

Práctica 2

Jesús Rodríguez Heras
Arantzazu Otañal Alberro

27 de marzo de 2019

1. Instalación de máquinas virtuales mediante Vagrant

En esta primera parte vamos a crear el entorno de trabajo, consistente en tres máquinas virtuales pertenecientes a una misma red privada. Las máquinas se tendrán que crear a partir de un mismo fichero Vagrant.

1. VM1, con IP 192.168.2.101
2. VM2, con IP 192.168.2.102
3. VM3, con IP 192.168.2.103

Las máquinas tendrán la siguiente configuración:

- nmap tiene que estar instalado en todos.
- iptables en la máquina VM1.
- ufw en la máquina VM1 (debería estar instalado por defecto).
- fwbuilder.

La instalación de los paquetes se deberá realizar mediante la provisión de Vagrant.

Para inicializar Vagrant usamos `vagrant init debian/jessie64` y luego abrimos y modificamos el archivo `Vagrantfile` de la siguiente forma:

```
1 Vagrant.configure("2") do |config|
2
3   config.vm.box = "debian/jessie64"
4   config.vm.provision :shell, path: "bootstrap.sh"
5
6   config.vm.define :vm1 do |vm1|
7     vm1.vm.box="debian/jessie64"
8     vm1.vm.hostname="VM1"
9     vm1.vm.network "private_network", ip: "192.168.2.101"
10  end
11
12  config.vm.define :vm2 do |vm2|
13    vm2.vm.box="debian/jessie64"
14    vm2.vm.hostname="VM2"
15    vm2.vm.network "private_network", ip: "192.168.2.102"
16  end
17
18  config.vm.define :vm3 do |vm3|
19    vm3.vm.box="debian/jessie64"
20    vm3.vm.hostname="VM3"
21    vm3.vm.network "private_network", ip: "192.168.2.103"
22  end
23
24 end
```

A continuación, iniciamos las tres máquinas virtuales en terminales diferentes con `vagrant up vmX` y nos conectamos a ellas mediante `vagrant ssh vmX` (siendo “X” el número de la máquina virtual comprendido entre 1 y 3).

Para instalar `nmap` en todas las máquinas usaremos el aprovisionamiento de Vagrant creando el archivo `bootstrap.sh` siguiente:

```
1 #!/usr/bin/env bash
2
3 apt-get update
4 apt-get install -y nmap
```

2. Visibilidad de las máquinas

Para los distintos ejercicios, se identifica a las máquinas como VM1, VM2 y VM3. Por comodidad, es recomendable poder usar nombres en las reglas. Para ello, se puede añadir en `/etc/hosts` una línea asociando un nombre y una IP con la siguiente sintaxis: IP NOMBRE ALIAS.

Para hacer esto, entramos en las tres máquinas virtuales y accedemos al archivo mencionado con `sudo nano /etc/hosts` y lo modificamos de la siguiente forma:

```
1 192.168.2.101 vm1
2 192.168.2.102 vm2
3 192.168.2.103 vm3
```

3. Configuraciones IPtables

3.1. Primeras pruebas

En este ejercicio se pide testear VM1 desde VM2, realizando los siguientes ejercicios:

1. Desde VM2 comprobar los puertos que VM1 tiene abiertos.

Para comprobar los puertos usamos: `nmap vm1`.

2. Prohibir el acceso por ssh.

Para ello usaremos: `sudo iptables -A INPUT -p tcp --dport 22 -j DROP`

3. Responde a las siguientes preguntas:

- **¿Qué ha pasado?**

La consola se queda bloqueada sin poder establecer conexión por ssh.

- **¿Puedo crear una nueva conexión?**

Es imposible.

- **¿La consola sigue funcionando?**

No, se queda bloqueada y no responde.

3.2. Configuración mínima

En los ejercicios siguientes, siempre debe partir de esta configuración:

- **Permitir conexiones locales.**

```
sudo iptables -A INPUT -i lo -j ACCEPT
```

- **Permitir conexiones ya establecidas.**

```
sudo iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT
```

- **Políticas por defecto de rechazar en input.**

```
sudo iptables -A INPUT -j DROP
```

Para comprobar estas configuraciones hicimos ping entre las máquinas para ver la conectividad.

3.3. Configurando servidor web completo

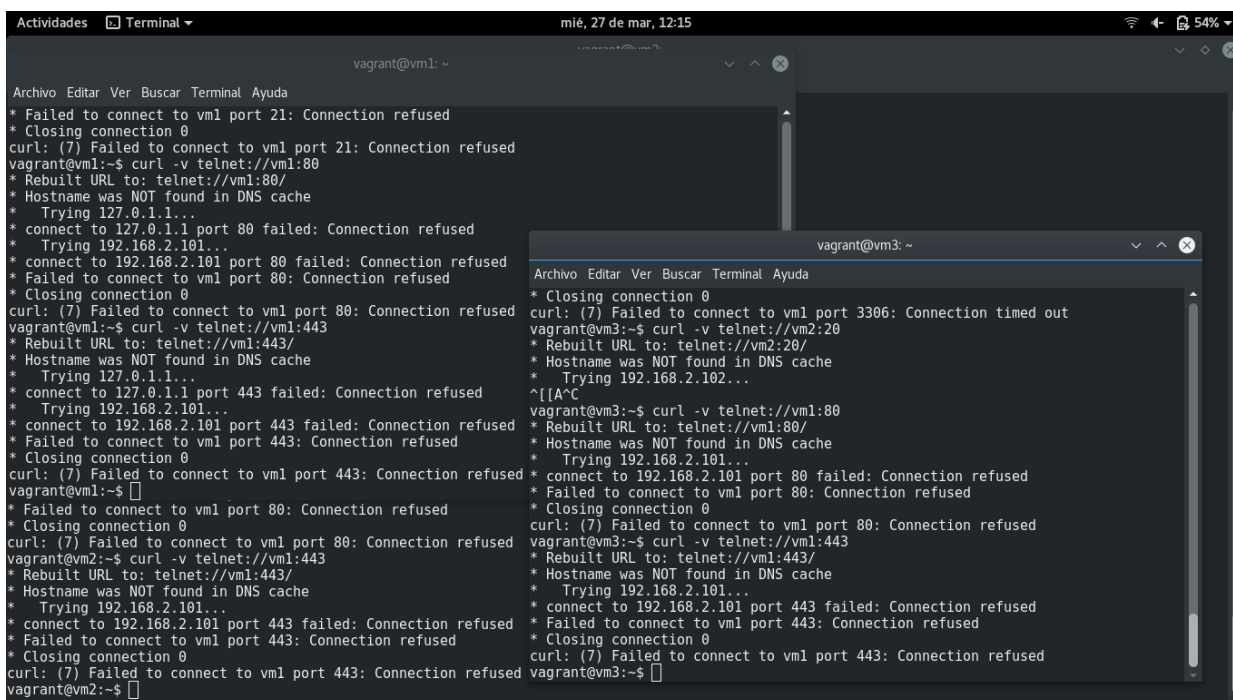
Configurar VM1 para que tenga la configuración de un servidor web, permitiendo:

- **Todos se conecten a los puertos http y https.**

Primero mostramos las iptables con `sudo iptables -L` y, si tenemos la anterior que no permitía conexiones entrantes, usamos `sudo iptables -F` para borrarlas todas¹ (con el inconveniente de que tendremos que reescribir las que queramos).

Para habilitar http usamos: `sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT`

Para habilitar https usamos: `sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT`



```
Actividades Terminal mié, 27 de mar, 12:15

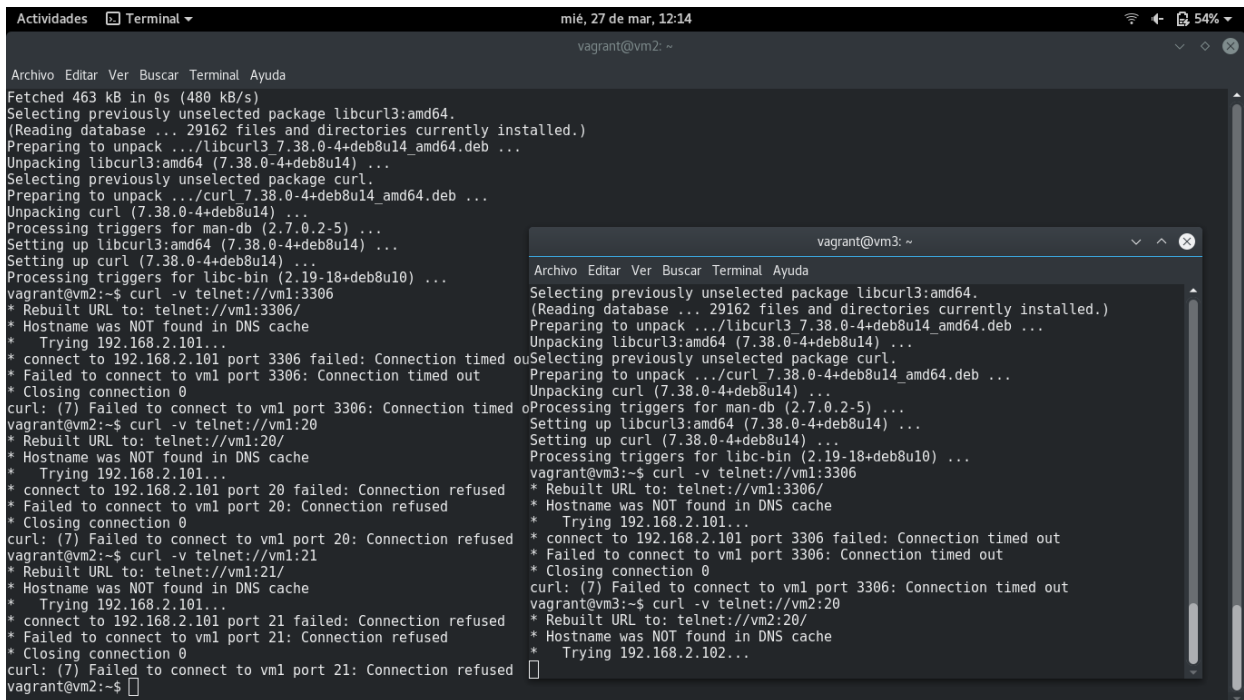
vagrant@vm1: ~
Archivo Editar Ver Buscar Terminal Ayuda
* Failed to connect to vm1 port 21: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 21: Connection refused
vagrant@vm1:~$ curl -v telnet://vm1:80
* Rebuilt URL to: telnet://vm1:80/
* Hostname was NOT found in DNS cache
* Trying 127.0.1.1...
* connect to 127.0.1.1 port 80 failed: Connection refused
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 80 failed: Connection refused
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm1:~$ curl -v telnet://vm1:443
* Rebuilt URL to: telnet://vm1:443/
* Hostname was NOT found in DNS cache
* Trying 127.0.1.1...
* connect to 127.0.1.1 port 443 failed: Connection refused
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 443 failed: Connection refused
* Failed to connect to vm1 port 443: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 443: Connection refused
vagrant@vm1:~$
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:443
* Rebuilt URL to: telnet://vm1:443/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 443 failed: Connection refused
* Failed to connect to vm1 port 443: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 443: Connection refused
vagrant@vm2:~$

vagrant@vm3: ~
Archivo Editar Ver Buscar Terminal Ayuda
* Closing connection 0
curl: (7) Failed to connect to vm1 port 3306: Connection timed out
vagrant@vm3:~$ curl -v telnet://vm2:20
* Rebuilt URL to: telnet://vm2:20/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.102...
^[[A^C
vagrant@vm3:~$ curl -v telnet://vm1:80
* Rebuilt URL to: telnet://vm1:80/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 80 failed: Connection refused
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm3:~$ curl -v telnet://vm1:443
* Rebuilt URL to: telnet://vm1:443/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 443 failed: Connection refused
* Failed to connect to vm1 port 443: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 443: Connection refused
vagrant@vm3:~$
```

¹Si queremos borrar solo una regla, mostramos todas las reglas existentes con `sudo iptables -L -line-numbers` y para borrar la que queramos usamos `sudo iptables -D INPUT numeroderegla`

■ Conexión únicamente por parte de VM2 al servidor ftp.

```
sudo iptables -A INPUT -p tcp --dport 20 -s 192.168.2.102 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 21 -s 192.168.2.102 -j ACCEPT
```



The image shows two terminal windows. The top window, titled 'vagrant@vm2: ~', shows the installation of curl and libcurl3:amd64. It then shows three failed telnet attempts from vm2 to vm1 on ports 3306, 20, and 21. The bottom window, titled 'vagrant@vm3: ~', shows the installation of curl and libcurl3:amd64. It then shows three failed telnet attempts from vm3 to vm1 on ports 3306, 20, and 21. The output of the telnet commands in both windows is identical, showing connection failures due to timeouts or refused connections.

```
Archivo Editar Ver Buscar Terminal Ayuda
Fetched 463 kB in 0s (480 kB/s)
Selecting previously unselected package libcurl3:amd64.
(Reading database ... 29162 files and directories currently installed.)
Preparing to unpack .../libcurl3_7.38.0-4+deb8u14_amd64.deb ...
Unpacking libcurl3:amd64 (7.38.0-4+deb8u14) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.38.0-4+deb8u14_amd64.deb ...
Unpacking curl (7.38.0-4+deb8u14) ...
Processing triggers for man-db (2.7.0.2-5) ...
Setting up libcurl3:amd64 (7.38.0-4+deb8u14) ...
Setting up curl (7.38.0-4+deb8u14) ...
Processing triggers for libc-bin (2.19-18+deb8u10) ...
vagrant@vm2:~$ curl -v telnet://vm1:3306
* Rebuilt URL to: telnet://vm1:3306/
* Hostname was NOT found in DNS cache
*   Trying 192.168.2.101...
* connect to 192.168.2.101 port 3306 failed: Connection timed out
* Closing connection 0
curl: (7) Failed to connect to vm1 port 3306: Connection timed out
vagrant@vm2:~$ curl -v telnet://vm1:20
* Rebuilt URL to: telnet://vm1:20/
* Hostname was NOT found in DNS cache
*   Trying 192.168.2.101...
* connect to 192.168.2.101 port 20 failed: Connection refused
* Failed to connect to vm1 port 20: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 20: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:21
* Rebuilt URL to: telnet://vm1:21/
* Hostname was NOT found in DNS cache
*   Trying 192.168.2.101...
* connect to 192.168.2.101 port 21 failed: Connection refused
* Failed to connect to vm1 port 21: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 21: Connection refused
vagrant@vm2:~$

Archivo Editar Ver Buscar Terminal Ayuda
Selecting previously unselected package libcurl3:amd64.
(Reading database ... 29162 files and directories currently installed.)
Preparing to unpack .../libcurl3_7.38.0-4+deb8u14_amd64.deb ...
Unpacking libcurl3:amd64 (7.38.0-4+deb8u14) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.38.0-4+deb8u14_amd64.deb ...
Unpacking curl (7.38.0-4+deb8u14) ...
Processing triggers for man-db (2.7.0.2-5) ...
Setting up libcurl3:amd64 (7.38.0-4+deb8u14) ...
Setting up curl (7.38.0-4+deb8u14) ...
Processing triggers for libc-bin (2.19-18+deb8u10) ...
vagrant@vm3:~$ curl -v telnet://vm1:3306
* Rebuilt URL to: telnet://vm1:3306/
* Hostname was NOT found in DNS cache
*   Trying 192.168.2.101...
* connect to 192.168.2.101 port 3306 failed: Connection timed out
* Failed to connect to vm1 port 3306: Connection timed out
* Closing connection 0
curl: (7) Failed to connect to vm1 port 3306: Connection timed out
vagrant@vm3:~$ curl -v telnet://vm2:20
* Rebuilt URL to: telnet://vm2:20/
* Hostname was NOT found in DNS cache
*   Trying 192.168.2.102...
```

- **Configurar VM1 para que sólo se pueda conectar localmente a mysql.**

```
sudo iptables -A INPUT -p tcp -i lo --dport 3306 -j ACCEPT
```

```

Archivo Editar Ver Buscar Terminal Ayuda
Get:2 http://security.debian.org/ jessie/updates/main curl amd64 7.38.0-4+deb8u14 [202 kB]
Fetched 463 kB in 0s (533 kB/s)
Selecting previously unselected package libcurl3:amd64.
(Reading database ... 29162 files and directories currently installed.)
Preparing to unpack .../libcurl3_7.38.0-4+deb8u14_amd64.deb ...
Unpacking libcurl3:amd64 (7.38.0-4+deb8u14) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.38.0-4+deb8u14_amd64.deb ...
Unpacking curl (7.38.0-4+deb8u14) ...
Processing triggers for man-db (2.7.0.2-5) ...
Setting up libcurl3:amd64 (7.38.0-4+deb8u14) ...
Setting up curl (7.38.0-4+deb8u14) ...
Processing triggers for libc-bin (2.19-18+deb8u10) ...
vagrant@vm1:~$ curl -v telnet://vm1:3306
* Rebuilt URL to: telnet://vm1:3306/
* Hostname was NOT found in DNS cache
*   Trying 127.0.1.1...
* connect to 127.0.1.1 port 3306 failed: Connection refused
*   Trying 192.168.2.101...
* connect to 192.168.2.101 port 3306 failed: Connection refused
* Failed to connect to vm1 port 3306: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 3306: Connection refused
vagrant@vm1:~$
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.38.0-4+deb8u14_amd64.deb ...
Unpacking curl (7.38.0-4+deb8u14) ...
Processing triggers for man-db (2.7.0.2-5) ...
Setting up libcurl3:amd64 (7.38.0-4+deb8u14) ...
Setting up curl (7.38.0-4+deb8u14) ...
Processing triggers for libc-bin (2.19-18+deb8u10) ...
vagrant@vm2:~$ curl -v telnet://vm1:3306
* Rebuilt URL to: telnet://vm1:3306/
* Hostname was NOT found in DNS cache
*   Trying 192.168.2.101...

```

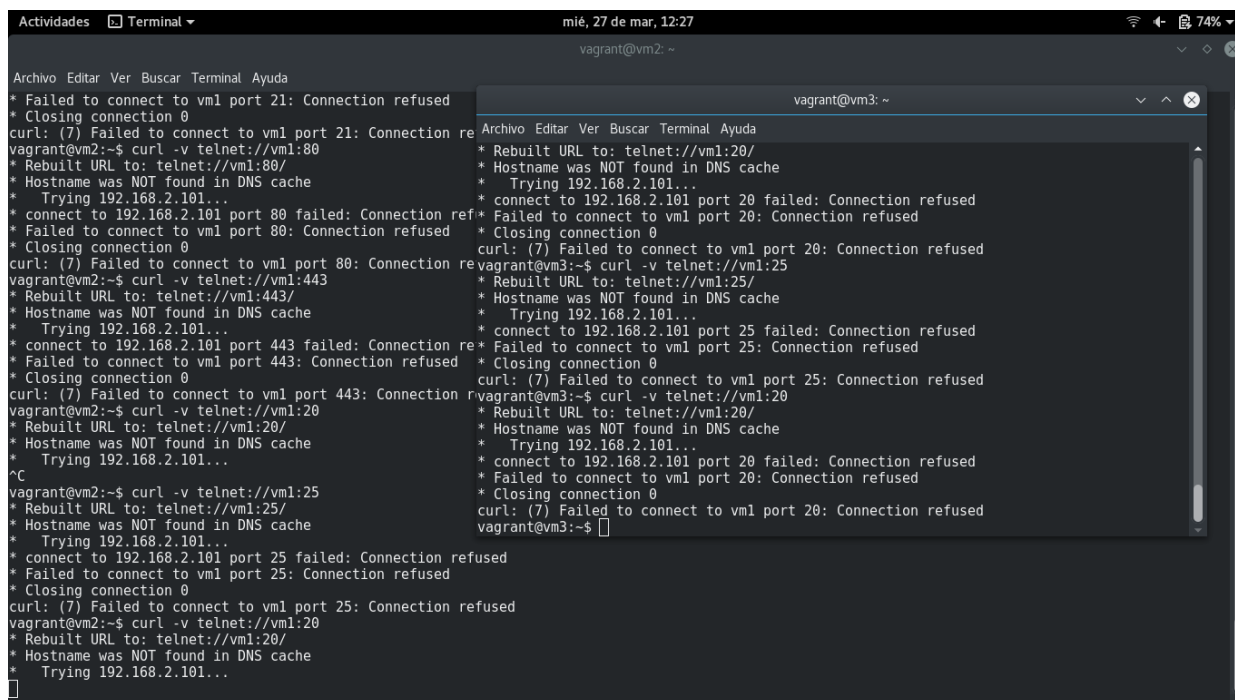
Otra opción para comprobar los puertos abiertos es poner el comando `nc -l numerodepuerto` &. Eso nos abrirá el puerto que queramos y lo mandará a segundo plano con la finalidad de que el nmap o el telnet nos indique que ese puerto está abierto y a la escucha. Para cerrarlo, tendremos que poner el comando `kill -9 PID`, siendo PID el PID del proceso que nos mantiene el puerto abierto (que se nos muestra al mandar a segundo plano el comando `nc`).5

3.4. Poniendo excepciones

Permitir conectar a VM1 desde VM2 y VM3 el acceso a los puertos desde 1:1000, con la excepción de que VM2 no se puede conectar por FTP.

Para VM2: `sudo iptables -A INPUT -p tcp --dport 20:21 -s 192.168.2.102 -j DROP`; `sudo iptables -A INPUT -p tcp --dport 1:1000 -s 192.168.2.101 -j ACCEPT`

Para VM3: `sudo iptables -A INPUT -p tcp --dport 1:1000 -s 192.168.2.103 -j ACCEPT`



```
Archivos Editor Ver Buscar Terminal Ayuda
* Failed to connect to vm1 port 21: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 21: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:80
* Rebuilt URL to: telnet://vm1:80/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 80 failed: Connection refused
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:443
* Rebuilt URL to: telnet://vm1:443/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 443 failed: Connection refused
* Failed to connect to vm1 port 443: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 443: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:20
* Rebuilt URL to: telnet://vm1:20/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 20 failed: Connection refused
* Failed to connect to vm1 port 20: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 20: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:25
* Rebuilt URL to: telnet://vm1:25/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 25 failed: Connection refused
* Failed to connect to vm1 port 25: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 25: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:20
* Rebuilt URL to: telnet://vm1:20/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 20 failed: Connection refused
* Failed to connect to vm1 port 20: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 20: Connection refused
vagrant@vm3:~$
```

4. UFW

Configurar VM1 para que tenga la configuración de un servidor web, permitiendo:

- Todos se conecten a los puertos http y https.

Para habilitar http usamos: `sudo ufw allow http`

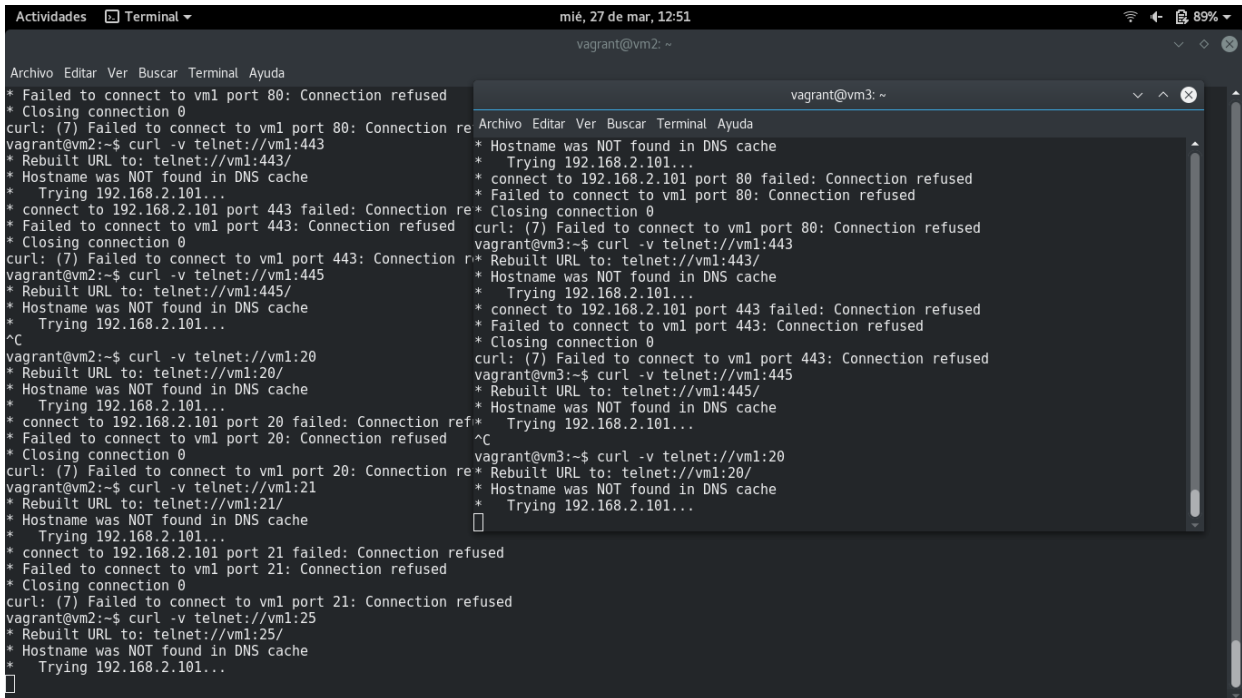
Para habilitar https usamos: `sudo ufw allow https`

```
Actividades Terminal mié, 27 de mar, 12:46 86%
vagrant@vm2: ~
Archivo Editar Ver Buscar Terminal Ayuda
update-rc.d: warning: start and stop actions are no longer supported by
Processing triggers for libc-bin (2.19-18+deb8u10) ...
Processing triggers for systemd (215-17+deb8u7) ...
vagrant@vm2:~$ curl -v telnet://vm1:80
* Rebuilt URL to: telnet://vm1:80/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 80 failed: Connection refused
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:81
* Rebuilt URL to: telnet://vm1:81/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 80 failed: Connection refused
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:443
* Rebuilt URL to: telnet://vm1:443/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 443 failed: Connection refused
* Failed to connect to vm1 port 443: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 443: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:445
* Rebuilt URL to: telnet://vm1:445/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 445 failed: Connection refused
* Failed to connect to vm1 port 445: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 445: Connection refused

vagrant@vm3: ~
Archivo Editar Ver Buscar Terminal Ayuda
defaults
Processing triggers for libc-bin (2.19-18+deb8u10) ...
Processing triggers for systemd (215-17+deb8u7) ...
vagrant@vm3:~$ curl -v telnet://vm1:80
* Rebuilt URL to: telnet://vm1:80/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 80 failed: Connection refused
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm3:~$ curl -v telnet://vm1:443
* Rebuilt URL to: telnet://vm1:443/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 443 failed: Connection refused
* Failed to connect to vm1 port 443: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 443: Connection refused
vagrant@vm3:~$ curl -v telnet://vm1:445
* Rebuilt URL to: telnet://vm1:445/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
```


■ **Conexión únicamente por parte de VM2 al servidor ftp.**

```
sudo ufw allow from 192.168.2.102 to any port 20
sudo ufw allow from 192.168.2.102 to any port 21
```



The screenshot shows two terminal windows. The left window is titled 'vagrant@vm2: ~' and the right window is titled 'vagrant@vm3: ~'. Both windows show the output of telnet commands attempting to connect to VM1 on various ports. The left window shows attempts on ports 80, 443, 445, 20, 21, and 25, all resulting in connection refusals. The right window shows attempts on ports 80, 443, and 20, also resulting in connection refusals. The output includes details about DNS cache misses and connection failures.

```
Archivo Editar Ver Buscar Terminal Ayuda
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:443
* Rebuilt URL to: telnet://vm1:443/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 443 failed: Connection refused
* Failed to connect to vm1 port 443: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 443: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:445
* Rebuilt URL to: telnet://vm1:445/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
^C
vagrant@vm2:~$ curl -v telnet://vm1:20
* Rebuilt URL to: telnet://vm1:20/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 20 failed: Connection refused
* Failed to connect to vm1 port 20: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 20: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:21
* Rebuilt URL to: telnet://vm1:21/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 21 failed: Connection refused
* Failed to connect to vm1 port 21: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 21: Connection refused
vagrant@vm2:~$ curl -v telnet://vm1:25
* Rebuilt URL to: telnet://vm1:25/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
^C

Archivo Editar Ver Buscar Terminal Ayuda
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 80 failed: Connection refused
* Failed to connect to vm1 port 80: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 80: Connection refused
vagrant@vm3:~$ curl -v telnet://vm1:443
* Rebuilt URL to: telnet://vm1:443/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
* connect to 192.168.2.101 port 443 failed: Connection refused
* Failed to connect to vm1 port 443: Connection refused
* Closing connection 0
curl: (7) Failed to connect to vm1 port 443: Connection refused
vagrant@vm3:~$ curl -v telnet://vm1:445
* Rebuilt URL to: telnet://vm1:445/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
^C
vagrant@vm3:~$ curl -v telnet://vm1:20
* Rebuilt URL to: telnet://vm1:20/
* Hostname was NOT found in DNS cache
* Trying 192.168.2.101...
^C
```

- **Configurar VM1 para que sólo se pueda conectar localmente a mysql.**

```
sudo ufw allow from 192.168.2.101 to any port 3306
```

[illegible]