

**MAYO DE 2021**

UGR CURSO 2020-2021

ELENA ORTIZ MORENO

CORREO: elena97om@correo.ugr.es

# **PRÁCTICA 4: ASEGURAR LA GRANJA WEB**

SERVIDORES WEB DE  
ALTAS PRESTACIONES

## ÍNDICE:

- |   |      |
|---|------|
| 1.Instalar un certificado SSL autofirmado para configurar el acceso por HTTPS | 3-7  |
| 2.Configuración del cortafuegos   | 8-9  |
| 3.Opciones avanzadas  | 9-14 |

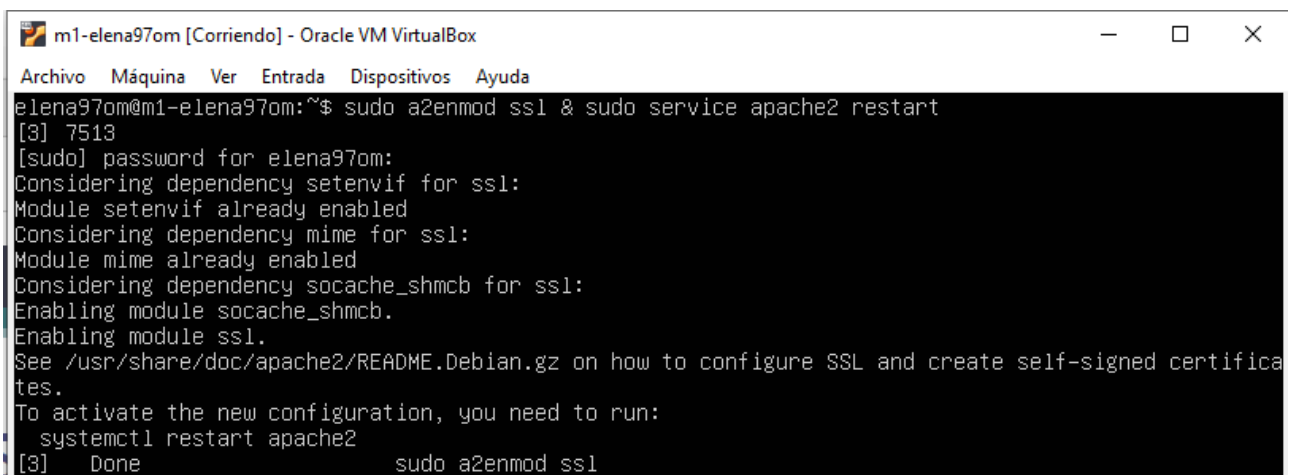
# 1.Instalar un certificado SSL autofirmado para configurar el acceso por HTTPS:

Un certificado SSL sirve para dar seguridad a los visitantes de la página web.

El protocolo SSL proporciona servicios de autenticación, integridad y privacidad.

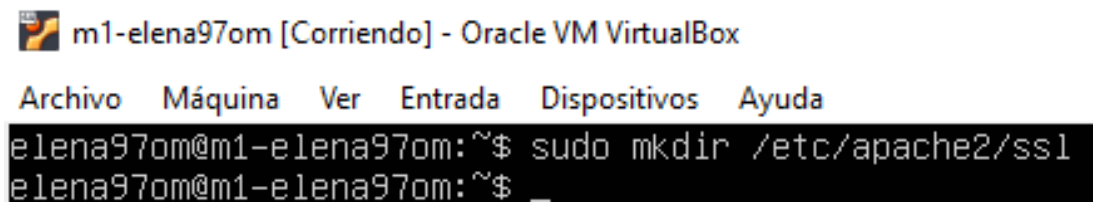
Para generar un certificado SSL debemos activar el módulo SSL de Apache, generar los certificados y especificar la ruta, todo ello de la siguiente manera:

## 1. Activamos el módulo SSL.



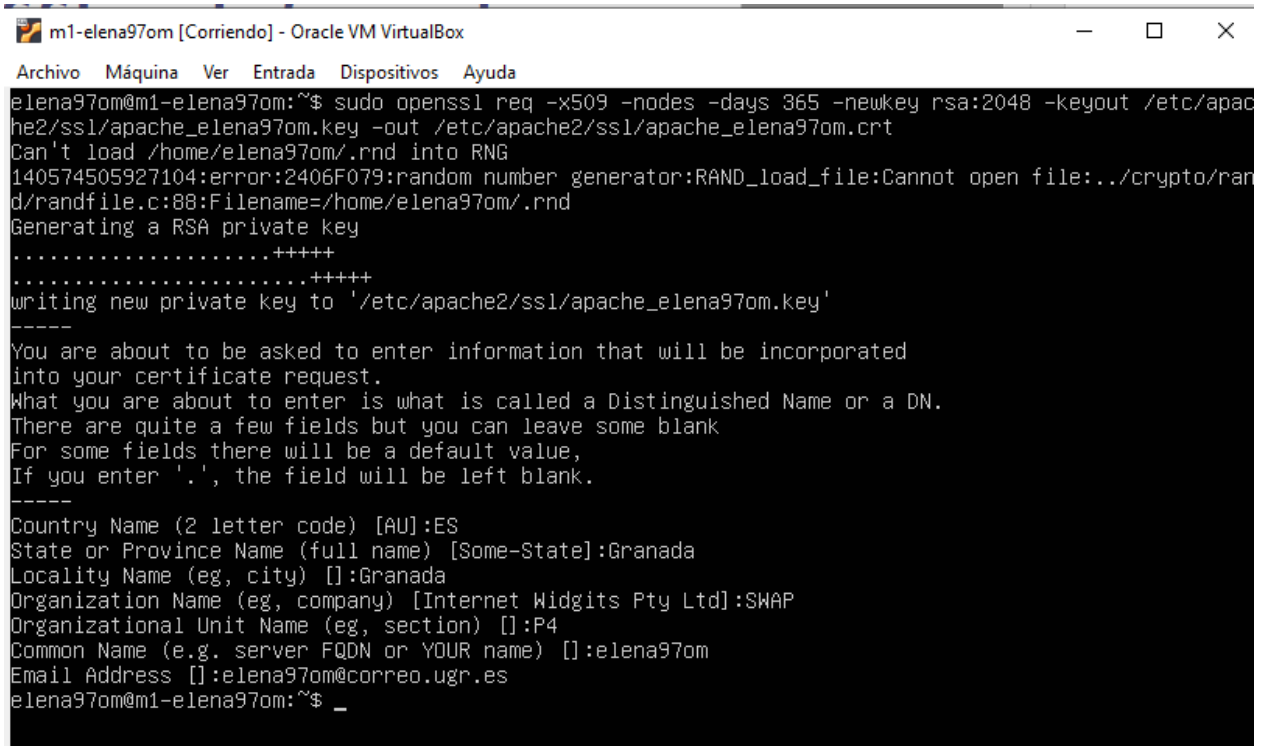
```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m1-elena97om:~$ sudo a2enmod ssl & sudo service apache2 restart
[3] 7513
[sudo] password for elena97om:
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
[3] Done                                sudo a2enmod ssl
```

## 2. Creamos el directorio SSL



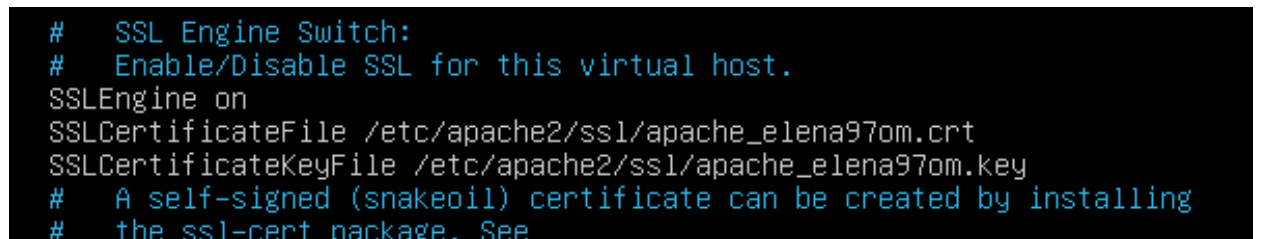
```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m1-elena97om:~$ sudo mkdir /etc/apache2/ssl
elena97om@m1-elena97om:~$ _
```

### 3. Generamos el certificado y lo configuramos



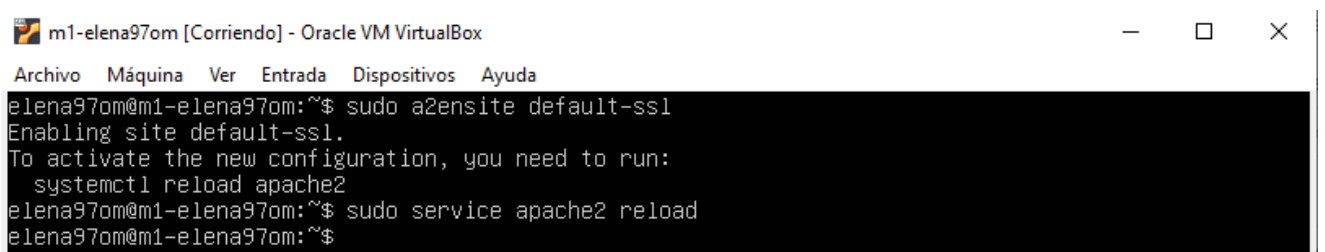
```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m1-elena97om:~$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apac
he2/ssl/apache_elena97om.key -out /etc/apache2/ssl/apache_elena97om.crt
Can't load /home/elena97om/.rnd into RNG
140574505927104:error:2406F079:random number generator:RAND_load_file:Cannot open file:../crypto/ran
d/randfile.c:88:Filename=/home/elena97om/.rnd
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/apache2/ssl/apache_elena97om.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:ES
State or Province Name (full name) [Some-State]:Granada
Locality Name (eg, city) []:Granada
Organization Name (eg, company) [Internet Widgits Pty Ltd]:SWAP
Organizational Unit Name (eg, section) []:P4
Common Name (e.g. server FQDN or YOUR name) []:elena97om
Email Address []:elena97om@correo.ugr.es
elena97om@m1-elena97om:~$ _
```

### 4. Editamos el archivo de configuración del sitio default-ssl y agregamos la ruta de los certificados



```
# SSL Engine Switch:
# Enable/Disable SSL for this virtual host.
SSLEngine on
SSLCertificateFile /etc/apache2/ssl/apache_elena97om.crt
SSLCertificateKeyFile /etc/apache2/ssl/apache_elena97om.key
# A self-signed (snakeoil) certificate can be created by installing
# the ssl-cert package. See
```

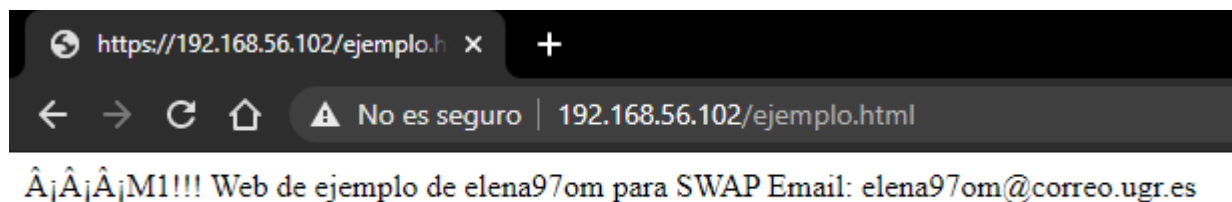
### 5. Activamos el sitio default-ssl y reiniciamos Apache



```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m1-elena97om:~$ sudo a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
  systemctl reload apache2
elena97om@m1-elena97om:~$ sudo service apache2 reload
elena97om@m1-elena97om:~$
```

## 6. Comprobamos que el certificado se ha creado correctamente

```
elena@DESKTOP-ADLCI6E: ~  
elena@DESKTOP-ADLCI6E:~$ curl -k https://192.168.56.102/ejemplo.html  
<HTML>  
<BODY>  
¡¡¡M1!!!  
Web de ejemplo de elena97om para SWAP  
Email: elena97om@correo.ugr.es  
</BODY>  
</HTML>  
elena@DESKTOP-ADLCI6E:~$
```



## 7. Por último, configuramos el balanceador para que acepte también este tráfico. Para ello, copiamos los archivos .crt y .key al resto de máquinas de la granja web.

```
m1-elena97om [Corriendo] - Oracle VM VirtualBox  
Archivo Máquina Ver Entrada Dispositivos Ayuda  
elena97om@m1-elena97om:~$ sudo scp /etc/apache2/ssl/apache_elena97om.crt elena97om@192.168.56.103:/home/elena97om/apache_elena97om.crt  
elena97om@192.168.56.103's password:  
apache_elena97om.crt 100% 1432 766.6KB/s 00:00  
elena97om@m1-elena97om:~$ sudo scp /etc/apache2/ssl/apache_elena97om.key elena97om@192.168.56.103:/home/elena97om/apache_elena97om.key  
elena97om@192.168.56.103's password:  
apache_elena97om.key 100% 1704 900.4KB/s 00:00  
elena97om@m1-elena97om:~$ _
```

```
m1-elena97om [Corriendo] - Oracle VM VirtualBox  
Archivo Máquina Ver Entrada Dispositivos Ayuda  
elena97om@m1-elena97om:~$ sudo scp /etc/apache2/ssl/apache_elena97om.crt elena97om@192.168.56.104:/home/elena97om/apache_elena97om.crt  
elena97om@192.168.56.104's password:  
apache_elena97om.crt 100% 1432 704.3KB/s 00:00  
elena97om@m1-elena97om:~$ sudo scp /etc/apache2/ssl/apache_elena97om.key elena97om@192.168.56.104:/home/elena97om/apache_elena97om.key  
elena97om@192.168.56.104's password:  
apache_elena97om.key 100% 1704 682.1KB/s 00:00  
elena97om@m1-elena97om:~$ _
```

8. Tras copiar los dos archivos a mi home de M2 y M3, creamos la carpeta ssl en /etc/apache2 y movemos los archivos en M2

```
m2-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m2-elena97om:/etc/apache2$ sudo mkdir ssl
elena97om@m2-elena97om:/etc/apache2$ sudo mv /home/elena97om/apache_elena97om.* /etc/apache2/ssl/
elena97om@m2-elena97om:/etc/apache2$ cd ssl/
elena97om@m2-elena97om:/etc/apache2/ssl$ ls
apache_elena97om.crt  apache_elena97om.key
elena97om@m2-elena97om:/etc/apache2/ssl$ _
```

9. Ahora configuramos en M2 default-ssl.conf, activamos el sitio default-ssl y reiniciamos apache

```
m2-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m2-elena97om:/ $ sudo a2enmod ssl & sudo service apache2 restart
[1] 2774
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
[1]+  Done                  sudo a2enmod ssl
elena97om@m2-elena97om:/ $
```

```
m2-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m2-elena97om:/ $ sudo a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
  systemctl reload apache2
elena97om@m2-elena97om:/ $ systemctl reload apache2
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to reload 'apache2.service'.
Authenticating as: Elena Ortiz Moreno (elena97om)
Password:
==== AUTHENTICATION COMPLETE ====
elena97om@m2-elena97om:/ $ sudo a2ensite default-ssl
Site default-ssl already enabled
```

10. Finalmente modificamos en M3 el archivo `/etc/nginx/conf.d/default.conf` añadiendo lo siguiente

```
listen 80;
listen 443 ssl;
ssl on;
ssl_certificate /home/elena97om/ssl/apache_elena97om.crt;
ssl_certificate_key /home/elena97om/ssl/apache_elena97om.key;
server_name balanceador_elena97om;
```

11. Y comprobamos que podemos hacer peticiones por https a la IP del balanceador M3

```
elena@DESKTOP-ADLCI6E: ~
elena@DESKTOP-ADLCI6E:~$ curl -k https://192.168.56.104/ejemplo.html
<HTML>
<BODY>
!!!M2!!!
Web de ejemplo de elena97om para SWAP
Email: elena97om@correo.ugr.es
</BODY>
</HTML>
elena@DESKTOP-ADLCI6E:~$ curl -k https://192.168.56.104/ejemplo.html
<HTML>
<BODY>
!!!M1!!!
Web de ejemplo de elena97om para SWAP
Email: elena97om@correo.ugr.es
</BODY>
</HTML>
elena@DESKTOP-ADLCI6E:~$ curl -k https://192.168.56.104/ejemplo.html
<HTML>
<BODY>
!!!M2!!!
Web de ejemplo de elena97om para SWAP
Email: elena97om@correo.ugr.es
</BODY>
</HTML>
elena@DESKTOP-ADLCI6E:~$ curl -k https://192.168.56.104/ejemplo.html
<HTML>
<BODY>
!!!M1!!!
Web de ejemplo de elena97om para SWAP
Email: elena97om@correo.ugr.es
</BODY>
</HTML>
```

## 2. Configuración del cortafuegos:

Un cortafuegos es un dispositivo colocado entre subredes que se encarga de diferentes tareas para el manejo de paquetes. Es esencial para proteger la granja web de accesos indebidos, ya que permite el tráfico autorizado y deniega el resto.

Iptables es una herramienta para definir reglas de filtrado de paquetes, de traducción de direcciones de red y para mantener registros de log.

Para configurar un cortafuegos vamos a crear un script en M1 que se ejecutará cada vez que arranque el sistema.

```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 2.9.3 /home/elena97om/iptables.sh

#!/bin/bash

#(1) Eliminar todas las reglas (configuración limpia)
iptables -F
iptables -X
iptables -Z
iptables -t nat -F

#(2) Política por defecto: denegar todo el tráfico entrante
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
iptables -A INPUT -m state --state NEW,ESTABLISHED -j ACCEPT

#(3) Permitir cualquier acceso desde localhost (interface lo)
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

#(4) Abrir el puerto 22 para permitir el acceso por SSH
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 22 -j ACCEPT

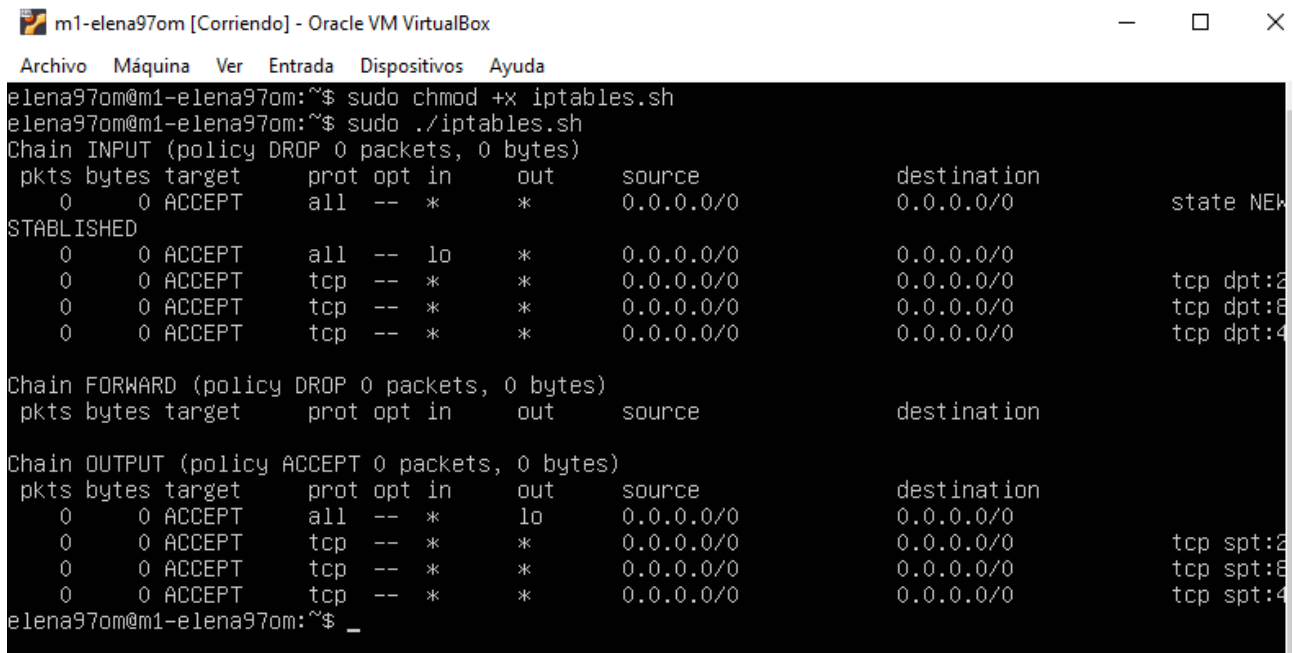
#(5) Permitir el tráfico por el puerto 80 (HTTP)
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 80 -j ACCEPT

#(6) Permitir el tráfico por el puerto 443 (HTTPS)
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 443 -j ACCEPT

iptables -L -n -v
```



Ahora ejecutamos el script y comprobamos si se han aplicado las reglas correctamente.



```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m1-elena97om:~$ sudo chmod +x iptables.sh
elena97om@m1-elena97om:~$ sudo ./iptables.sh
Chain INPUT (policy DROP 0 packets, 0 bytes)
pkts bytes target      prot opt in     out     source            destination
 0      0 ACCEPT      all  --  *      *       0.0.0.0/0         0.0.0.0/0         state NEW
STABLISHED
 0      0 ACCEPT      all  --  lo     *       0.0.0.0/0         0.0.0.0/0
 0      0 ACCEPT      tcp  --  *      *       0.0.0.0/0         0.0.0.0/0         tcp dpt:2
 0      0 ACCEPT      tcp  --  *      *       0.0.0.0/0         0.0.0.0/0         tcp dpt:8
 0      0 ACCEPT      tcp  --  *      *       0.0.0.0/0         0.0.0.0/0         tcp dpt:4
Chain FORWARD (policy DROP 0 packets, 0 bytes)
pkts bytes target      prot opt in     out     source            destination
Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in     out     source            destination
 0      0 ACCEPT      all  --  *      lo     0.0.0.0/0         0.0.0.0/0
 0      0 ACCEPT      tcp  --  *      *       0.0.0.0/0         0.0.0.0/0         tcp spt:2
 0      0 ACCEPT      tcp  --  *      *       0.0.0.0/0         0.0.0.0/0         tcp spt:8
 0      0 ACCEPT      tcp  --  *      *       0.0.0.0/0         0.0.0.0/0         tcp spt:4
elena97om@m1-elena97om:~$ _
```

## 4. Opciones avanzadas:

1. Configurar M3 para que sea el único que acepte peticiones HTTP y HTTPS mientras que M1 y M2 solo aceptan peticiones si vienen de M3.

Primero configuramos M1 y M2 para que solo acete peticiones de M3, editando el script de M1 y creando uno igual en M2.

Añadimos parámetros a las reglas ya descritas donde especificamos la IP origen (-s) y la IP destino (-d), correspondientes a M3. Utilizamos los parámetros -m multiport para especificar los tres puertos en la misma regla.

```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 2.9.3 iptables.sh

#!/bin/bash

#(1) Eliminar todas las reglas (configuración limpia)
iptables -F
iptables -X
iptables -Z
iptables -t nat -F

#(2) Política por defecto: denegar todo el tráfico entrante
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP

#(4) Permitir el tráfico con el balanceador por SSH, HTTP y HTTPS
iptables -A INPUT -s 192.168.56.104 -p tcp -m multiport --dport 22,80,443 -j ACCEPT
iptables -A OUTPUT -d 192.168.56.104 -p tcp -m multiport --sport 22,80,443 -j ACCEPT

iptables -L -n -v
```

Lo ejecutamos de nuevo y comprobamos en la salida que todas las reglas se han aplicado correctamente.

```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m1-elena97om:~$ sudo ./iptables.sh
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source            destination
  0      0 ACCEPT    tcp -- *      *       192.168.56.104    0.0.0.0/0         multiport
 ports 22,80,443

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source            destination

Chain OUTPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source            destination
  0      0 ACCEPT    tcp -- *      *       0.0.0.0/0         192.168.56.104    multiport
 ports 22,80,443
elena97om@m1-elena97om:~$ _
```

Comprobamos que Solo podemos establecer una conexión desde M3

 elena@DESKTOP-ADLCI6E: ~

```
elena@DESKTOP-ADLCI6E:~$ curl http://192.168.56.102/ejemplo.html
curl: (7) Failed to connect to 192.168.56.102 port 80: Connection refused
elena@DESKTOP-ADLCI6E:~$
```

 m3-elena97om [Corriendo] - Oracle VM VirtualBox


Archivo Máquina Ver Entrada Dispositivos Ayuda

```
elena97om@m3-elena97om:~$ curl http://192.168.56.102/ejemplo.html
<HTML>
<BODY>
iiiM1!!!
Web de ejemplo de elena97om para SWAP
Email: elena97om@correo.ugr.es
</BODY>
</HTML>
elena97om@m3-elena97om:~$
```

Y al comprobar que funciona hacemos lo mismo en M2.

 elena@DESKTOP-ADLCI6E: ~

```
elena@DESKTOP-ADLCI6E:~$ curl http://192.168.56.103/ejemplo.html
curl: (7) Failed to connect to 192.168.56.103 port 80: Connection refused
elena@DESKTOP-ADLCI6E:~$
```

 m3-elena97om [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

```
elena97om@m3-elena97om:~$ curl http://192.168.56.103/ejemplo.html
<HTML>
<BODY>
iiiM2!!!
Web de ejemplo de elena97om para SWAP
Email: elena97om@correo.ugr.es
</BODY>
</HTML>
elena97om@m3-elena97om:~$
```

Ahora vamos a configurar M3 para que sea el único que reciba peticiones HTTP y HTTPS. Para ello creamos el siguiente script, que será muy parecido al primero que hemos creado.

```
m3-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 2.9.3  iptables.sh  Modified

#!/bin/bash

#(1) Eliminar todas las reglas
iptables -F
iptables -X
iptables -Z
iptables -t nat -F

#(2) Denegar todo el tráfico entrante
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
iptables -A INPUT -m state --state NEW,ESTABLISHED -j ACCEPT

#(3) Permitir tráfico HTTP y HTTPS
iptables -I INPUT -p tcp -m multiport --dports 80,443 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -I OUTPUT -p tcp -m multiport --sports 80,443 -m state --state NEW,ESTABLISHED -j ACCEPT

iptables -L -n -v
```

Y tras ejecutarlo comprobamos que las reglas son correctas.

```
m3-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m3-elena97om:~$ sudo ./iptables.sh
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source    destination
  0      0 ACCEPT    tcp  --  *      *       0.0.0.0/0  0.0.0.0/0      multiport d
ports 80,443 state NEW,ESTABLISHED
  0      0 ACCEPT    all  --  *      *       0.0.0.0/0  0.0.0.0/0      state NEW,E
STABLISHED

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source    destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source    destination
  0      0 ACCEPT    tcp  --  *      *       0.0.0.0/0  0.0.0.0/0      multiport s
ports 80,443 state NEW,ESTABLISHED
elena97om@m3-elena97om:~$ _
```

Comprobamos que se realizan bien las peticiones.

```
elena@DESKTOP-ADLCI6E: ~  
elena@DESKTOP-ADLCI6E:~$ curl http://192.168.56.104  
<html>  
<head><title>400 The plain HTTP request was sent to HTTPS port</title></head>  
<body bgcolor="white">  
<center><h1>400 Bad Request</h1></center>  
<center>The plain HTTP request was sent to HTTPS port</center>  
<hr><center>nginx/1.14.0 (Ubuntu)</center>  
</body>  
</html>  
elena@DESKTOP-ADLCI6E:~$ curl -k https://192.168.56.104/ejemplo.html  
<HTML>  
<BODY>  
!!!M2!!!  
Web de ejemplo de elena97om para SWAP  
Email: elena97om@correo.ugr.es  
</BODY>  
</HTML>  
elena@DESKTOP-ADLCI6E:~$ curl -k https://192.168.56.104/ejemplo.html  
<HTML>  
<BODY>  
!!!M1!!!  
Web de ejemplo de elena97om para SWAP  
Email: elena97om@correo.ugr.es  
</BODY>  
</HTML>  
elena@DESKTOP-ADLCI6E:~$
```

2. Por último, vamos a hacer que la configuración del cortafuegos se ejecute al arranque del sistema en todas las máquinas. Creamos el servicio o demonio de la siguiente manera.

```
m1-elena97om [Corriendo] - Oracle VM VirtualBox  
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda  
GNU nano 2.9.3  iptables.service  
  
[Unit]  
After=network-online.target  
  
[Service]  
ExecStart=/home/elena97om/iptables.sh  
  
[Install]  
WantedBy=default.target
```

Y activamos el demonio con los siguientes comandos.

```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m1-elena97om:/etc/systemd/system$ sudo systemctl daemon-reload
elena97om@m1-elena97om:/etc/systemd/system$ sudo systemctl enable iptables.service
elena97om@m1-elena97om:/etc/systemd/system$ _
```

Ahora reiniciamos M1 y comprobamos los puertos que se están escuchando y la configuración del cortafuegos, que son correctos.

```
m1-elena97om [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
elena97om@m1-elena97om:~$ sudo iptables -L -n -v
[sudo] password for elena97om:
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source               destination
  0      0 ACCEPT    tcp  --  *      *       192.168.56.104        0.0.0.0/0             multiport d
ports 22,80,443

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source               destination

Chain OUTPUT (policy DROP 80 packets, 5880 bytes)
 pkts bytes target    prot opt in     out     source               destination
  0      0 ACCEPT    tcp  --  *      *       0.0.0.0/0            192.168.56.104        multiport s
ports 22,80,443
elena97om@m1-elena97om:~$ _
```

Esta configuración del arranque se lleva a cabo igualmente en M2 y M3.