ventana de screen ha sido *detached*. Veamos las sesiones de screen que existen en este momento(comando screen -ls):



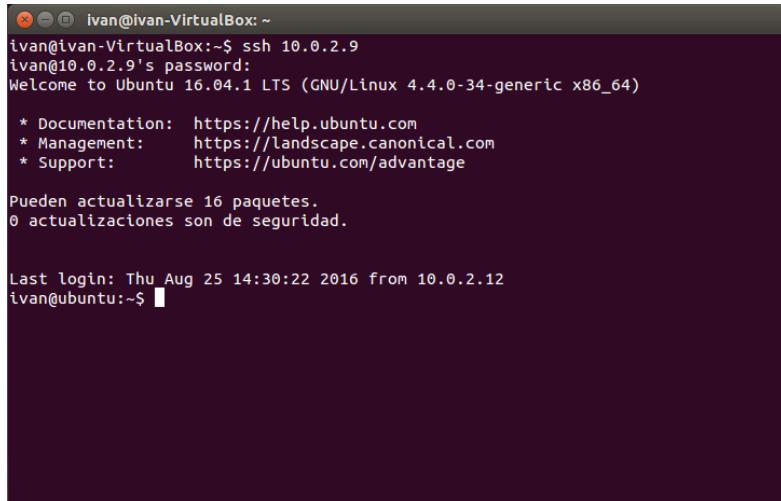
```
ivan@ivan-VirtualBox:~ [detached from 5422.ivanscreen]
ivan@ivan-VirtualBox:~$ screen -ls
There is a screen on:
      5422.ivanscreen (25/08/16 17:12:15)  (Detached)
1 Socket in /var/run/screen/S-ivan.

ivan@ivan-VirtualBox:~$
```

Figura 2.16: Mensaje de screen detached y muestra de las sesiones con screen que existen.

Cabe mencionar que hemos salido de screen ya que no funcionan los atajos de teclado. Ahora, para reanudar la sesión, utilizamos el comando screen -r con el nombre de la sesión:

```
screen -r ivanscreen
```



```
ivan@ivan-VirtualBox:~ $ ssh 10.0.2.9
ivan@10.0.2.9's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-34-generic x86_64)

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

Pueden actualizarse 16 paquetes.
0 actualizaciones son de seguridad.

Last login: Thu Aug 25 14:30:22 2016 from 10.0.2.12
ivan@ubuntu:~$
```

Figura 2.17: Sesión retomada de screen con ssh abierto.

2.4. Un poco de seguridad de acceso:fail2ban.

- Instale el servicio y pruebe su funcionamiento.

La herramienta fail2ban[15] sirve para detectar los accesos repetitivos y erroneos. Esto es, si una máquina accede de forma errónea un número determinado de veces, fail2ban no deja que esta misma máquina vuelva a acceder. Prevenimos así el intento de encontrar contraseña por prueba y error.

3. Instalación de un servidor Web básico.

Esta sección está dedicada a la instalación de un servidor web básico en distintos sistemas operativos. En concreto, instalaremos los servicios LAMP(Apache,mysql o mariadb,PHP o python).

3.1. Servidores de Ubuntu y CentOS.

- 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).

En Ubuntu, basta con ejecutar la siguiente orden en la consola:

```
sudo apt install lamp-server^
```

tras lo cual solo tendremos que configurar la contraseña de mysql, como se muestra en la siguiente imagen:

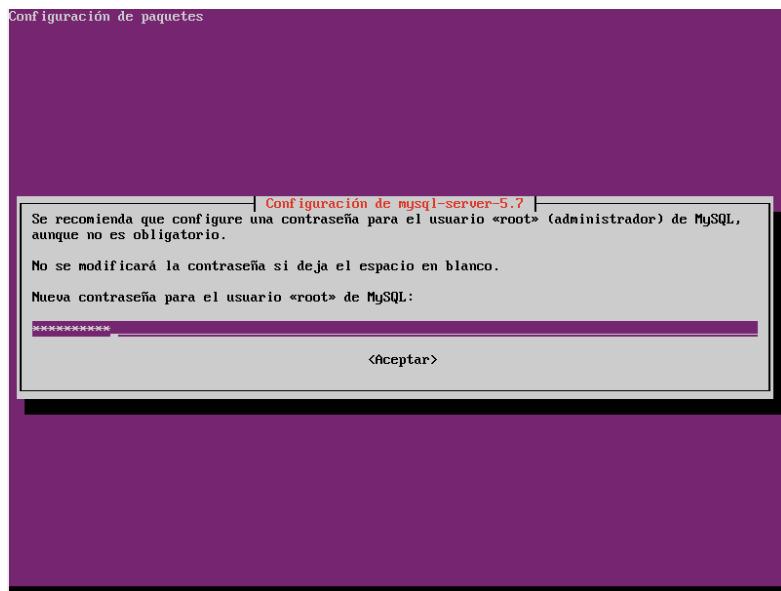


Figura 3.1: Configurar contraseña en la instalación de mysql en Ubuntu.

Aquí vemos, desde otra máquina, que el servidor web está corriendo:

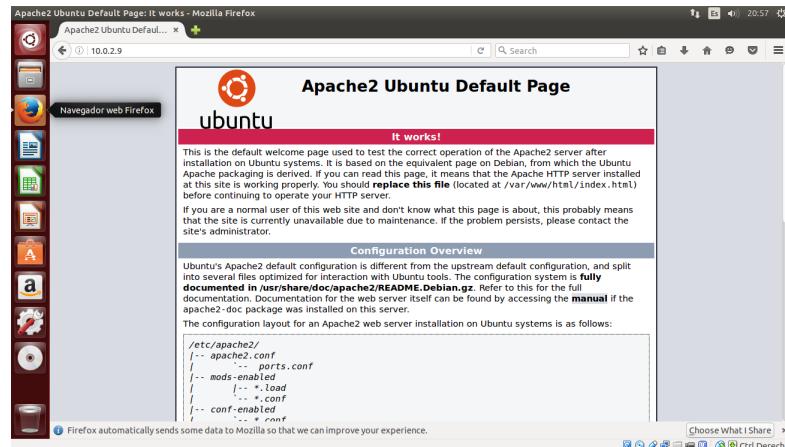


Figura 3.2: Acceso a la web del servidor de Ubuntu localizado en la IP 10.0.2.9 desde una máquina con entorno gráfico.

En CentOS, no existe un paquete que englobe todo lo necesario para el servidor lamp, sin embargo si se puede descargar parte a parte. Para ello, los paquetes que instalaremos serán httpd, mariadb, php y php-mysql. Con el siguiente comando, descargaremos todo:

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

A modo de confirmación, aquí dejo la salida de la terminal al introducir el comando:

```
[root@localhost ~]# sudo yum install mariadb
[sudo] password for ivan:
Complementos cargados:fastestmirror
http://dl.fedoraproject.org/mirrorlist?repo=epel&arch=x86_64
extras                                         : 3.6 kB  00:00
updates                                         : 3.4 kB  00:00
updates                                         : 3.4 kB  00:00
[Cañete]/>x86_64/primary_d 16x {-->          } 280 kB/s  1 1.1 MB  00:30 ETH
Saliendo por cancelación del usuario
[root@localhost ~]# sudo yum install httpd mariadb php php-mysql
Complementos cargados:fastestmirror
Loading mirror speeds from cached hostfile
 * base: ftp.up.pt
 * epel: mirrors.kiwi.fcpt
 * updates: ftp.up.pt
updates/7/x86_64/primary_d 47x {=====-->          } 213 kB/s  3.4 MB  00:17 ETH
```

Figura 3.3: Instalación del servidor lamp en CentOS.

Al entrar desde la misma máquina con el entorno gráfico en la dirección IP del servidor CentOS, este es el resultado:

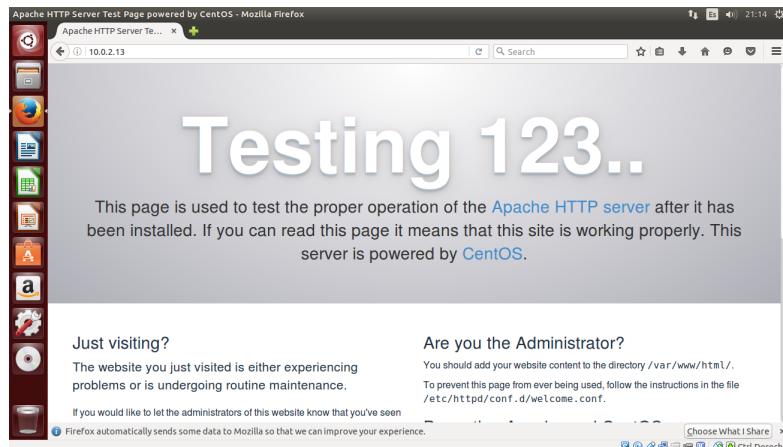


Figura 3.4: Acceso a la máquina que corre con CentOS localizada en la IP 10.0.2.13.

Figura 3.5:

En Ubuntu, para que el servidor esté completamente en orden, no hacía falta mas que la descarga del paquete. Sin embargo, en CentOS hay que poner en orden algunos puntos. Para ello tenemos que habilitar el servicio httpd, abrir los puertos del mismo y reiniciar el servicio. Por orden, estos son los comandos que hay que ejecutar:

Habilitar el servicio web:

```
sudo systemctl enable httpd.service
```

Abrir el puerto del servidor web:

```
sudo firewall-cmd --permanent --add-service http
```

Reiniciar el firewall(ya que hemos modificado los puertos que el demonio de firewall debe escuchar):

```
sudo systemctl restart firewalld.service
```

- Enumere otros servidores web y las páginas de sus proyectos (mínimo 3 sin considerar Apache, IIS ni nginx)

Para contestar esta pregunta, se ha recurrido a la página oficial de w3techs[16], la cual hace mediciones del número de servidores que utilizan un tipo u otro de servidor web entre otros datos(como el sistema operativo que se utiliza). Entre los servidores web que listan, se encuentran los siguientes:

- El proyecto LiteSpeed Web Server[17] es un proyecto de la misma compañía que vende compatibilidad con apache, mas seguridad y menos coste.
- Apache Tomcat[18] es un proyecto que desarrolla mucho software de Java. Es un proyecto de código abierto y se jacta de ser un proyecto colaborativo.
- Por último, Node.js[19] tiene un servidor web con alrededor del 0.2 por ciento de los sitios webs de la red. Cabe destacar que si estudiamos que sitios webs lo usan, estos tienen la característica de tener mucho tráfico.

3.2. Windows IIS

- Compruebe que el servicio está funcionando accediendo a la MV a través de la anfitriona.

Para instalar los servicios necesarios en Windows, vamos a abrir el panel de Administrador de servicios:

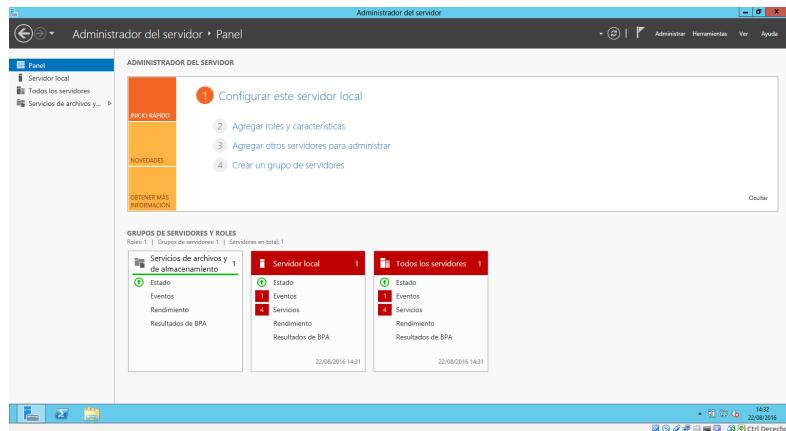


Figura 3.6: Panel de administrador de servicios de Windows.

Clicamos en Agregar roles y características .

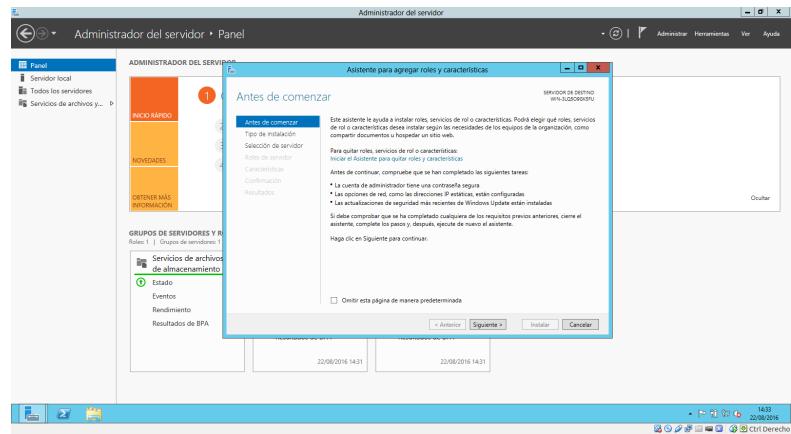


Figura 3.7: Ventana inicial del asistente para agregar roles.

Entonces, tras pinchar Siguiente unas cuantas veces, seleccionamos el servicio IIS dicho anteriormente en el paso de Roles de servidor:

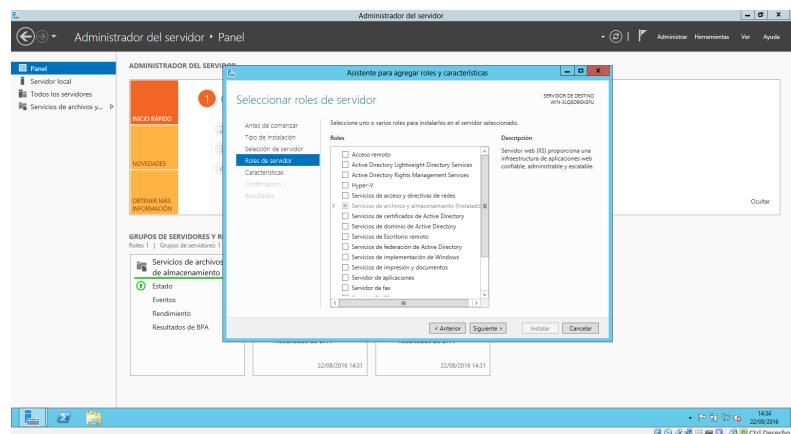


Figura 3.8: Lista de roles posibles en Windows. Buscamos el servicio IIS.

Los servicios que debemos instalar tras esto, que son los servicios de “Estado y Diagnóstico”: “Herramientas de registro” y “Seguimiento”. También instalaremos “Scripts y herramientas de administración de IIS” y el “Servicio de administración”, que se encuentran en otra lista posterior:

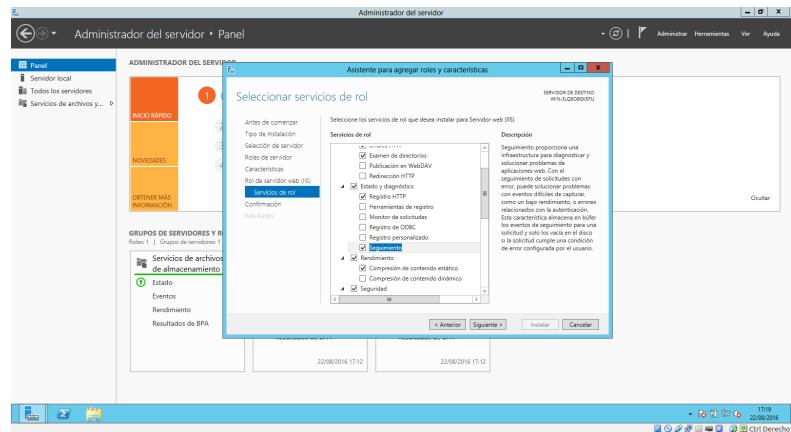


Figura 3.9: Servicios que ofrece el rol de IIS

Como último requisito para las prácticas, instalaremos FTP también:

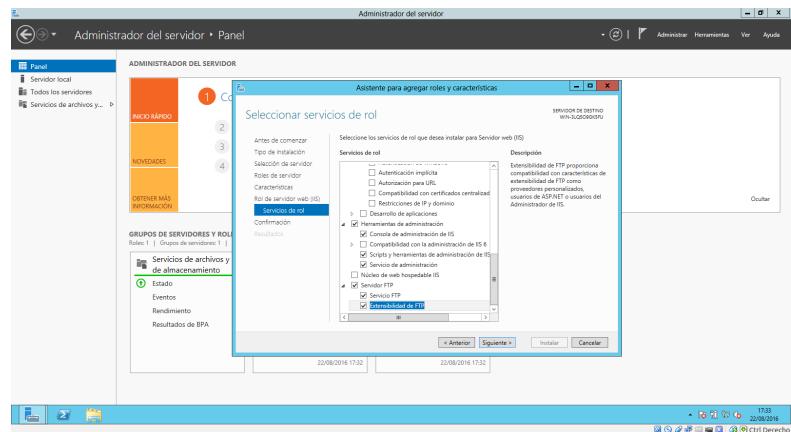


Figura 3.10: Servicio de rol FTP en Windows

Por último, se pide confirmación para la habilitación de servicios:

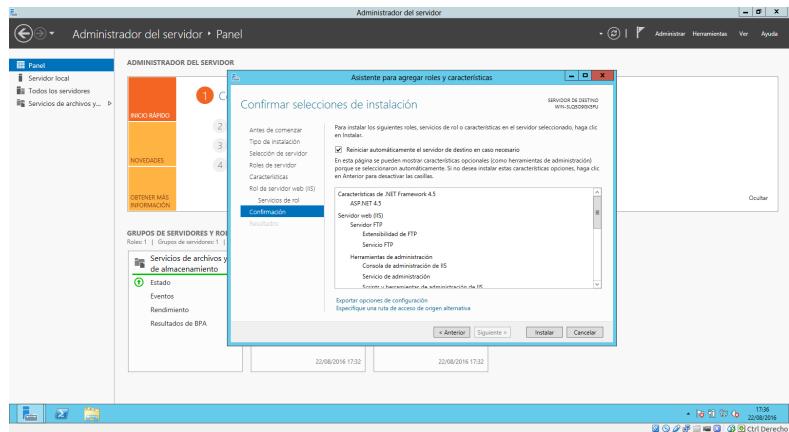


Figura 3.11: Confirmación para la instalación de los servicios.

Ya tenemos el servidor apunto. Para comprobar que realmente funciona la página, accedemos desde otra máquina con entorno gráfico:



Figura 3.12: Pagina web con IIS como servidor web.

Para habilitar ftp en Windows Server 2012 hay que seguir los siguientes pasos, sacados del servicio técnico de Microsoft[20]:

- **Crear un usuario en el servidor Windows.**

Para ello, en Panel de control->Cuentas de usuario->Administrar cuentas, clicamos en Agregar una cuenta de usuario. Rellenamos los campos de usuario, contraseña e indicio y ya tenemos creado nuestro usuario:

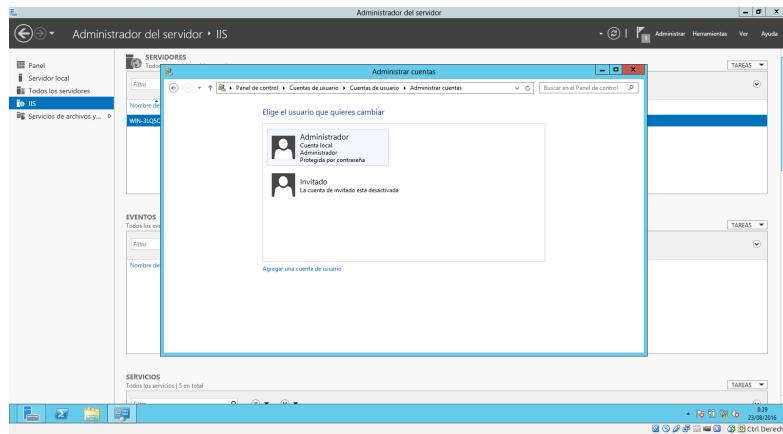


Figura 3.13: Ventana en la que se encuentra el asistente para crear cuentas de usuario.

- **Crear un sitio FTP al cual acceder.**

Para ello, en el panel de administrador de roles, en la sección IIS, utilizamos la herramienta de administrador de IIS. Tras esto, expandimos el nodo conexiones y en el nodo sitios, aplicamos la acción Agregar sitio FTP. Un asistente de nos irá guiando en la creación del sitio FTP:

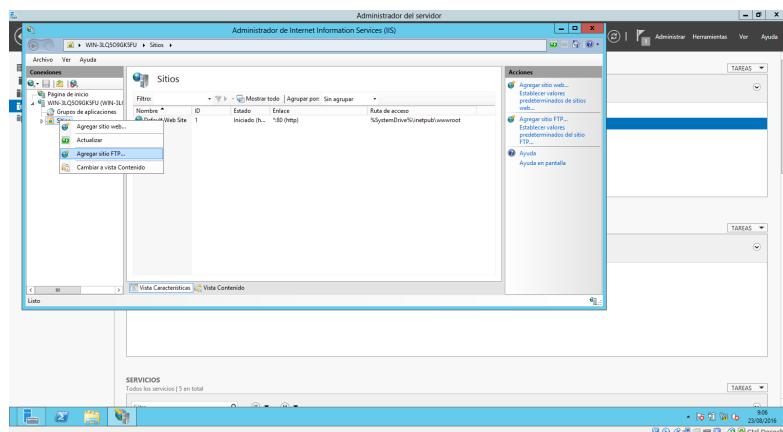


Figura 3.14: Nodos a expandir para llamar al asistente de creación de sitio FTP.

Tendremos que ponerle nombre a nuestro sitio y una ruta a la que acceder. En nuestro caso lo llamaremos ivan-ftp y lo localizaremos en la carpeta FTP en el disco C :

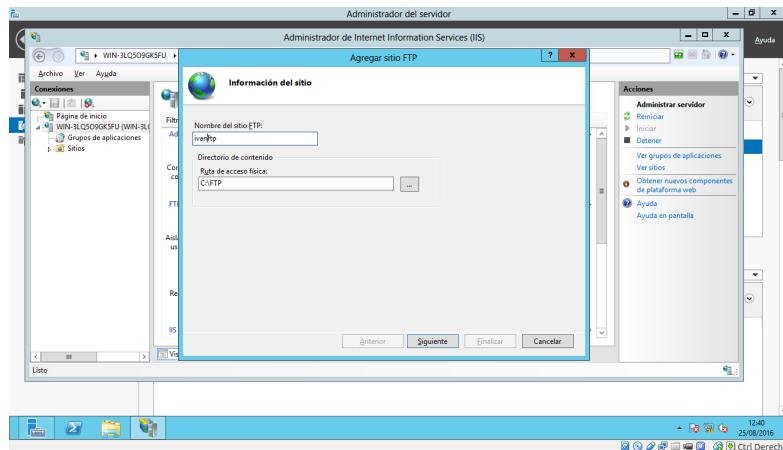


Figura 3.15: Nombre y localización del sitio FTP.

En la configuración de enlace y SSL, pinchamos en Sin SSL para no requerir el cifrado SSL. Los demás parámetros por defecto no será necesario modificarlos:

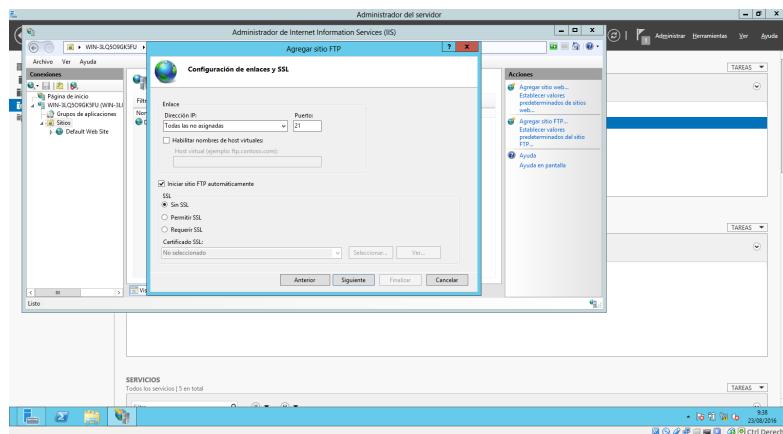


Figura 3.16: Ventana de configuración de enlace y SSL.

Por último, incluimos al usuario ivan para que pueda leer y escribir los archivos del sitio FTP:

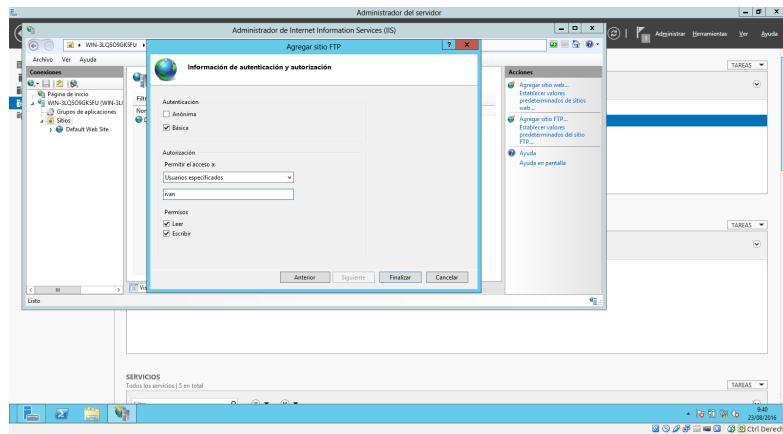


Figura 3.17: Añadir al usuario como lector y escritor del sitio FTP.

Así queda la configuración de sitios Web:

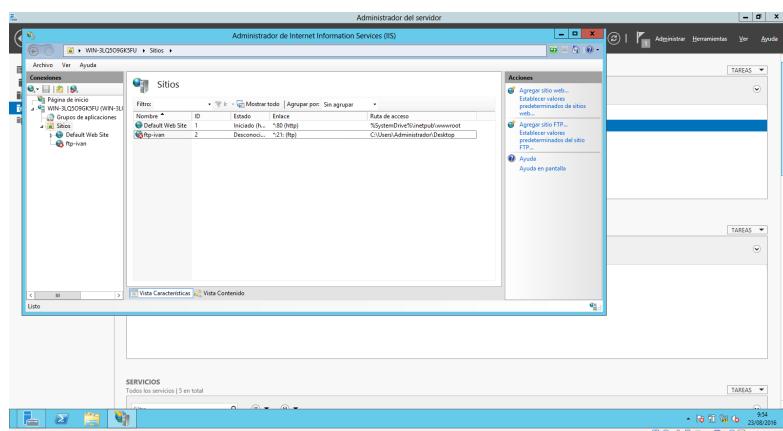


Figura 3.18: Configuración de sitios web tras la inclusión del sitio FTP.

- **Modificar el firewall de Windows para que la conexión ftp pueda llevarse a cabo.**

La herramienta firewall se encuentra en Herramientas Administrativas. Solo hay que añadir una regla de entrada que acepte las peticiones al puerto 21, el puerto por defecto de FTP. Para ello, abrimos el FireWall de Windows y creamos una regla de entrada en la que se acepte la conexión. La conexión se llevará a cabo a través del puerto 21. El asistente tiene las siguientes ventanas, explicadas en su descripción:

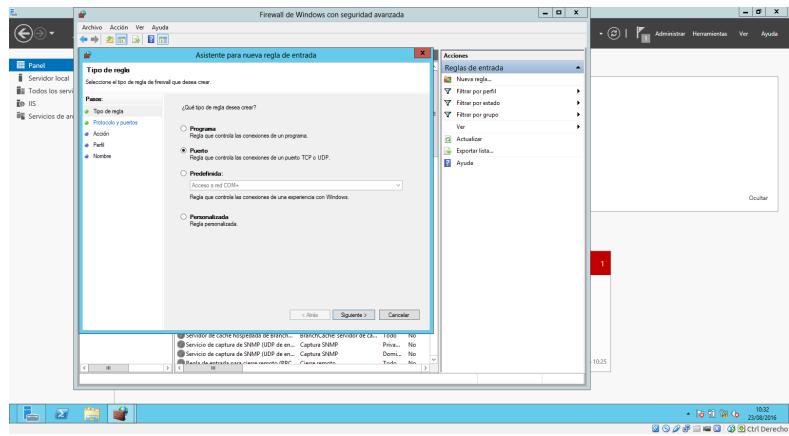


Figura 3.19: Tipo de regla que vamos a crear. Se elije la configuración puerto, ya que para que se escuche las peticiones ftp, vanta que se abra el puerto 21.

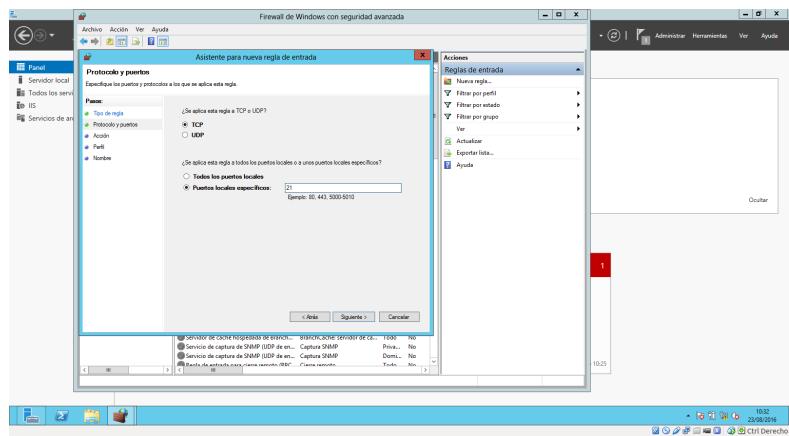


Figura 3.20: Selección del protocolo a seguir y el puerto que se va a abrir. Escogemos TCP y el puerto 21.

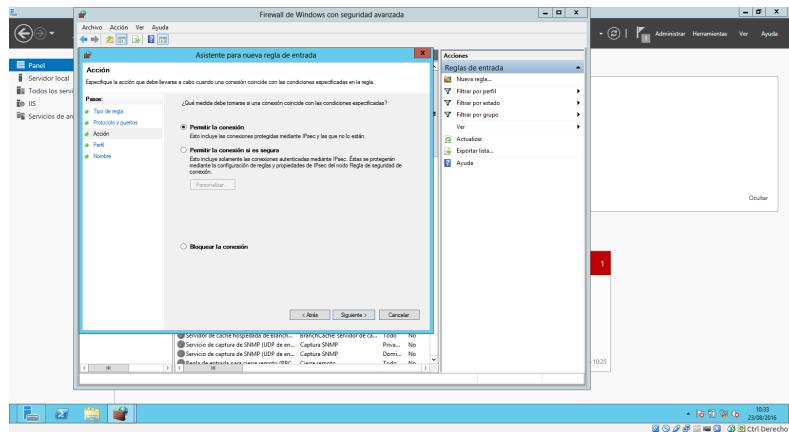


Figura 3.21: Acción a realizar. Marcamos la opción de permitir la conexión ftp.

Tras estas ventanas, tenemos que ponerle nombre y añadir una descripción a la regla. No creo necesario incluir captura de pantalla de esta parte. Ya tenemos la regla de entrada configurada. Solo queda comprobar que funciona. Realizaremos ahora la conexión ftp desde otra máquina:

4. Manteniendo los servicios actualizados

- **Ejemplo de uso de el comando patch.** Muestre un ejemplo de uso del comando patch, por ejemplo en el link dado en la referencia [21].

Existe un comando, diff[22], el cual extrae la diferencia entre dos archivos y lo codifica en un archivo .diff. El comando patch[21] sirve para aplicar esos cambios a uno de los archivos para cambiarlo al otro. Esto es muy útil, ya que si se ha distribuido una versión de un código compilado con un error no detectado, no vamos a volver a distribuir de nuevo el ejecutable entero, si no que obtendremos con diff la diferencia entre el código original con errores y el nuevo código y pediremos que se aplique este parche al código compilado.

En el link se explica como instalar VMWare. En el mismo se contemplan varios errores, entre ellos uno en el que se le puede aplicar un parche descargable. Esto se soluciona al aplicar el comando patch. Mostramos aquí el script que ofrecen para solucionarlo:

```
// Descarga del parche en un directorio temporal:  
$ curl http://pastie.org/pastes/8672356/download -o /tmp/vmware-netfilter.patch
```

```

// Cambio del directorio y descompresión del archivo a modificar.
$ cd /usr/lib/vmware/modules/source
# tar -xvf vmnet.tar

// Aplicación del parche.
# patch -p0 -i /tmp/vmware-netfilter.patch

// Compresión del archivo modificado
# tar -cf vmnet.tar vmnet-only

// Instalar lo que quedaba de VMWare.
# rm -r vmnet-only
# vmware-modconfig --console --install-all

```

5. Administración web.

- Realice la instalación de webadmin y pruebe a modificar algún parámetro de algún servicio. Muestre las capturas de pantalla pertinentes así como el proceso de instalación.

Para instalar webadmin vamos a seguir los pasos detallados en la página de webadmin para la instalación de paquetes[?]:

Primero debemos modificar el archivo */etc/apt/sources.list* y añadir la siguiente linea:

```
deb http://download.webmin.com/download/repository sarge contrib
```

```

ivan@ivan-VirtualBox: ~
GNU nano 2.2.6      Archivo: /etc/apt/sources.list

# deb-src http://archive.canonical.com/ubuntu trusty partner
## This software is not part of Ubuntu, but is offered by third-party
## developers who want to ship their latest software.
deb http://extras.ubuntu.com/ubuntu trusty main
deb-src http://extras.ubuntu.com/ubuntu trusty main

## Añadido por el usuario Ivan

deb http://download.webmin.com/download/repository sarge contrib
[ 60 líneas escritas ]
^G Ver ayuda ^O Guardar   ^R Leer fich.^Y Pág. ant. ^K Cortar Tex^C Posición
^X Salir    ^J Justificar^W Buscar    ^V Pág. sig. ^U PegarTxt ^T Ortografía

```

Figura 5.1: Repositorio añadido a apt.

Tras lo cual debemos instalar la clave para poder descargar el paquete:

```

cd /root
wget http://www.webmin.com/jcameron-key.asc
apt-key add jcameron-key.asc

```

```

ivan@ivan-VirtualBox: ~
ivan@ivan-VirtualBox:-$ cd /root
bash: cd: /root: Permiso denegado
ivan@ivan-VirtualBox:-$ wget http://www.webmin.com/jcameron-key.asc
--2016-08-23 16:03:09-- http://www.webmin.com/jcameron-key.asc
Resolviendo www.webmin.com (www.webmin.com)... 216.34.181.97
Conectando con www.webmin.com (www.webmin.com)[216.34.181.97]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 1320 (1,3K) [text/plain]
Grabando a: "jcameron-key.asc"

100%[=====] 1.320      ---K/s  en 0,004s
2016-08-23 16:03:11 (303 KB/s) - "jcameron-key.asc" guardado [1320/1320]

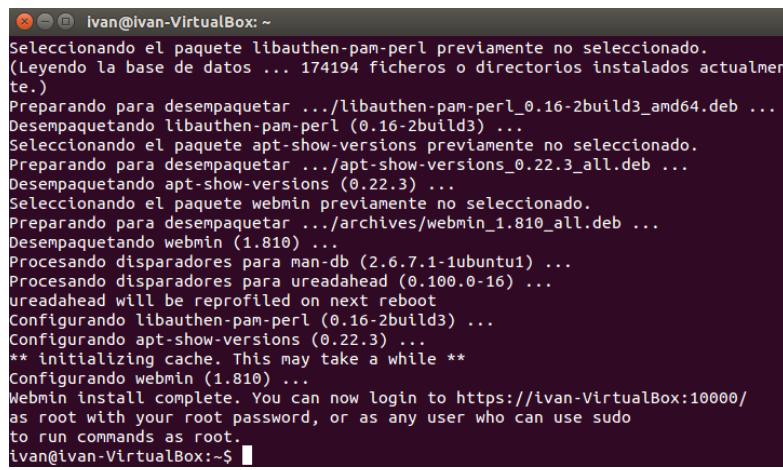
ivan@ivan-VirtualBox:-$ apt-key add jcameron-key.asc
ERROR: This command can only be used by root.
ivan@ivan-VirtualBox:-$ sudo apt-key add jcameron-key.asc
OK
ivan@ivan-VirtualBox:-$ 

```

Figura 5.2: Añadir la clave necesaria para instalar webadmin

Una vez hecho esto, se puede instalar webmin desde repositorio, para lo cual antes tenemos que actualizar la lista de paquetes(apt update) e instalar el paquete de

webadmin:



```
ivan@ivan-VirtualBox:~$ 
Seleccionando el paquete libauthen-pam-perl previamente no seleccionado.
(Leyendo la base de datos ... 174194 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../libauthen-pam-perl_0.16-2build3_amd64.deb ...
Desempaquetando libauthen-pam-perl (0.16-2build3) ...
Seleccionando el paquete apt-show-versions previamente no seleccionado.
Preparando para desempaquetar .../apt-show-versions_0.22.3_all.deb ...
Desempaquetando apt-show-versions (0.22.3) ...
Seleccionando el paquete webmin previamente no seleccionado.
Preparando para desempaquetar .../archives/webmin_1.810_all.deb ...
Desempaquetando webmin (1.810) ...
Procesando disparadores para man-db (2.6.7.1-1ubuntu1) ...
Procesando disparadores para ureadahead (0.100.0-16) ...
ureadahead will be reprofiled on next reboot
Configurando libauthen-pam-perl (0.16-2build3) ...
Configurando apt-show-versions (0.22.3) ...
** initializing cache. This may take a while **
Configurando webmin (1.810) ...
Webmin install complete. You can now login to https://ivan-VirtualBox:10000/
as root with your root password, or as any user who can use sudo
to run commands as root.
ivan@ivan-VirtualBox:~$
```

Figura 5.3: Instalación de webmin llevada a cabo a través de repositorio.

Aquí vemos Webadmin accesible en el puerto 10000 del servidor:

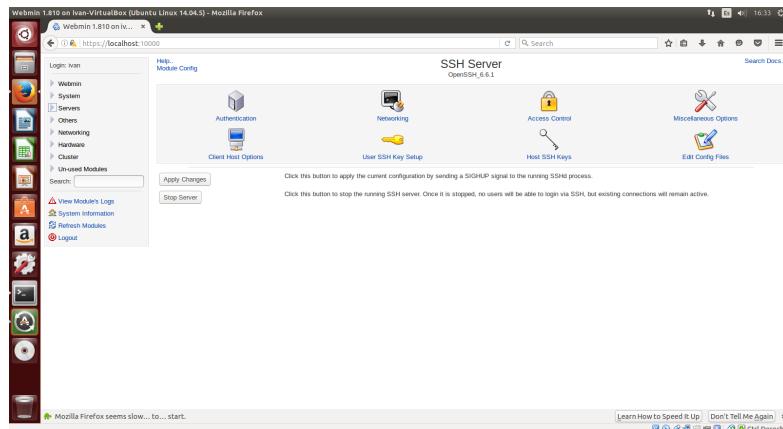


Figura 5.4: Webmin accesible en el navegador. Por defecto, escucha el puerto 10000.

Se nos pide que modifiquemos algún parámetro de algún servicio y mostremos que funciona. Nosotros modificaremos el puerto de escucha del servidor de apache. Para ello, extendemos el nodo Servers y seleccionamos la configuración global de Apache:

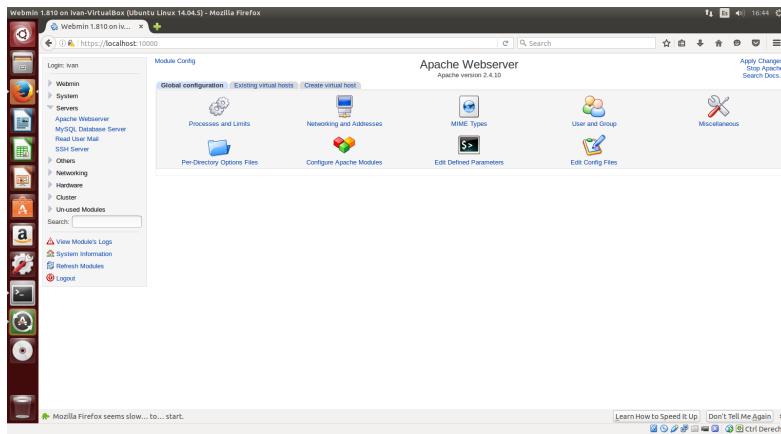


Figura 5.5: Configuración global de Apache ofrecida por Webmin.

donde seleccionaremos redes y direcciones(Network and Addressess). Entonces, en la casilla del puerto de escucha, cambiamos a 8080:

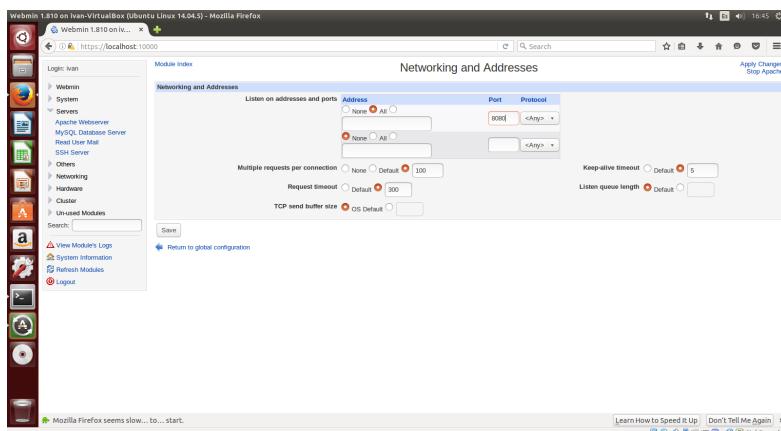


Figura 5.6: Puerto del servicio Apache modificado por Webmin al 8080.

Al guardar y aplicar los cambios, vemos que podemos acceder a la página web de Apache en el puerto 8080:

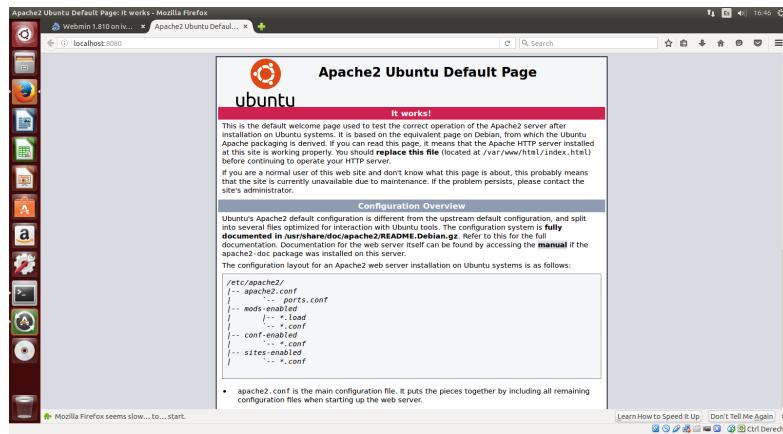


Figura 5.7: Página de apache ofrecida en el puerto 8080.

- Instale phpMyAdmin, indique cómo lo ha realizado y muestre algunas capturas de pantalla. Configure PHP para poder importar BDs mayores de 8MiB (límite por defecto). Indique cómo ha realizado el proceso y muestre capturas de pantalla.

Según la página oficial de phpMyAdmin[24], podemos añadir el repositorio escribiendo el siguiente comando:

```
sudo add-apt-repository ppa:nijel/phpmyadmin
```

así que tras actualizar el sistema(*apt-get update*), debemos de tener el paquete *phpmyadmin* en la lista de paquetes que podemos descargar. Ejecutamos el siguiente comando:

```
sudo apt install phpmyadmin
```

el cual nos dejará configurar algunas opciones de *phpmyadmin* mientras se instala:

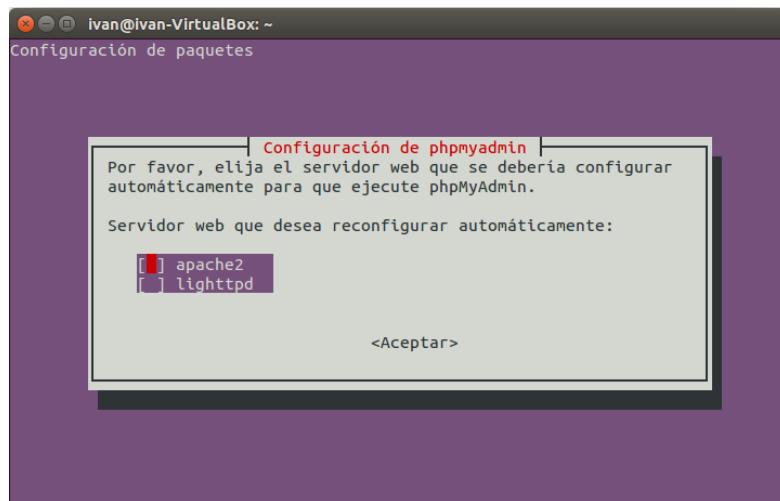


Figura 5.8: Servidor web que configurar por defecto. Nosotros elegiremos apache2.

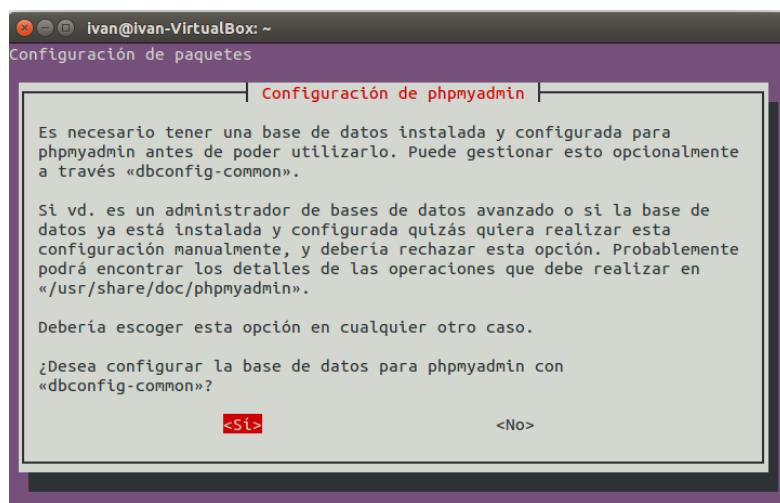


Figura 5.9: Configuración por defecto para la base de datos. Por defecto, y como no somos expertos en la materia, tendremos esta configuración.

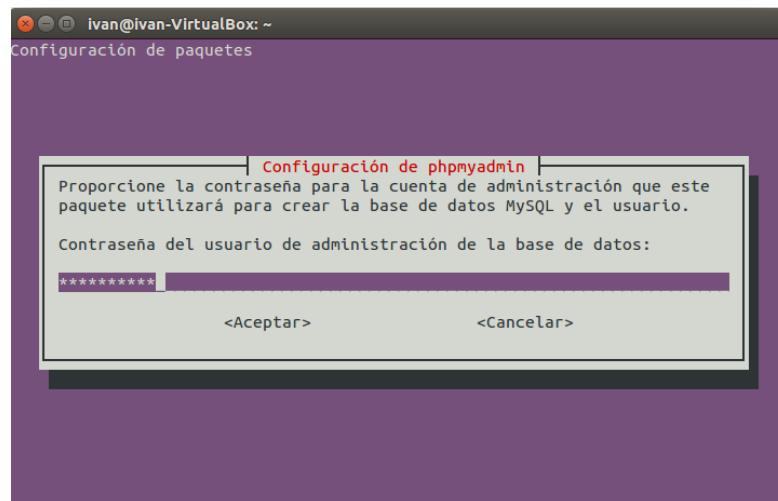


Figura 5.10: Configurar la contraseña para la base de datos.

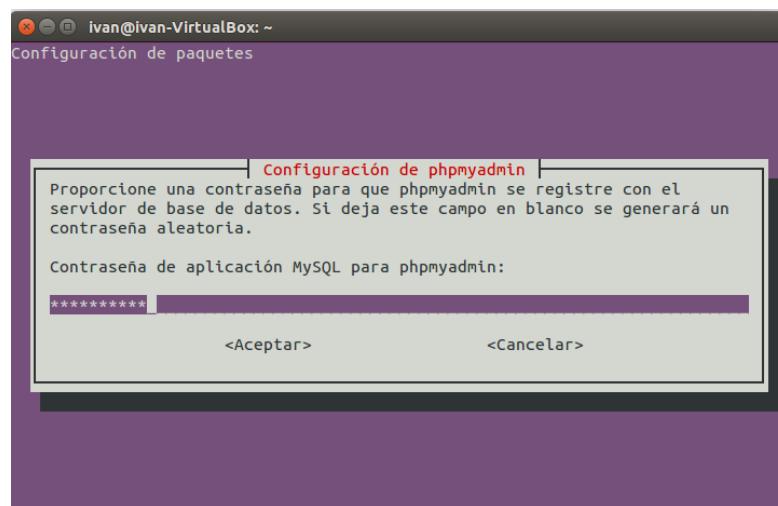


Figura 5.11: Contraseña que pide phpmyadmin para la base de datos.

Ya tenemos instalada la herramienta PhpMyAdmin. Para acceder a ella en un navegador accedemos a la dirección *localhost/phpmyadmin* :

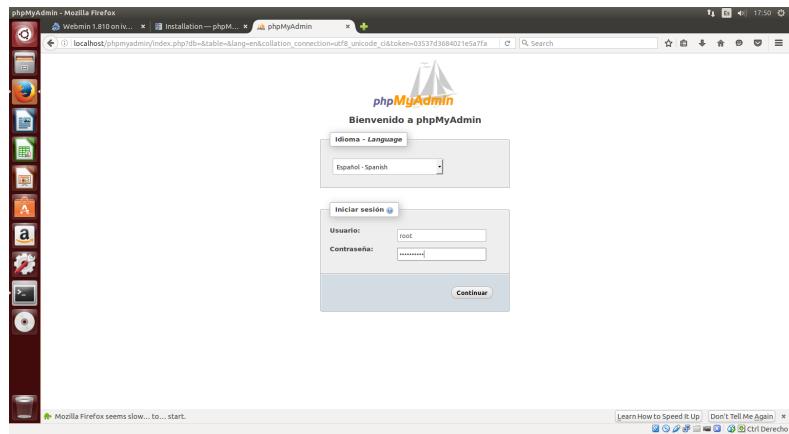


Figura 5.12: Pantalla de inicio de PhpMyAdmin

La siguiente es la página de inicio que se abre tras autenticarse como usuario root de phpmyadmin:

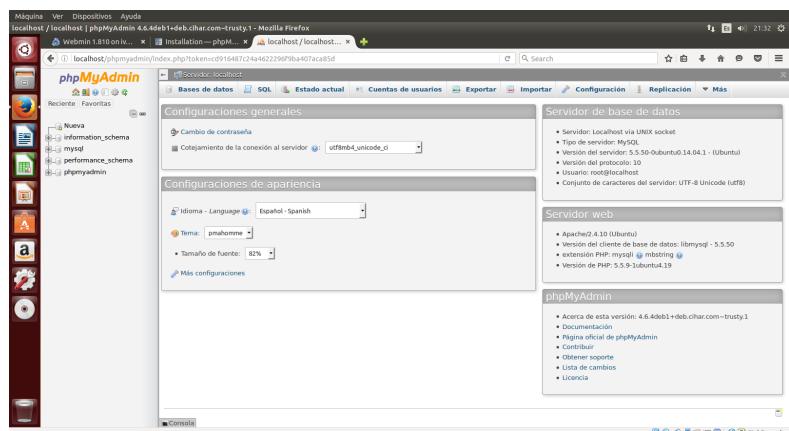


Figura 5.13: Pantalla de inicio de phpmyadmin tras autenticarse como root.

Si nos ponemos a fisgar en la herramienta, nos encontramos con la pestaña de base de datos:

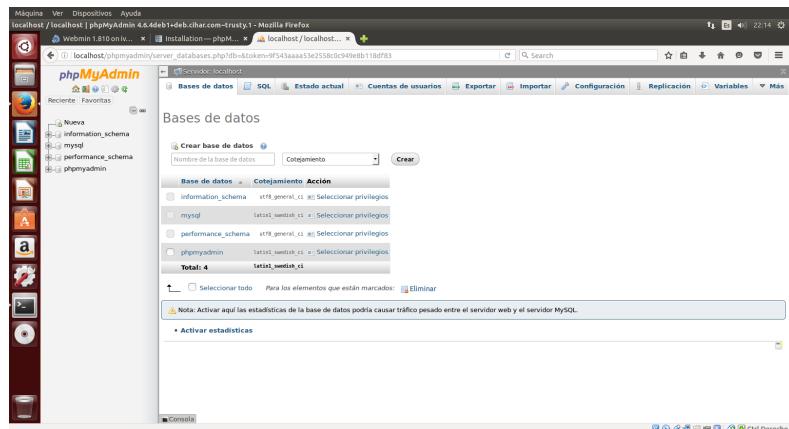
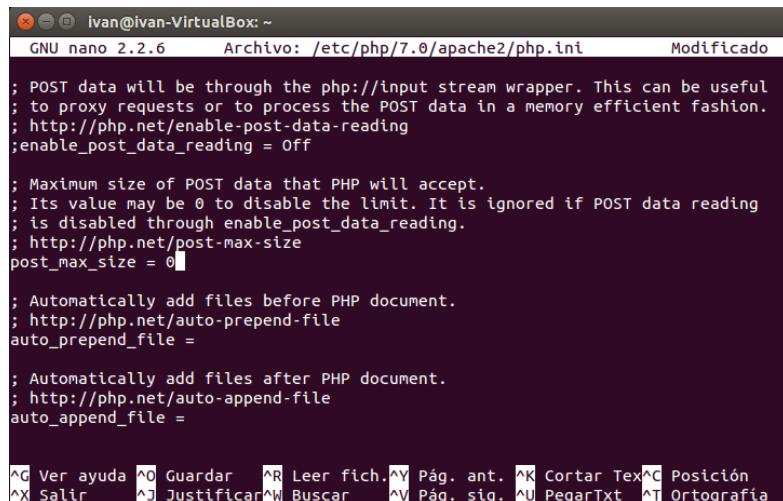


Figura 5.14: Bases de datos por defecto en phpmyadmin.

de las cuales se pueden ver estadísticas como la cantidad de datos que estas poseen, que como vemos, ninguna sobrepasa la cantidad por defecto. Esto nos lleva a la segunda parte de la pregunta:

¿Cómo modiflico el tamaño máximo de base de datos que puedo administrar en phpmyadmin?

La primera respuesta que encontré estaba en stackoverflow, tras la cual busqué en php el parámetro que tenía que modificar(las páginas web consultadas están en [25]). Según ambas respuestas, el parámetro que tengo que modificar se encuentra en el directorio `/etc/php/7.0/apache2/` en el archivo **php.ini**. Tal parámetro es `post_max_size`, el cual define el tamaño máximo de datos permitido. También afecta a la subida de ficheros, lo que andábamos buscando. En principio, para terminar esta parte de la pregunta nos basta responder que con cambiar el valor de 8Mib a un valor mayor podremos subir bases de datos mayores. Sin embargo el parámetro también se puede cambiar a cero, lo que haría que se admitiese cualquier tamaño de base de datos para la subida.



```
GNU nano 2.2.6      Archivo: /etc/php/7.0/apache2/php.ini      Modificado

; 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 = 0

; 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 =


^G Ver ayuda ^O Guardar ^R Leer fich.^Y Pág. ant. ^K Cortar Tex^C Posición
^X Salir ^J Justificar^W Buscar ^V Pág. sig. ^U PegarTxt ^T Ortografía
```

Figura 5.15: Modificación del parámetro post_max_size en el archivo php.ini.

Reiniciamos el servicio de apache2 y ya estará en funcionamiento.

5.1. Otros servidores.

- Visite al menos una de las webs de los software mencionados en la práctica y pruebe las demos que ofrecen realizando capturas de pantalla y comentando qué está realizando.

Vamos a visitar la página web de DirectAdmin[26], en la cual probaremos los servicios que ofrece como administrador web:

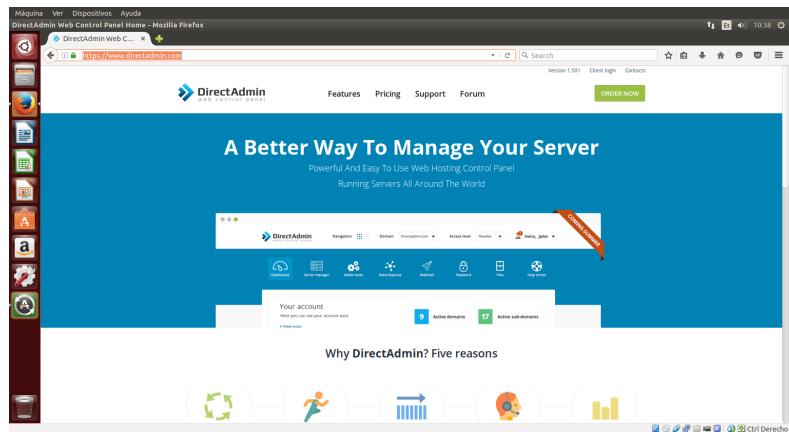


Figura 5.16: Página web de DirectAdmin

Figura 5.17:

en la cual, podremos probar una demo de administrador:

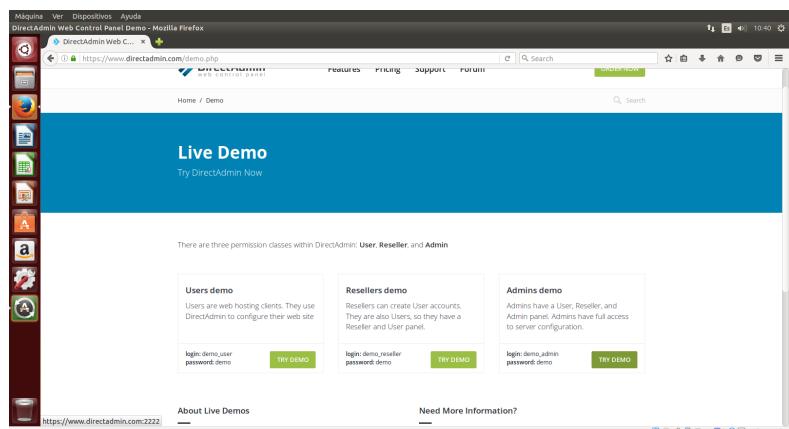


Figura 5.18: Probar las demos. Login:demo_admin, Password:demo

La página de inicio del servicio es la siguiente:

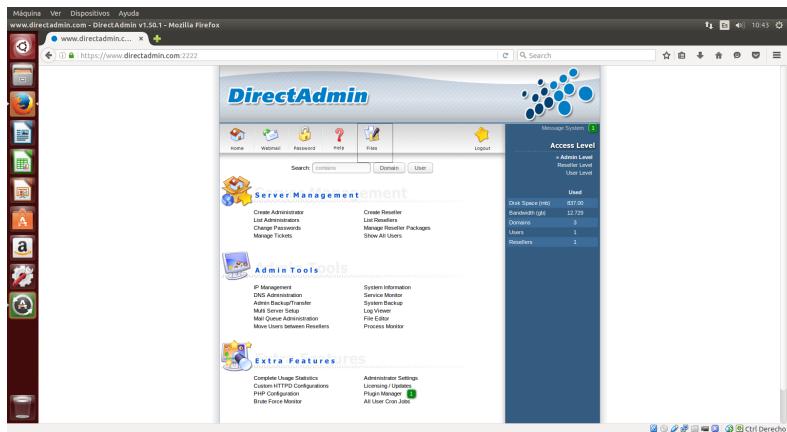


Figura 5.19: Página de inicio de Directadmin como administrador.

Vamos a ir mostrando distintas funcionalidades de la demo. Un ejemplo de ello es el monitor de servicios, desde el cual se pueden reiniciar o parar:



Figura 5.20: Monitor de servicios, localizado en herramientas de administrador(Admin Tool).

También nos encontramos con un editor de texto. En la web no podemos usarlo ya que está deshabilitado, pero con él podremos editar archivos sensibles para la configuración de nuestra página web, como sería httpd.conf:

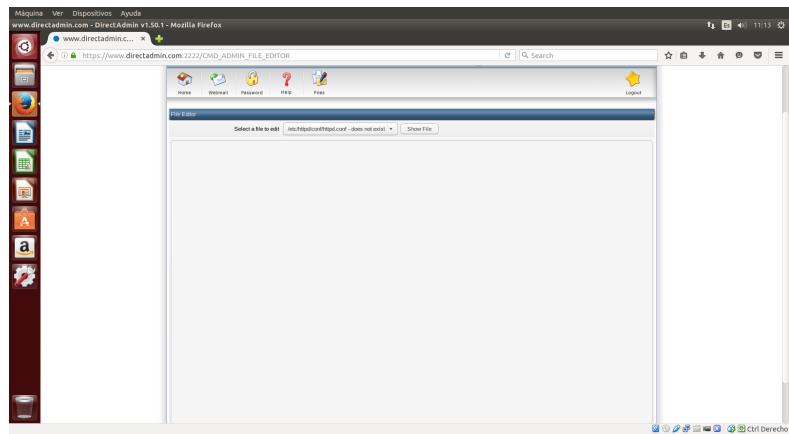


Figura 5.21: Editor de texto para modificar archivos sensibles de la configuración. No disponible en la demo.

También podemos crear cuentas de toda clase, como por ejemplo de administrador. Sin embargo, en la demo la creación está deshabilitada:



Figura 5.22: Creación de la cuenta de administrador, con login ivan y cuenta de correo fake@fakemail.com. Se encuentra en Server Management.

Por último, en la sección Extra Features, nos podemos encontrar con los datos de las actividades periódicas programadas con cron de todos los usuarios:



Figura 5.23: Trabajos periódicos programados por cron de todos los usuarios.

6. Automatización de tareas.

En esta sección se aborda la creación de scripts, el uso de la shell y algunos comandos útiles para la automatización de tareas. Un ejemplo de comandos son *grep*[27], *find*[28], *awk*[29] y *sed*[30].

- Ejecute los ejemplos de *find*, *grep* y escriba el script que haga uso de *sed* para cambiar la configuración de ssh y reiniciar el servicio.

Ejecución del ejemplo de *find*:

```
ivan@ivan-HP-ENVY-dv6-Notebook-PC: ~
ivan@~$ find /home/ivan/Documentos/ -name '*pdf' -exec cp {} ~/PDFs/ \;
cp: se omite el directorio «/home/ivan/Documentos/3º_Doble_Grado/IA/Curso_2015-2016-Doble-Grado/temas pdf»
ivan@~$
```

Figura 6.1: Ejecución del ejemplo de *Find*, en donde se recopilan los Pdfs dentro del directorio Documentos en el directorio PDFs

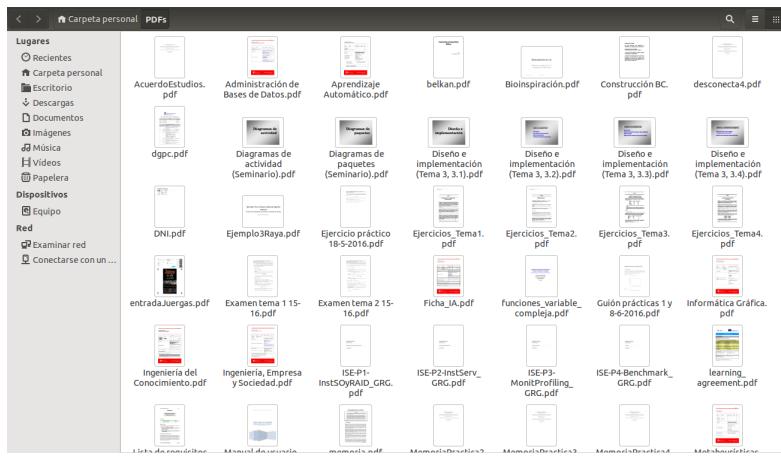


Figura 6.2: Directorio PDFs tras la ejecución de find.

Ejecución del ejemplo de grep:

```
ivan@-~$ ps -Af | grep firefox
ivan 315 32682 1 12:17 ? 00:00:00 /usr/lib/firefox/plugin-container -greomni /usr/lib/firefo
x/omni.ja -appomni /usr/lib/firefox/browser/omni.ja -appdir /usr/lib/firefox/browser 32682 true tab
ivan 375 31381 0 12:18 pts/9 00:00:00 grep --color=auto firefox
ivan 32682 23594 33 12:17 ? 00:00:20 /usr/lib/firefox/firefox
ivan@-~$
```

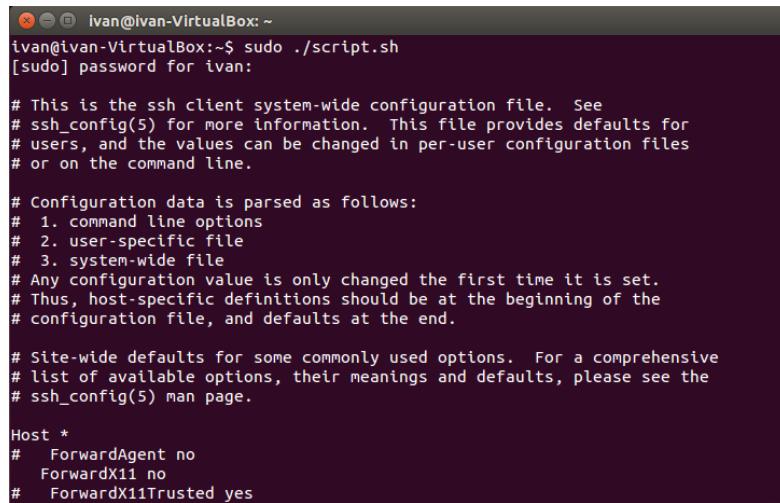
Figura 6.3: Ejecución del ejemplo de grep. En el se busca la información de los procesos que tengan que ver con firefox.

El script que usaremos será el siguiente, que deberá ser ejecutado como administrador:

```
cp /etc/ssh/ssh_config /etc/ssh/ssh_config.original
sed 's/# ForwardX11 no/ ForwardX11 no/g' /etc/ssh/ssh_config
service ssh restart
```

En el, primero copiamos la configuración original por si acaso en un archivo aparte. Luego, en la segunda linea, se sustituye la linea comentada en la que se invalida

la posibilidad de poder usar interfaz gráfica en conexiones ssh por la misma linea descomentada. En la última linea se reinicia el servicio. Aquí está la ejecución del script:



```
ivan@ivan-VirtualBox:~$ sudo ./script.sh
[sudo] password for ivan:

# This is the ssh client system-wide configuration file. See
# ssh_config(5) for more information. This file provides defaults for
# users, and the values can be changed in per-user configuration files
# or on the command line.

# Configuration data is parsed as follows:
# 1. command line options
# 2. user-specific file
# 3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options. For a comprehensive
# list of available options, their meanings and defaults, please see the
# ssh_config(5) man page.

Host *
#   ForwardAgent no
#   ForwardX11 no
#   ForwardX11Trusted yes
```

Figura 6.4: Salida producida por el script. En el se ve que se modifica la linea deseada.

- **Escriba el script para cambiar el acceso a ssh usando PHP o Python.**
Yo he escogido python.???
- **Windows Powershell.** Abra una consola de Powershell y pruebe a parar un programa en ejecución (p.ej), realice capturas de pantalla y comente lo que muestra.

Para averiguar que comando es el necesario para parar un proceso, escribimos en la powershell el comando Get-Command -verb stop:

CommandType	Name	ModuleName
Function	Stop-Dtc	MsDtc
Function	Stop-DtcTransactionsTraceSession	MsDtc
Function	Stop-PerformanceCollector	ServerManagerTasks
Function	Stop-PhysicalDesktopCollectionJob	RemoteManagement
Function	Stop-ScheduledTask	ScheduledTasks
Function	Stop-Trace	PSDiagnostics
Cmdlet	Stop-Computer	Microsoft.PowerShell.Management
Cmdlet	Stop-DtcDiagnosticResourceManager	MsDtc
Cmdlet	Stop-Process	Microsoft.PowerShell.Core
Cmdlet	Stop-Service	Microsoft.PowerShell.Management
Cmdlet	Stop-Transcript	Microsoft.PowerShell.Management
Cmdlet	Stop-WebAppPool	WebAdministration
Cmdlet	Stop-WebCommitDelay	WebAdministration
Cmdlet	Stop-Website	WebAdministration
Cmdlet	Stop-WebItem	WebAdministration
Cmdlet	Stop-Website	WebAdministration

Figura 6.5: Comando get-command de la powershell para encontrar el comando que para un programa.

que como sospechábamos, era Stop-Process. Ahora probamos a obtener los procesos activos y vamos a eliminar el nuestro:

Handles	NPM(K)	PM(K)	UEL(K)	IMOD	CPU(s)	Id	ProcessName
47	5	1816	7452	52	0.03	2188	conhost
294	12	1168	2832	339	0.13	1798	taskhostw
294	12	1168	15726	175	0.06	428	svcs
297	12	1168	15726	175	0.06	428	svcs
297	12	1168	27118	137	0.09	769	devlayer
297	12	1168	27118	137	0.09	769	devlayer
291	21	2344	2728	36	0.47	524	iexplorer
427	24	60708	74756	607	0.72	7228	powershell
214	11	3098	4112	7	0.06	515	services
214	11	3098	4112	7	0.06	515	services
161	14	5144	1700	35	1.40	104	svchost
415	15	5205	2689	54	2.17	703	svchost
415	15	5205	2689	54	2.17	703	svchost
415	15	5205	2689	54	2.17	703	svchost
172	44	26972	26975	135	1.97	824	svchost
644	21	11252	6048	1374	2.54	740	svchost
207	15	11722	2544	41	0.11	2987	svchost
208	15	11722	2544	41	0.11	2987	svchost
177	8	3120	2623	3	21.21	1208	System
127	13	22548	2876	56	16.20	644	UserService
131	18	22548	2876	56	16.20	644	UserService
131	18	22548	2876	56	16.20	644	UserService
137	7	1154	1484	49	0.13	406	wia
137	7	1154	1484	49	0.13	406	wia
276	16	4972	1044	40	0.19	2528	WinPvTE

Figura 6.6: Información de los procesos que lleva a cabo windows. Información suministrada por el comando Get-Process de PowerShell. ID del proceso encargado del Server Manager:1560.

Puesto que estamos ejecutando el Server Manager, con Pid 1560, para pararlo tenemos que usar el comando antes buscado: stop-process 1560, tras lo cual, paramos el proceso:

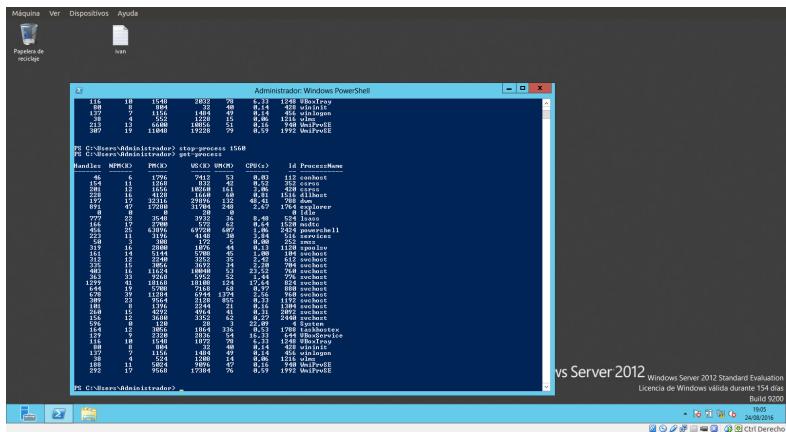


Figura 6.7: Comando stop-process de la powershell sobre el server manager de windows.
Tras esto, hemos vuelto a usar get-process.

Como vemos en la ventana, hemos parado el proceso y no vemos rastro de la ventana ni del nombre del proceso entre los procesos que lista get-process.

Referencias

- [1] <http://linux.die.net/man/8/yum>
- [2] <http://linux.die.net/man/5/yum.conf>
- [3] <http://linux.die.net/man/8/apt>
- [4] <http://linux.die.net/man/5/apt.conf>
- [5] <https://es.opensuse.org>
- [6] <http://linux.die.net/man/1/telnet>
- [7] <http://linux.die.net/man/1/ssh>
- [8] <http://linux.die.net/man/1/ssh>
- [9] <http://linux.die.net/man/1/ssh-keygen>
- [10] <http://linux.die.net/man/1/ssh-copy-id>
- [11] [http://www.freebsd.org/cgi/man.cgi?sshd\(8\)](http://www.freebsd.org/cgi/man.cgi?sshd(8))
- [12] [https://www.freebsd.org/cgi/man.cgi?sshd_config\(5\)](https://www.freebsd.org/cgi/man.cgi?sshd_config(5))
- [13] <http://manpages.ubuntu.com/manpages/precise/man1/terminator.1.html>
- [14] <https://www.gnu.org/software/screen/manual/screen.html>
- [15] http://www.fail2ban.org/wiki/index.php/MANUAL_0_8
- [16] <https://w3techs.com>
- [17] <https://www.litespeedtech.com/products/litespeed-web-server/overview>
- [18] <http://tomcat.apache.org/>
- [19] <https://nodejs.org/en/>
- [20] [https://technet.microsoft.com/es-es/library/hh831655\(v=ws.11\).aspx#Step2](https://technet.microsoft.com/es-es/library/hh831655(v=ws.11).aspx#Step2)
- [21] <http://fedoraproject.org/wiki/VMWare>
- [22] <http://man7.org/linux/man-pages/man1/diff.1.html>
- [23] <http://www.webmin.com/deb.html>
- [24] <https://docs.phpmyadmin.net>
- [25] <http://php.net/manual/es/ini.core.php#ini.post-max-size>
<http://stackoverflow.com/questions/3958615/import-file-size-limit-in-phpmyadmin>

- [26] <https://www.directadmin.com/>
- [27] <https://www.gnu.org/software/grep/manual/grep.html>
- [28] <http://man7.org/linux/man-pages/man1/find.1.html>
- [29] <https://www.gnu.org/software/gawk/manual/gawk.html#Running-gawk>
- [30] <http://linux.die.net/man/1/sed>