

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES

Seguridad Informática

20/11/2014

Brais López Yáñez

ÍNDICE

Objetivos de la práctica	3
Desarrollo.....	3
Ejercicio 1.....	5
Script final ("iptables-drop2.sh")	9
Explicación de las reglas IPTABLES en el script final	13
Ejercicio 2: opcional.....	15
Conclusiones Finales	18

OBJETIVOS DE LA PRÁCTICA

Se proponen una serie de ejercicios para verificar la configuración y operación del cortafuegos incorporado en el kernel de Linux: NETFILTER/IPTABLES.

El entorno de trabajo consiste en un cortafuegos básico separando una red interna segura de una red externa no segura. Sobre este cortafuegos se implementarán distintas políticas de filtrado y traducción de direcciones.

DESARROLLO

Lo primero que vamos a hacer es montar toda la infraestructura necesaria para realizar la práctica, con los datos propuestos en la actividad.

- Para esto, vamos a reutilizar las imágenes utilizadas en la práctica anterior texto.dvi y grafico.dvi.
- Asignamos los discos como inmutable, en las máquinas externo e interno.
- Configuramos la red de cada máquina, con las IPs de la siguiente imagen.

Infraestructura de la práctica:

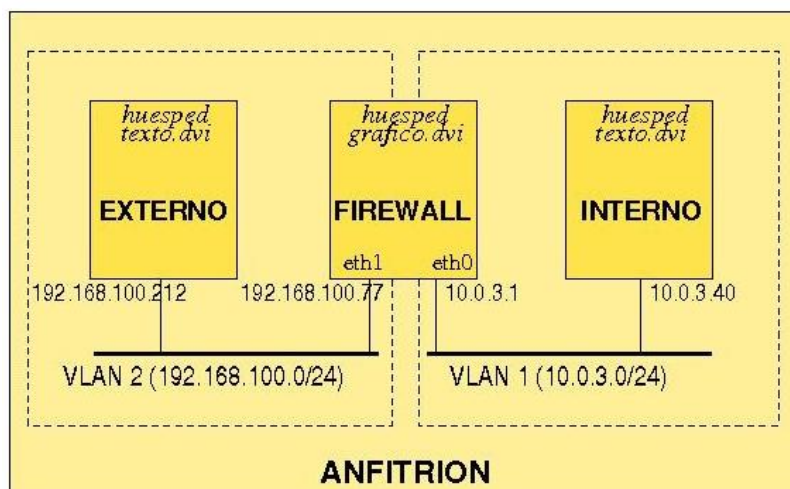
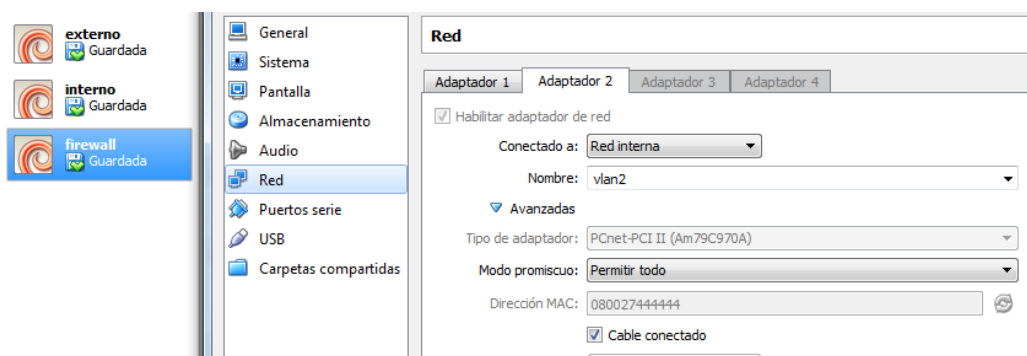


Imagen de la configuración:



Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES

2014

Tras la configuración inicial, ejecutamos los siguientes comandos en cada una de ellas para finalizar con la configuración requerida:

```
interno
MAC 08:00:27:11:11:11 (eth0)
# ifconfig eth0 10.0.3.40
# hostname interno
# route add default gw 10.0.3.1
```

```
externo
MAC 08:00:27:22:22:22 (eth0)
# ifconfig eth0 192.168.100.212
# hostname externo
```

```
firewall
MAC 08:00:27:33:33:33 (eth0)
MAC 08:00:27:44:44:44 (eth1)
# ifconfig eth0 10.0.3.1
# ifconfig eth1 192.168.100.77
# hostname firewall
# echo 1 > /proc/sys/net/ipv4/ip_forward
```

Una vez configuradas las máquinas, lo que haremos será iniciar los servicios necesarios: APACHE y TELNET.

```
externo:~# /etc/init.d/inetutils-inetd start
Starting internet superserver: inetd.
externo:~# /etc/init.d/apache2 start
Starting web server: apache2
apache2: apr_sockaddr_info_get() failed for externo
apache2: Could not reliably determine the server's fully qualified domain name,
using 127.0.0.1 for ServerName
```

```
interno:~# /etc/init.d/apache2 start
Starting web server: apache2
apache2: apr_sockaddr_info_get() failed for interno
apache2: Could not reliably determine the server's fully qualified domain name,
using 127.0.0.1 for ServerName
```

Modificación de la página por defecto de APACHE, en la máquina externo.

```
<html>
<head>
<title> Pagina web de prueba </title>
<body>
<h1>Pagina web de prueba en texto.dvi</h1>
<ul>
<li> Aquí externo! Hola amigo! Soy externo! </li>
<li> eN La casa de externo </li>
</ul>
</body>
</html>
```

[11 líneas escritas]

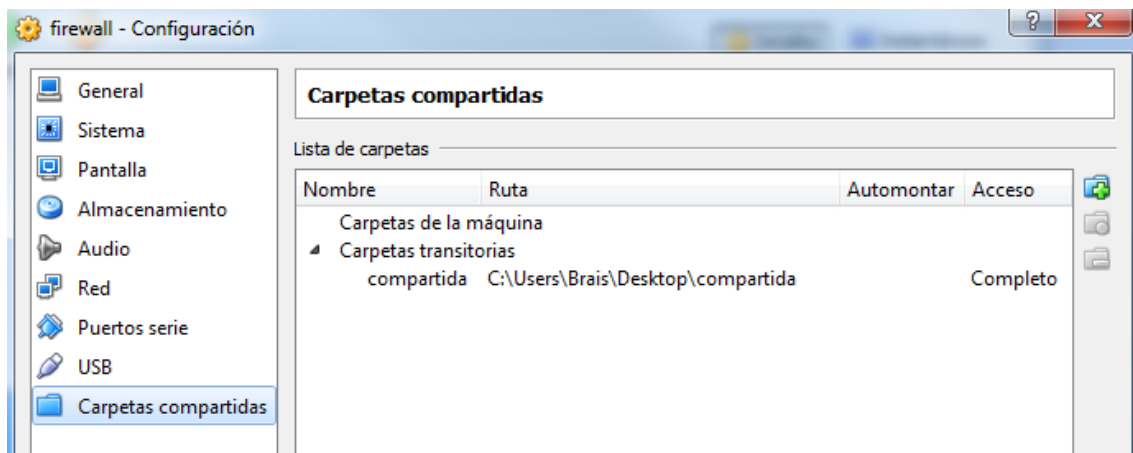
```
externo:~# _
```

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES 2014

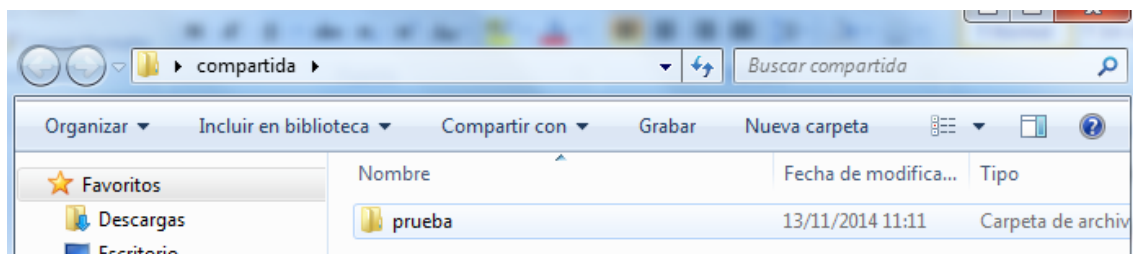
Configuración y montaje de la carpeta compartida en firewall:

```
firewall:~# mkdir /mnt/compartida
firewall:~# mount -t vboxsf compartida /mnt/compartida
firewall:~# mkdir /mnt/compartida/prueba
firewall:~# _
```

Además de esto, en el virtual también añadimos la ruta correspondiente a la carpeta que se compartirá:



Estado inicial de la carpeta compartida.



EJERCICIO 1

Una vez tenemos la configuración finalizada, ya procedemos a realizar los ejercicios correspondientes. Lo primero que haremos será ejecutar el script de restauración:

iptables-inicial.sh

Que va a ser el script de punto de partida, y que si hacemos algo mal, volveremos al paso inicial mediante su ejecución.

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES 2014

Tras esto verificamos la configuración actual, tras ejecutar el script:

```
firewall:/# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
firewall:/#
```

Luego de probar, el script de restauración procedemos a seguir los pasos de la práctica y creamos un nuevo script:

iptables-drop-1.sh

Este script, lo almacenamos en la carpeta compartida, ya que así lo podemos editar desde Windows para completarlo. Tras finalizar con el script, procedemos a ejecutarlo en la máquina firewall.

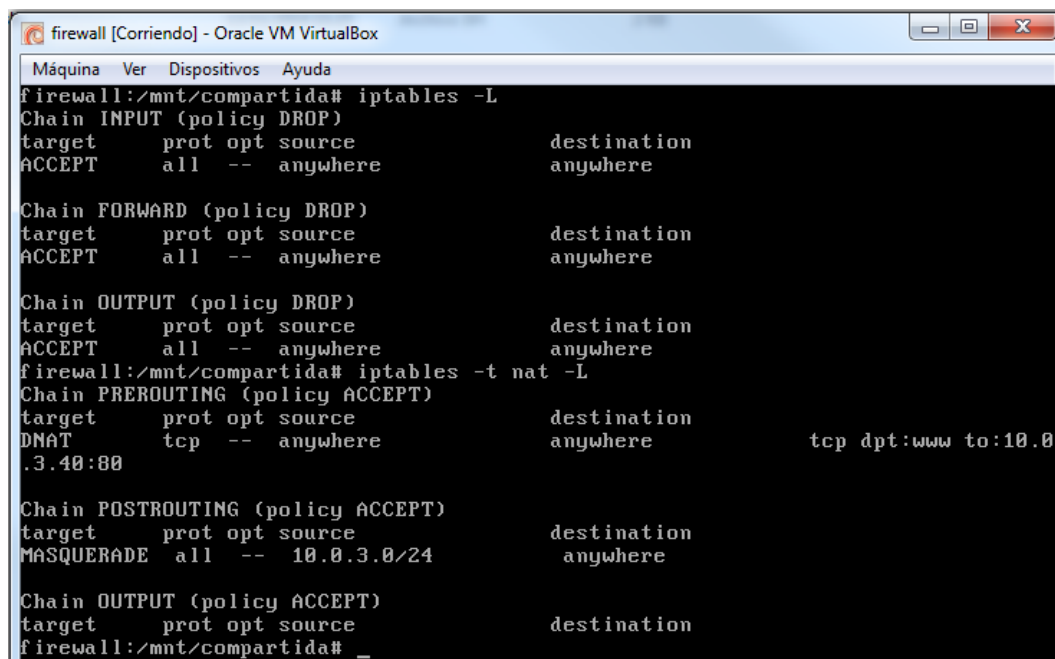
```
#!/bin/sh
# Establecer variables: red interna y red externa
export INTERNAL_NETWORK=eth0
export EXTERNAL_NETWORK=eth1
# Vaciar y reiniciar tablas
iptables -F
iptables -X
iptables -Z
iptables -t nat -F
# Establecer políticas por defecto (denegar por defecto: DROP)
iptables -P INPUT DROP # discard firewall inputs
iptables -P OUTPUT DROP # discard firewall outputs
iptables -P FORWARD DROP # discard forwarding traffic through the firewall
iptables -t nat -P PREROUTING ACCEPT
iptables -t nat -P POSTROUTING ACCEPT
# Permitirle todo al localhost (firewall)
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

iptables -t nat -A POSTROUTING -s 10.0.3.0/24 -o $EXTERNAL_NETWORK -j MASQUERADE
# DNAT (servicio HTTP [puerto 80] redireccionado a la red interna)
iptables -t nat -A PREROUTING -i $EXTERNAL_NETWORK -p tcp --dport 80 -j DNAT --to-destination 10.0.3.40:80
# Habilitar redireccionamiento de paquetes
echo 1 > /proc/sys/net/ipv4/ip_forward
# REGLA TEMPORAL: para pruebas de NAT
iptables -A FORWARD -j ACCEPT
```

Para asegurarnos del correcto funcionamiento, procedemos a realizar la verificación de la configuración actual del firewall:

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES

2014



```
firewall:/mnt/compartida# iptables -L
Chain INPUT (policy DROP)
target     prot opt source                destination
ACCEPT     all  --  anywhere              anywhere

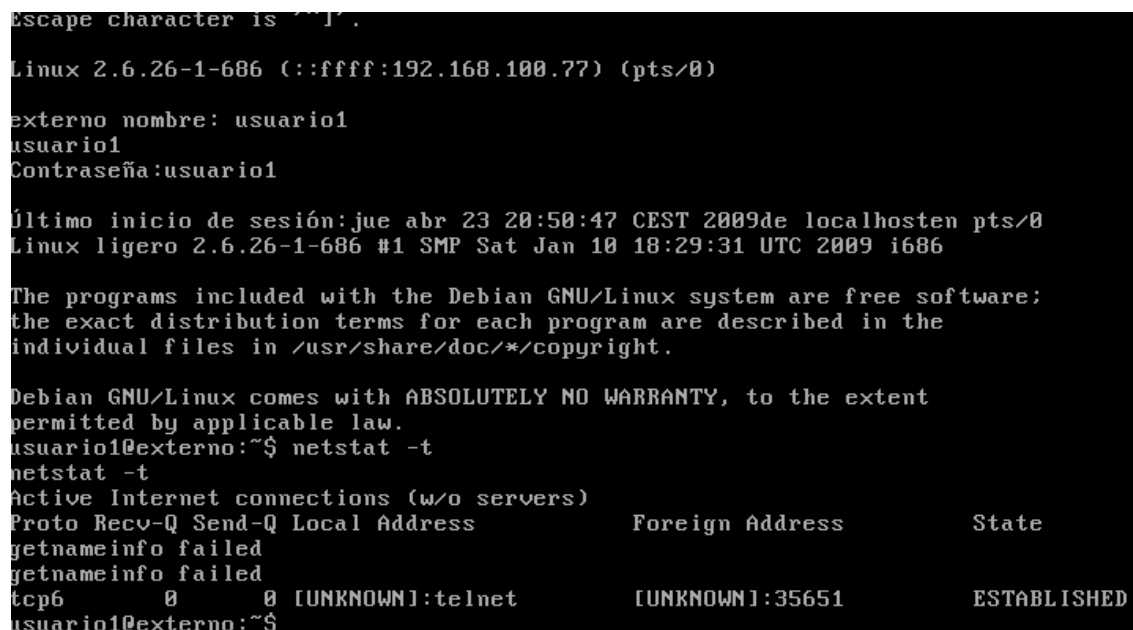
Chain FORWARD (policy DROP)
target     prot opt source                destination
ACCEPT     all  --  anywhere              anywhere

Chain OUTPUT (policy DROP)
target     prot opt source                destination
ACCEPT     all  --  anywhere              anywhere
firewall:/mnt/compartida# iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source                destination
DNAT       tcp  --  anywhere              anywhere            tcp dpt:www to:10.0.3.40:80

Chain POSTROUTING (policy ACCEPT)
target     prot opt source                destination
MASQUERADE all  --  10.0.3.0/24           anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
firewall:/mnt/compartida#
```

Establecemos una conexión telnet desde interno (10.0.3.40) a externo (192.168.100.212)



```
Linux 2.6.26-1-686 (::ffff:192.168.100.77) (pts/0)
externo nombre: usuario1
usuario1
Contraseña:usuario1

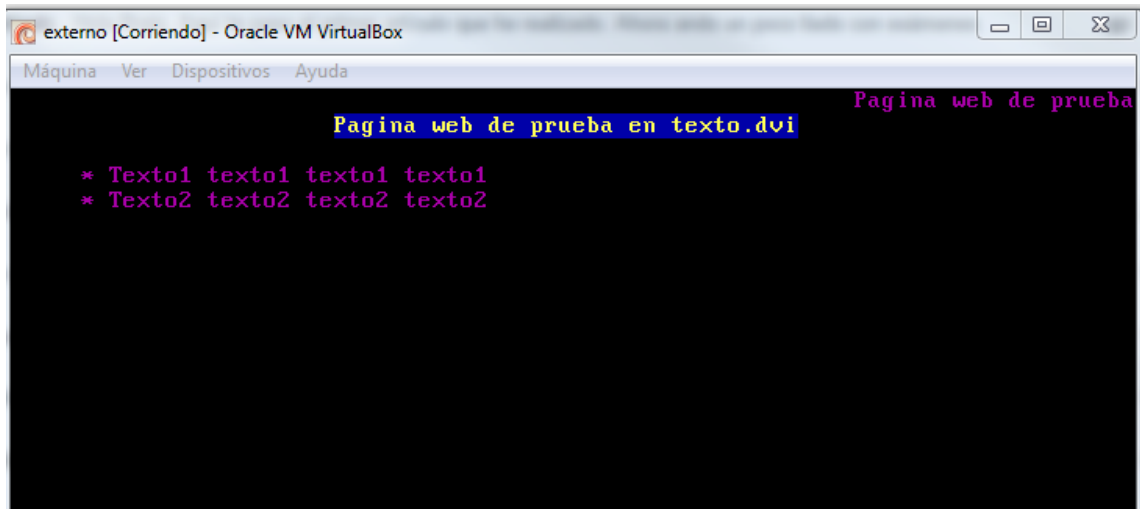
Último inicio de sesión:jue abr 23 20:50:47 CEST 2009de localhosten pts/0
Linux ligero 2.6.26-1-686 #1 SMP Sat Jan 10 18:29:31 UTC 2009 i686

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

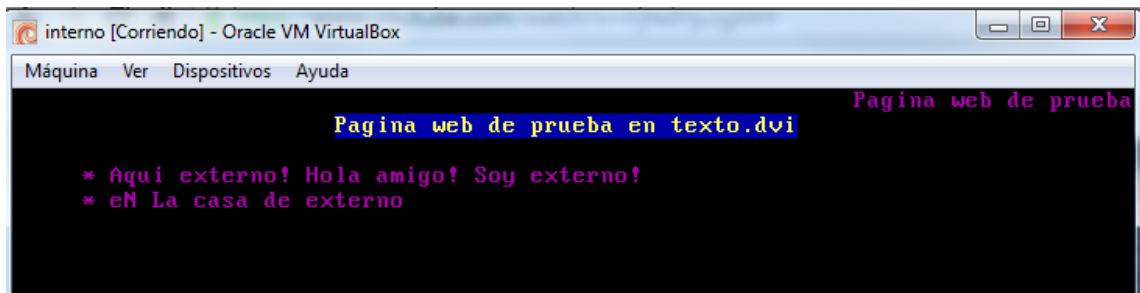
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
usuario1@externo:~$ netstat -t
netstat -t
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address          State
tcp6      0      0 [UNKNOWN]:telnet        [UNKNOWN]:35651         ESTABLISHED
usuario1@externo:~$
```

Establecemos una conexión web desde externo (192.168.100.212) al firewall (192.168.100.77)

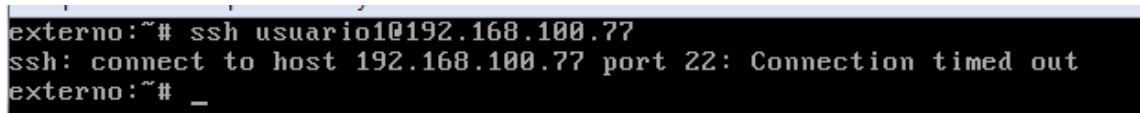
Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES 2014



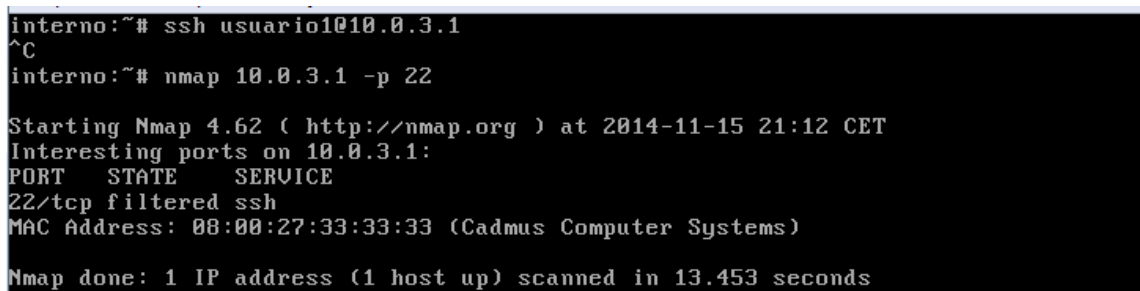
Establecemos conexión web desde interno (10.0.3.40) a externo (192.168.100.212)



Conexión al firewall mediante SSH desde externo (192.168.100.212) e interno (10.0.3.40)



La conexión de externo al firewall es demasiada lenta, caduca antes de realizarse.



Otra posibilidad (más rápida): escaneo de puertos con NMAP (puerto 22).

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES

2014

Ejemplo de nmap en externo:

```
externo:~# nmap 192.168.100.77 -p 22

Starting Nmap 4.62 ( http://nmap.org ) at 2014-11-15 21:18 CET
Interesting ports on 192.168.100.77:
PORT      STATE      SERVICE
22/tcp    filtered  ssh
MAC Address: 08:00:27:44:44:44 (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 13.419 seconds
```

SCRIPT FINAL ("IPTABLES-DROP2.SH")

Una vez completado el ejercicio1, y por lo tanto las actividades correspondientes a iptables-drop1.sh procedemos a realizar el siguiente script, con los pasos planteados en la práctica.

Que además de lo que contiene el script del ejercicio 1, al script *iptables-drop2.sh*, se le añaden una serie de restricciones y permisos.

```
# Limitar tráfico ICMP (permitir como máximo 5 peticiones/second)
iptables -A INPUT -p icmp -m limit --limit 5/second -j ACCEPT
iptables -A OUTPUT -p icmp -m limit --limit 5/second -j ACCEPT
iptables -A FORWARD -p icmp -m limit --limit 5/second -j ACCEPT

# FILTRO DE ENTRADA PARA LA RED INTERNA
# - permitir redireccionamiento de servicios (peticiones + sus respuestas)
# [necesario porque usamos DROP por defecto]
iptables -A FORWARD -i $EXTERNAL_NETWORK -d 10.0.3.40 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.40 -p tcp --sport 80 -m state --state ESTABLISHED,RELATED -j ACCEPT
# - registrar (log) los otros accesos a la red interna (denegados por defecto)
iptables -A FORWARD -i $EXTERNAL_NETWORK -d 10.0.3.0/24 -j LOG --log-prefix "Acceso a la red interna:"

# # FILTRO DE SALIDA PARA LA RED INTERNA
# - permitir conexiones HTTP salientes + sus respuestas
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.0/24 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -o $INTERNAL_NETWORK -d 10.0.3.0/24 -p tcp --sport 80 -m state --state ESTABLISHED,RELATED -j ACCEPT
# - permitir peticiones DNS salientes + sus respuestas
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.0/24 -p tcp --dport 53 -j ACCEPT
iptables -A FORWARD -o $INTERNAL_NETWORK -d 10.0.3.0/24 -p tcp --sport 53 -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.0/24 -p udp --dport 53 -j ACCEPT
iptables -A FORWARD -o $INTERNAL_NETWORK -d 10.0.3.0/24 -p udp --sport 53 -m state --state ESTABLISHED,RELATED -j ACCEPT
# - rechazar cualquier otro intento de salida (DROP por defecto)
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.0/24 -j REJECT --reject-with icmp-port-unreachable

# FILTRADO DE CONEXIONES HACIA EL FIREWALL
# - permitir tráfico SSH de entrada y salida desde la red interna,
# el resto, bloqueado por la política por defecto
iptables -A INPUT -i $INTERNAL_NETWORK -s 10.0.3/24 -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -o $INTERNAL_NETWORK -d 10.0.3/24 -p tcp --sport 22 -j ACCEPT
# - registrar (log) los intentos de acceso al firewall desde la red externa
# (serán denegados)
iptables -A INPUT -i $EXTERNAL_NETWORK -j LOG --log-prefix "Acceso al firewall:"
```

Una vez tenemos el script listo, procedemos a su verificación:

Verificar la configuración actual del firewall

firewall:~# iptables -L

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES 2014

```
Chain INPUT (policy DROP)
target     prot opt source                destination
ACCEPT     all  --  anywhere               anywhere
ACCEPT     icmp --  anywhere               anywhere           limit: avg 5/sec bu
rst 5
ACCEPT     tcp  --  10.0.3.0/24            anywhere           tcp dpt:ssh
LOG         all  --  anywhere               anywhere           LOG level warning p
refix 'Acceso al firewall:'

Chain FORWARD (policy DROP)
target     prot opt source                destination
ACCEPT     icmp --  anywhere               anywhere           limit: avg 5/sec bu
rst 5
ACCEPT     tcp  --  anywhere               10.0.3.40          tcp dpt:www
ACCEPT     tcp  --  10.0.3.40              anywhere           tcp spt:www state R
ELATED,ESTABLISHED
LOG         all  --  anywhere               10.0.3.0/24        LOG level warning p
refix 'Acceso a la red interna:'
ACCEPT     tcp  --  10.0.3.0/24            anywhere           tcp dpt:www
ACCEPT     tcp  --  anywhere               10.0.3.0/24        tcp spt:www state R
ELATED,ESTABLISHED
ACCEPT     tcp  --  10.0.3.0/24            anywhere           tcp dpt:domain
ACCEPT     tcp  --  anywhere               10.0.3.0/24        tcp spt:domain stat
e RELATED,ESTABLISHED
:_
```

firewall:~# iptables -t nat -L

```
firewall:~# iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source                destination
DNAT       tcp  --  anywhere               anywhere           tcp dpt:www to:10.0
.3.40:80

Chain POSTROUTING (policy ACCEPT)
target     prot opt source                destination
MASQUERADE all  --  10.0.3.0/24            anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
firewall:~# _
```

Conectarse al firewall mediante SSH desde externo (192.168.100.212) e interno (10.0.3.40)

interno:~# ssh usuario1@10.0.3.1

```
interno:~# ssh usuario1@10.0.3.1
ssh: connect to host 10.0.3.1 port 22: Connection timed out
interno:~# _
```

Al revés del anterior ejercicio, justo cuando queremos hacer la conexión desde interno a firewall, tenemos un fallo de tiempo. Esto es debido a una restricción que implantamos en el script, que no deja conectarse al firewall desde la red externa.

externo:~# ssh usuario1@192.168.100.77

```
externo:~# ssh usuario1@192.168.100.77
ssh: connect to host 192.168.100.77 port 22: Connection timed out
externo:~# nmap 192.168.100.77 -p 22

Starting Nmap 4.62 ( http://nmap.org ) at 2014-11-15 21:18 CET
Interesting ports on 192.168.100.77:
PORT      STATE      SERVICE
22/tcp    filtered  ssh
MAC Address: 08:00:27:44:44:44 (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 13.419 seconds
externo:~# ssh usuario1@192.168.100.77
ssh: connect to host 192.168.100.77 port 22: Connection timed out
externo:~# _
```

Otra opción más rápida: escaneo de puertos con NMAP (puerto 22).

interno:~# nmap 10.0.3.1 -p 22

```
interno:~# nmap 10.0.3.1 -p 22

Starting Nmap 4.62 ( http://nmap.org ) at 2014-11-16 00:52 CET
Interesting ports on 10.0.3.1:
PORT      STATE      SERVICE
22/tcp    filtered  ssh
MAC Address: 08:00:27:33:33:33 (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 13.389 seconds
interno:~# _
```

externo:~# nmap 192.168.100.77 -p 22

```
externo:~# nmap 192.168.100.77 -p 22

Starting Nmap 4.62 ( http://nmap.org ) at 2014-11-16 00:54 CET
Interesting ports on 192.168.100.77:
PORT      STATE      SERVICE
22/tcp    filtered  ssh
MAC Address: 08:00:27:44:44:44 (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 13.387 seconds
externo:~# _
```

Conexión TELNET desde interno (10.0.3.40) a externo (192.168.100.212).

interno:~# telnet 192.168.100.212

```
interno:~# telnet 192.168.100.212
Trying 192.168.100.212...
telnet: Unable to connect to remote host: Connection timed out
interno:~# _
```

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES 2014

Al lanzar la conexión telnet, desde interno a externo podemos observar que nos sale otro error, y no es imposible de conectar. Esto es debido a que solo estamos permitiendo tráfico SSH de entrada y de salida desde la red interna.

Conexión web desde interno (10.0.3.40) a externo (192.168.100.212) y desde externo al firewall.

interno:~# lynx 192.168.100.212

```
Alert!: Unable to connect to remote host.

Looking up 192.168.100.212 first
Looking up 192.168.100.212
Making HTTP connection to 192.168.100.212
Alert!: Unable to connect to remote host.

lynx: Can't access startfile http://192.168.100.212/
```

```
Alert!: Unable to connect to remote host.

Looking up 192.168.100.77 first
Looking up 192.168.100.77
Making HTTP connection to 192.168.100.77
Alert!: Unable to connect to remote host.

lynx: Can't access startfile http://192.168.100.77/
externo:~#
```

Lo que acabamos de explicar, es el motivo que tampoco nos deje conectarnos mediante web a firewall. La conexión permanece bloqueada y no deja que nos conectemos.

Para concluir la práctica comprobamos el registro de accesos (/var/log/syslog) en firewall:

firewall:~# less /var/log/syslog

```
100.77 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=30819 DF PROTO=TCP SPT=36915 DPT=22 W
INDOW=5840 RES=0x00 SYN URG=0
Nov 16 00:38:19 observador kernel: [20714.988539] Acceso al firewall:IN=eth1 OUT
= MAC=08:00:27:44:44:08:00:27:22:22:22:08:00 SRC=192.168.100.212 DST=192.168.
100.77 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=30820 DF PROTO=TCP SPT=36915 DPT=22 W
INDOW=5840 RES=0x00 SYN URG=0
Nov 16 00:38:43 observador kernel: [20738.931988] Acceso al firewall:IN=eth1 OUT
= MAC=08:00:27:44:44:08:00:27:22:22:22:08:00 SRC=192.168.100.212 DST=192.168.
100.77 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=30821 DF PROTO=TCP SPT=36915 DPT=22 W
INDOW=5840 RES=0x00 SYN URG=0
Nov 16 00:39:30 observador kernel: [20786.818034] Acceso al firewall:IN=eth1 OUT
= MAC=08:00:27:44:44:08:00:27:22:22:22:08:00 SRC=192.168.100.212 DST=192.168.
100.77 LEN=60 TOS=0x00 PREC=0x00 TTL=64 ID=30822 DF PROTO=TCP SPT=36915 DPT=22 W
INDOW=5840 RES=0x00 SYN URG=0
Nov 16 00:52:28 observador kernel: [21564.297177] Acceso al firewall:IN=eth1 OUT
= MAC=08:00:27:44:44:08:00:27:22:22:22:08:00 SRC=192.168.100.212 DST=192.168.
100.77 LEN=44 TOS=0x00 PREC=0x00 TTL=56 ID=25108 PROTO=TCP SPT=46553 DPT=22 WIND
OW=1024 RES=0x00 SYN URG=0
Nov 16 00:52:28 observador kernel: [21564.348075] Acceso al firewall:IN=eth1 OUT
= MAC=08:00:27:44:44:08:00:27:22:22:22:08:00 SRC=192.168.100.212 DST=192.168.
100.77 LEN=44 TOS=0x00 PREC=0x00 TTL=47 ID=56855 PROTO=TCP SPT=46554 DPT=22 WIND
OW=4096 RES=0x00 SYN URG=0
Nov 16 01:17:01 observador /usr/sbin/cron[2371]: (root) CMD ( cd / && run-part
s --report /etc/cron.hourly)
firewall:/# _
```

En este archivo, se guardan todos los accesos al firewall y a la propia red de este (la red interna). Aquí quedan tanto las conexiones que se han realizado, como las que no.

EXPLICACIÓN DE LAS REGLAS IPTABLES EN EL SCRIPT FINAL

```
#!/bin/sh
# Establecer variables: red interna y red externa
export INTERNAL_NETWORK=eth0
export EXTERNAL_NETWORK=eth1
# Vaciar y reiniciar tablas
iptables -F
iptables -X
iptables -Z
iptables -t nat -F
# Establecer políticas por defecto (denegar por defecto: DROP)
iptables -P INPUT DROP # discard firewall inputs
iptables -P OUTPUT DROP # discard firewall outputs
iptables -P FORWARD DROP # discard forwarding traffic through the firewall
iptables -t nat -P PREROUTING ACCEPT
iptables -t nat -P POSTROUTING ACCEPT
# Permitirle todo al localhost (firewall)
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

iptables -t nat -A POSTROUTING -s 10.0.3.0/24 -o $EXTERNAL_NETWORK -j MASQUERADE
# DNAT (servicio HTTP [puerto 80] redireccionado a la red interna)
iptables -t nat -A PREROUTING -i $EXTERNAL_NETWORK -p tcp --dport 80 -j DNAT --to-destination
10.0.3.40:80
# Habilitar redireccionamiento de paquetes
echo 1 > /proc/sys/net/ipv4/ip_forward

# Limitar tráfico ICMP (permitir como máximo 5 peticiones/second)
iptables -A INPUT -p icmp -m limit --limit 5/second -j ACCEPT
iptables -A OUTPUT -p icmp -m limit --limit 5/second -j ACCEPT
```

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES

2014

```
iptables -A FORWARD -p icmp -m limit --limit 5/second -j ACCEPT
# FILTRO DE ENTRADA PARA LA RED INTERNA
# - permitir redireccionamiento de servicios (peticiones + sus respuestas)
# [necesario porque usamos DROP por defecto]
iptables -A FORWARD -i $EXTERNAL_NETWORK -d 10.0.3.40 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.40 -p tcp --sport 80 -m state --state ESTABLISHED,RELATED -j ACCEPT
# - registrar (log) los otros accesos a la red interna (denegados por defecto)
iptables -A FORWARD -i $EXTERNAL_NETWORK -d 10.0.3.0/24 -j LOG --log-prefix "Acceso a la red interna:"
## FILTRO DE SALIDA PARA LA RED INTERNA
# - permitir conexiones HTTP salientes + sus respuestas
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.0/24 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -o $INTERNAL_NETWORK -d 10.0.3.0/24 -p tcp --sport 80 -m state --state ESTABLISHED,RELATED -j ACCEPT
# - permitir peticiones DNS salientes + sus respuestas
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.0/24 -p tcp --dport 53 -j ACCEPT
iptables -A FORWARD -o $INTERNAL_NETWORK -d 10.0.3.0/24 -p tcp --sport 53 -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.0/24 -p udp --dport 53 -j ACCEPT
iptables -A FORWARD -o $INTERNAL_NETWORK -d 10.0.3.0/24 -p udp --sport 53 -m state --state ESTABLISHED,RELATED -j ACCEPT
# - rechazar cualquier otro intento de salida (DROP por defecto)
iptables -A FORWARD -i $INTERNAL_NETWORK -s 10.0.3.0/24 -j REJECT --reject-with icmp-port-unreachable
# FILTRADO DE CONEXIONES HACIA EL FIREWALL
# - permitir tráfico SSH de entrada y salida desde la red interna,
# el resto, bloqueado por la política por defecto
iptables -A INPUT -i $INTERNAL_NETWORK -s 10.0.3/24 -p tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -o $INTERNAL_NETWORK -d 10.0.3/24 -p tcp --sport 22 -j ACCEPT
# - registrar (log) los intentos de acceso al firewall desde la red externa
# (serán denegados)
iptables -A INPUT -i $EXTERNAL_NETWORK -j LOG --log-prefix "Acceso al firewall:"
```

- El script presentado, contiene una serie de reglas que forman la interfaz a utilizar por la máquina virtual Firewall.
- Al principio se designan una serie de políticas donde se deniegan las entradas y salidas del Firewall, y el reenvío de tráfico a través de este. Se enmascaran los paquetes de la red interna a la externa, utilizando el puerto 80 con HTTP.
- Para conseguir no sobrecargar la red, se implementa una regla de limitación del tráfico ICMP que permite como máximo 5 peticiones/segundo.
- Se establece un filtro de entrada interna donde existirá un redireccionamiento de peticiones y respuestas sobre la red, así como, la forma en la que va a registrarse en el archivo del log.
- Se establece la realización de conexiones HTTP y peticiones DNS de salida con la red interna, especificando el puerto correspondiente para cada una (HTTP, puerto 80. DNS, puerto 53). Además de esto, se rechazan el resto de peticiones.
- En la última parte del script, se permite el tráfico de las conexiones SSH de entrada y salida desde la red interna hacia el firewall, almacenando todos los registros y estableciendo el puerto 22.

EJERCICIO 2: OPCIONAL

El objetivo del ejercicio opcional es conocer el cortafuegos del entorno gráfico, que nos facilitará las tareas, al ser de un uso muy sencillo.

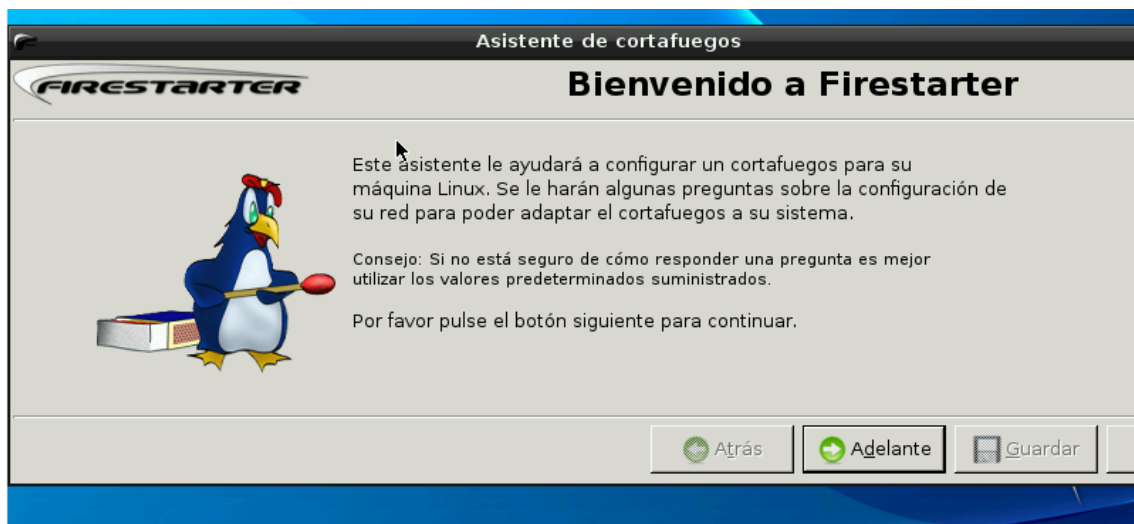
Para la realización del ejercicio, arrancamos el interfaz gráfico de la máquina virtual: firewall.

1. Arrancar la interfaz gráfica

```
firewall:~# startx
```

2. En el entorno gráfico:

[Inicio] -> Herramientas del sistema -> Firestarter

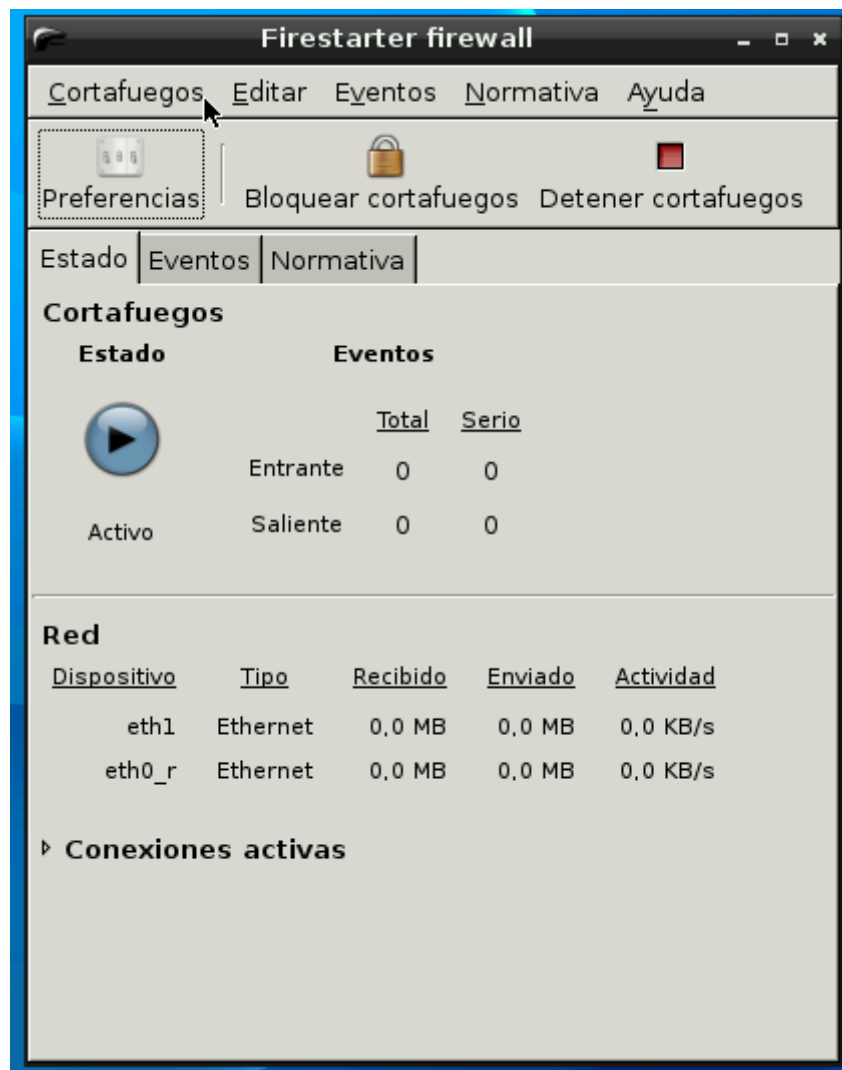


Uso de firestarter firewall

Una vez configuramos el firestarter firewall, accedemos al programa en si. Y ya podemos empezar a hacer las diferentes pruebas, indicadas en el enunciado del ejercicio.

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES

2014



Una vez, tenemos funcionando el firestarter, procedemos a la verificación de datos:

La lista de reglas introducida se puede comprobar en línea de comandos: [Inicio] -> Accesorios -> LXTerminal

```
firewall:~# iptables -L
```


Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES 2014

```
firewall:~# iptables -L
Chain INPUT (policy DROP)
target     prot opt source                destination            tcp flags:!FIN,SYN,RST,ACK/SYN
ACCEPT     tcp  --  10.0.2.3                anywhere
ACCEPT     udp  --  10.0.2.3                anywhere
ACCEPT     all  --  anywhere                anywhere
ACCEPT     icmp --  anywhere                anywhere                limit: avg 10/sec burst 5
DROP       all  --  anywhere                255.255.255.255
DROP       all  --  anywhere                10.255.255.255
DROP       all  --  224.0.0.0/8             anywhere
DROP       all  --  anywhere                224.0.0.0/8
DROP       all  --  255.255.255.255         anywhere
DROP       all  --  anywhere                default
DROP       all  --  anywhere                anywhere                state INVALID
LSI        all  -f  anywhere                anywhere                limit: avg 10/min burst 5
```

firewall:~# iptables -t nat -L

```
firewall:~# iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source                destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source                destination
MASQUERADE all  --  anywhere                anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
firewall:~#
```

Los scripts IPTABLES utilizados/configurados por firestarter están en la carpeta

"/etc/firestarter".

```
firewall:~# ls /etc/firestarter
configuration      firestarter.sh  non-routables  user-post
events-filter-hosts firewall        outbound       user-pre
events-filter-ports inbound         sysctl-tuning
firewall:~#
```

Desde externo [192.168.100.212] comenzamos un escaneo de puertos NMAP sobre el firewallly observamos los intentos detectados por firestarter.

o En externo: ejecutamos # nmap 192.168.100.77

Actividad 9: Configuración de un cortafuegos usando NETFILTER/IPTABLES

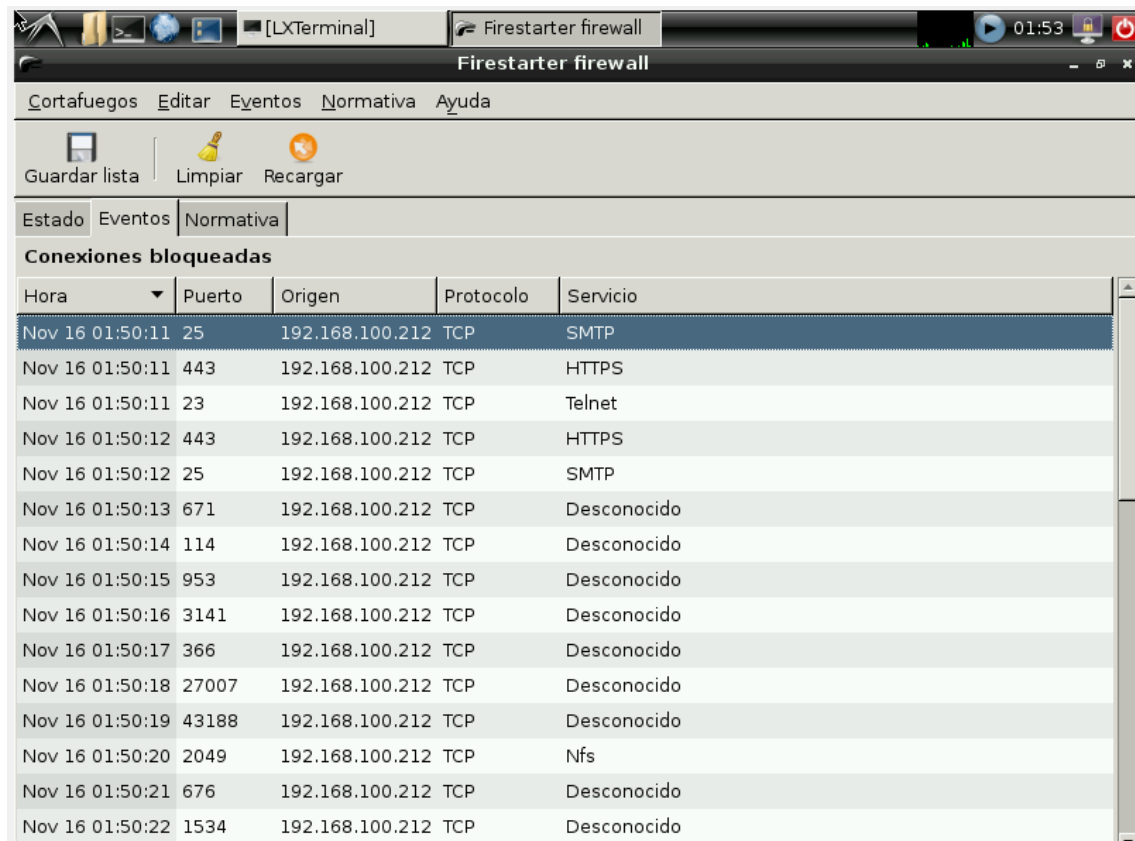
2014

```
externo:~# nmap 192.168.100.77

Starting Nmap 4.62 ( http://nmap.org ) at 2014-11-16 01:55 CET
All 1715 scanned ports on 192.168.100.77 are filtered
MAC Address: 08:00:27:44:44:44 (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 49.612 seconds
externo:~# _
```

o En firewall: abrimos la pestaña Eventos en firestarter para ver los intentos de conexión producidos durante la operación de escaneo.



Hora	Puerto	Origen	Protocolo	Servicio
Nov 16 01:50:11	25	192.168.100.212	TCP	SMTP
Nov 16 01:50:11	443	192.168.100.212	TCP	HTTPS
Nov 16 01:50:11	23	192.168.100.212	TCP	Telnet
Nov 16 01:50:12	443	192.168.100.212	TCP	HTTPS
Nov 16 01:50:12	25	192.168.100.212	TCP	SMTP
Nov 16 01:50:13	671	192.168.100.212	TCP	Desconocido
Nov 16 01:50:14	114	192.168.100.212	TCP	Desconocido
Nov 16 01:50:15	953	192.168.100.212	TCP	Desconocido
Nov 16 01:50:16	3141	192.168.100.212	TCP	Desconocido
Nov 16 01:50:17	366	192.168.100.212	TCP	Desconocido
Nov 16 01:50:18	27007	192.168.100.212	TCP	Desconocido
Nov 16 01:50:19	43188	192.168.100.212	TCP	Desconocido
Nov 16 01:50:20	2049	192.168.100.212	TCP	Nfs
Nov 16 01:50:21	676	192.168.100.212	TCP	Desconocido
Nov 16 01:50:22	1534	192.168.100.212	TCP	Desconocido

Tras realizar todos los pasos del ejercicio opcional, comprobamos los eventos ocurridos con el firestarter, verificando todos los intentos de conexión y conexiones realizadas.

CONCLUSIONES FINALES

Para concluir con la realización de la actividad 9, vamos a definir lo que me ha parecido:

Con Iptables conseguimos establecer una red segura, que nos servirá para trabajar con total confianza y seguridad. Aún así tenemos que tener cuidado con la aplicación de las reglas que ejecutemos, ya que, en función de la regla que introduciremos, podemos limitar el tráfico de la red filtrando las peticiones que se establezcan. Además de esto, nos servirá para llevar un

registro de todas las conexiones que se produzcan, como los intentos denegados clasificándolos según nuestra preferencia.

Asimismo también comprendemos la importancia de permitir las conexiones de la propia red y restringir las conexiones de redes ajenas, todo esto, apoyándonos en un firewall que controle dicho tráfico, descartando y aprobando los paquetes que considere aptos, utilizando un puerto elegido para cada función.