Práctica 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 (ip_forwarding).
- Revisiones frecuentes y actualizaciones y parcheos.

Suelen ser equipos con algún tipo de servicio a las redes externas (Servidores Web, de aplicaciones, ftp, anclas, 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.

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.

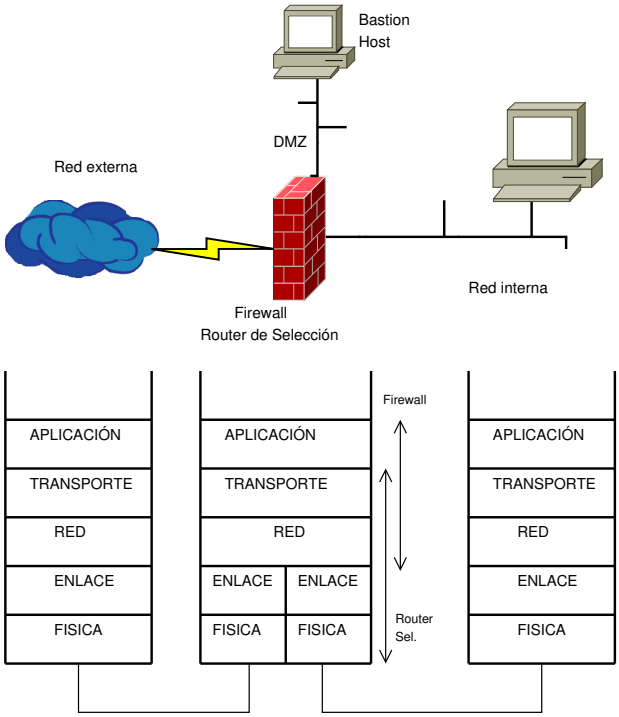

Figura 1: Capas de firewalling

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. El Kernel de Linux provee de una interfaz que permite filtrar los paquetes de entrada y de salida mediante "tablas de filtros de paquetes". En un sentido amplio, *iptables* es una aplicación en línea de comandos para gestionar dichas "tablas de filtros de paquetes". Cada tabla puede contener múltiples cadenas, y cada cadena es un conjunto de reglas. Cuando un paquete se ajusta a una regla, se le da un destino o una acción (TARGET). Un TARGET puede ser otra cadena o valores como ACCEPT (permitido que pase), DROP (eliminado), REJECT (rechazado) o RETURN (salta la cadena actual y vuelve a la siguiente regla de la cadena desde la que viene el paquete) entre otros.

Las tablas que vienen por defecto son:

Tabla 1: Tablas mas comunes de IPTABLES

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.

Las cadenas por defecto se presentan en la tabla 2.

Tabla 2: Cadenas por defecto de IPTABLES

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.

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.
- 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

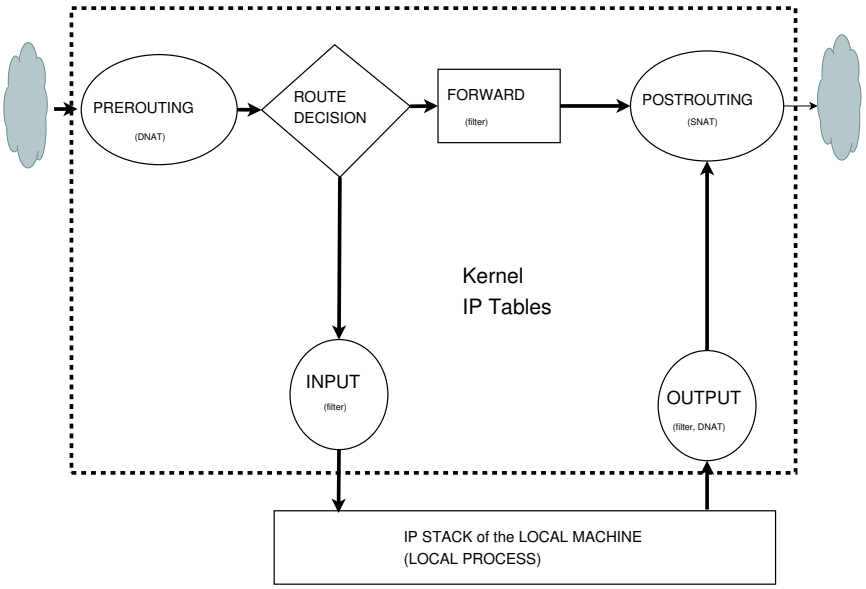

Figura 2: Esquema de las cadenas IPTABLES

iptables tiene también acciones definidas (TARGET). Suelen activarse con la etiqueta "-j" Las mas comunes se presentan en la tabla 3.

Tabla 3: Acciones mas comunes de *iptables*

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.

Las acciones se utilizan en diversos campos del paquete, algunos muy comunes se muestran en la tabla 4.

1.5 Ejemplos de uso

Un ejemplo genérico para añadir una regla a una cadena tendría un formato :

Tabla 4: campos 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.

```
PCX#  iptables -A <CHAIN> -i <input_int> -o <output_int> -p <protocol(tcp/udp)> -s <source> --dport <↩
    port>  -j <target>
```

donde la etiqueta "-A" se utiliza para "añadir/add",

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
```

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
```

- Habilitar y deshabilitar los "icmp request (icmp tipo 8)" e "icmp response (icmp tipo 0)" tanto de salida como de entrada.

  Para habilitar los icmp 8 (ping) de **salida** (con sus respectivas respuestas, icmp 0 pong):

```
PCX# iptables -A OUTPUT -p icmp --icmp-type 8 -s IPdelEQUIPO -d 0/0 -j ACCEPT
PCX# iptables -A INPUT -p icmp --icmp-type 0 -s 0/0 -d IPdelEQUIPO -j ACCEPT
```

  Para habilitar los icmp 8 (ping) de **entrada** (y sus respectivas respuestas, icmp 0 pong):

```
PCX# iptables -A OUTPUT -p icmp --icmp-type 0 -s IPdelEQUIPO -d 0/0 -j ACCEPT
PCX# iptables -A INPUT -p icmp --icmp-type 8 -s 0/0 -d IPdelEQUIPO -j ACCEPT
```

  Para deshabilitar habrá que cambiar la acción de ACCEPT a REJECT.

- 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 activar el forwarding (*# echo 1 > /proc/sys/net/ipv4/ip_forwarding*), 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. Manteniendo el ejemplo anterior, se pretende que el puerto tcp80 del equipo A sea redirigido al puerto tcp8080 de un equipo de la red privada, por ejemplo el equipo cuya IP es *192.168.1.10*.

```
HostA# iptables -t nat -A PREROUTING -p tcp -d 193.147.87.88 --dport 80 -j DNAT --to-↩
    destination 192.168.1.10:8080
```

- Para eliminar determinadas reglas de una tabla:

```
HostA# iptables -L --line-numbers
```

  y seleccionar la cadena y el número de línea de la regla que se quiere eliminar, por ejemplo la 3 de la cadena INPUT:

```
HostA# iptables -D INPUT 3
```

Estos comandos pueden escribirse dentro de un fichero que haría de script *iptables*.
También es posible hacer estos cambios permanentes mediante:

```
HostA# /sbin/iptables-save
```

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.

IMPORTANTE!!!: En el caso en que se hayan hecho cambios en la práctica, conviene revertirlos mediante:

```
HostA# iptables -F
HostA# /sbin/iptables-save
```

1.6 Frontends de IPTABLES

1.6.1 shorewall

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

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 (redes locales). Para ello se dispone del RouterBoard de Mikrotik RB2011 que se configurará como firewall (FX), y sistema operativo de red OpenWrt. Se dispone además de un equipo para administración remota (PCX), e interconexión con otras redes según el esquema. Los equipos PCX están instalados con sistema operativo Linux Debian. La versión de OpenWrt es la 19.07, y en su web se pueden encontrar manuales (instalación y uso) y abundante documentación.

Esta práctica está diseñada para su realización de forma individual y en colaboración con otros. Las pruebas finales de funcionamiento dependerán en gran medida del funcionamiento del resto de la red.

2.1 Objetivos

El objetivo es conseguir gestionar la conectividad de los equipos de la red local hacia o desde Internet (o la intranet).

De forma mas práctica, los objetivos consisten en la configuración de los firewalls y los equipos de gestión con el direccionamiento deseado y presentado en el esquema general de la práctica.

Para ello se seguirá una lista de tareas, que de forma resumida son:

- Configuración de las interfaces de los equipos.
  - Configuración de las interfaces del equipo gestor.
  - Configuración de las interfaces del firewall.
- Configuración del firewall

  - Zonas, políticas, reglas.
  - SNAT, DNAT

- Pruebas

2.2 Situación de partida

Dadas las características de compartición del laboratorio y por tanto de sus equipos, la configuración inicial de los equipos puede variar, por lo que habrá que uniformarla en la medida de lo posible.

- Situación de partida de los PC's

  Los PCs deberán arrancar con un direccionamiento de red acorde a la red plana del laboratorio (192.168.29.0/24) y con salida a Internet a través de F0 (192.168.29.1). Por ejemplo: el PC13 arrancará con la dirección IP 192.168.29.113/24 y tendrá como puerta de enlace (Gateway) a 192.168.29.1 (F0).

- Situación de partida de los firewall (RBM)

  Los RouterBoardsMicrotik (RBM) arrancan con la dirección ip 192.168.29.A siendo A el número de RBM escrito en su carcasa, por ejemplo, el RBM21 arranca con la dirección IP 192.168.29.21. Esta dirección es accesible en la interfaz "br-lan" de la figura 3, es decir, el PC gestor de cada RBM deberá conectarse con el latiguillo con conectores RJ45 a cualquier puerto de la interfaz "br-lan".

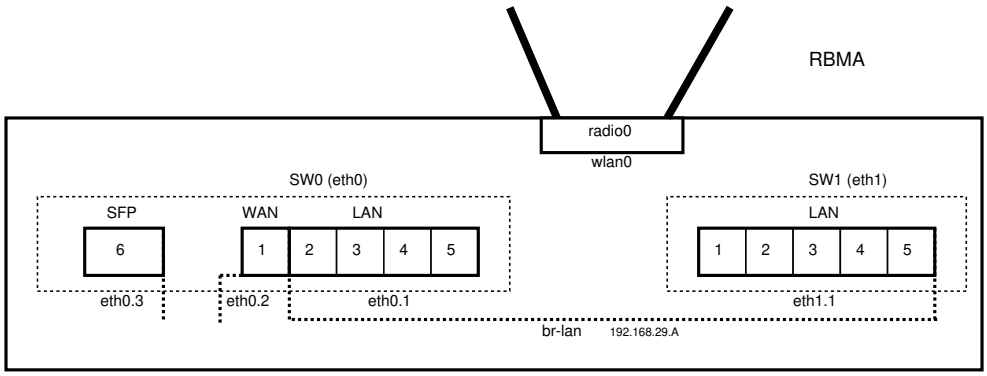

Figura 3: Situación de partida

´

Para que funcionen como firewall, será necesario acceder a ellos y ejecutar el comando:

Comandos 1: Conversión en firewall.

```
root@RBMA:~# ./isa_firewall.sh
```

Desde ese momento, la configuración de las interfaces cambia, así como el direccionamiento IP, según se muestra en la figura 3 y en la tabla 5. Los RouterBoardMicrotik pasarán a convertirse en routers y tomarán el nombre de FA, siendo A el mismo número asociado al RBMA anterior.

El sistema operativo OpenWRT incluye además de los demonios de enrutamiento y las utilidades básicas de los sistemas Linux, otras utilidades mas complejas, como lo son el firewall "shorewall", la weblet para visualizar el funcionamiento del router a través de un navegador y el demonio "sshd" para permitir conexiones remotas seguras y poder prescindir en los equipos con OpenWRT de tarjeta gráfica y monitor. El RBMA dispone de un puerto serie con interfaz RJ45 para conectividad directa.

  - Tabla con información de la situación de partida de RBMA (atención a las interfaces y sus direcciones IP)

    Incluye servidor shell seguro (dropbear) configurado para acceder desde cualquier direccion. Incluye paquete shorewall. Se puede acceder a traves de puerto serie.

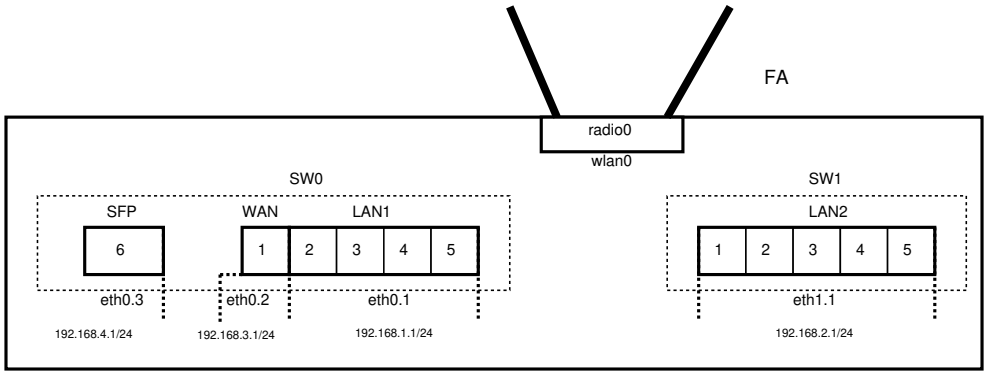

Figura 4: Situación de partida como firewall.

Tabla 5: Situación de partida

Interfaz	**Dirección IP**
eth0.1	192.168.1.1/24
eth1.1	192.168.2.1/24
eth0.2	192.168.3.1/24
eth0.3	192.168.4.1/24
login	**passwd**
root	provisional

Es muy importante resaltar que TODOS los firewalls tienen el mismo direccionamiento IP, por lo que la misma dirección IP estará repetida. Esto implica que habrá que modificar el direccionamiento de todas las interfaces de los routers y ponerlas según el objetivo de direccionamiento de la figura 6.

IMPORTANTE: Todos los RBMs adquieren la misma configuración y el mismo direccionamiento IP, por lo que será necesario extremar las precauciones para no interferir entre equipos con la misma dirección IP.

3 Desarrollo

Se necesitará utilizar los comandos específicos:

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

Cuyo uso mas común tendrá como ejemplo:

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

ejemplo# ip  addr  add 192.168.12.112/24  dev  eth1

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

ejemplo# ip  route  add  192.168.12.0 via 10.10.12.1  dev eth0.1
```

Además será necesario editar mediante el editor "vi" los ficheros del entorno "shorewall":

- /etc/shorewall/zones
- /etc/shorewall/interfaces
- /etc/shorewall/rules
- /etc/shorewall/policy

3.1 Acceso inicial a cada uno de los firewalls desde los equipos de gestión

Será necesario realizar el conexionado adecuado a la topología de la figura 5. En ella, las líneas negras son latiguillos de par de cobre con conectores RJ45, la "nube oscura" es la red conmutada de nuestro laboratorio, la "nube clara" sería la zona DMZ que de base no se configurará. FA es el RouterBoardMicrotik del puesto de usuario en el laboratorio, y el PC-A es el PC de usuario.

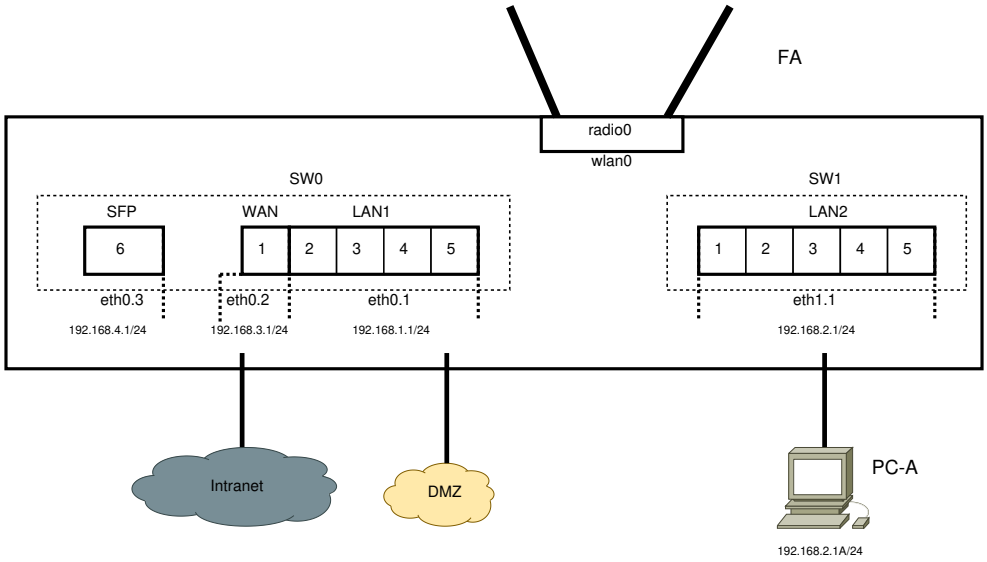

Figura 5: Conexionado

Para que cada equipo de gestión (PC-A) acceda a su Firewall tendrá que realizar además de un cambio en su conectividad hacia el firewall también en su direccionamiento, de forma que para conectarse al firewall FA a través de la interfaz eth1.1 con la dirección IP de partida 192.168.2.1/24, el PC-A tendrá que configurarse con la IP 192.168.2.1A/24.

```
PC-A# ip addr add 192.168.2.1A/24 dev eth0
```

Nota: la interfaz en PC-A puede tener un nombre distinto a "eth0". Será necesario sustituirlo en el comando anterior.

Tras ello se realizará la conexión por ssh a la interfaz eth1 de FA. El usuario será "root" y la clave la que aparece en las condiciones iniciales según corresponda.

```
PC-A # ssh -l root 192.168.2.1
```

3.2 Configuración de las interfaces de los Firewalls

Será necesario, tras tener el control del firewall, configurar sus interfaces según la figura 6, para lo cual se usará el comando "ip addr" en los firewalls.

Un ejemplo indicativo de cómo quedaría el direccionamiento con los firewalls F20 y F21 puede verse en la figura 7

Se deberá probar la conectividad de las interfaces con equipos próximos y otros firewalls.

El objetivo es conseguir gestionar el acceso a internet de cualquier subred internas del laboratorio según el esquema general de la práctica.

Por tanto, se recomienda que se configuren las interfaces de los PC's gestores para conectarse a la interfaz eth1.1 de los firewalls, así

En PCA se realiza el cambio de dirección IP con el comando "ip addr".

```
PCA# ip addr add 192.168.2.1A/24 dev eth0
```

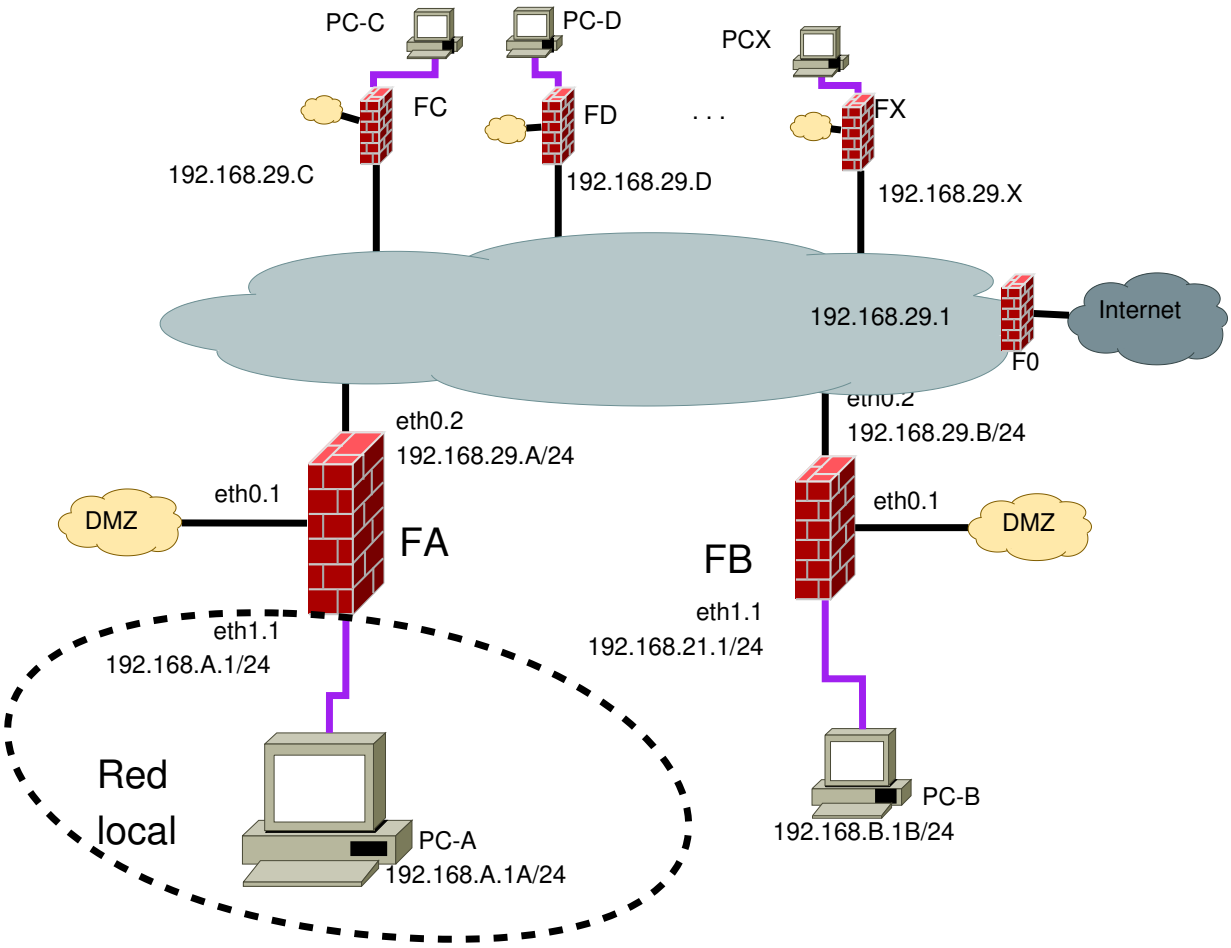

Figura 6: Objetivo de direccionamiento

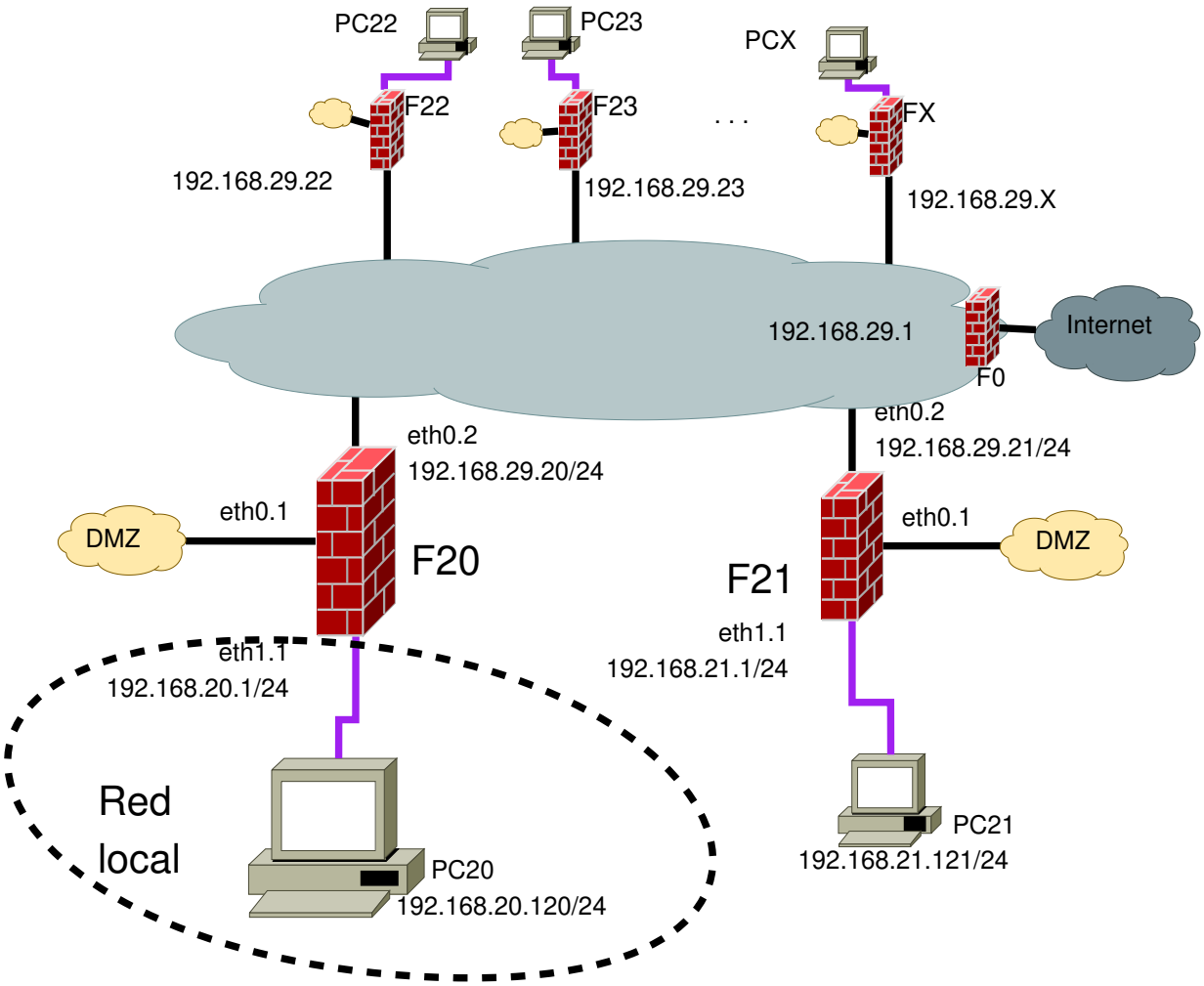

Figura 7: Ejemplo de direccionamiento

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

```
PCA # ssh -l root 192.168.2.1
```

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.

Es posible que los Firewalls no dispongan de los mismos métodos de intercambio de claves que los host origen, en cuyo caso pueden rechazarse las conexiones con el mensaje:

```
PCA$ ssh -l root 192.168.2.1
Unable to negotiate with ::1 port 22: no matching key exchange method found.
  Their offer: diffie-hellman-group1-sha1
```

En ese caso podrá realizarse la conexión con la opción:

```
PCA # ssh -o KexAlgorithms=+diffie-hellman-group1-sha1 -l root 192.168.2.1
```

También es posible que los Firewalls no dispongan de los mismos algoritmos de cifrado que los host origen, en cuyo caso pueden rechazarse las conexiones con el mensaje:

```
PCA$ ssh -l root 192.168.2.1
Unable to negotiate with 192.168.2.1 port 22: no matching cipher found.
  Their offer: aes128-cbc,3des-cbc,aes192-cbc,aes256-cbc
```

En ese caso, será necesario establecer la conexión con unos algoritmos de cifrado específicos aceptados por el servidor, por ejemplo el aes128-cbc. Esto se haría:

```
PCA # ssh -c aes128-cbc -l root 192.168.2.1
```

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 los PC gestores, como se describe a continuación

```
FA# ip addr add 192.168.29.A/24 dev eth0.2
FA# ip addr add 192.168.20.1/24 dev eth1.1
```

Tras ello, FA dispondrá de dos direcciones IP en sus interfaces eth0 y eth1. Antes de eliminar las direcciones IP originales de ambos en eth0.2 y eth1.1, será necesario cambiar también la dirección IP de PCA para que pueda acceder con el direccionamiento final.

```
PCA# ip addr add 192.168.20.1A/24 dev eth0
PCA# ssh -l root 192.168.20.1/24
```

Una vez que este acceso sea efectivo se podrán eliminar las direcciones iniciales del firewall FA y del PCA.

En FA:

```
FA# ip addr del 192.168.2.1/24 dev eth1.1
FA# ip addr del 192.168.3.1/24 dev eth0.2
```

En PC-A:

```
PCA# ip addr del 192.168.29.1A/24 dev eth0
PCA# ip addr del 192.168.2.A/24 dev eth0
```

y la ruta por defecto.

```
FA# ip route del default
```

```
PCA#  ip route del default
```

Será necesario por tanto, dar la nueva ruta por defecto según aparece en la figura 8.

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

```
PCA#  ip route add default via 192.168.A.1
```

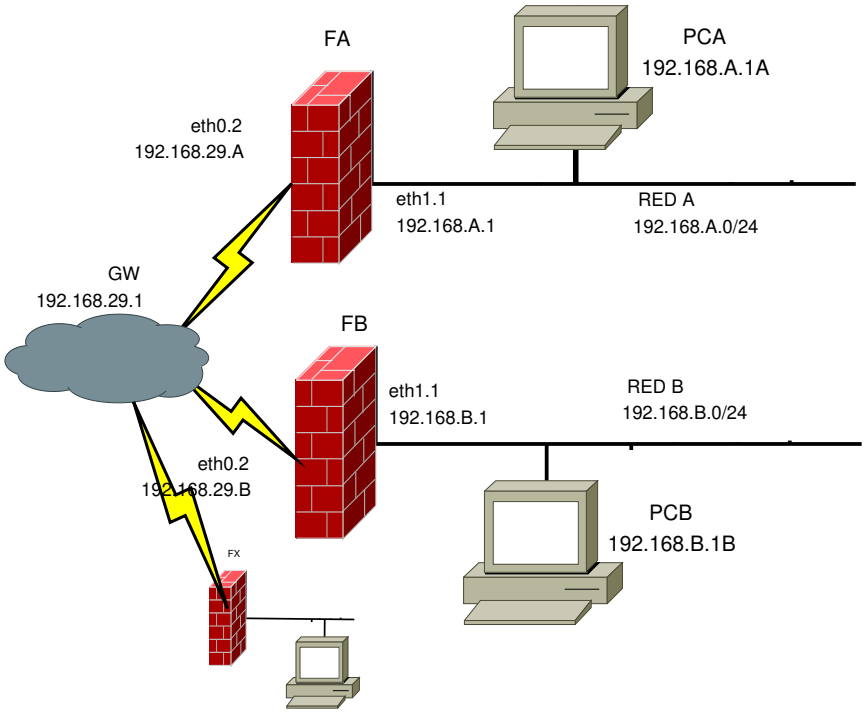

Figura 8: Visión general del direccionamiento

NOTA: es posible que si ha habido algún paso anterior fallido se pierda la conexión entre los PCs y los firewalls, debiéndose reconectar a las nuevas interfaces ethernet de PCA con la dirección IP marcada en la arquitectura de la figura (192.168.A.1A), para recuperar esa conexión.

4 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.

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.

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      eth0.2       detect   routefilter
lan      eth1.1       detect   routefilter
```

- /etc/shorewall/snat

```
#########################################################################################    ...
#ACTION                  SOURCE                  DEST          PROTO    IPSEC    MARK     ...
MASQUERADE               192.168.A.0/24          eth0.2
```

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
    #
    ?SECTION ALL
    ACCEPT   fw   net      udp 53
    ACCEPT   fw   net      tcp 53
    ACCEPT   lan  fw       tcp 22
    ACCEPT   lan  fw       tcp 80
    DROP     lan:192.168.A.1A     net      all
    ?SECTION ESTABLISHED
    ?SECTION RELATED
    ?SECTION INVALID
    ?SECTION UNTRACKED
    ?SECTION NEW
```

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.A.1A* 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.A.1A     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:

- FA: ssh (TCP 22): PCA
- FA: HTTP (TCP 80): PCA

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

- /etc/shorewall/rules

```
#ACTION SOURCE   DEST              PROTO    DEST     SOURCE   ORIGINAL
#                          PORT     PORT
?SECTION NEW
DNAT    net lan:192.168.A.1A     tcp 22
DNAT    net lan:192.168.A.1A     tcp 80
```

Habrá que comprobar la integridad de los cambios.

```
FA#  shorewall check
```

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

```
FA#  shorewall restart
```

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

```
FA#  shorewall clear
```

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

```
FA#  shorewall stop
```

5 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/snat*, donde el campo "DEST" es la interfaz de salida de los paquetes enmascarados, "SOURCE" es la red de entrada de los paquetes a enmascarar y "ACTION" la acción de enmascaramiento (pues existen otras acciones posibles).

```
##############################################################################################  ...
#ACTION                  SOURCE                  DEST            PROTO    IPSEC     MARK     ...
MASQUERADE               192.168.A.0/24          eth0.2
MASQUERADE               192.168.202.0/24          eth0.2
MASQUERADE               192.168.203.0/24          eth0.2
MASQUERADE               192.168.204.0/24          eth0.2
```

En el ejemplo listado, existirán también las redes 192.168.202.0/24, 192.168.203.0/24, 192.168.204.0/24 como redes internas (accesibles desde la interfaz eth1.1) y cuya dirección origen será también necesario enmascarar en FA.

No obstante, 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*.

6 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.

- FA <–> PCA
- FA <–> FB,FC ...

- FA <–> PCB, PCC ... (puertos 22 y 80)
- FA <–> INTERNET
- PCA <–> INTERNET