



*ugr*

Universidad  
de **Granada**

**Grado en Ingeniería Informática.**

## **Práctica 2.**

---

**Nombre de la asignatura:**

Ingeniería de Servidores.

**Realizado por:**

Néstor Rodríguez Vico



**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS  
INFORMÁTICA Y DE TELECOMUNICACIÓN.**

---

Granada, 30 de noviembre de 2016.

# Índice

<b>1. Cuestión 1:</b>	<b>7</b>
1.1. a) Liste los argumentos de yum necesarios para instalar, buscar y eliminar paquetes. . . . .	7
1.2. b) ¿Qué ha de hacer para que yum pueda tener acceso a Internet en el PC del aula?(Pistas: archivo de configuración en /etc, proxy: stargate.ugr.es:3128). . . . .	7
1.3. c) ¿Cómo añadimos un nuevo repositorio? . . . . .	8
<b>2. Cuestión 2:</b>	<b>8</b>
2.1. a) Liste los argumentos de apt necesarios para instalar, buscar y eliminar paquetes. . . . .	8
2.2. b)¿Qué ha de hacer para que apt pueda tener acceso a Internet en el PC del aula?(Pistas: archivo de configuración en /etc, proxy: stargate.ugr.es:3128) . . . . .	8
2.3. c)¿Cómo añadimos un nuevo repositorio? . . . . .	9
<b>3. Cuestión 3:</b>	<b>9</b>
3.1. a) ¿Con qué comando puede abrir/cerrar un puerto usando ufw? Muestre un ejemplo de cómo lo ha hecho. . . . .	9
3.2. b) ¿Con qué comando puede abrir/cerrar un puerto usando firewall-cmd en CentOS? Muestre un ejemplo de cómo lo ha hecho. . . . .	9
3.3. c) Utilice el comando nmap para ver que, efectivamente, los puertos están accesibles. . . . .	10
<b>4. Cuestión 4: ¿Qué diferencia hay entre telnet y ssh?</b>	<b>11</b>
<b>5. Cuestión 5:</b>	<b>11</b>
5.1. a) ¿Para qué sirve la opción -X? . . . . .	11
5.2. b) Ejecute remotamente, es decir, desde la máquina anfitriona (si tiene Linux) o desde la otra máquina virtual, el comando gedit en una sesión abierta con ssh. ¿Qué ocurre? . . . . .	11
<b>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).</b>	<b>13</b>
<b>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.</b>	<b>14</b>

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.	15
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.	16
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.	21
11.Cuestión 11: Muestre un ejemplo de uso del comando.	21
12.Cuestión 12: Realice la instalación de esta aplicación y pruebe a modificar algún parámetro de algún servicio. Muestre las capturas de pantalla pertinentes así como el proceso de instalación.	22
13.Cuestión 13 : Instale phpMyAdmin, indique cómo lo ha realizado y muestre algunas capturas de pantalla. Configure PHP para poder importar BDs de hasta 25MiB (en vez de los 8 MiB de límite por defecto). Indique cómo ha realizado el proceso y muestre capturas de pantalla.	25
14.Cuestión 14: Viste al menos una de las webs de los software mencionados y pruebe las demos que ofrecen realizando capturas de pantalla y comentando qué está realizando.	28
15.Cuestión 15:	30
15.1. a) Ejecute los ejemplos de find, grep. . . . .	30
15.2. b) Escriba el script que haga uso de sed para cambiar la configuración de ssh y reiniciar el servicio. . . . .	30
15.3. c) Muestre un ejemplo de uso para awk. . . . .	31
16.Cuestión 16: Escriba el script para cambiar el acceso a ssh usando PHP o Python.	31
17.Cuestión 17: Abra una consola de Powershell y pruebe a parar un programa en ejecución (p.ej), realice capturas de pantalla y comente lo que muestra.	32
18.Cuestión opcional 1: Instale y pruebe terminator y/o tmux. Con screen, pruebe su funcionamiento dejando sesiones ssh abiertas en el servidor y recuperándolas posteriormente.	34
19.Cuestión opcional 2: Instale el servicio y pruebe su funcionamiento.	37
20.Cuestión opcional 3: Instale el servicio y pruebe su funcionamiento.	38

- 21. Cuestión opcional 4: Realice la instalación de uno de estos dos “web containers” y pruebe su ejecución. 39**
- 22. Cuestión opcional 5: Realice la instalación de MongoDB en alguna de sus máquinas virtuales. Cree una colección de documentos y haga una consulta sobre ellos. 42**

## Índice de figuras

1.1. Error con el proxy añadido en mi ordenador (yum). . . . .	7
2.1. Error con el proxy añadido en mi ordenador (apt). . . . .	9
3.1. Puerto 10 habilitado y análisis de los puertos con nmap (Ubuntu Server). . . . .	10
3.2. Puerto 10 habilitado y análisis de los puertos con nmap (Ubuntu Server). . . . .	10
5.1. Error al ejecutar gedit en remoto (máquinas conectadas en modo <i>bridge</i> ). . . . .	11
5.2. Error tras instalar gedit en remoto (máquinas conectadas en modo <i>bridge</i> ). . . . .	12
5.3. gedit funcionando (máquinas conectadas en modo <i>bridge</i> ). . . . .	12
6.1. Conexión ssh sin contraseña (máquinas conectadas en modo <i>bridge</i> ). . . . .	13
7.1. Conexión no permitida al usuario root (máquinas conectadas en modo <i>host-only</i> ). . . . .	14
7.2. Conexión ssh con el puerto cambiado al puerto 23 (máquinas conectadas en modo <i>bridge</i> ). . . . .	15
8.1. Reinicio del servicio ssh. . . . .	16
8.2. Reinicio del servicio httpd. . . . .	16
9.1. Servidor web en funcionamiento (máquinas conectadas en modo <i>bridge</i> ). . . . .	17
9.2. php en funcionamiento (máquinas conectadas en modo <i>host-only</i> ). . . . .	18
9.3. mysql en funcionamiento (máquinas conectadas en modo <i>host-only</i> ). . . . .	18
9.4. Servidor web en funcionamiento (CentOS) (máquinas conectadas en modo <i>bridge</i> ). . . . .	19
9.5. php en funcionamiento (máquinas conectadas en modo <i>host-only</i> ). . . . .	20
9.6. mysql en funcionamiento (máquinas conectadas en modo <i>host-only</i> ). . . . .	20
10.1. Servidor web en funcionamiento (Windows Server). . . . .	21
11.1. Proceso de “parcheo” con el comando <i>patch</i> . . . . .	22
12.1. Repositorio añadido. . . . .	23
12.2. Obtención e instalación de la clave GPG. . . . .	23
12.3. Actualizamos los repositorios e instalamos Webmin. . . . .	23
12.4. Webmin en funcionamiento (máquinas conectadas en modo <i>bridge</i> ). . . . .	24
12.5. Manipulación de servicios con Webmin (máquinas conectadas en modo <i>bridge</i> ). . . . .	24
12.6. Ejecución de comandos mediante Webmin (máquinas conectadas en modo <i>bridge</i> ). . . . .	25
13.1. Archivo modificado. . . . .	26
13.2. phpMyAdmin funcionando (máquinas conectadas en modo <i>bridge</i> ). . . . .	26
13.3. <code>upload_max_filesize</code> con el nuevo valor. . . . .	27
13.4. <code>post_max_size</code> con el nuevo valor. . . . .	27
13.5. El tamaño máximo permitido es 25MiB (máquinas conectadas en modo <i>bridge</i> ). . . . .	27
14.1. Inicio Parallels Plesk. . . . .	28
14.2. Estadísticas del servidor desde Parallels Plesk. . . . .	29
14.3. Ajustes y herramientas del servidor desde Parallels Plesk. . . . .	29
15.1. Ejecución de <code>grep</code> y <code>find</code> . . . . .	30

15.2. Ejecución de <i>awk</i> . . . . .	31
17.1. Procesos activos. . . . .	32
17.2. Paramos el servicio <i>VBoxService</i> . . . . .	33
17.3. Procesos activos tras parar el proceso <i>VBoxService</i> . . . . .	33
18.1. Creación de las sesiones de <i>screen</i> . . . . .	35
18.2. Restauración de las sesiones de <i>screen</i> . . . . .	35
18.3. Sesiones de <i>screen</i> restauradas. . . . .	36
18.4. Sesiones de <i>screen</i> cerradas. . . . .	36
19.1. Copia del archivo <i>/etc/fail2ban/jail.conf</i> . . . . .	37
19.2. Acceso incorrecto (máquinas conectadas en modo <i>host-only</i> ). . . . .	37
19.3. Dirección IP bloqueada (máquinas conectadas en modo <b><i>bridge</i></b> ). . . . .	38
20.1. Creación de la base de datos para <i>rkhunter</i> . . . . .	38
20.2. Resultado del análisis del sistema con <i>rkhunter</i> . . . . .	39
21.1. Java no se encuentra instalado de Ubuntu Server. . . . .	40
21.2. Dirección IP de mi servidor. . . . .	40
21.3. Servicio <i>tomcat</i> funcionando correctamente (máquinas conectadas en modo <i>host-only</i> ). . . . .	41
21.4. Sesión iniciada en <i>tomcat</i> (máquinas conectadas en modo <i>host-only</i> ). . . . .	41
22.1. Instalación de <i>MongoDB</i> . . . . .	42
22.2. Cambio de <i>SELINUX</i> . . . . .	43
22.3. Creación de la colección, inserción de documentos y consulta. . . . .	43

## 1. Cuestión 1:

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

Como podemos ver en la página de RedHat [1], los comandos son los siguientes:

- Instalar paquetes: *yum install <nombre del paquete/s>*
- Buscar paquetes: *yum search <palabra clave>*
- Eliminar paquetes: *yum remove <nombre del paquete/s>*

Por lo tanto los argumentos son:

- Instalar paquetes: *install*
- Buscar paquetes: *search*
- Eliminar paquetes: *remove*

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

Para tener acceso a Internet, tenemos que configurar un proxy. Según CentOS [2] hay que indicar el proxy en el archivo */etc/yum.conf*. Para tener acceso desde los ordenadores de la UGR, debemos añadir la siguiente línea: *proxy=http://astargate.ugr.es:3128*

Una manera de comprobar si funciona correctamente es añadir el proxy en mi ordenador. Si hacemos esto, al ejecutar por ejemplo *sudo yum install httpd* nos da error. Da error porque intenta conectarse hacia afuera usando ese proxy. De esta manera podemos ver que funciona correctamente. En la figura 1.1 podemos ver el error que obtenemos y como, efectivamente, está el proxy activado.

```
Error downloading packages:
mailcap-2.1.41-2.el7.noarch: [Errno 256] No more mirrors to try.
apr-1.4.8-3.el7.x86_64: [Errno 256] No more mirrors to try.
httpd-tools-2.4.6-40.el7.centos.4.x86_64: [Errno 256] No more mirrors to try.
apr-util-1.5.2-6.el7.x86_64: [Errno 256] No more mirrors to try.
httpd-2.4.6-40.el7.centos.4.x86_64: [Errno 256] No more mirrors to try.

nrv/2016-10-29:~$ cat /etc/yum.conf | grep proxy
proxy=http://astargate.ugr.es:3128
nrv/2016-10-29:~$
```

Figura 1.1: Error con el proxy añadido en mi ordenador (yum).

### 1.3. c) ¿Cómo añadimos un nuevo repositorio?

Como podemos ver en la página de Centos [2] para añadir un repositorio tenemos que ubicar el archivo de definiciones en el directorio `/etc/yum.repos.d/`. Los proveedores de paquetes ponen los archivos de definiciones en sus páginas web. para añadir un archivo de definiciones hay que tener acceso root. Para copiar el archivo en `textit/etc/yum.repos.d/` debemos ejecutar el comando `su -c 'cp example.repo /etc/yum.repos.d/'`.

## 2. Cuestión 2:

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

En la página de Ubuntu [3] podemos ver los comandos para realizar las distintas tareas:

- Instalar paquetes: `sudo apt-get install <nombre del paquete/s>`
- Buscar paquetes: `apt-cache search <palabra clave>`
- Eliminar paquetes: `sudo apt-get remove <nombre del paquete/s>`

Por lo tanto los argumentos son:

- Instalar paquetes: `install`
- Buscar paquetes: `search`
- Eliminar paquetes: `remove`

Nota: `-get` y `-cache` no son necesarios en la versiones más modernas de `apt`.

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

Para tener acceso a Internet, tenemos que configurar un proxy. Según Ubuntu [3] hay que indicar el proxy en el archivo `/etc/apt/apt.conf`. Para tener acceso desde los ordenadores de la UGR, hay que añadir la línea: `Acquire::http::Proxy "astargate.ugr.es:3128";`

Una manera de comprobar si funciona correctamente es añadir el proxy en mi ordenador. Si hacemos esto, al ejecutar por ejemplo `sudo apt update` nos da error. Da error porque intenta conectarse hacia afuera usando ese proxy. De esta manera podemos ver que funciona correctamente. En la figura 2.1 podemos ver el error que obtenemos y como, efectivamente, está el proxy activado.



```

W: Imposible obtener http://es.archive.ubuntu.com/ubuntu/dists/trusty-backports/universe/binary-i386/
Packages No puedo iniciar la conexión a 3128:80 (0.0.12.56). - connect (22: Argumento inválido)

W: Imposible obtener http://es.archive.ubuntu.com/ubuntu/dists/trusty-backports/multiverse/binary-i386/
Packages No puedo iniciar la conexión a 3128:80 (0.0.12.56). - connect (22: Argumento inválido)

E: No se han podido descargar algunos archivos de índice, se han omitido, o se han utilizado unos an-
tiguos en su lugar.
nrv/2016-10-29:/etc/apt$ cat apt.conf
Acquire::http::Proxy "astargate.ugr.es:3128";
nrv/2016-10-29:/etc/apt$

```

Figura 2.1: Error con el proxy añadido en mi ordenador (apt).

### 2.3. c) ¿Cómo añadimos un nuevo repositorio?

Según Ubuntu [4] hay dos maneras de hacerlo:

- Si el repositorio se encuentra en el fichero */etc/apt/sources.list*, tenemos que descomentarlo, es decir, quitar el símbolo *#* que precede a las líneas correspondientes al repositorio o ejecutando la orden *sudo add-apt-repository <repositorio a añadir>*
- Si el repositorio es PPA, debemos ejecutar la orden *sudo add-apt-repository ppa:<nombre del repositorio>*

## 3. Cuestión 3:

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

Cómo podemos ver en la página de Ubuntu [5], podemos usar los siguientes comandos para abrir y cerrar puertos, ambos deben ser ejecutado como *root*:

- Abrir un puerto: *ufw allow <puerto>/<opcional: protocolo>*
- Cerrar un puerto: *ufw deny <puerto>/<opcional: protocolo>*

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

Cómo podemos ver en la página de RedHat [6] y en la de Fedora [7], podemos usar los siguientes comandos para abrir y cerrar puertos, ambos deben ser ejecutado como *root*:

- Abrir un puerto: *firewall-cmd [--zone=<zona>] --add-port=<puerto>/<protocolo>*
- Cerrar un puerto: *firewall-cmd [--zone=<zona>] --remove-port=<puerto>/<protocolo>*

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

Para entender los resultados que nos devuelve nmap lo mejor es ver la página de ellos [8]. Hay que tener especial cuidado con el estado *closed*, ya que cómo podemos ver en la página de nmap puede haber puertos abiertos pero que no hay ningún programa escuchando dicho puerto y por lo tanto el estado de dicho puerto es *closed*. En ambos sistemas operativos (Ubuntu Server y CentOS) voy a abrir el puerto 10.

```
nrν/2016-11-03:~$ sudo ufw allow 10
Regla añadida
Regla añadida (v6)
nrν/2016-11-03:~$ sudo nmap -sS localhost

Starting Nmap 6.40 ( http://nmap.org ) at 2016-11-03 11:11 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000050s latency).
Other addresses for localhost (not scanned): 127.0.0.1
Not shown: 996 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
3306/tcp  open  mysql
10000/tcp open  snet-sensor-mgmt

Nmap done: 1 IP address (1 host up) scanned in 1.58 seconds
nrν/2016-11-03:~$ _
```

Figura 3.1: Puerto 10 habilitado y análisis de los puertos con nmap (Ubuntu Server).

```
nrν/2016-11-03:~$ sudo firewall-cmd --add-port 10/tcp
success
nrν/2016-11-03:~$ sudo nmap -sS localhost

Starting Nmap 6.40 ( http://nmap.org ) at 2016-11-03 15:18 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (-450s latency).
Other addresses for localhost (not scanned): 127.0.0.1
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    open  smtp

Nmap done: 1 IP address (1 host up) scanned in 0.05 seconds
nrν/2016-11-03:~$ _
```

Figura 3.2: Puerto 10 habilitado y análisis de los puertos con nmap (Ubuntu Server).

Como podemos ver, ni en la figura 3.1 ni en la figura 3.2 aparece el puerto abierto tras la ejecución del comando `sudo nmap -sS localhost` pero están abiertos correctamente. Esto se debe a la explicación anterior.

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

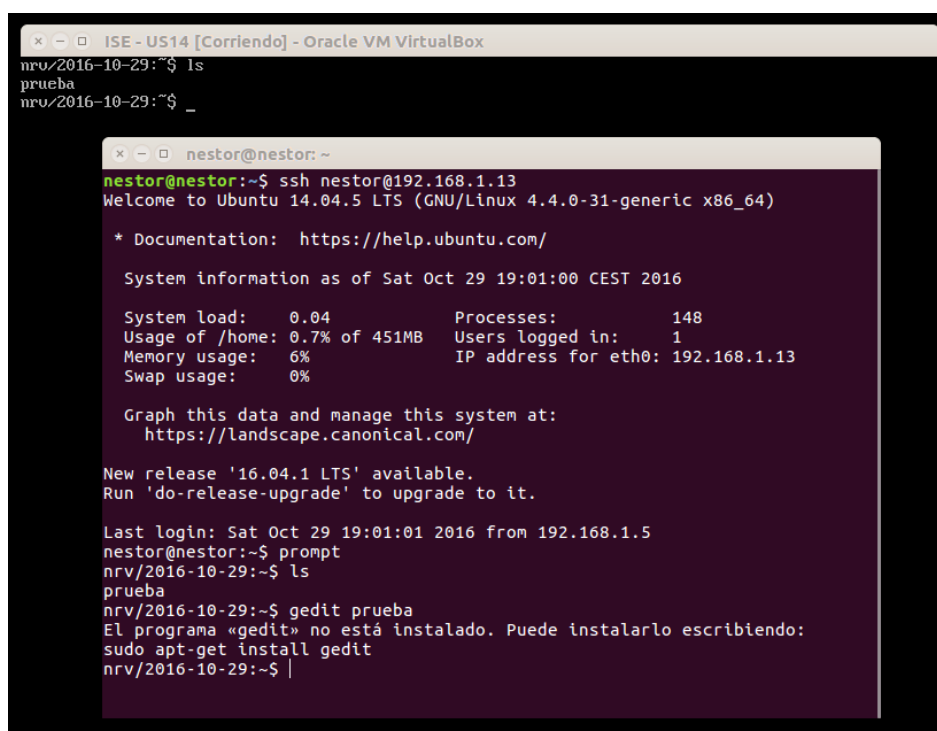
Comparando las páginas de Telnet [9] y la de OpenSSH [10] podemos ver que la mayor diferencia reside en la seguridad. ssh cifra los datos para mantener la comunicación segura mientras que telnet no.

## 5. Cuestión 5:

### 5.1. a) ¿Para qué sirve la opción -X?

Como podemos ver en las páginas de manual de ssh [11], la opción -X sirve para activar el reenvío X11.

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



```
x - □ ISE - US14 [Corriendo] - Oracle VM VirtualBox
nrv/2016-10-29:~$ ls
prueba
nrv/2016-10-29:~$ _

x - □ nestor@nestor:~
nestor@nestor:~$ ssh nestor@192.168.1.13
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information as of Sat Oct 29 19:01:00 CEST 2016

System load:  0.04               Processes:    148
Usage of /home: 0.7% of 451MB    Users logged in: 1
Memory usage:  6%               IP address for eth0: 192.168.1.13
Swap usage:    0%

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

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

Last login: Sat Oct 29 19:01:01 2016 from 192.168.1.5
nestor@nestor:~$ prompt
nrv/2016-10-29:~$ ls
prueba
nrv/2016-10-29:~$ gedit prueba
El programa «gedit» no está instalado. Puede instalarlo escribiendo:
sudo apt-get install gedit
nrv/2016-10-29:~$ |
```

Figura 5.1: Error al ejecutar gedit en remoto (máquinas conectadas en modo *bridge*).

El error se debe a que gedit no se encuentra instalado en Ubuntu Server y por eso no se puede ejecutar. Para instalarlo ejecutamos *sudo apt install gedit*. Una vez instalado, probamos de nuevo a ejecutar *gedit prueba* pero nos sigue dando error, como podemos ver en la figura 5.2.

```

nrv/2016-11-04:~$ ifconfig
eth0      Link encap:Ethernet direcciónHW 08:00:27:be:e0:b6
          Direc. inet:10.0.2.15 Difus.:10.0.2.255 Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:febe:e0b6/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
          Paquetes RX:44244 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:20811 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:59316701 (59.3 MB) TX bytes:1282753 (1.2 MB)

eth1      Link encap:Ethernet direcciónHW 08:00:27:fb:d2:b0
          Direc. inet:192.168.56.101 Difus.:192.168.56.255 Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:feb:d2b0/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
          Paquetes RX:6628 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:6613 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:596737 (596.7 KB) TX bytes:2499552 (2.4 MB)

lo        Link encap:Bucle local
          Direc. inet:127.0.0.1 Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Amfitrión
          ACTIVO BUCLE FUNCIONANDO MTU:65536 Métrica:1
          Paquetes RX:0 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:0 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1
          Bytes RX:0 (0.0 B) TX bytes:0 (0.0 B)

nrv/2016-11-04:~$ _

nestor@nestor:~$ ssh nestor@192.168.56.101
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information as of Fri Nov  4 17:42:20 CET 2016

System load:  0.0      Processes:    151
Usage of /home: 0.7% of 451MB Users logged in:  1
Memory usage:  11%    IP address for eth0: 10.0.2.15
Swap usage:    0%     IP address for eth1: 192.168.56.101

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

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

Last login: Fri Nov  4 17:42:20 2016 from 192.168.56.1
nestor@nestor:~$ gedit prueba
(gedit:9336): Gtk-WARNING **: cannot open display:
nestor@nestor:~$

```

Figura 5.2: Error tras instalar gedit en remoto (máquinas conectadas en modo *bridge*).

Para arreglarlo, debemos conectarnos a ssh con el argumento *-X*, tal y como vimos en el apartado anterior de esta misma pregunta. Podemos ver que ahora si se abre gedit correctamente, como se ve en la figura 5.3.

```

nrv/2016-11-04:~$ ifconfig
eth0      Link encap:Ethernet direcciónHW 08:00:27:be:e0:b6
          Direc. inet:10.0.2.15 Difus.:10.0.2.255 Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:febe:e0b6/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
          Paquetes RX:44244 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:20811 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:59316701 (59.3 MB) TX bytes:1282753 (1.2 MB)

eth1      Link encap:Ethernet direcciónHW 08:00:27:fb:d2:b0
          Direc. inet:192.168.56.101 Difus.:192.168.56.255 Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:feb:d2b0/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
          Paquetes RX:6628 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:6613 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:596737 (596.7 KB) TX bytes:2499552 (2.4 MB)

lo        Link encap:Bucle local
          Direc. inet:127.0.0.1 Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Amfitrión
          ACTIVO BUCLE FUNCIONANDO MTU:65536 Métrica:1
          Paquetes RX:0 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:0 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1
          Bytes RX:0 (0.0 B) TX bytes:0 (0.0 B)

nrv/2016-11-04:~$ _

nestor@nestor:~$ ssh nestor@192.168.56.101 -X
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information as of Fri Nov  4 17:42:26 CET 2016

System load:  0.0      Processes:    151
Usage of /home: 0.7% of 451MB Users logged in:  1
Memory usage:  11%    IP address for eth0: 10.0.2.15
Swap usage:    0%     IP address for eth1: 192.168.56.101

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

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

Last login: Fri Nov  4 17:42:27 2016 from 192.168.56.1
nestor@nestor:~$ gedit prueba

```

Figura 5.3: gedit funcionando (máquinas conectadas en modo *bridge*).

## 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`).

Gracias a la información adquirida en la página de CentOS [12] y en la página de Debian [13], los pasos que he seguido yo son:

1. Crear el par clave pública/clave privada en el cliente, en mi caso es mi ordenador, con el comando `ssh-keygen -t rsa`
2. Para mejorar la seguridad de el par de claves generadas, ejecutamos en el cliente el comando `chmod 700 ~/.ssh` y el comando `chmod 600 ~/.ssh/id_rsa`
3. Copiamos la llave pública generada en el servidor. El nombre de usuario de mi servidor es *nestor* y la dirección IP es *192.168.1.135*. Para copiar la llave pública ejecutamos desde el cliente `ssh-copy-id -i ~/.ssh/id_rsa.pub nestor@192.168.1.13`
4. Al igual que hicimos en el cliente, cambiamos los permisos de los archivos en el servidor ejecutando el comando `chmod 700 ~/.ssh` y el comando `chmod 600 ~/.ssh/id_rsa`

Siguiendo estos pasos, podemos conectarnos a nuestro servidor ssh sin tener que introducir las contraseña, como podemos ver en la figura 6.1:

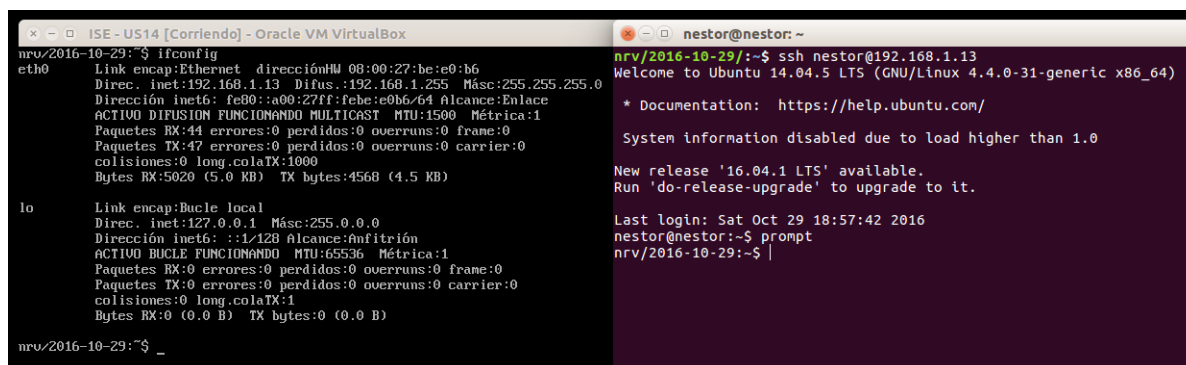
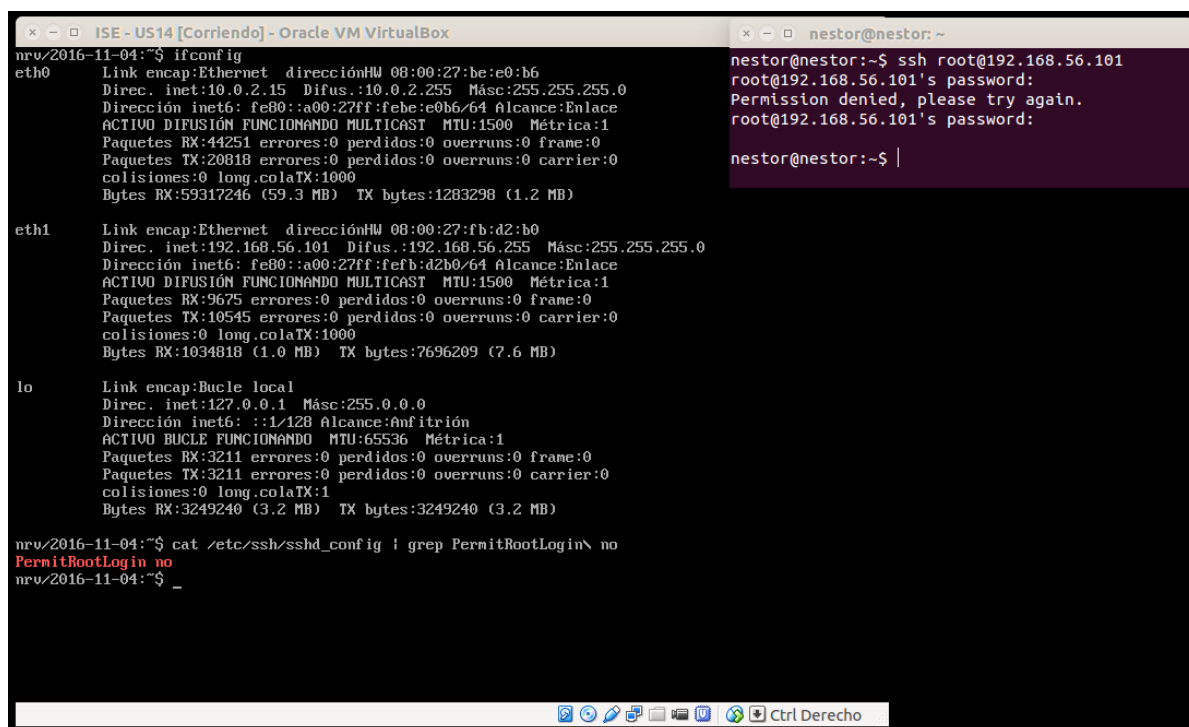


Figura 6.1: Conexión ssh sin contraseña (máquinas conectadas en modo *bridge*).

## 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.

El archivo que tiene la configuración del servicio ssh es `/etc/ssh/sshd_config`. [14]

Como podemos ver en las páginas de manual de `sshd_config` [14] para evitar que el usuario acceda hay que cambiar el parámetro `PermitRootLogin` que se encuentra en el archivo `/etc/ssh/sshd_config`. Inicialmente vemos `PermitRootLogin without-password` y debemos escribir `PermitRootLogin no`. Una vez hecho el cambio, debemos reiniciar el servicio con `sudo service ssh restart`. Como podemos ver en la figura 7.1 no se permite la conexión al usuario root.



```
nrw/2016-11-04:~$ ifconfig
eth0      Link encap:Ethernet  direcciónHW 08:00:27:be:e0:b6
          Direc. inet:10.0.2.15  Difus.:10.0.2.255  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:febe:e0b6/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
          Paquetes RX:44251 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:20818 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colatx:1000
          Bytes RX:59317246 (59.3 MB)  TX bytes:1283298 (1.2 MB)

eth1      Link encap:Ethernet  direcciónHW 08:00:27:fb:d2:b0
          Direc. inet:192.168.56.101  Difus.:192.168.56.255  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:febf:d2b0/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
          Paquetes RX:9675 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:10545 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colatx:1000
          Bytes RX:1034818 (1.0 MB)  TX bytes:7696209 (7.6 MB)

lo        Link encap:Bucle local
          Direc. inet:127.0.0.1  Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Amfitrión
          ACTIVO BUCLE FUNCIONANDO  MTU:65536  Métrica:1
          Paquetes RX:3211 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:3211 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colatx:1
          Bytes RX:3249240 (3.2 MB)  TX bytes:3249240 (3.2 MB)

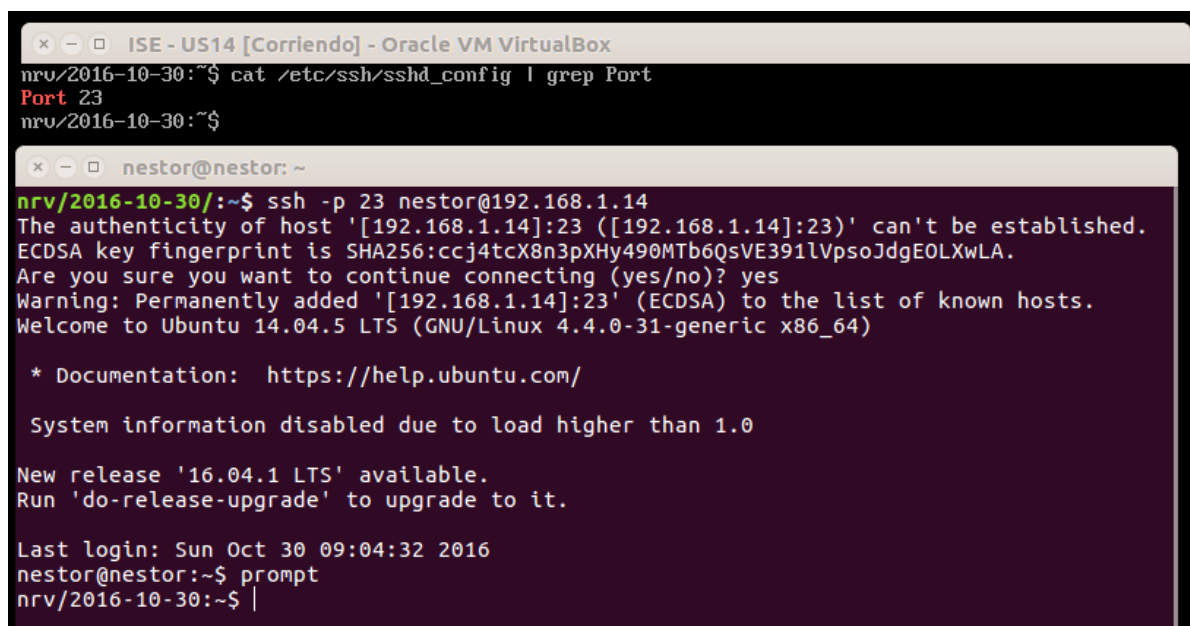
nrw/2016-11-04:~$ cat /etc/ssh/sshd_config | grep PermitRootLogin\ no
PermitRootLogin no
nrw/2016-11-04:~$ _
```

```
nestor@nestor:~$ 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:
nestor@nestor:~$ |
```

Figura 7.1: Conexión no permitida al usuario root (máquinas conectadas en modo *host-only*).

Como podemos ver en las páginas de manual de `sshd_config` [14] para cambiar el puerto por defecto hay que cambiar el parámetro `Port` que se encuentra en `/etc/ssh/sshd_config`. Inicialmente vemos `Port 22` y debemos escribir `Port <puerto_deseado>`. Una vez hecho el cambio, debemos reiniciar el servicio con `sudo service ssh restart`. Ahora, para acceder

debemos indicar el puerto con la opción *-p*. La orden para conectarnos sería *ssh -p puerto usuario@dirección*. Yo he cambiado el puerto 22 por el puerto 23 como podemos ver en la figura 7.2. Puedo seguir conectándome como podemos ver en la figura 7.2.



```
ISE - US14 [Corriendo] - Oracle VM VirtualBox
nrv/2016-10-30:~$ cat /etc/ssh/sshd_config | grep Port
Port 23
nrv/2016-10-30:~$

nestor@nestor: ~
nrv/2016-10-30/~$ ssh -p 23 nestor@192.168.1.14
The authenticity of host '[192.168.1.14]:23 ([192.168.1.14]:23)' can't be established.
ECDSA key fingerprint is SHA256:ccj4tcX8n3pXHy490MTb6QsVE391lVpsoJdgEOLXwLA.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[192.168.1.14]:23' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information disabled due to load higher than 1.0

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

Last login: Sun Oct 30 09:04:32 2016
nestor@nestor:~$ prompt
nrv/2016-10-30:~$ |
```

Figura 7.2: Conexión ssh con el puerto cambiado al puerto 23 (máquinas conectadas en modo *bridge*).

## 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.

Es necesario reiniciar un servicio cada vez que se cambié algo de la configuración de dicho servicio. En caso de que no se reiniciase, los cambios no sería visible hasta el próximo reinicio.

Para reiniciar un servicio hay que ejecutar un comando distinto según si estamos en Ubuntu [15] o en CentOS [16]:

- Ubuntu (ver figura 8.1): *sudo service nombre\_servicio restart*
- CentOS (ver figura 8.2): *systemctl restart nombre\_servicio*

```
nrv/2016-11-03:~$ sudo service ssh restart
[sudo] password for nedor:
ssh stop/waiting
ssh start/running, process 2090
nrv/2016-11-03:~$
```

Figura 8.1: Reinicio del servicio ssh.

```
nrv/2016-11-03:~$ systemctl restart httpd.service
=== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ===
Authentication is required to manage system services or units.
Authenticating as: Nestor Rodriguez Vico (nrv)
Password:
=== AUTHENTICATION COMPLETE ===
nrv/2016-11-03:~$ _
```

Figura 8.2: Reinicio del servicio httpd.

## 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.

Primero vamos a realizar la instalación en Ubuntu de Apache + MySQL + Python + PHP.<sup>1</sup> Los pasos a seguir son:

1. Instalamos apache [17] ejecutando: *sudo apt install apache2*
2. Instalamos MySQL [18] ejecutando: *sudo apt install mysql-server*
3. Instalamos PHP [19] ejecutando: *sudo apt-get install php5 libapache2-mod-php5 php5-mysql*<sup>2</sup>

Para ver si la instalación de Apache ha funcionado correctamente, desde la máquina anfitriona debemos acceder desde un navegador a la dirección IP de nuestro servidor.

---

<sup>1</sup>Python ya viene instalado por defecto en Ubuntu Server, por eso voy a instalar PHP.

<sup>2</sup>php5-mysql es para usar PHP con MySQL.



En la figura 9.1 podemos ver que la dirección IP es *192.168.1.19* y que funciona correctamente, ya que podemos acceder desde un navegador en la máquina anfitrión.<sup>3</sup>

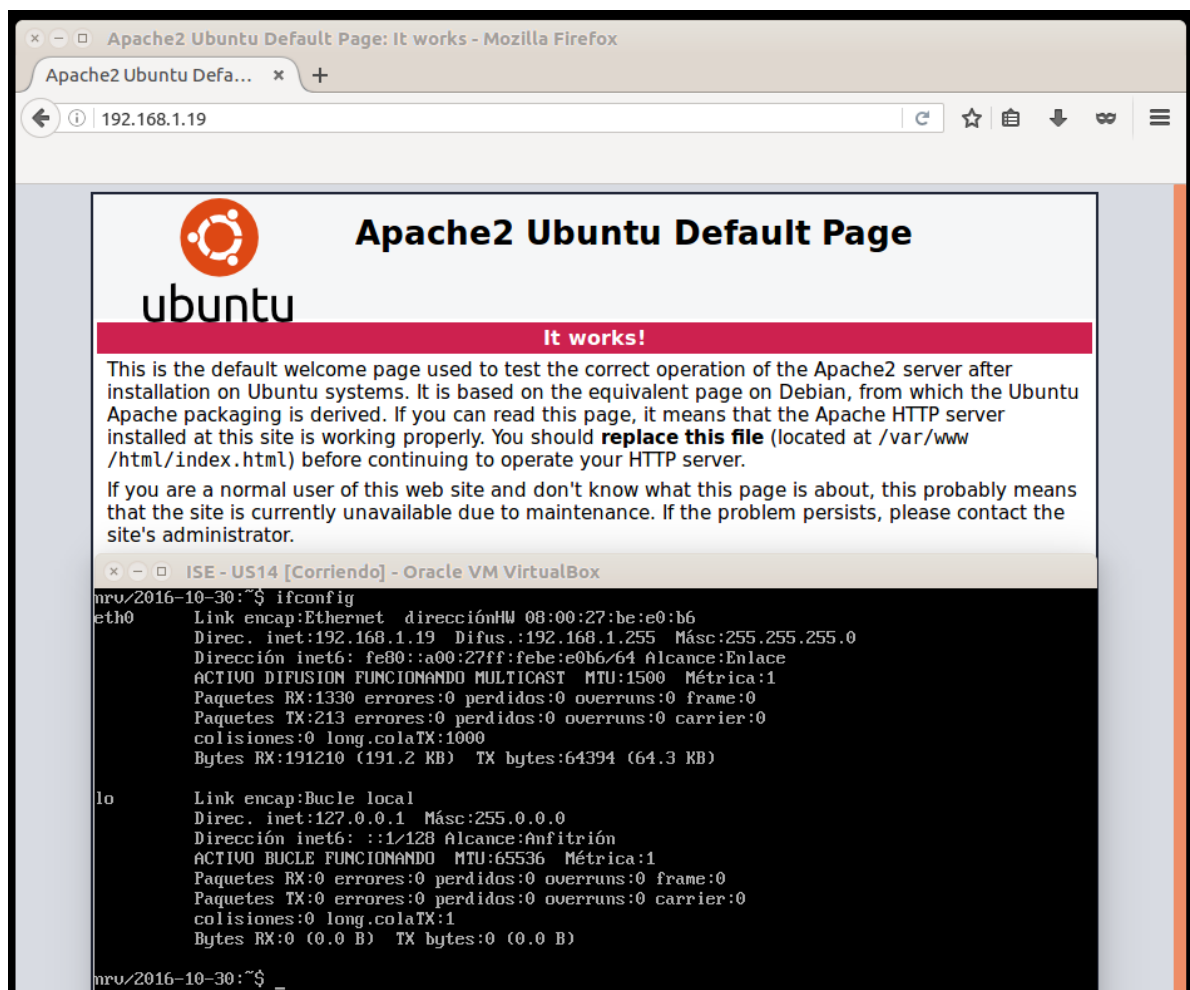


Figura 9.1: Servidor web en funcionamiento (máquinas conectadas en modo *bridge*).

Para ver si la instalación de php ha funcionado correctamente, he creado un archivos en */var/www/html* y he accedido desde mi máquina anfitriona, como se puede ver en la figura 9.2.<sup>4</sup>

<sup>3</sup>Para que funcione correctamente he tenido que habilitar el puerto 80 y el 8080 con *ufw*.

<sup>4</sup>Para poder acceder desde la máquina anfitriona a Ubuntu Server he tenido que levantar la interfaz de red *eth1*, para ello he ejecutado el comando *sudo ifconfig eth1* seguido de *sudo dhclient*. Doy por hecho que de aquí en adelante cada vez que necesite acceder a Ubuntu Server desde mi máquina anfitriona estos dos comandos han sido ejecutados previamente.

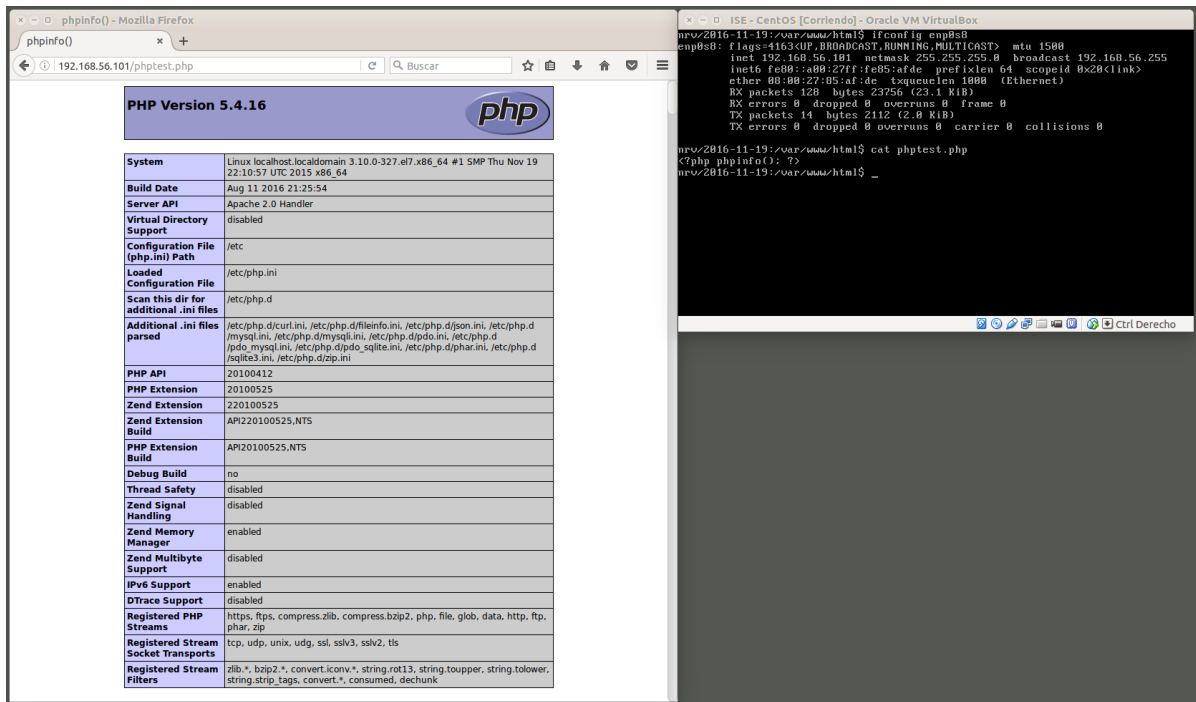


Figura 9.2: php en funcionamiento (máquinas conectadas en modo *host-only*).

Para ver si la instalación de MySQL ha funcionado correctamente, debemos ejecutar el comando `mysql -u root -p` e introducir la contraseña que se usó para la instalación. Una vez hecho esto, se accede a la consola de MySQL, como se puede ver en la figura 9.3.

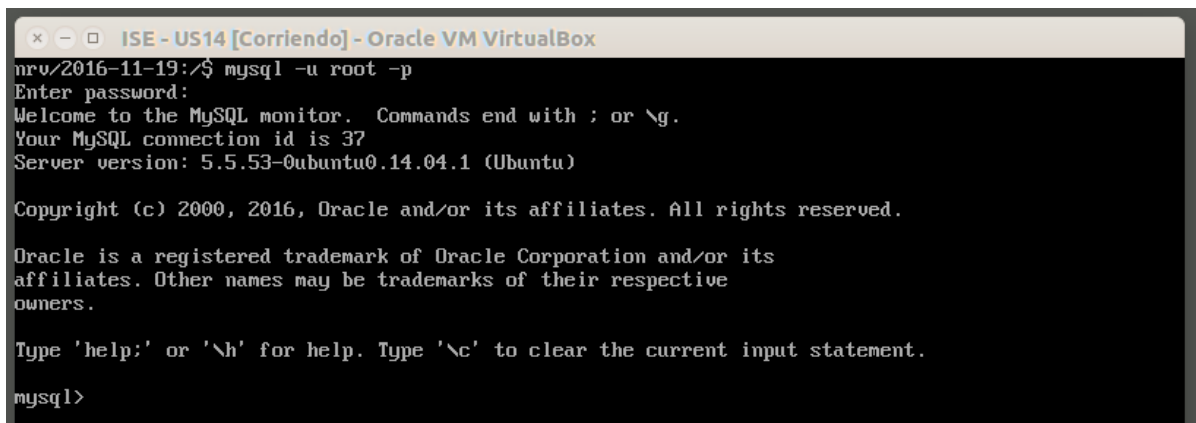


Figura 9.3: mysql en funcionamiento (máquinas conectadas en modo *host-only*).

A continuación vamos a realizar la instalación en CentOS de Apache + MySQL + Python. Los pasos a seguir son [20]:

1. Instalamos Apache ejecutando: `sudo yum install httpd`  
Una vez instalado Apache, debemos arrancar el servidor. Para ello ejecutamos: `sudo systemctl start httpd.service`

2. Instalamos MySQL (MariaDB <sup>5</sup>) con: `sudo yum install mariadb-server mariadb`  
Una vez instalado MariaDB, debemos arrancar el servidor. Para ello ejecutamos: `sudo systemctl start mariadb`
3. Instalamos PHP ejecutando: `sudo yum install php php-mysql`  
Una vez instalado PHP, debemos arrancar el servidor. Para ello ejecutamos: `sudo systemctl start httpd.service`

Para ver si la instalación de Apache ha funcionado correctamente, desde la máquina anfitriona debemos acceder desde un navegador a la dirección IP de nuestro servidor. En la figura 9.4 podemos ver que la dirección IP es *192.168.1.21* y que funciona correctamente, ya que podemos acceder desde un navegador en la máquina anfitrión. <sup>6</sup>



Figura 9.4: Servidor web en funcionamiento (CentOS) (máquinas conectadas en modo *bridge*).

Para ver si la instalación de php ha funcionado correctamente, he creado un archivos en `/var/www/html` y he accedido desde mi máquina anfitriona, como se puede ver en la figura 9.5.

<sup>5</sup>MariaDB es un reemplazo de MySQL.

<sup>6</sup>Para que funcione correctamente he tenido que habilitar el puerto 80 y el 8080 con `firewall-cmd`.

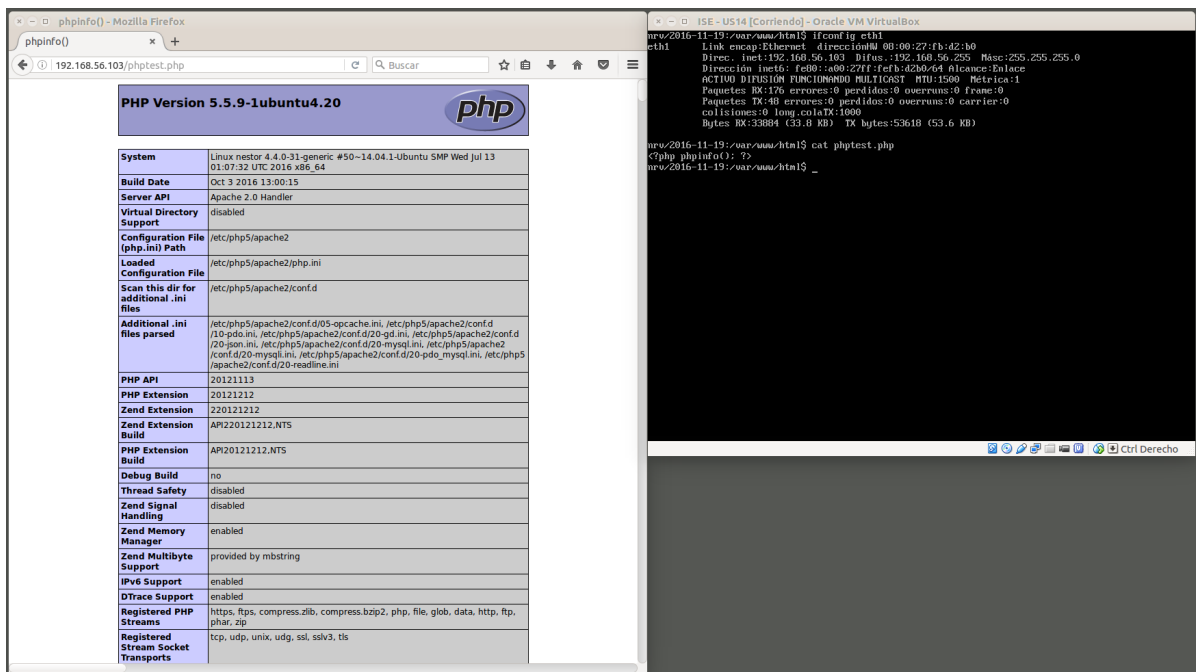


Figura 9.5: php en funcionamiento (máquinas conectadas en modo *host-only*).

Para ver si la instalación de MySQL ha funcionado correctamente, debemos ejecutar el comando `mysql -u root -p`. Cuando nos pida una contraseña, no introducimos ninguna. Una vez hecho esto, se accede a la consola de MariaDB, que como ya dije antes, es un reemplazo de MySQL, como se puede ver en la figura 9.6.

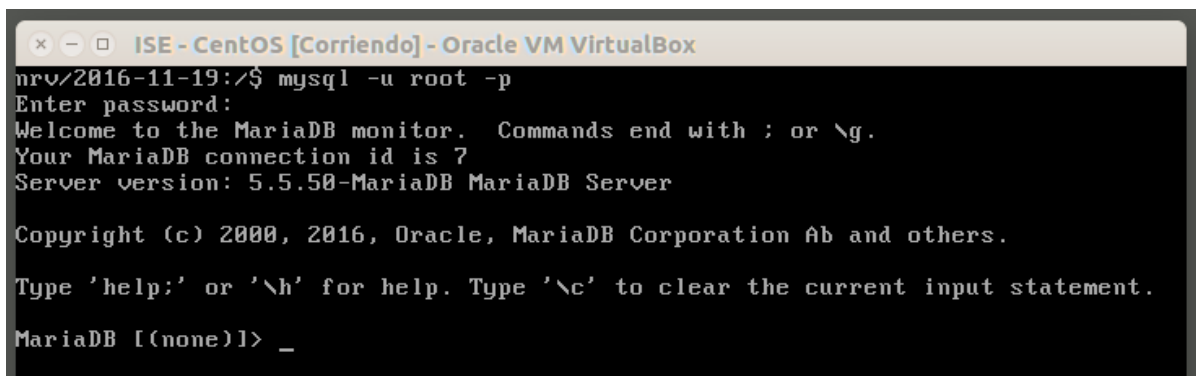


Figura 9.6: mysql en funcionamiento (máquinas conectadas en modo *host-only*).

## 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 anfitrióna.

Para ver si ha funcionado correctamente, desde la máquina anfitrióna debemos acceder desde un navegador a la dirección IP de nuestro servidor. En la figura 10.1 podemos ver que la dirección IP es *192.168.1.20* y que funciona correctamente, ya que podemos acceder desde un navegador en la máquina anfitrión.



Figura 10.1: Servidor web en funcionamiento (Windows Server).

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

Como podemos ver en la página de Fedora [21] un ejemplo de uso del comando sería `patch -p0 -i /tmp/vmware-netfilter.patch`

Como podemos ver en las páginas de manual del comando `patch` [22]:

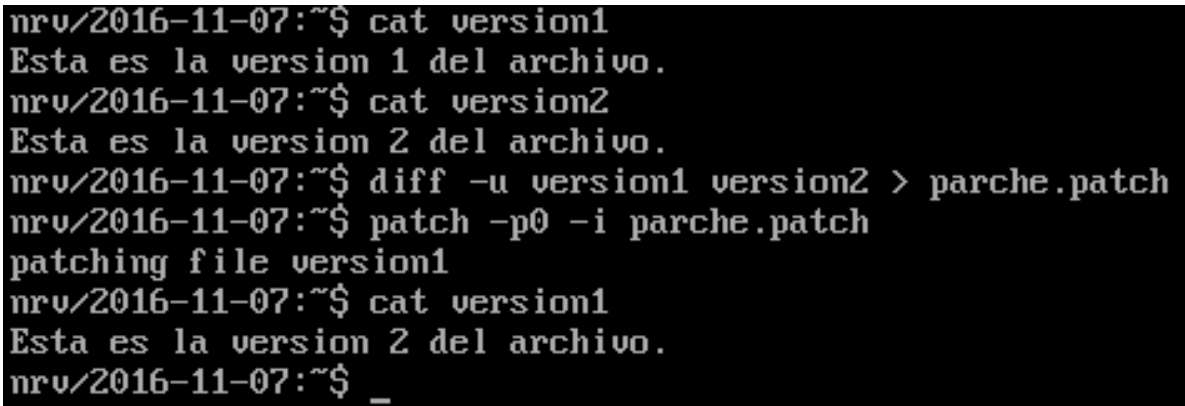
- El argumento `-p0` sirve para dar el nombre del fichero sin modificar.
- El argumento `-i` sirve para indicar que debe leer el parche del archivo que tiene a continuación, `/tmp/vmware-nerfilter.patch` en este caso.

Para mostrar como funciona en la práctica, voy a realizar un ejemplo pequeño. Ya que no tengo ningún archivo que parchear ni ningún parche, voy a inventármelos yo.

1. Creamos dos archivos, *version1* y *version2*.

2. Vemos las diferencias de ambos archivos y creamos el parche a aplicar ejecutando `diff -u version1 version2 > parche.patch`
3. Parcheamos el archivo `version1` ejecutando `patch -p0 -i parche.patch`

Todo este proceso se puede ver en la figura 11.1.



```
nrv/2016-11-07:~$ cat version1
Esta es la version 1 del archivo.
nrv/2016-11-07:~$ cat version2
Esta es la version 2 del archivo.
nrv/2016-11-07:~$ diff -u version1 version2 > parche.patch
nrv/2016-11-07:~$ patch -p0 -i parche.patch
patching file version1
nrv/2016-11-07:~$ cat version1
Esta es la version 2 del archivo.
nrv/2016-11-07:~$ _
```

Figura 11.1: Proceso de “parcheo” con el comando `patch`.

## 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.

El proceso de instalación lo podemos ver en la página oficial de Webmin [23]. Los pasos a seguir son:

1. Editar el archivo `/etc/apt/sources.list` y añadir siguiente línea, cómo podemos ver en la figura 12.1.  
`deb http://download.webmin.com/download/repository sarge contrib`
2. Obtenemos la clave GPG con la cual el repositorio está firmado y la instalamos. Para ellos ejecutamos los siguientes comandos en modo `root` como podemos ver en la figura 12.2:
  - a) `cd /root`
  - b) `wget http://www.webmin.com/jcameron-key.asc`
  - c) `apt-key add jcameron-key.asc`
3. Actualizamos los repositorios con `sudo apt update` e instalar Webmin con el comando `sudo apt install webmin`, como podemos ver en la figura 12.3.

```

nrv/2016-10-30:~$ cat /etc/apt/sources.list | grep webmin
deb http://download.webmin.com/download/repository sarge contrib
nrv/2016-10-30:~$ _

```

Figura 12.1: Repositorio añadido.

```

nrv/2016-10-30:/$
nrv/2016-10-30:/$ sudo su
root@nestor:~# cd root/
root@nestor:~# wget http://www.webmin.com/jcameron-key.asc
--2016-10-30 13:31:53-- 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.1"

100%[=====>] 1.320 --.-K/s en 0s

2016-10-30 13:31:54 (9,35 MB/s) - "jcameron-key.asc.1" guardado [1320/1320]

root@nestor:~# apt-key add jcameron-key.asc
OK
root@nestor:~# exit
exit
nrv/2016-10-30:/$ _

```

Figura 12.2: Obtención e instalación de la clave GPG.

```

nrv/2016-10-30:/$
nrv/2016-10-30:/$ sudo apt update && sudo apt install webmin_

```

Figura 12.3: Actualizamos los repositorios e instalamos Webmin.

Una vez hemos instalado Webmin, vamos a proceder a cambiar algunos de los parámetros de nuestro servidor. Como podemos ver en las *faq* de Webmin [24] debemos acceder desde el navegador a la dirección IP de nuestro servidor pero accediendo al puerto 10000. Puede ser que esto nos de problemas, en caso de que sea así, en vez de acceder a mediante *http* debemos acceder mediante *https*. En la figura 12.4 podemos ver que la dirección IP de mi servidor es *192.168.1.29*, por lo tanto debemos acceder a través de la dirección *https://192.168.1.29:10000*. Una vez en la página, los datos para acceder son el nombre de usuario de nuestro servidos y la contraseña del mismo.

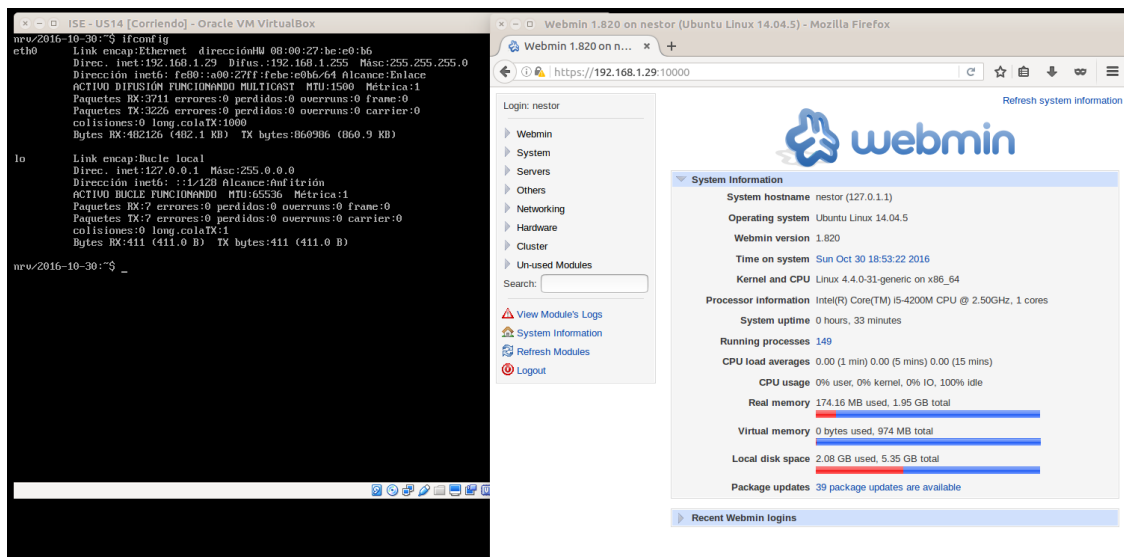


Figura 12.4: Webmin en funcionamiento (máquinas conectadas en modo *bridge*).

En el menú de la izquierda hay múltiples opciones que nos permiten realizar distintas tareas. Hay dos de ellas que me parecen realmente interesantes. La primera se encuentra dentro del menú *System*, en la sección *Bootup and Shutdown*. Esta sección nos permite manipular los servicios que se están ejecutando en nuestro servidor, permitiéndonos, por ejemplo, parar un servicio, reiniciarlo e incluso elegir si debe iniciarse o no al encender el sistema. Como podemos ver en la figura 12.5, podría parar mi servidor ssh.

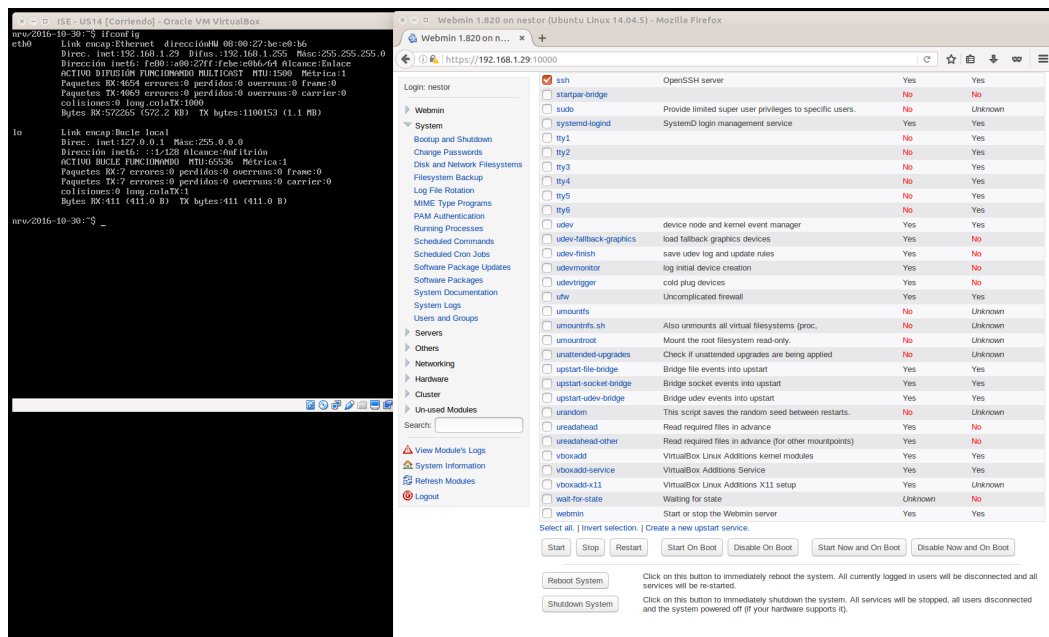


Figura 12.5: Manipulación de servicios con Webmin (máquinas conectadas en modo *bridge*).



La segunda característica se encuentra dentro del menú *Others*, en la sección *Command Shell*. Esta sección nos permite ejecutar comandos en nuestro servidor desde Webmin como podemos ver en la figura <sup>7</sup> 12.6.

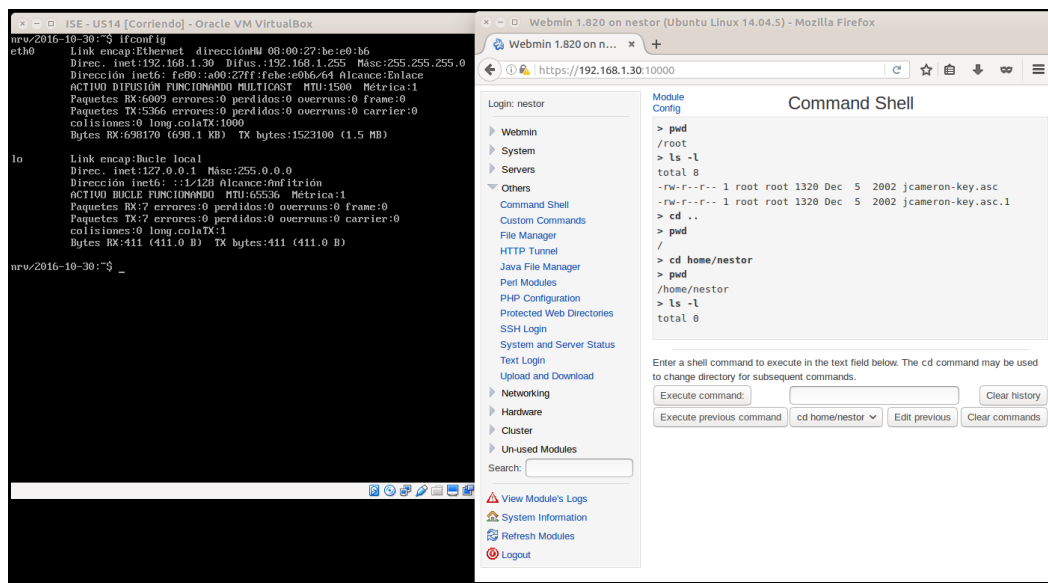


Figura 12.6: Ejecución de comandos mediante Webmin (máquinas conectadas en modo *bridge*).

### 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 vamos a seguir los pasos que podemos ver en la página de Ubuntu, tanto en la documentación oficial [25] como en la Wiki de la comunidad [26]:

1. Primero instalamos phpMyAdmin con el comando `sudo apt install phpmyadmin`. Cuando nos pregunte que servidor web vamos a usar, elegimos `apache2`.
2. Ahora tenemos que modificar el archivo `/etc/apache2/apache2.conf` y añadir la línea `Include /etc/phpmyadmin/apache.conf`. El resultado lo podemos ver en la figura 13.1.

<sup>7</sup>La dirección IP de mi servidor ha cambiado, ahora es `192.168.1.30` como se puede ver en la figura.

3. Ahora podemos acceder desde nuestra máquina anfitriona escribiendo en el navegador la dirección IP de nuestro servidor seguido de `/phpmyadmin`. En mi caso, la dirección IP es `192.168.1.31` así que para entrar debemos escribir en la barra del navegador `http://192.168.1.32/phpmyadmin`. Podemos entrar perfectamente, como vemos en la figura 13.2. Una vez en la página, el nombre de usuario para acceder es `root` y la contraseña es la de nuestro servidor.

```
nrv/2016-10-30:~$ cat /etc/apache2/apache2.conf | grep phpmyadmin
Include /etc/phpmyadmin/apache.conf
nrv/2016-10-30:~$
```

Figura 13.1: Archivo modificado.

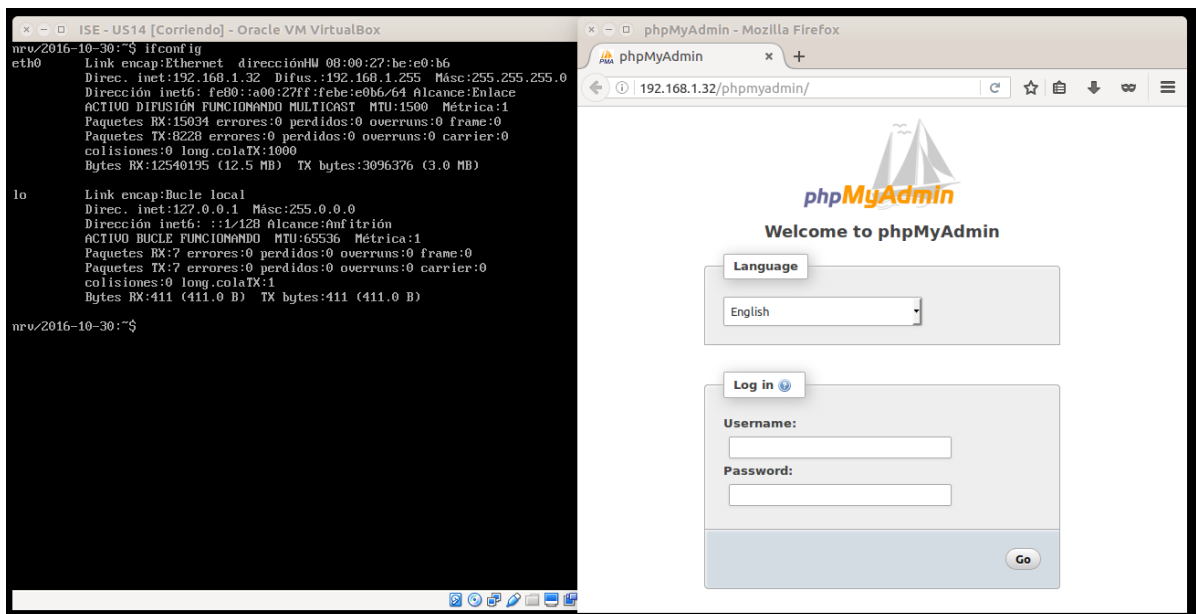


Figura 13.2: phpMyAdmin funcionando (máquinas conectadas en modo *bridge*).

El proceso para cambiar el tamaño máximo de las bases de datos lo podemos ver en la página de phpMyAdmin [27]. Tenemos que modificar el archivo `/etc/php5/apache2/php.ini`:

- Debemos cambiar el valor de `upload_max_filesize`. El valor que trae por defecto es de 2M, nosotros lo cambiamos por 25M como podemos ver en la figura 13.3.
- Debemos cambiar el valor de `post_max_size`. El valor que trae por defecto es de 8M, nosotros lo cambiamos a 25M como podemos ver en la figura 13.4.

```

nrv/2016-10-30:~$ cat /etc/php5/apache2/php.ini | grep upload_max_filesize
upload_max_filesize = 25M
nrv/2016-10-30:~$

```

Figura 13.3: upload\_max\_filesize con el nuevo valor.

```

nrv/2016-10-30:~$ cat /etc/php5/apache2/php.ini | grep post_max_size
post_max_size = 25M
nrv/2016-10-30:~$

```

Figura 13.4: post\_max\_size con el nuevo valor.

Una vez hemos aplicado los cambios, debemos reiniciar el servicio apache2 con el comando `sudo service apache2 restart`. Si ahora nos dirigimos a *phpMyAdmin* desde el navegador e intentamos subir una archivo, podemos ver que el tamaño máximo permitido es 25MiB, tal y cómo se ve en la figura 13.5.

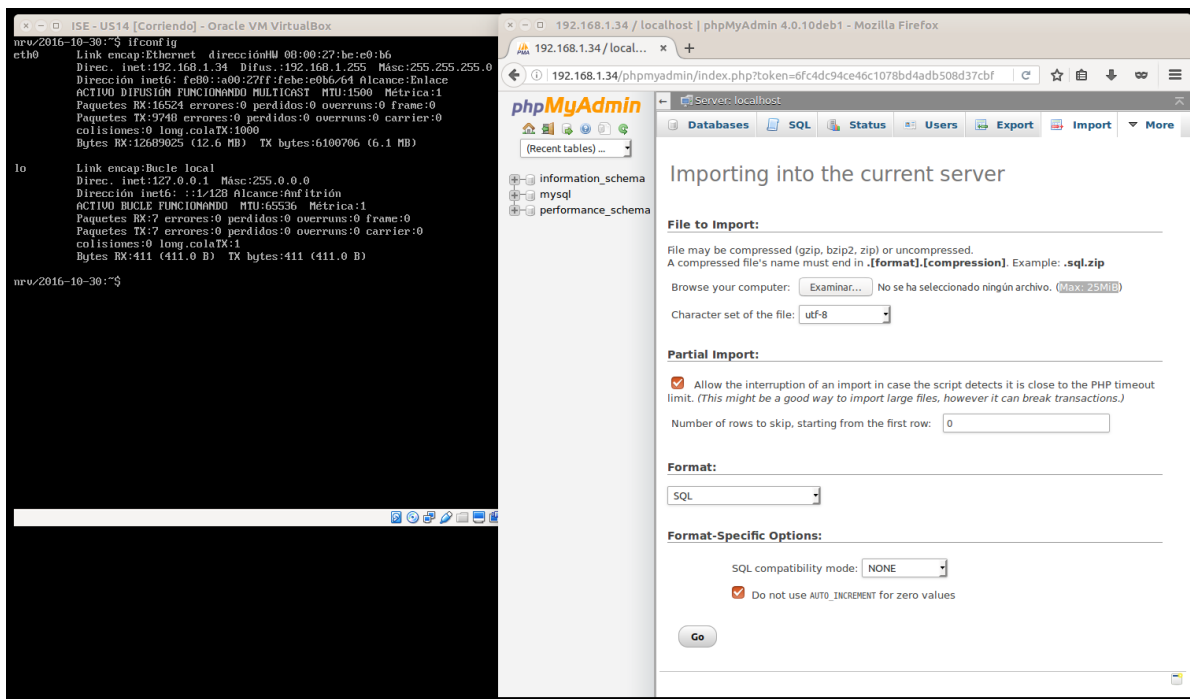


Figura 13.5: El tamaño máximo permitido es 25MiB (máquinas conectadas en modo *bridge*).

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

Yo voy a probar Parallels Plesk [?]. En la página de Parallels Plesk podemos probar una demo para ver las opciones que tiene. Lo primero que vemos al acceder es lo que se muestra en la figura 14.1.

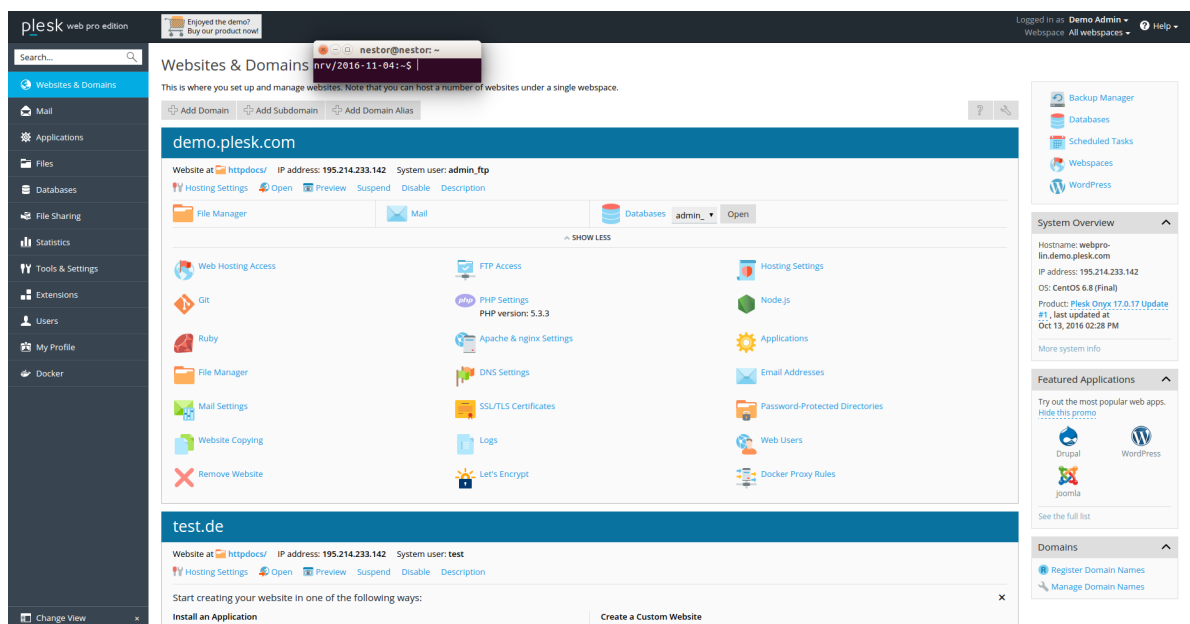


Figura 14.1: Inicio Parallels Plesk.

Cómo podemos ver, se pueden hacer muchas cosas distintas. Las menús más interesantes para mí son *Files*, *Statistics* y *Tools & Settings*. Veamos cada una de ellas más en profundidad.

El menú *Statistics*, como podemos ver en la figura 14.2, nos permite ver estadísticas de nuestro servidor, como puede ser el ancho de banda consumido o el espacio en disco usado.

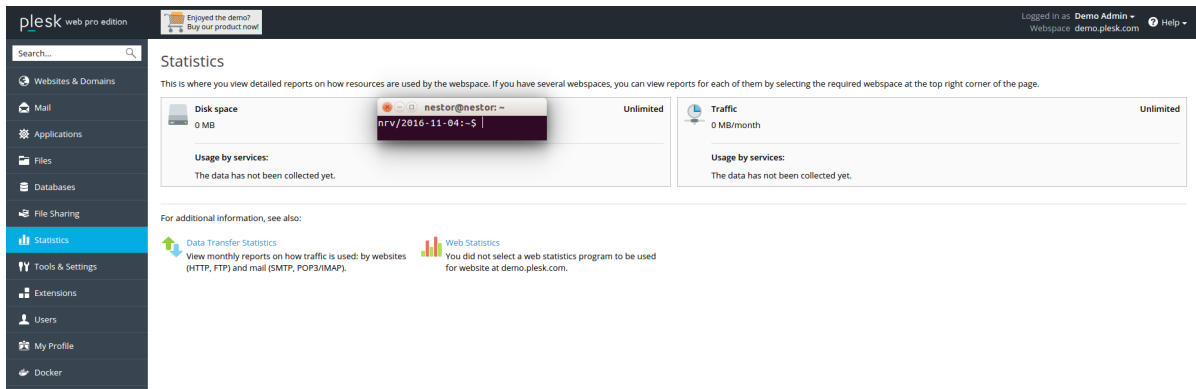


Figura 14.2: Estadísticas del servidor desde Parallels Plesk.

El menú *Tools & Settings*, como podemos ver en la figura 14.3, nos permite configurar y usar distintas opciones para nuestro servidor, como podría ser adquirir un certificado SSL, activar un filtro de Spam para el correo y obtener información del servidor.

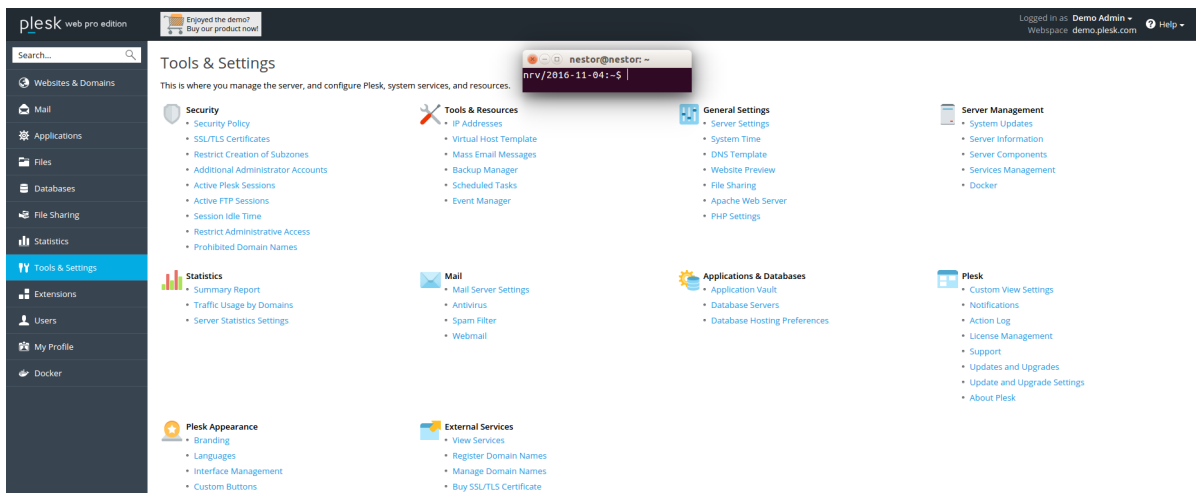
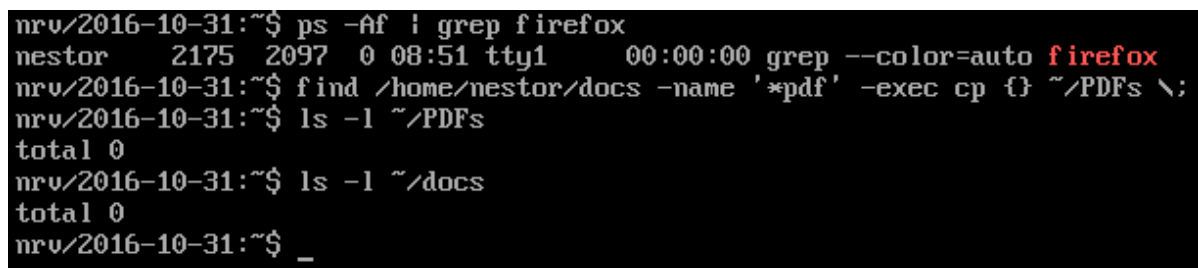


Figura 14.3: Ajustes y herramientas del servidor desde Parallels Plesk.

## 15. Cuestión 15:

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



```
nrv/2016-10-31:~$ ps -Af | grep firefox
nestor    2175  2097  0 08:51 tty1      00:00:00 grep --color=auto firefox
nrv/2016-10-31:~$ find /home/nestor/docs -name '*pdf' -exec cp {} ~/PDFs \;
nrv/2016-10-31:~$ ls -l ~/PDFs
total 0
nrv/2016-10-31:~$ ls -l ~/docs
total 0
nrv/2016-10-31:~$ _
```

Figura 15.1: Ejecución de grep y find.

La ejecución del comando *find* no proporciona ningún resultado, porque como podemos ver en la figura 15.1 no había ningún pdf en */home/nestor/docs*.

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

Para ver el funcionamiento de *sed* podemos visitar las páginas de manual del comando [28]. Una vez sabemos como funciona, el resultado lo podemos ver en el script <sup>8</sup> 15.2.

```
#!/bin/bash/
sed -i 's/.*PasswordAuthentication no.*/PasswordAuthentication yes/'
    /etc/ssh/sshd_config
service ssh restart
sleep 180
sed -i 's/.*PasswordAuthentication yes.*/PasswordAuthentication no/'
    /etc/ssh/sshd_config
service ssh restart
```

---

<sup>8</sup>El script se encuentra dentro de la carpeta *Archivos auxiliares*.

### 15.3. c) Muestre un ejemplo de uso para awk.

```
nrν/2016-11-01:~$ cat pruebaawk.txt
L 10 11
M 12 13
X 14 15
J 16 17
V 18 19
S 20 21
D 22 23
L 24 25
M 26 27
X 28 29
J 30 31
V 32 33
S 34 35
D 36 37
L 38 39
nrν/2016-11-01:~$ awk '/L/ { print $0 }' pruebaawk.txt
L 10 11
L 24 25
L 38 39
nrν/2016-11-01:~$
```

Figura 15.2: Ejecución de awk.

Con este comando hemos conseguido mostrar por pantalla sólo las líneas correspondientes a los lunes, es decir, los que empiezan por *L*.

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

El resultado obtenido lo podemos ver en el script <sup>9</sup> 16.

```
#!/usr/bin/python
import subprocess
subprocess.call(['sudo sed -i 's/.*/PasswordAuthentication_no.*/
    PasswordAuthentication_yes/' sshd_config'], shell=True)
subprocess.call(['sudo service ssh restart'], shell=True)
```

---

<sup>9</sup>El script se encuentra dentro de la carpeta *Archivos auxiliares*.

```
subprocess.call(['sleep 180'], shell=True)
subprocess.call(['sudo sed -i 's/.*/PasswordAuthentication_yes.*/
    PasswordAuthentication_no/' sshd_config'], shell=True)
subprocess.call(['sudo service ssh restart'], shell=True)
```

## 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.

Para ver que comandos debo usar, he visitado la página de Microsoft [29].

Los pasos para parar un programa en ejecución son:

1. Con el comando *Get-Process* obtenemos un listado de los procesos en ejecución, como podemos ver en la figura 17.1.
2. Con el comando *Stop-Process -Name VBoxService* paramos el proceso *VBoxService*, como podemos ver en la figura 17.2. Nos pedirá una confirmación, así que le decimos que sí estamos seguro.
3. Podemos comprobar que se ha parado correctamente ejecutando de nuevo el comando *Get-Process*, como podemos ver en la figura 17.3

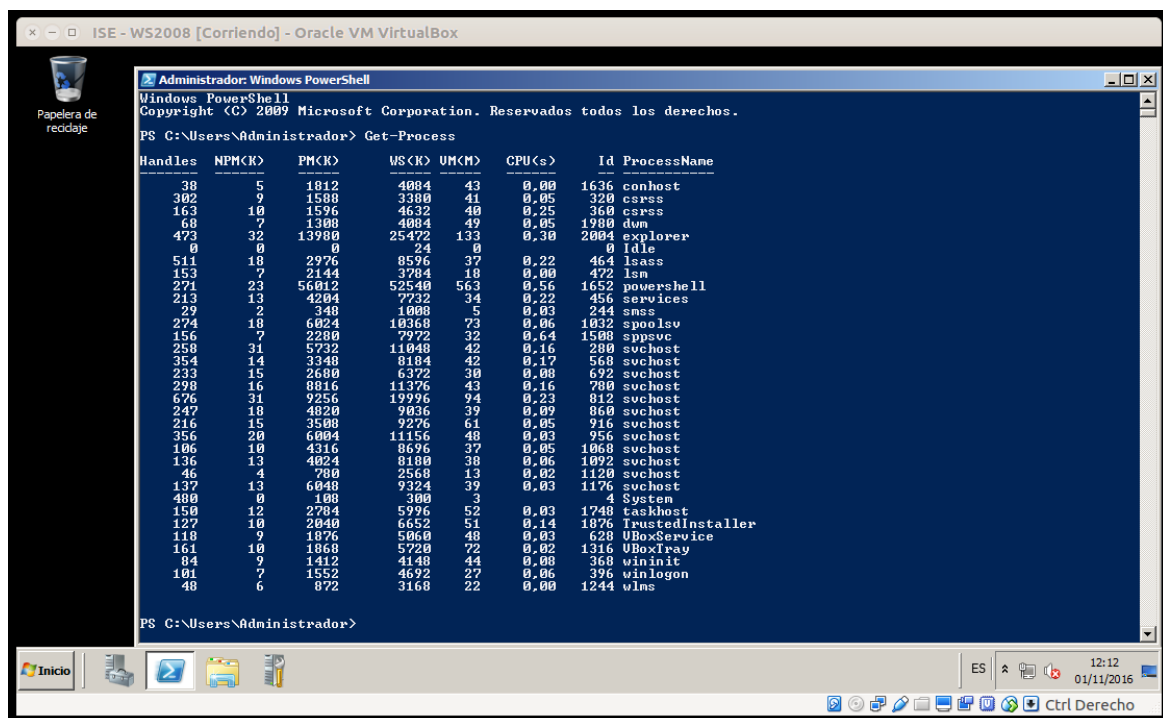


Figura 17.1: Procesos activos.



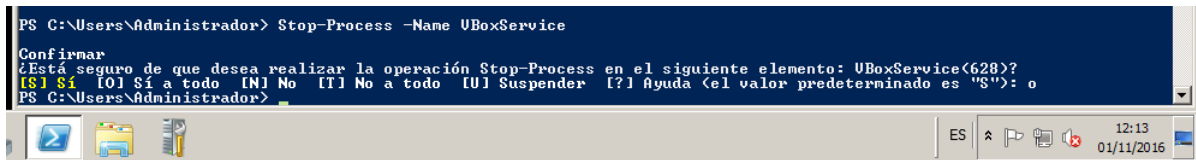


Figura 17.2: Paramos el servicio VBoxService.

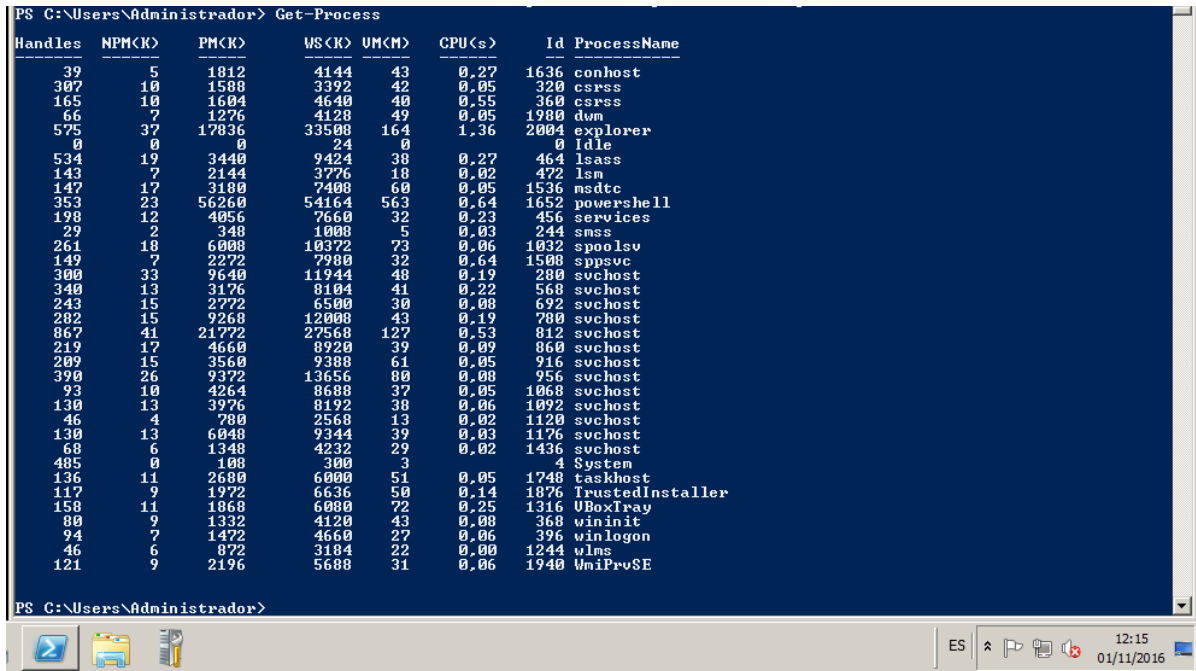


Figura 17.3: Procesos activos tras parar el proceso VBoxService.

## 18. Cuestión opcional 1: Instale y pruebe terminator y/o tmux. Con screen, pruebe su funcionamiento dejando sesiones ssh abiertas en el servidor y recuperándolas posteriormente.

Para instalar *terminator* debemos ejecutar `sudo apt install terminator`. Para instalar *screen* debemos ejecutar `sudo apt install screen`. Aprender a usar *terminator*, a un nivel suficiente para la realización de este ejercicio, es fácil. Pero para aprender a usar *screen* he consultado las páginas del manual [30].

1. Lo primero que he hecho ha sido crear tres terminales en *terminator*, dos para usar *screen* y realizar dos conexiones ssh a mi servidor y una tercera para comprobar el estado de las dos anteriores. En las dos terminales superiores que se ven en la figura 18.1 he abierto dos sesiones de *screen*. Para ello he ejecutado `screen -S ssh1` en la terminal de la izquierda y `screen -S ssh2` en la terminal de la derecha. Para corroborar que estamos en *screen*, he ejecutado en ambas `echo $TERM` y efectivamente podemos ver que estamos en *screen* mientras que en la terminal inferior no. También podemos ver en la terminal inferior como hay dos sesiones de *screen* abiertas. En las terminales superiores he realizado dos conexiones ssh, y he ejecutado dos comandos para ver que, efectivamente, funcionan. Todo este proceso se puede ver en la figura 18.1.
2. A continuación, cerramos *terminator*. Lo abrimos de nuevo y creamos las mismas tres terminales. Podemos comprobar que no estamos en una sesión de *screen* ejecutando `echo $TERM` y vemos que no estamos en una sesión de *screen*. Para restaurar las sesiones, ejecutamos `screen -r 9417.ssh2` y `screen -r 9403.ssh1`, como podemos ver en la figura 18.2.
3. Tras ejecutar los dos comandos anteriores, podemos ver que se restauran las sesiones de *screen* y las conexiones ssh. Una vez más ejecutamos `echo $TERM` para ver que, correctamente, estamos en las sesiones de *screen*. Todo este proceso lo podemos ver en la figura 18.3.
4. Para cerrar la sesión de *screen* ejecutamos `exit` dos veces, la primera de ellas para cerrar la conexión ssh y la segunda para cerrar la conexión de *screen*. Finalmente, ejecutamos de nuevo `echo $TERM` para ver que no estamos en una sesión de *screen*. En la terminal inferior ejecutamos `screen -ls` para ver que no queda ninguna sesión de *screen* abierta. Todo lo comentado podemos verlo en la figura 18.4.

```
nestor@nestor:~ 77x29
nrv/2016-11-07:~$ echo $TERM
screen
nrv/2016-11-07:~$ ssh 192.168.56.101
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information as of Mon Nov  7 10:28:59 CET 2016

System load:  0.0      Processes:    152
Usage of /home: 0.7% of 451MB   Users logged in:  1
Memory usage:  12%      IP address for eth0: 10.0.2.15
Swap usage:    0%       IP address for eth1: 192.168.56.101

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

43 packages can be updated.
28 updates are security updates.

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

Last login: Mon Nov  7 10:28:59 2016 from 192.168.56.1
nestor@nestor:~$ prompt
nrv/2016-11-07:~$ pwd
/home/nestor
nrv/2016-11-07:~$

nestor@nestor:~ 76x29
nrv/2016-11-07:~$ echo $TERM
screen
nrv/2016-11-07:~$ ssh 192.168.56.101
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information as of Mon Nov  7 10:31:37 CET 2016

System load:  0.0      Processes:    148
Usage of /home: 0.7% of 451MB   Users logged in:  1
Memory usage:  12%      IP address for eth0: 10.0.2.15
Swap usage:    0%       IP address for eth1: 192.168.56.101

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

43 packages can be updated.
28 updates are security updates.

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

Last login: Mon Nov  7 10:31:37 2016 from 192.168.56.1
nestor@nestor:~$ prompt
nrv/2016-11-07:~$ ls
prueba
nrv/2016-11-07:~$

nestor@nestor:~ 155x23
nrv/2016-11-07:~$ screen -ls
There are screens on:
  9417.ssh2      (07/11/16 10:30:47)  (Attached)
  9403.ssh1      (07/11/16 10:30:41)  (Attached)
2 Sockets in /var/run/screen/S-nestor.

nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$
```

Figura 18.1: Creación de las sesiones de *screen*.

```
nestor@nestor:~ 70x21
nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$ screen -r 9417.ssh2

nestor@nestor:~ 69x21
nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$ screen -r 9403.ssh1

nestor@nestor:~ 141x20
nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$ screen -ls
There are screens on:
  9417.ssh2      (07/11/16 10:30:47)  (Detached)
  9403.ssh1      (07/11/16 10:30:41)  (Detached)
2 Sockets in /var/run/screen/S-nestor.

nrv/2016-11-07:~$
```

Figura 18.2: Restauración de las sesiones de *screen*.

```
nestor@nestor: ~
nestor@nestor: ~ 70x30
nrv/2016-11-07:~$ echo $TERM
screen
nrv/2016-11-07:~$ ssh 192.168.56.101
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information as of Mon Nov  7 10:28:59 CET 2016

System load:   0.0      Processes:    152
Usage of /home: 0.7% of 451MB   Users logged in: 1
Memory usage:  12%      IP address for eth0: 10.0.2.15
Swap usage:    0%        IP address for eth1: 192.168.56.101

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

43 packages can be updated.
28 updates are security updates.

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

Last login: Mon Nov  7 10:28:59 2016 from 192.168.56.1
nestor@nestor:~$ prompt
nrv/2016-11-07:~$ pwd
/home/nestor
nrv/2016-11-07:~$ echo $TERM
screen
nrv/2016-11-07:~$

nestor@nestor: ~ 69x30
nrv/2016-11-07:~$ echo $TERM
screen
nrv/2016-11-07:~$ ssh 192.168.56.101
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

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

System information as of Mon Nov  7 10:31:37 CET 2016

System load:   0.0      Processes:    148
Usage of /home: 0.7% of 451MB   Users logged in: 1
Memory usage:  12%      IP address for eth0: 10.0.2.15
Swap usage:    0%        IP address for eth1: 192.168.56.101

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

43 packages can be updated.
28 updates are security updates.

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

Last login: Mon Nov  7 10:31:37 2016 from 192.168.56.1
nestor@nestor:~$ prompt
nrv/2016-11-07:~$ ls
prueba
nrv/2016-11-07:~$ echo $TERM
screen
nrv/2016-11-07:~$

nestor@nestor: ~ 141x12
nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$ screen -ls
There are screens on:
  9417.ssh2      (07/11/16 10:30:47)  (Detached)
  9403.ssh1      (07/11/16 10:30:41)  (Detached)
2 Sockets in /var/run/screen/S-nestor.

nrv/2016-11-07:~$
```

Figura 18.3: Sesiones de *screen* restauradas.

```
nestor@nestor: ~
nestor@nestor: ~ 70x25
nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$ screen -r 9403.ssh1
[screen is terminating]
nrv/2016-11-07:~$

nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$

nestor@nestor: ~ 69x25
nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$ screen -r 9417.ssh2
[screen is terminating]
nrv/2016-11-07:~$

nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$

nestor@nestor: ~ 141x21
nrv/2016-11-07:~$ echo $TERM
xterm
nrv/2016-11-07:~$ screen -ls
There are screens on:
  9417.ssh2      (07/11/16 10:30:47)  (Detached)
  9403.ssh1      (07/11/16 10:30:41)  (Detached)
2 Sockets in /var/run/screen/S-nestor.

nrv/2016-11-07:~$ screen -ls
No Sockets found in /var/run/screen/S-nestor.

nrv/2016-11-07:~$
```

Figura 18.4: Sesiones de *screen* cerradas.

## 19. Cuestión opcional 2: Instale el servicio y pruebe su funcionamiento.

Para ver el funcionamiento de *fail2ban* podemos visitar la página oficial de dicho servicio [31]. Como página complementaria voy a usar Digital Ocean [32]. Para instalar *fail2ban* debemos ejecutar el comando `sudo apt install fail2ban`. Una vez instalado, lo recomendable es copiar el archivo `/etc/fail2ban/jail.conf` para evitar problemas en futuras actualizaciones. Para ello, ejecutamos el comando `sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local`, como podemos ver en la figura 19.1.

```
nrv/2016-11-05:~$ sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
[sudo] password for nesor:
nrv/2016-11-05:~$ ls -l /etc/fail2ban/ | grep jail
-rw-r--r-- 1 root root 11885 nov 18 2013 jail.conf
drwxr-xr-x 2 root root 4096 nov 18 2013 jail.d
-rw-r--r-- 1 root root 11885 nov 5 20:20 jail.local
nrv/2016-11-05:~$
```

Figura 19.1: Copia del archivo `/etc/fail2ban/jail.conf`.

Para ver su funcionamiento, voy a intentar conectarme a mi servidor *ssh* pero fallaré la contraseña para ver que *fail2ban* funciona correctamente. Como en mi máquina anfitriona tengo configurado *ssh* para permitir el acceso sin contraseña, voy a realizar el intento de acceso desde CentOS, como podemos ver en la figura 19.2.

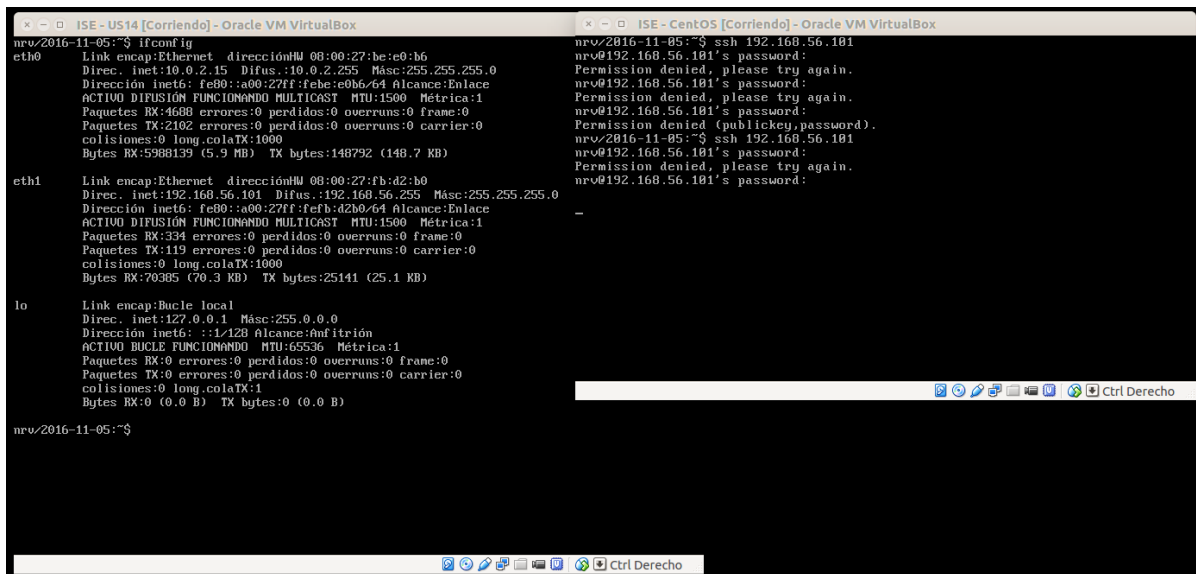
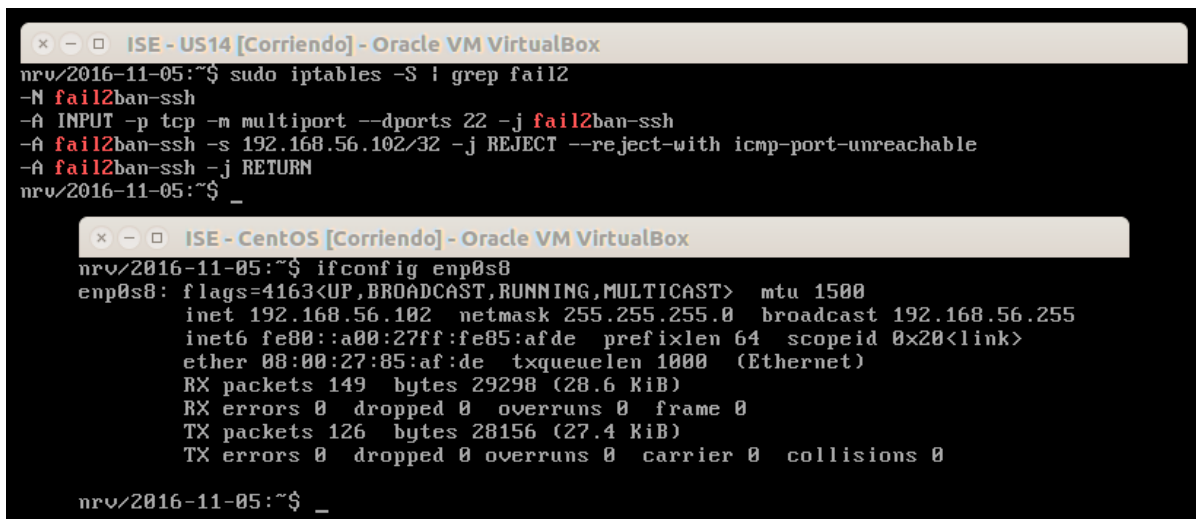


Figura 19.2: Acceso incorrecto (máquinas conectadas en modo *host-only*).

Desde nuestro servidor, ejecutando el comando `sudo iptables -S` vemos que se ha bloqueado la conexión *ssh* para la IP que es la IP de mi máquina virtual de CentOS, como podemos ver en la figura 19.3.



```
nrv/2016-11-05:~$ sudo iptables -S | grep fail2
-N fail2ban-ssh
-A INPUT -p tcp -m multiport --dports 22 -j fail2ban-ssh
-A fail2ban-ssh -s 192.168.56.102/32 -j REJECT --reject-with icmp-port-unreachable
-A fail2ban-ssh -j RETURN
nrv/2016-11-05:~$ _

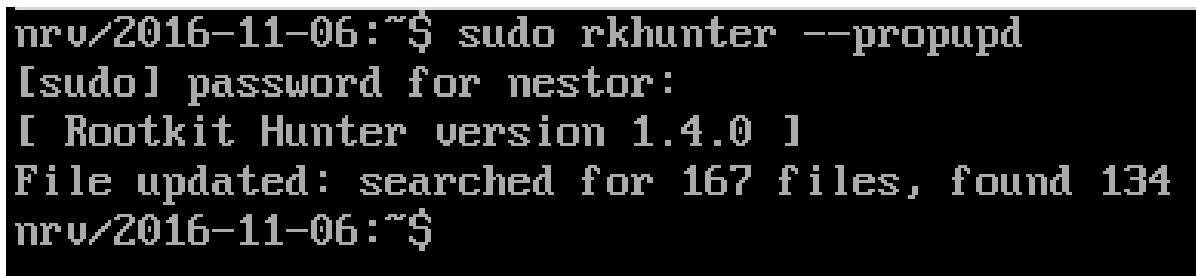
ISE - CentOS [Corriendo] - Oracle VM VirtualBox
nrv/2016-11-05:~$ ifconfig enp0s8
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.102 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a00:27ff:fe85:afde prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:85:af:de txqueuelen 1000 (Ethernet)
    RX packets 149 bytes 29298 (28.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 126 bytes 28156 (27.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

nrv/2016-11-05:~$ _
```

Figura 19.3: Dirección IP bloqueada (máquinas conectadas en modo *bridge*).

## 20. Cuestión opcional 3: Instale el servicio y pruebe su funcionamiento.

Para ver el funcionamiento de *rkhunter* podemos visitar la página oficial de dicho servicio [33]. Para instalar el servicio ejecutamos `sudo apt install rkhunter`. Lo primero que hacemos es crear una base de datos de como se encuentra nuestro sistema en el momento actual, para luego usarla como referencia. Para ello ejecutamos el comando `sudo rkhunter --propupd`, como podemos ver en la figura 20.1.



```
nrv/2016-11-06:~$ sudo rkhunter --propupd
[sudo] password for nestor:
[ Rootkit Hunter version 1.4.0 ]
File updated: searched for 167 files, found 134
nrv/2016-11-06:~$
```

Figura 20.1: Creación de la base de datos para *rkhunter*.

Una vez hemos creado la base de datos, podemos analizar nuestro sistema ejecutando `sudo rkhunter -c --enable all`. Una vez acabado el análisis, podemos ver un resumen del análisis, como podemos ver en la figura 20.2.

```

System checks summary
=====

File properties checks...
  Files checked: 134
  Suspect files: 1

Rootkit checks...
  Rootkits checked : 307
  Possible rootkits: 0

Applications checks...
  All checks skipped

The system checks took: 55 seconds

All results have been written to the log file (/var/log/rkhunter.log)

One or more warnings have been found while checking the system.
Please check the log file (/var/log/rkhunter.log)

nrv/2016-11-06:~$ _

```

Figura 20.2: Resultado del análisis del sistema con *rkhunter*.

Como podemos ver en la figura 20.2, si queremos ver el resultado completo debemos ver el contenido del archivo */var/log/rkhunter.log*. Finalmente, si queremos actualizar la base de datos que creamos anteriormente, debemos ejecutar el comando *sudo rkhunter --update*

## 21. Cuestión opcional 4: Realice la instalación de uno de estos dos “web containers” y pruebe su ejecución.

Voy a probar Apache Tomcat [34] en Ubuntu Server. Para instalar Apache Tomat, primero debemos ver si tenemos instalado Java instalado. Para ello, ejecutamos el comando *java -version*. Como podemos ver en la figura 21.1 Java no se encuentra instalado, para instalarlo ejecutamos *sudo apt-get install default-jdk*

```

nrv/2016-11-06:~$ java -version
El programa «java» puede encontrarse en los siguientes paquetes:
* default-jre
* gcj-4.8-jre-headless
* openjdk-7-jre-headless
* gcj-4.6-jre-headless
* openjdk-6-jre-headless
Intente: sudo apt-get install <paquete seleccionado>
nrv/2016-11-06:~$ _

```

Figura 21.1: Java no se encuentra instalado de Ubuntu Server.

Una vez hemos instalado java, ejecutamos *sudo apt install tomcat7* para instalar Apache Tomcat. Para ver su funcionamiento, desde la máquina anfitriona vamos a un navegador y escribimos la dirección IP de nuestro servidor web seguido de *:8080*. En mi caso, como podemos ver en la figura 21.2, la dirección IP de mi servidor es *192.168.56.101*, por lo tanto en el navegador debemos ir a la dirección *192.168.56.101:8080*. Como podemos ver en la figura 21.3, el servicio funciona correctamente.

```

nrv/2016-11-06:~$ ifconfig eth1
eth1      Link encap:Ethernet  direcciónHW 08:00:27:fb:d2:b0
          Direc. inet:192.168.56.101  Difus.:192.168.56.255  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:fe8b:d2b0/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
          Paquetes RX:316 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:82 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:70162 (70.1 KB)  TX bytes:27043 (27.0 KB)

nrv/2016-11-06:~$

```

Figura 21.2: Dirección IP de mi servidor.



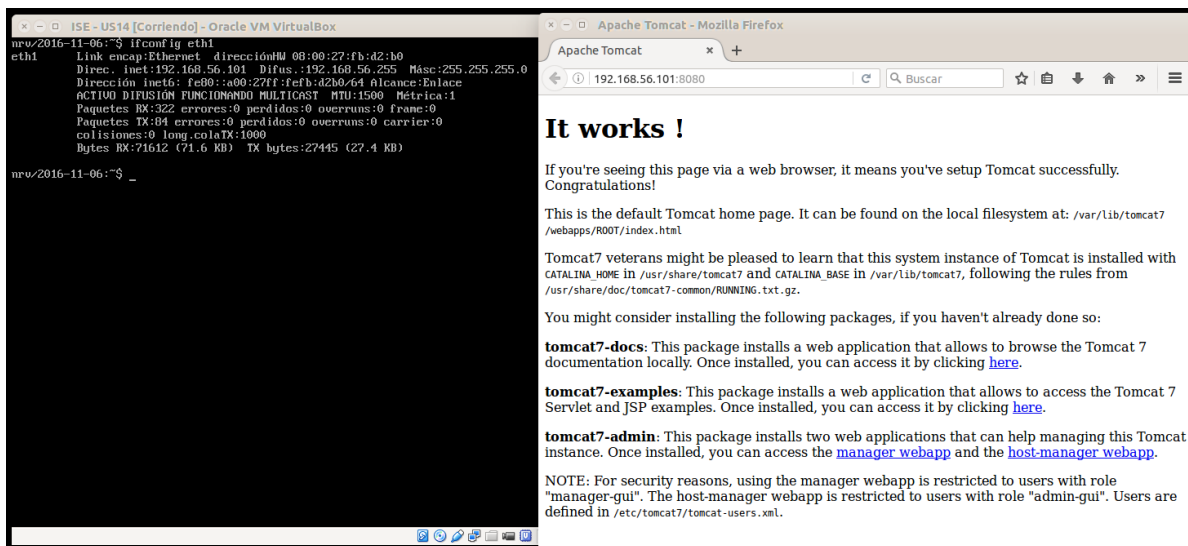


Figura 21.3: Servicio *tomcat* funcionando correctamente (máquinas conectadas en modo *host-only*).

Para acceder al gestor de aplicaciones, como podemos ver en la página de Apache Tomcat [35] debemos acceder a la dirección `192.168.56.101:8080/manager/html`. Una vez en esa dirección, nos pide un usuario y una contraseña. Para acceder he creado un usuario con los roles que se puede ver en la figura 21.4 y he accedido con el. Dentro de tomcat podemos ver que se puede gestionar diferentes parámetros, como puede ser el tiempo que tardará en expirar una sesión inactiva, como se puede ver en la figura 21.4.

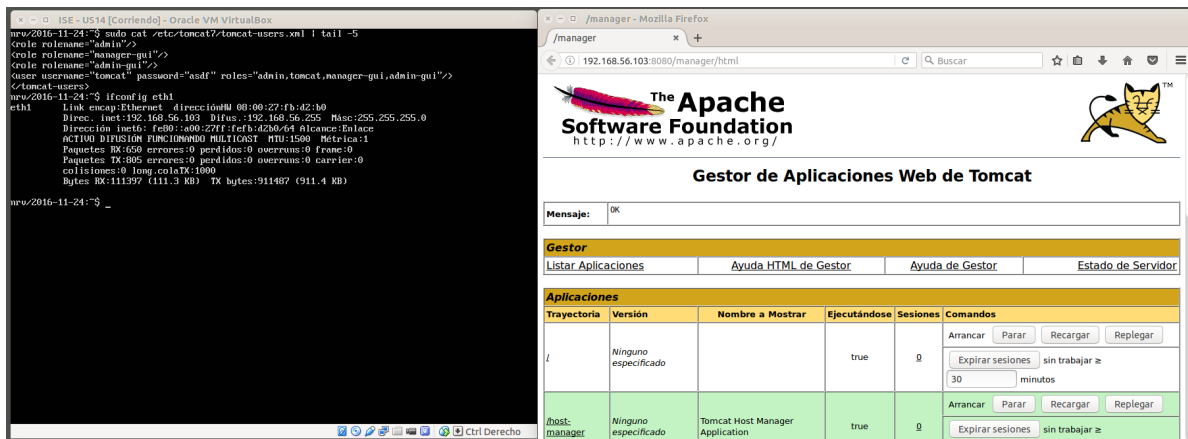


Figura 21.4: Sesión iniciada en *tomcat* (máquinas conectadas en modo *host-only*).

## 22. Cuestión opcional 5: Realice la instalación de MongoDB en alguna de sus máquinas virtuales. Cree una colección de documentos y haga una consulta sobre ellos.

Voy a instalar MongoDB en CentoOS. Para ello voy a seguir los pasos de la documentación oficial [36].

1. Creamos el archivo `/etc/yum.repos.d/mongodb-org-3.2.repo` y añadimos las siguientes líneas:

```
[mongodb-org-3.2]
name=MongoDB Repository
baseurl=https://repo.mongodb.org/yum/redhat/$releasever/mongodb-org/3.2/x86_64/
gpgcheck=1
enabled=1
gpgkey=https://www.mongodb.org/static/pgp/server-3.2.asc
```

El resultado de este paso lo podemos ver en la figura 22.1.

2. Instalamos MongoDB ejecutando `sudo yum install -y mongodb-org`, como podemos ver en la figura 22.1.

Una vez instalado, debemos cambiar el valor de *SELINUX*. Para ello editamos el fichero `/etc/selinux/config` y cambiamos el valor de *SELINUX* a *permissive*, como podemos ver en la figura 22.2. Una vez cambiado, debemos reiniciar CentOS.<sup>10</sup>

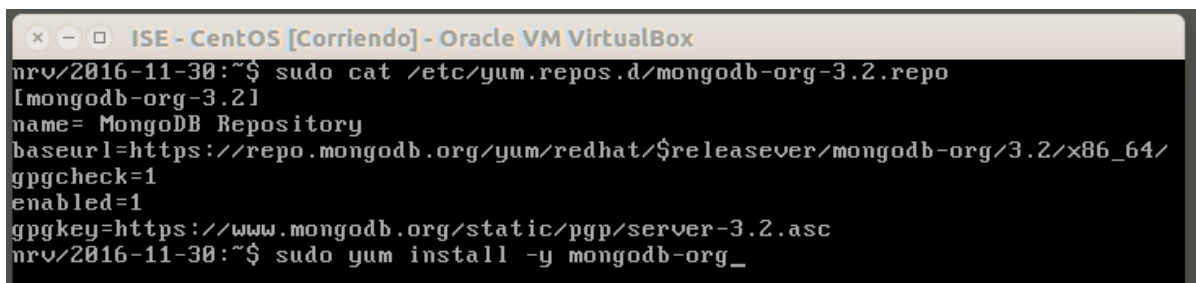
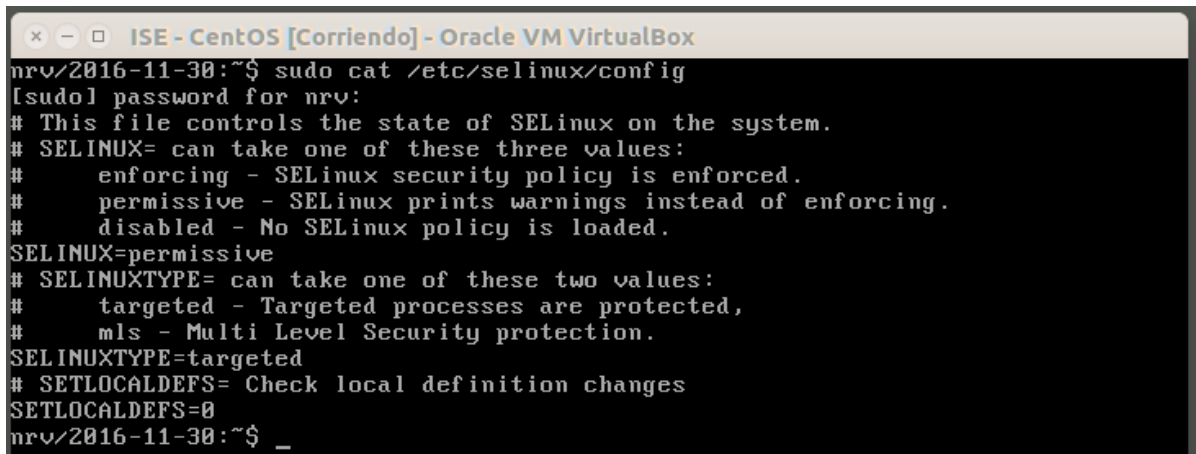


Figura 22.1: Instalación de *MongoDB*.

<sup>10</sup>Se podría usar *setenforce* pero el cambio no sería permanente.



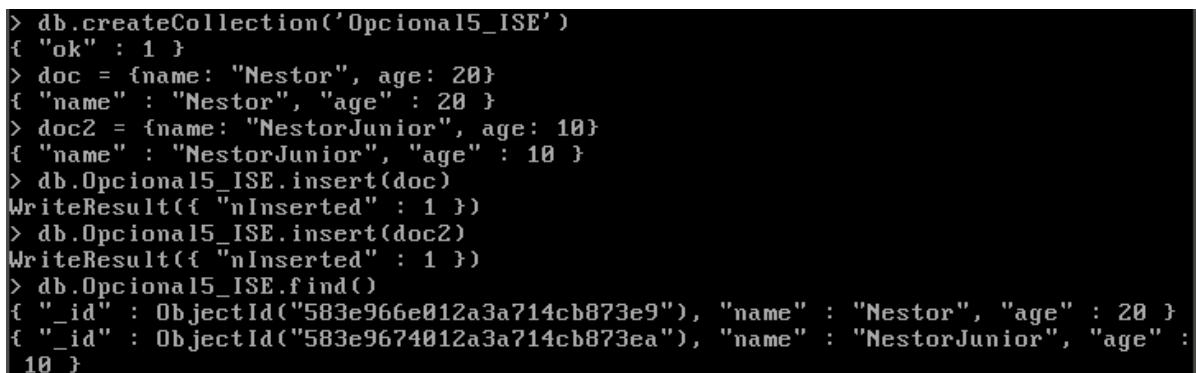
```
nr@2016-11-30:~$ sudo cat /etc/selinux/config
[sudo] password for nr:
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=permissive
# SELINUXTYPE= can take one of these two values:
#   targeted - Targeted processes are protected,
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
# SETLOCALDEFs= Check local definition changes
SETLOCALDEFs=0
nr@2016-11-30:~$ _
```

Figura 22.2: Cambio de *SELINUX*.

Para entrar en MongoDB ejecutamos *mongo*. Una vez dentro de MongoDB seguimos los siguientes pasos:

1. Creamos la colección que vamos a usar, *Opcional5\_ISE*, para ello ejecutamos *db.createCollection('Opcional5\_ISE')*.
2. Creamos dos documentos para luego insertarlos en la colección que hemos creado en el paso anterior. Para ello ejecutamos:
  - a) *doc = { name: "Nestor", age: 20 }*
  - b) *doc2 = { name: "NestorJunior", age: 10 }*
3. Insertamos los documentos en la colección *Opcional5\_ISE*. Para ello ejecutamos:
  - a) *db.Opcional5\_ISE.insert(doc)*
  - b) *db.Opcional5\_ISE.insert(doc2)*
4. Finalmente realizamos la consulta en la colección ejecutando *db.Opcional5\_ISE.find()*

El resultado de este proceso lo podemos ver en la figura 22.3.



```
> db.createCollection('Opcional5_ISE')
{ "ok" : 1 }
> doc = {name: "Nestor", age: 20}
{ "name" : "Nestor", "age" : 20 }
> doc2 = {name: "NestorJunior", age: 10}
{ "name" : "NestorJunior", "age" : 10 }
> db.Opcional5_ISE.insert(doc)
WriteResult({ "nInserted" : 1 })
> db.Opcional5_ISE.insert(doc2)
WriteResult({ "nInserted" : 1 })
> db.Opcional5_ISE.find()
{ "_id" : ObjectId("583e966e012a3a714cb873e9"), "name" : "Nestor", "age" : 20 }
{ "_id" : ObjectId("583e9674012a3a714cb873ea"), "name" : "NestorJunior", "age" : 10 }
```

Figura 22.3: Creación de la colección, inserción de documentos y consulta.

## Referencias

- [1] [https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/5/html/Deployment\\_Guide/s1-yum-useful-commands.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/s1-yum-useful-commands.html). Fecha de acceso: 28/10/2016.
- [2] <https://www.centos.org/docs/5/html/yum/sn-yum-proxy-server.html>. Fecha de acceso: 28/10/2016.
- [3] <https://help.ubuntu.com/community/AptGet/Howto>. Año de publicación/última edición: 2016 - Fecha de acceso: 28/10/2016.
- [4] <https://help.ubuntu.com/community/AptGet/Howto>. Año de publicación/última edición: 2016 - Fecha de acceso: 28/10/2016.
- [5] <https://help.ubuntu.com/community/UFW>. Año de publicación/última edición: 2015 - Fecha de acceso: 28/10/2016.
- [6] [https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/Security\\_Guide/sec-Using\\_Firewalls.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/sec-Using_Firewalls.html). Año de publicación/última edición: 2015 - Fecha de acceso: 28/10/2016.
- [7] <https://fedoraproject.org/wiki/FirewallD?rd=FirewallD/>. Año de publicación/última edición: 2015 - Fecha de acceso: 28/10/2016.
- [8] <https://nmap.org/book/man-port-scanning-basics.html>. Fecha de acceso: 03/11/2016.
- [9] <http://www.telnet.org/htm/faq.htm>. Fecha de acceso: 29/10/2016.
- [10] <https://www.openssh.com/index.html>. Fecha de acceso: 29/10/2016.
- [11] <http://man.openbsd.org/ssh>. Año de publicación/última edición: 2016 - Fecha de acceso: 29/10/2016.
- [12] <https://wiki.centos.org/HowTos/Network/SecuringSSH>. Año de publicación/última edición: 2015 - Fecha de acceso: 29/10/2016.
- [13] [https://debian-administration.org/article/152/Password-less\\_logins\\_with\\_OpenSSH](https://debian-administration.org/article/152/Password-less_logins_with_OpenSSH). Año de publicación/última edición: 2005 - Fecha de acceso: 29/10/2016.
- [14] [http://man.openbsd.org/sshd\\_config](http://man.openbsd.org/sshd_config). Año de publicación/última edición: 2016 - Fecha de acceso: 29/10/2016.
- [15] <http://manpages.ubuntu.com/manpages/precise/man8/service.8.html>. Año de publicación/última edición: 2006 - Fecha de acceso: 29/10/2016.

- [16] [https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/System\\_Administrators\\_Guide/sect-Managing\\_Services\\_with\\_systemd-Services.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/sect-Managing_Services_with_systemd-Services.html). Fecha de acceso: 29/10/2016.
- [17] <https://help.ubuntu.com/lts/serverguide/httpd.html>. Fecha de acceso: 29/10/2016.
- [18] <https://help.ubuntu.com/12.04/serverguide/mysql.html>. Fecha de acceso: 29/10/2016.
- [19] [https://help.ubuntu.com/community/ApacheMySQLPHP#Installing\\_PHP\\_5](https://help.ubuntu.com/community/ApacheMySQLPHP#Installing_PHP_5). Año de publicación/última edición: 2015 - Fecha de acceso: 29/10/2016.
- [20] <https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-centos-7>. Año de publicación/última edición: 2015 - Fecha de acceso: 29/10/2016.
- [21] <http://fedoraproject.org/wiki/VMWare>. Año de publicación/última edición: 2015 - Fecha de acceso: 30/10/2016.
- [22] <https://linux.die.net/man/1/patch>. Año de publicación/última edición: 2015 - Fecha de acceso: 30/10/2016.
- [23] <http://webmin.com/deb.html>. Fecha de acceso: 30/10/2016.
- [24] <http://www.webmin.com/faq.html>. Fecha de acceso: 30/10/2016.
- [25] <https://help.ubuntu.com/lts/serverguide/phpmyadmin.html>. Fecha de acceso: 30/10/2016.
- [26] <https://help.ubuntu.com/community/phpMyAdmin>. Año de publicación/última edición: 2015 - Fecha de acceso: 30/10/2016.
- [27] <https://phpmyadmin.readthedocs.io/en/latest/faq.html>. Fecha de acceso: 30/10/2016.
- [28] <https://linux.die.net/man/1/sed>. Fecha de acceso: 31/10/2016.
- [29] <https://technet.microsoft.com/en-us/library/ff714569.aspx>. Fecha de acceso: 01/11/2016.
- [30] <http://ss64.com/bash/screen.html>. Fecha de acceso: 07/11/2016.
- [31] [http://www.fail2ban.org/wiki/index.php/Main\\_Page](http://www.fail2ban.org/wiki/index.php/Main_Page). Fecha de acceso: 05/11/2016.
- [32] <https://www.digitalocean.com/community/tutorials/how-to-protect-ssh-with-fail2ban-on-ubuntu-14-04>. Año de publicación/última edición: 2014 - Fecha de acceso: 05/11/2016.

- [33] <http://rkhunter.sourceforge.net/>. Fecha de acceso: 06/11/2016.
- [34] <http://tomcat.apache.org/>. Fecha de acceso: 06/11/2016.
- [35] <https://tomcat.apache.org/tomcat-7.0-doc/manager-howto.html>. Año de publicación/última edición: 2016 - Fecha de acceso: 06/11/2016.
- [36] <https://docs.mongodb.com/v3.2/tutorial/install-mongodb-on-red-hat/>. Fecha de acceso: 06/11/2016.