Práctica 5. Firewall

1. Intrucción teórica

1.1. Firewalling

Existen multiples definiciones de cortafuegos, pudiéndose resaltar de forma genérica, la dada por el Doctor Javier Areitio Bertolin:

"Un cortafuegos es un mecanismo de protección que se puede utilizar para controlar el acceso entre una red segura y una menos segura. Un cortafuegos (o firewall) no es un único componente, es una estrategia diseñada para proteger los recursos de una organización que se pueden alcanzar a través de Internet."

El concepto de firewall es bastante genérico, resumiendose en dos tipos.

Routers de selección: disponen de capacidad de seleccionar paquetes basándose en criterios como el protocolo, el puerto, dirección, campos de control etc. Se suelen fabricar en conjunto hardware-firmware y abarcan las 3 capas inferiores del modelo TCP/IP. A nivel Software están para Linux el "netfilter" cuyo plano de usuario "iptables" es mas conocido. Normalmente, los que vienen de fábrica, no tienen posibilidad de auditoria.

Gateways de Firewall: dejan subir los paquetes de la capa de transporte hasta la de aplicación para disponer de mas información. Suelen ser proxys y proporcionan mecanismos para requerir autentificación.

La figura 1 muestra el uso habitual de los routers de selección y de los gateways de firewall.

1.2. Arquitecturas básicas de seguridad

Bastion Host: equipo identificado por el administrador como punto crítico de seguridad. Requiere politicas de seguridad de sistema específicas, como:

- Eliminar cuentas de usuario superfluas.
- Eliminar demonios de apertura de puertos superfluos.
- Activar auditorias (logs).
- Desactivar las funciones de reenvio TCP/IP.
- Revisiones frecuentes y actualizaciones y parcheos.

Suelen ser equipos con algún tipo de servicio a las redes externas (Servidores Web, de aplicaciones, ftp, o incluso el propio firewall, etc.).

DMZ: (De-Militarized Zone ozona desmilitarizada) zona entre un router de selección y una red interna que tiene equipos no considerados críticos.

La figura 1 muestra la topología básica de seguridad con DMZ y Bastion Host.

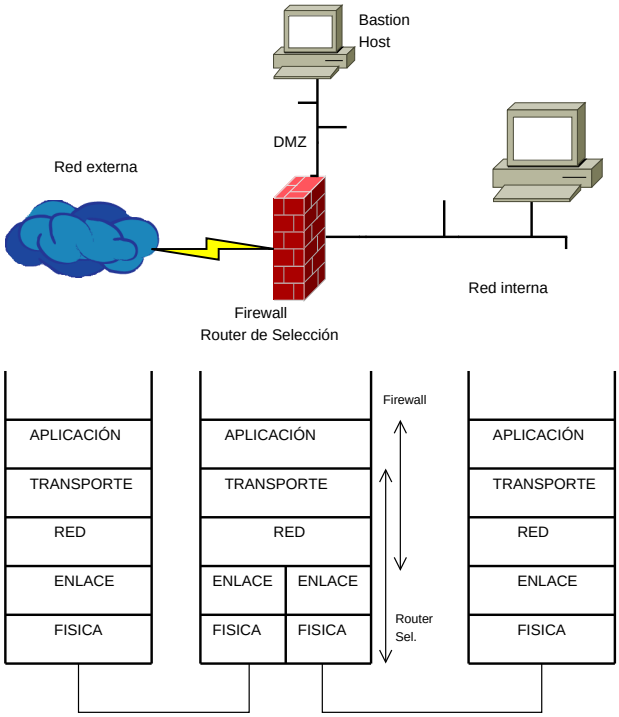

Figura 1: Capas de firewalling

1.3. Conceptos básicos

Regla: acción a realizar con un paquete o conexión que cumple una condición.

Política: regla general que se aplica cuando no existen otras reglas previas. Es el último recurso de qué hacer con un paquete

Orden de aplicación: en general, se ejecuta una regla con la primera condición que se cumpla. El orden de aplicación de las reglas es por tanto muy importante.

1.4. IPTABLES

IPTABLES es un sistema de filtrado de paquetes que afecta a prácticamente todos los niveles de la arquitectura TCP/IP, incluido en el Kernel de Linux. En un sentido amplio, *iptables* consiste en tablas, que a su vez consisten en cadenas a las que se les aplican reglas.

Las tablas que vienen por defecto son:

tabla	uso común
Filter	Sirve para el filtrado de paquetes.
Mangle	Es una tabla para la alteración de paquetes especiales. Se usa en QoS.
NAT	Se usa para el enmascaramiento de direcciones y puertos.
Raw	Sirve para excepciones de configuración.

Cuadro 1: Tablas mas comunes de IPTABLES

Las cadenas por defecto se presentan en la tabla 1.4
Se pueden definir cadenas a medida. La tabla por defecto en *iptables* es la tabla Filter, que utiliza tres de las cinco cadenas por defecto.

- INPUT: esta cadena se aplica a paquetes entrantes en la máquina local. Un ejemplo pueden ser las repuestas de solicitudes HTTP.

cadena	uso común
PREROUTING	Utilizada por las tablas raw, mangle y nat.
INPUT	Utilizada por las tablas mangle y filter.
FORWARD	Utilizada por las tablas mangle y filter.
OUTPUT	Utilizada por las tablas raw, mangle nat y filter.
POSTROUTING	Utilizada por las tablas mangle y nat.

Cuadro 2: Cadenas por defecto de IPTABLES

- OUTPUT: esta cadena se aplica a paquetes salientes de la máquina local. Un ejemplo pueden ser solicitudes HTTP a otras máquinas.
- FORWARD: esta cadena es para paquetes entrantes pero no dirigidos a la máquina local. Se utiliza básicamente en situaciones en que la máquina local sea un firewall o un gateway.

Puede verse un esquema básico de las cadenas en la figura 2

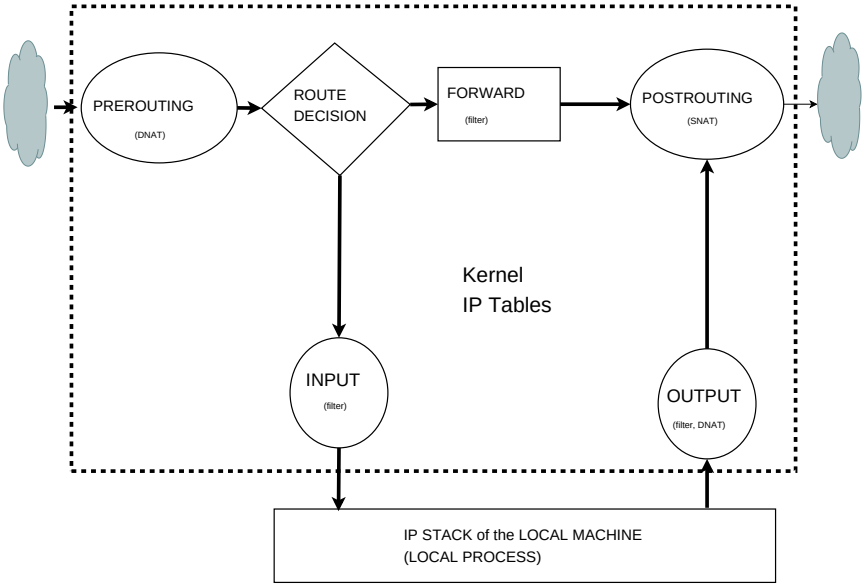

Figura 2: Esquema de las cadenas IPTABLES

iptables tiene también acciones definidas. Suelen activarse con la etiqueta "-j" Las mas comunes se presentan en la tabla 1.4.

Las acciones se utilizan en diversos campos del paquete, algunos muy comunes se muestran en la tabla 1.4.

1.5. Ejemplos de uso

Creación de algunas reglas básicas:

- Poner políticas por defecto para la tabla "filter" en las cadenas INPUT y OUTPUT, utilizando la etiqueta "-P"

```
PCX#  iptables -t filter -P INPUT DROP
PCX#  iptables -t filter -P OUTPUT DROP
```

- Permitir a la máquina local el envío sólo de solicitudes HTTP y SSH.

```
PCX# iptables -A OUTPUT -p tcp -o eth0 --dport 80 -j ACCEPT
PCX# iptables -A OUTPUT -p tcp -o eth0 --dport 22 -j ACCEPT
```

acción	uso común
ACCEPT	El paquete es aceptado.
DROP	El paquete es rechazado. No se informa a la fuente.
REJECT	El paquete es rechazado. Se le envía un mensaje de error a la fuente
LOG	Se guarda información en ficheros log.
DNAT	Enmascara la dirección IP de destino del paquete.
SNAT	Enmascara la dirección IP de origen del paquete.
QUEUE	Pasa el paquete al espacio de usuario.
RETURN	Para el viaje del paquete a través de la cadena donde esté la regla. Si la cadena es una subcadena, se devuelve el control a la cadena principal.

Cuadro 3: Acciones mas comunes de *iptables*

codificación	campo
-p <protocol>	campo protocolo, como tcp, udp, icmp, etc
-s <ip_addr>	IP origen
-d <ip_addr>	IP destino
–sport <port>	puerto origen
–dport <port>	puerto destino
-i <interface>	interfaz de entrada
-o <interface>	interfaz de salida
-m state	No es un campo. Indica que se tendrá en cuenta el estado de la conexión (–state)
–state ESTABLISHED	la conexión ya está establecida.
–state NEW	la conexión es nueva.

Cuadro 4: campos mas comunes de *iptables*

La etiqueta "-A" se utiliza para "añadir/add" esta regla a las ya existentes en la cadena (en este caso INPUT). Si no se utilizase desaparecerían las reglas anteriores

- Asumiendo que la máquina actúa como un servidor FTP, habría que permitir la entrada de conexiones a los puertos 20 y 21.

```
PCX# iptables -A INPUT -p tcp -i eth0 --dport 20 -j ACCEPT
PCX# iptables -A INPUT -p tcp -i eth0 --dport 21 -j ACCEPT
```

- Asumiendo que la máquina actúa como un servidor FTP, habría que permitir también la salida de conexiones de los puertos 20 y 21.

```
PCX# iptables -A OUTPUT -p tcp -o eth0 --sport 20 -j ACCEPT
PCX# iptables -A OUTPUT -p tcp -o eth0 --sport 21 -j ACCEPT
```

- Asumiendo que la máquina actúa como un servidor FTP, habría que permitir la salida de conexiones de los puertos 20 y 21, pero sólo para conexiones previamente establecidas desde un cliente.

```
PCX# iptables -A OUTPUT -p tcp -o eth0 --sport 20 -m state --state ESTABLISHED -j ACCEPT
PCX# iptables -A OUTPUT -p tcp -o eth0 --sport 21 -m state --state ESTABLISHED -j ACCEPT
```

- Asumiendo que la máquina actúa como un servidor interno y que quiere permitir el acceso a los miembros de la red local (192.168.1.0/24)

```
PCX# iptables -A OUTPUT -j ACCEPT -p all -d 192.168.1.0/24 -o eth0
PCX# iptables -A INPUT -j ACCEPT -p all -s 192.168.1.0/24 -i eth0
```

- Si se dispone de un host A con dirección pública y acceso a Internet, se puede permitir que cualquier equipo de una red privada salga a Internet a través de él. Es lo que se conoce como SNAT. Asumiendo que la dirección pública es *193.147.87.88* y que la red privada es *192.168.1.0/24* y que el host A puede configurarse además con una IP privada (p.ej. *192.168.1.1/24*) y forwaring (*# echo 1 >/proc/sys/net/ipv4/ip_forwar* en los equipos de la red privada se pone como pasarela de salida la dirección IP del host A (*192.168.1.1*) y en el host A se puede poner la regla de IPTABLES siguiente:

```
HostA# iptables -t nat -A POSTROUTING -j SNAT -s 192.168.1.0/24 --to-source 193.147.87.88 -↩
    o eth0
```

  o si la dirección IP pública del host A fuese dinámica:

```
HostA# iptables -t nat -A POSTROUTING -j MASQUERADING -s 192.168.1.0/24 -o eth0
```

- Si se dispone de un host A con dirección pública y acceso a Internet, se puede permitir que cualquier equipo de una red privada pueda ser accedido desde Internet a través de él. Es lo que se conoce como DNAT.

Estos comandos pueden escribirse dentro de un fichero que haría de script *iptables*.

Las reglas se leen secuencialmente, la última que se escribe será la primera de la lista. Cuando un paquete cumple una regla, se ejecuta y se para el proceso a la espera del siguiente paquete.

1.6. Frontends de IPTABLES

1.6.1. shorewall

Es el cortafuegos utilizado en esta práctica. Ver la página principal en *http://www.shorewall.net*.

1.6.2. csf & lfd

Este frontend de IPTABLES es un potente firewall de bloqueo dinámico de IP's. Según su página web, el "csf" (configserver security firewall) es un "A Stateful Packet Inspection (SPI) firewall, Login/Intrusion Detection and Security application for Linux servers. ".

El componente "lfd" (login failure daemon) permite un bloqueo rápido de IPs que realizan ataques de fuerza bruta. Este demonio corre cada pocos segundos y localiza las IPs que intentan conectarse de forma no autorizada de forma contínua.

Ver la página principal en *http://configserver.com/cp/csf.html*.

1.7. Intrusion Detection/Protection Systems (IDS/IPS)

1.7.1. snort

http://www.snort.org

2. Introducción

La práctica pretende adquirir cierta soltura en las tecnologías de cortafuegos y de enmascaramiento de IP's (SNAT y NAT reverso o DNAT), dentro de lo que es la problemática de acceso a Internet de redes con rangos de direccionamiento privados. Para ello se dispone de 2 dispositivos de red que se configurarán como firewalls (F1,F2) con placa Microtik RB433 (CPU Atheros a 300Mhz CPU, 64MB de RAM y tres puertos LAN, RB433), y sistema operativo de red OpenWrt. Se dispone además de dos equipos para administración remota (PC12, PC13), e interconexión con otras redes según el esquema. Los equipos PC12 y PC13 están instalados con sistema operativo Linux Debian. La versión de OpenWrt es la Kamikaze 8.09.02, y en su web se pueden encontrar manuales (instalación y uso) y abundante documentación.

Esta práctica está diseñada para 1 grupo de dos personas, y las pruebas finales de funcionamiento dependerán en gran medida del funcionamiento del resto de la red y en particular de la práctica de routing.

Grupo 1a: conjunto F1 y PC13 Grupo 1b: conjunto F2 y PC12.

El objetivo es conseguir dar acceso a internet a todas las redes/subredes internas del laboratorio según el esquema general del bloque 1 de prácticas, y poder seleccionar a posteriori las subredes que no tengan derecho a acceder a Internet. Para ello se deberá proceder a las siguientes actividades:

¡¡ IMPORTANTE !!: los firewalls disponen inicialmente de la siguiente configuración:

```
F1
OpenWrt Funcionando.
* OpenWrt Kamikaze 8.09.2, Shorewall 1.4 activado
* teclado espanol
* Interfaces, Direcciones, Zonas:
    eth0 (8139too.o , mii.o)     : 192.168.29.11/24 (net)
    eth1 (8139too.o , mii.o)     : 192.168.2.11/24 (lan)

* Incluye servidor shell seguro (dropbear) configurado para acceder desde cualquier direccion.
* login: root
* passwd: redes
```

```
F2
OpenWrt Funcionando.
* OpenWrt Kamikaze 8.09.2, Shorewall 1.4 activado
* teclado espanol
* Interfaces, Direcciones, Zonas:
    eth0 (8139too.o , mii.o)     : 192.168.29.12/24 (net)
    eth1 (8139too.o , mii.o)     : 192.168.2.12/24 (lan)

* Incluye servidor shell seguro (dropbear) configurado para acceder desde cualquier direccion.
* login: root
* passwd: redes
```

Por tanto, se recomienda que se configuren las interfaces de los PC's gestores para conectarse a la interfaz eth0 y/o eth1 de ambos firewalls, así

- PC13: Conexión a eth0-F1

  Desde Linux se realiza el cambio de dirección IP con el comando "ifconfig" o "ip addr".

  Desde Windows se realiza el cambio de dirección IP con el comando "netsh".

```
Linux:      PC13# ip addr add 192.168.29.113/24 dev eth0
Linux:      PC13# ifconfig  eth0  192.168.29.113/24

Windows:    C:\> netsh  interface  ip  set  address  local1  static 192.168.29.113 ↩
    255.255.255.0
```

  Nota: realmente PC13 ya está configurado así por defecto al arrancar la máquina, no obstante, siempre, en toda infraestructura con caracter compartido es mejor comprobarlo.

- PC12: Conexión a eth0-F2

  Desde Linux se realiza el cambio de dirección IP con el comando "ifconfig" o "ip addr" según se comentó anteriormente

  Desde Windows se realiza el cambio de dirección IP con el comando "netsh", también comentada anteriormente:

```
Linux:      PC12# ifconfig    eth0   192.168.29.112/24
Linux:      PC12# ip addr add   192.168.29.112/24  dev  eth0

Windows:    C:\> netsh interface ip set address local1 static 192.168.2.112  255.255.255.0
```

  Nota: realmente PC12 ya está configurado así por defecto al arrancar la máquina, no obstante, siempre, en toda infraestructura con caracter compartido es mejor comprobarlo.

Para conectarse desde el PC de gestión al firewall se usa el comando "ssh" de la forma:

```
PCXX # ssh  -l  root   <direccion IP del firewall>
```

Si se hiciese desde equipos Windows bastaría con ejecutar el programa "putty" y conectarse por el puerto 22 (SSH) a la dirección IP del firewall correspondiente dada anteriormente. Es recomendable hacer un "ping" previamente para asegurar que existe conectividad.

Será necesario, tras tener el control del firewall, configurar sus interfaces según la arquitectura de la figura, para lo cual se usará el comando "ip addr" desde cualquiera de los PC gestores, como se describe a continuación

En F1:

```
F1#  ip  addr  add  192.168.201.2/24  dev  eth0
F1#  ip  addr  add  192.168.29.11/24  dev  eth1
```

En F2:

```
F1#  ip  addr  add  192.168.202.2/24  dev  eth0
F1#  ip  addr  add  192.168.29.12/24  dev  eth1
```

Tras la ejecución de este comando, F1 y F2 dispondrán de dos direcciones IP en sus interfaces eth0 y eth1. Antes de eliminar las direcciones IP originales de ambos en eth0 y eth1, será necesario cambiar también las direcciones IP de PC12 y PC13 para que entren en el mismo rango que las de F2.

Para evitar complejidad en el desarrollo de la práctica, se eliminarán las direcciones originales en ambos firewalls,

En F1:

```
F1#  ip  addr  del  192.168.29.11/24  dev  eth0
F1#  ip  addr  del  192.168.2.11/24  dev  eth1
```

En F2:

```
F2#  ip  addr  del  192.168.29.12/24  dev  eth0
F2#  ip  addr  del  192.168.2.12/24  dev  eth1
```

y la ruta por defecto.

```
FX#  ip route del default
```

Será necesario por tanto, dar la nueva ruta por defecto según aparece en la figura 3.

```
FX#  ip route add default via 192.168.29.1
```

NOTA: es posible que se pierda la conexión entre los PCs y los firewalls, debiéndose reconectar a las nuevas interfaces ethernet de PC12 y PC13 con la dirección IP marcada en la arquitectura de la figura (192.168.201.113, 192.168.202.112), para recuperar esa conexión.

3. Desarrollo

OpenWrt incluye además de los demonios de enrutamiento y las utilidades básicas de los sistemas Linux, otras utilidades mas complejas, como lo es el firewall "shorewall" - http://www.shorewall.net/ - , un servidor http para visualizar el funcionamiento del firewall a través de un navegador y el demonio "sshd" para permitir conexiones remotas sin necesidad de conexión a través del puerto serie.

Los comandos específicos imprescindibles para esta práctica son:

```
#ip  addr [help]
#ip  route [help]
```

Cuyo uso mas común tendrá como ejemplo

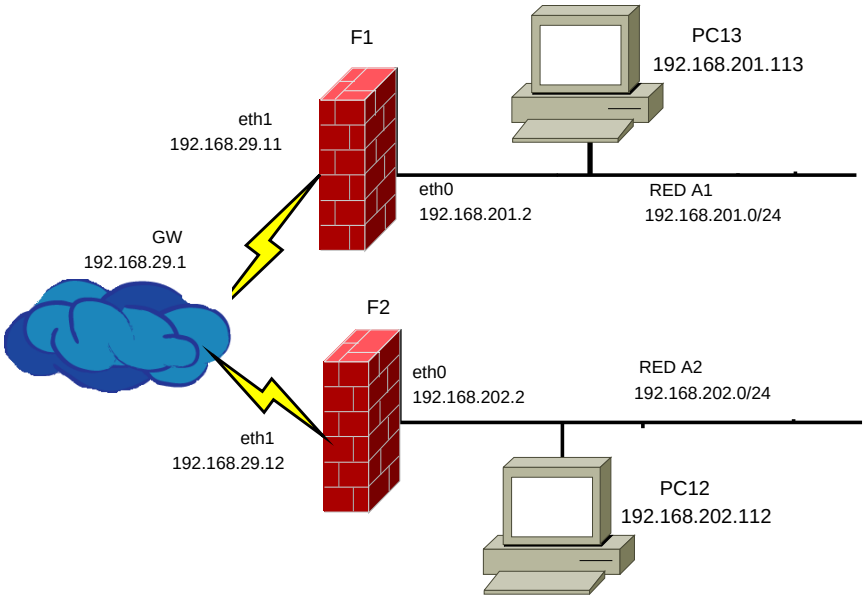

Figura 3: Visión general del direccionamiento

```
# ip addr add <direccion/mask> dev <ethx>

ejemplo# ip addr add 192.168.202.2/24 dev eth0

# ip route add <red destino/mascara> via <gateway> dev <interfaz de salida>

ejemplo# ip route add 192.168.201.0 via 192.168.202.1 dev eth0
```

La gestión de los firewalls se realizará desde sus equipos de gestión PC12 y PC13 mediante conexión ssh. Los cambios de direccionamiento IP en las interfaces se harán según lo descrito en la sección anterior.

Será además necesario configurar en los firewalls las rutas hacia el resto de redes a las que no están conectados.

En F1:

- A1: 192.168.201.0/24 de forma directa.
- A2: 192.168.202.0/24 a través de R1
- A2: 192.168.203.0/24 a través de R1
- A4: 192.168.204.0/24 a través de R1

En F2:

- A1: 192.168.201.0/24 a través de R2
- A2: 192.168.202.0/24 de forma directa
- A2: 192.168.203.0/24 a través de R2
- A4: 192.168.204.0/24 a través de R2

Para F1 el siguiente salto será la interfaz eth0 de R1 (192.168.201.1), al cual F1 está conectado directamente. Para F2 el siguiente salto será la interfaz eth0 de R2 (192.168.202.1), al cual F2 está conectado directamente.

Se puede estudiar la posibilidad de hacer una monitorización básica de los firewalls a través de un navegador web del equipo de gestión correspondiente, conectándose a la dirección IP del firewall teniendo, claro, permisos para ello. Esta parte se realizará a la finalización de la práctica si hay tiempo disponible.

Configuración direccionamiento IP en la interfaz "externa (eth1)" del firewall y de las rutas a las demás redes.

Ya se han mencionado el procedimiento y los comandos para hacerlo en la sección 2.

Comandos:

```
#ip  addr [help]
#ip  route [help]
```

Se deberá probar la conectividad de las interfaces con equipos próximos (equipos de gestión PC12 y PC13) y routers R1 y R2.

Se sugiere el uso del scripting para la creación de las rutas de forma estática en el arranque del sistema.

Se deberá probar la conectividad a internet y a las demás redes con los parámetros descritos en la sección "1. Descripción" de la práctica, recordando (depende de la correcta parametrización de las prácticas de routing).

Se deberá configurar el firewall F1 para limitar el acceso a internet de, por ejemplo, la subred A3 (192.168.203.0/24), y F2 limitará a la subred A4 (192.168.204.0/24). Probar con otras subredes.

La configuración de los cinco ficheros fundamentales de shorewall sería:

- /etc/shorewall/zones

```
###############################################################################
#ZONE    TYPE            OPTIONS         IN                      OUT
#                                       OPTIONS                 OPTIONS
net     ipv4
lan     ipv4
fw      firewall
#LAST LINE - ADD YOUR ENTRIES ABOVE THIS ONE - DO NOT REMOVE
```

- /etc/shorewall/interfaces

```
#ZONE   INTERFACE BROADCAST       OPTIONS
net     eth1      detect     routefilter
lan     eth0      detect     routefilter
```

- /etc/shorewall/masq (F1)

```
#INTERFACE    SUBNET ADDRESS
eth1          eth0          192.168.29.11
eth1     192.168.202.0/24     192.168.29.11
eth1     192.168.203.0/24     192.168.29.11
eth1     192.168.204.0/24     192.168.29.11
```

- /etc/shorewall/masq (F2)

```
#INTERFACE    SUBNET ADDRESS
eth1          eth0          192.168.29.12
eth1     192.168.201.0/24     192.168.29.12
eth1     192.168.203.0/24     192.168.29.12
eth1     192.168.204.0/24     192.168.29.12
```

El campo INTERFACE indica la interfaz de salida del Firewall, SUBNET indica la subred a la que se permite ser enmascarada y ADDRESS indica la IP con la que se enmascara, que deberá coincidir en la mayoría de los casos con la <direccion ip de la interfaz de salida>. Hay que hacer esto para cada una de las redes que se conecten a Internet.

- /etc/shorewall/policy

```
        #SOURCE         DEST            POLICY          LOG             BURST:LIMIT
        #                                               LEVEL
        lan             fw              ACCEPT
        lan             net             ACCEPT
```

```
fw              net             ACCEPT
net             all             DROP        info
#
# THE FOLLOWING POLICY MUST BE LAST
#
all             all             REJECT          info
```

¡Es muy importante no olvidarse de la primera línea como medida de seguridad para acceder al firewall, antes de reiniciar el shorewall.!

- /etc/shorewall/rules

```
#ACTION SOURCE  DEST            PROTO   DEST    SOURCE  ORIGINAL
#
ACCEPT  fw  net     udp 53
ACCEPT  fw  net     tcp 53
ACCEPT  lan fw      tcp 22
ACCEPT  lan fw      tcp 80
DROP    lan:192.168.202.112 net     all
DROP    lan:192.168.201.113 net     all
```

Es posible seleccionar equipos origen o destino por url, aunque en equipos que utilicen IP dinámica puede dar lugar a errores. Estos errores vienen dados porque shorewall (e iptables) consulta el DNS la primera vez que se aplica la regla, y ya da de alta la línea iptables con la dirección IP encontrada sin modificarla en ningún otro momento. Dos ejemplos:

1. Se quiere limitar el acceso a la web "www.as.com" y a la web "as.com" para todos los protocolos desde toda la lan. En principio, si el DNS de "as.com" o de "www.as.com" es estable, quedaría bloqueado. Para saber si es estable un dns (entendido como que la asociación "nombre-ip" es estable) basta con ejecutar varias veces `dig www.as.com` y fijarse en la columna del TTL (primera columna numérica) que da una idea del tiempo estimado de mantenimiento del registro tipo CNAME o A.

```
#ACTION SOURCE  DEST            PROTO   DEST    SOURCE  ORIGINAL
#
DROP    lan net:www.as.com      all
DROP    lan net:as.com.     all
```

¡OJO! Hay que fijarse en el punto final "." cuando solo hay dos componentes en el nombre, es decir, el nombre dns tiene que tener al menos dos puntos ".".

2. Se quiere limitar el acceso a la web *www.google.es* desde el equipo *192.168.201.123* de la lan. Se procede como antes, pero la web posiblemente sólo quede bloqueada unos minutos o ni siquiera eso, ya que la dirección IP asociada a ese nombre tiene una vida muy corta. Shorewall resuelve el DNS y activa la regla iptables. Si la IP asociada al DNS ha cambiado, la regla no funcionará hasta que se reinicie el shorewall.

```
#ACTION SOURCE  DEST            PROTO   DEST    SOURCE  ORIGINAL
#
DROP    lan:192.168.201.123 net:www.google.es   all
```

En estos últimos casos, se recomienda (shorewall también lo hace) deshabilitar rangos de IPs, en lugar de nombres. Para utilizar rangos de direcciones IP se recomienda utilizar el formato "dirección máscara" que se mantiene desde versiones antiguas de shorewall. Para versiones modernas se pueden utilizar rangos estándar de direcciones IP (dirección_inicial-dirección_final). Shorewall.net no recomienda en ningún momento el uso de nombres DNS en lugar de IPs. Se reproducen una parte de la FAQ de shorewall.net.

```
If your firewall rules include DNS names then:
- If your /etc/resolv.conf is wrong then your firewall won't start.
- If your /etc/nsswitch.conf is wrong then your firewall won't start.
- If your Name Server(s) is(are) down then your firewall won't start.
```

```
    - If your startup scripts try to start your firewall before starting your DNS server ↩
      then your firewall won't start.
    - Factors totally outside your control (your ISP's router is down for example), can ↩
      prevent your firewall from starting.
    - You must bring up your network interfaces prior to starting your firewall.
    Each DNS name must be fully qualified and include a minimum of two periods (although ↩
      one may be trailing). This restriction is imposed by Shorewall to insure backward↩
       compatibility with existing configuration files.
```

Para controlar el acceso a www.google.es, será necesario por tanto conocer el rango de direcciones IP que responden a esa URL; conocerlo es relativamente sencillo haciendo cada cinco minutos un *$ dig www.google.es* y fijarse en los registros tip A y CNAME. A modo de ejemplo mas o menos serio, se puede limitar algo el acceso a www.google.es de la siguiente forma:

```
        #ACTION SOURCE   DEST            PROTO    DEST     SOURCE   ORIGINAL
        #
    DROP     lan net:173.194.45.64/27     all
```

Se deberán configurar los firewalls para poder acceder desde Internet a los siguientes servicios:

- F1: ssh (TCP 22): PC12
- F2: HTTP (TCP 80): PC13

Esto se puede hacer añadiendo en el fichero /etc/shorewall/rules.

- /etc/shorewall/rules (F1)

```
        #ACTION SOURCE   DEST            PROTO    DEST     SOURCE   ORIGINAL
        #                                PORT     PORT
    DNAT     net lan:192.168.201.113 tcp 22
```

- /etc/shorewall/rules (F2)

```
        #ACTION SOURCE   DEST            PROTO    DEST     SOURCE   ORIGINAL
        #                                PORT     PORT
    DNAT     net lan:192.168.202.112 tcp 80
```

Habrá que comprobar la integridad de los cambios.

```
    FX#  shorewall check
```

Si los mensajes son correctos, habrá que reiniciar el firewall.

```
    FX#  shorewall restart
```

Para no aplicar las reglas ni políticas del firewall y dejar acceso a todo el sistema

```
    FX#  shorewall clear
```

Para cerrar el firewall, ¡OJO!, se perdería el acceso. Habría que entrar desde puerto serie, o reiniciar el shorewall perdiendose los cambios.

```
    FX#  shorewall stop
```

4. Sistemas NAT

Un firewall no se encarga únicamente de filtrar paquetes entre unas redes (zonas) u otras en función de unas reglas o políticas, sino que además, suele disponer de la técnica de "Traducción de direcciones" o NAT (Network Address Translation). Básicamente, la técnica NAT consiste en cambiar la dirección de origen de un paquete para que este sea reconocido en una red externa. Además, el firewall NAT (o router NAT) mantiene una tabla en la que guarda diferentes datos de conexión del paquete (en general identificadores de la conexión interna y de la conexión externa) para que la réplica o respuesta al paquete pueda ser finalmente redirigida a su verdadero origen.

En el frontend *shorewall*, el fichero que se encarga de la configuración del sistema NAT es el fichero */etc/shorewall/masq*, donde el campo "INTERFACE" es la interfaz de salida de los paquetes enmascarados, "SUBNET" es la red de entrada de los paquetes a enmascarar y "ADDRESS" la dirección con la que finalmente se enmascara.

```
#INTERFACE    SUBNET  ADDRESS
eth1          eth0          192.168.29.11
eth1     192.168.202.0/24     192.168.29.11
eth1     192.168.203.0/24     192.168.29.11
eth1     192.168.204.0/24     192.168.29.11
```

No obstante, el la técnica de NAT no está limitada a los firewalls, de hecho, a los dispositivos que implementan el NAT se les sigue llamando routers NAT.

La suite de comandos *iproute2* dispone del subcomando *ip rule add from <origen>nat <mascara>table <tabla>prio <* que realiza exáctamente la misma acción.

Se invita al alumno a no hacer el sistema NAT por *iptables* sino mediante *iproute2*.

5. Pruebas

Finalmente, se comprobará la conectividad (con el comando "ping") con el resto de la red, así como hacia internet, dependiendo esto de la correcta implementación de las demás prácticas implicadas.

- F1,F2 <–>PC13,PC12
- F1,F2 <–>R1, R2, R3, R4
- F1,F2 <–>Resto de equipos
- F1,F2 <–>INTERNET
- PC12,PC13 <–>INTERNET

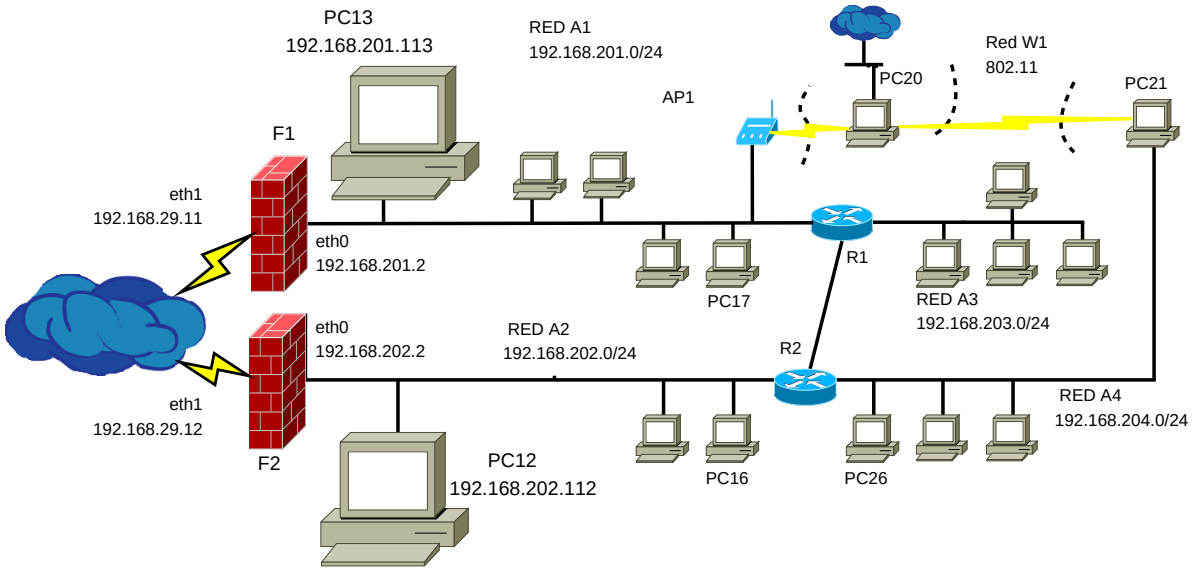

Figura 4: Visión general de la práctica