TEMAS 7 CORTAFUEGOS - 8 PROXY

1. CORTAFUEGOS

CONOCIMIENTOS PREVIOS NECESARIOS

MODELO OSI
MODELO TCP/IP
PROTOCOLOS DE RED
PUERTOS TCP/UDP

COMANDOS DE RED WINDOWS
COMANDOS DE RED LINUX

CONFIGURACIÓN DE RED WINDOWS
CONFIGURACIÓN DE RED LINUX

ROUTER XP
ROUTER LINUX
ROUTER 2008 SERVER
ROUTER DISPOSITIVO MULTIFUNCIÓN
ROUTER CISCO
NAT
PAT
ENRUTAMIENTO ESTÁTICO
ENRUTAMIENTO DINÁMICO

1.1 TIPOS DE CORTAFUEGOS

Cortafuegos ***Firewall*** ***Proxy*** ***Application level firewall***

Un cortafuegos (firewall) es un dispositivo de control de acceso a redes que está diseñado para denegar todo el tráfico, exceptuando el que permita explícitamente.
Un firewall puede tener componentes hardware y software

Router: dispositivo de red que está diseñado para dirigir el tráfico tan rápido como sea posible.

Un cortafuegos es un dispositivo de seguridad que puede permitir que el tráfico apropiado fluya, mientras un router es un dispositivo de red que puede ser configurado para denegar cierto tráfico.

Existen dos **tipos** generales **de cortafuegos**: cortafuegos de capa **de aplicación** y cortafuegos **de filtrado de paquetes**. La forma en que son implementados los dos tipos tiene impacto en cómo se hace cumplir la ***política de seguridad***.

Un cortafuegos tendrá interfaces múltiples para cada red a la que se encuentre conectado.

Un conjunto de ***reglas*** de política define como se transporta el tráfico de una red hacia cualquier otra. Si una regla no permite especialmente que fluya el tráfico, el cortafuegos denegará o retirará los paquetes.

¿Los cortafuegos son utilizados únicamente en conexiones a Internet?
Un cortafuegos es un dispositivo de control de acceso a la red que puede ser utilizado en cualquier lugar donde el acceso deba ser controlado. Esto incluye las redes internas que deberían estar protegidas de otros sistemas internos.

Ejercicio: Buscar información (características y precio) de algún cortafuegos hardware.

Primera generación: cortafuegos de red – filtrado de paquetes
Segunda generación: cortafuegos de inspección e estado del paquete (si el paquete inicia una nueva conexión o forma parte de una conexión existente)
Tercera generación: cortafuegos de aplicación

1.1.1 CORTAFUEGOS DE INSPECCIÓN DE PAQUETES

Netfilter / iptables

Los cortafuegos de filtrado de paquetes también pueden ser paquetes de software que se ubican en la parte superior de los sistemas operativos de propósito general o sobre dispositivos hardware de cortafuegos.

El cortafuegos tendrá interfaces múltiples, una para cada red a la cual se encuentre conectado.

Un conjunto de reglas de política define como se transporta el tráfico de una red hacia cualquier otra. Si una regla no permite específicamente que fluya el tráfico, el cortafuegos denegará los paquetes o los retirará.

Las reglas de la política se hacen cumplir a través del uso de filtros de inspección de paquete. Los filtros examinan los paquetes y determinan si el tráfico está permitido con base en las reglas de la política y el estado del protocolo (esto se conoce como inspección de estado). Esto significa que cuando el protocolo se encuentra en cierto estado, solamente son esperados ciertos paquetes.

En este estado, se pueden esperar uno o dos paquetes, si aparece cualquier otro paquete para esta conexión, el cortafuegos lo retirará o lo denegará, ya que es incorrecto para el estado de conexión, incluso si la conexión es permitida por el conjunto de reglas.

Si el protocolo de aplicación se está ejecutando sobre TCP, la determinación del estado es relativamente fácil, en la medida en que el TCP mismo mantiene el estado.

Con un cortafuegos de filtrado de paquetes las conexiones no terminan en el cortafuegos, sino que viajan directamente hacia el sistema destino. A medida que los paquetes llegan al cortafuegos, éste determinará si el paquete y el estado de la conexión están permitidos por las reglas de la política. Si así fuera el paquete es puesto en camino. De no ser así el paquete es denegado o retirado.

Los firewall de filtrado de paquetes no utilizan proxies, de modo que el tráfico generado por un cliente es enviado directamente hacia el servidor.

Son más rápidos que los firewall de capa de aplicación.

1.1.2 CORTAFUEGOS DE CAPA DE APLICACIÓN: PROXY

Application level firewall

Los cortafuegos de capa de aplicación (también denominados ***cortafuegos Proxy***) son paquetes de software que se ubican en la parte superior de los sistemas operativos de propósito general o sobre dispositivos hardware de cortafuegos.

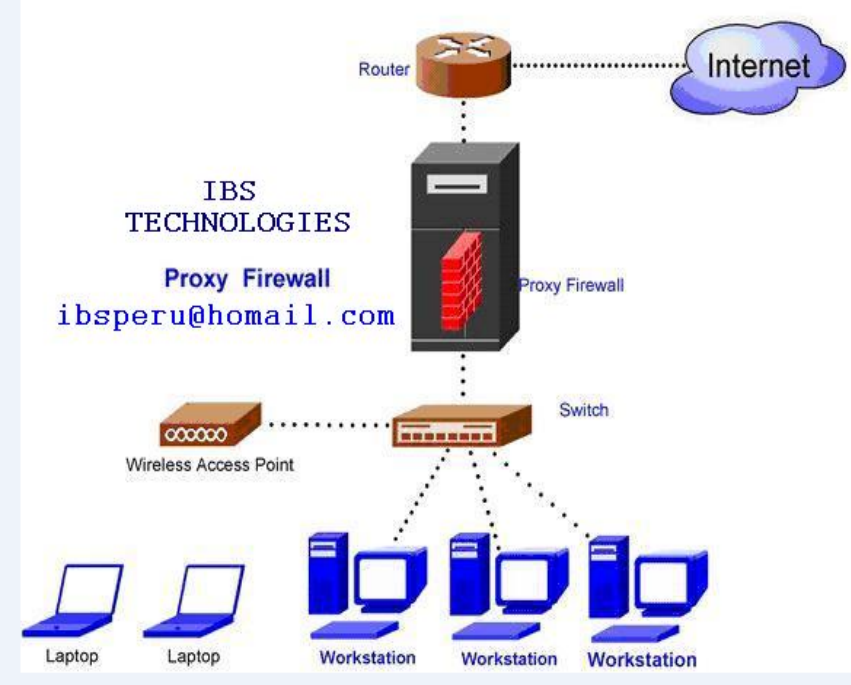
Las reglas de la política se hacen cumplir a través del uso de proxies. En cortafuegos de capa de aplicación cada protocolo que sea permitido debe tener su propio proxy. Los mejores proxies son aquellos que están construidos específicamente para el protocolo que será permitido.

Con un cortafuegos de capa de aplicación, todas las conexiones terminan en el cortafuegos. Una conexión se inicia en el sistema cliente y se dirige hacia la interfaz interna del cortafuegos. El cortafuegos acepta la conexión, analiza el contenido del paquete y el protocolo que será utilizado, y determina si las reglas de la política permiten el tráfico. Si es así, el cortafuegos inicia una nueva conexión desde su interfaz externa hacia el sistema servidor.

Los firewall de capa de aplicación también utilizan proxies para conexiones entrantes. El Proxy en el cortafuegos recibirá la conexión entrante y procesará los comandos antes que el tráfico se envíe hacia el sistema destino. De este modo el cortafuegos puede proteger a los sistemas de ataques iniciados por medio de las aplicaciones.

Los cortafuegos de capa de aplicación tendrán proxies para los protocolos utilizados con mayor frecuencia, como HTTP, SMTP y FTP. Otros proxies pueden no estar disponibles. Si un Proxy no está disponible, el protocolo no puede ser utilizado a través del cortafuegos.

El cortafuegos también oculta las direcciones de los sistemas detrás del cortafuegos de capa de aplicación. Puesto que todas las conexiones se originan y terminan en las interfaces del cortafuegos, los sistemas internos no son directamente visibles hacia el exterior y, de este modo, el esquema de direccionamiento interno puede quedar oculto.



FUNCIONAMIENTO DEL PROXY

Los equipos que pertenecen a una red que tiene instalado un proxy, cuando se comunican con el exterior a través de uno de los protocolos activos en el proxy, en realidad están intercambiando paquetes con el proxy, y es el proxy el que intercambia paquetes con los equipos del exterior, de forma que cuando este recibe respuesta, a su vez contesta a los equipos externos.

¿Cómo sabe que destino le corresponde?

Utiliza una ***tabla de estado*** que crea el proxy para saber a qué equipo de la red interna le corresponde cada paquete que llega de la red externa.

RED INTERNA	RED EXTERNA
IP:PUERTO	IP:PUERTO
…	…

Características:

Permiten el acceso a máquinas privadas (con una dirección IP privada) que no están conectadas directamente a internet.

Controlan el acceso aplicando reglas o normas.

Registran el tráfico que pasa por el proxy.

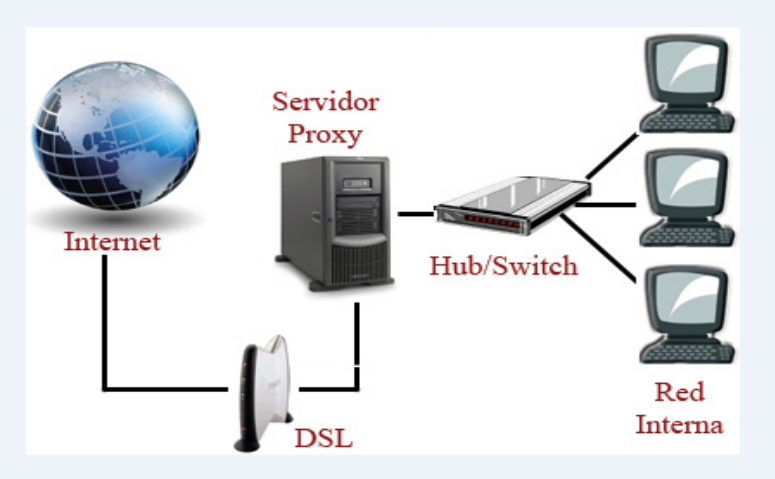
Controlan el contenido solicitado y descargado para detectar la presencia de posibles ataques a través de virus,...

Controlan la seguridad de la red local ante posibles ataques o intrusiones en el sistema.

Funcionan como una caché: almacenando los contenidos descargados, de tal modo que no sea necesaria una segunda descarga si hay una segunda solicitud.

VENTAJAS DE LA UTILIZACIÓN DE UN SERVIDOR PROXY-CACHÉ

Mayor velocidad de navegación.
Uso más eficiente de la línea de conexión a internet.
Cortafuegos de contenidos.
Filtrado de servicios.



Diferencias entre router, cortafuegos y proxy

TIPOS DE PROXIES – CLASIFICACIONES POSIBLES:

Proxy como **intermediario** entre el proceso cliente y el proceso servidor.

Programa que hace de intermediario entre un programa cliente y un programa servidor.
Protocolo TCP: establecimiento de conexión entre el cliente y el servidor. Proxy como intermediario en esta conexión.

Tipos de proxy **en función del servicio** al que hacen de intermediario: **proxy web**, **proxy arp**, …

Casi todos los protocolos de la capa de aplicación son cliente-servidor, si utilizamos un programa intermedio podremos hablar de proxy HHTP (web), proxy FTP, proxy SMTP, POP3 (correo electrónico), IMAP, …

Proxy Web – proxy HTTP
Proxy DNS

Otros protocolos de capas mas bajas también pueden trabajar con proxies para ampliar su área de acción (proxy ARP, proxy DHCP,…) y saltar al otro lado del router…

Proxy **caché**

Proxy caché: el proxy dispone de una zona de memoria donde guarda las respuestas de los servidores a las solicitudes que realiza y, en el caso de que otro cliente vuelva a pedir lo mismo, el proxy busca y utiliza la información de su memoria en vez de pasar la solicitud al servidor.
Proxy no caché: el proxy no guarda información sobre los resultados de las consultas que los clientes piden a través de él.

Proxy **transparente** – Proxy configurado en el cliente.

Proxy transparente: el cliente no es consciente de estar utilizando un proxy, no ha tenido que modificar la configuración (pero sus solicitudes pasan a través de un proxy).

Proxy no transparente: en el cliente tenemos que poner la dirección del proxy, tenemos que configurar el cliente para que utilice el proxy.

Proxy **LAN** – proxy **remoto**

Proxy ubicado en nuestra LAN: es nuestra empresa la que configura y administra el proxy de acuerdo con sus necesidades…

Proxy remoto – proxy abierto (open): utilizo un proxy que no está en mi LAN y que tampoco está en la red del servidor… está en Internet… alternativa utilizada para saltarse el cortafuegos de la empresa, para un "pretendido" anonimato en la navegación por internet.

Proxy en la red del cliente (LAN) – proxy **inverso** (en la red del servidor)

Proxy ubicado en nuestra LAN: para controlar los servicios que se solicitan desde los puestos de usuario, denegar permisos de acceso a determinados servicios, auditar los servicios solicitados, cachear las solicitudes para agilizar las respuestas…

Proxy ubicado en la misma red que el servidor (**proxy inverso – reverse proxy**): para proteger el servidor (evitando que le lleguen determinadas solicitudes – filtrando las solicitudes), para balancear la carga cuando el servicio es ofrecido por varios servidores.

Proxy como elemento de **seguridad** informática.

ACL: en función de unas reglas incluidas en listas y asociadas a las interfaces del router filtramos el tráfico que puede y que no puede enrutarse.

Listas de control de acceso
Reglas de control de acceso
Tipos de filtro

Proxy **NAT / enmascaramiento**: Para ocultar las direcciones internas – privadas (y puertos PAT).

VPN: utilizamos una conexíon VPN como forma de acceder a una conexión a Internet desde una red insegura.

Proxy **por anonimato**: utilizado para ocultar al servidor quien es el cliente que solicita el servicio.

WAF: proxy inverso especializado en la detección de tráfico web malintencionado con el fin de proteger a los servidores web objeto de la solicitud.

Log: Utilización adecuada de los log de la actividad del proxy.

Proxy como elemento de **inseguridad** informática.

La mayoría de los ataques en los que el atacante y la víctima comparten la misma red se basan en técnicas de "man in de middle" que no son más que distintas formas de desviar el tráfico de la víctima a través del equipo del atacante que utilizando distintos tipos de proxies evitará que la víctima se de cuenta. (Suplantación de la puerta de enlace, suplantación del servidor DNS, suplantación de un servidor web,…)

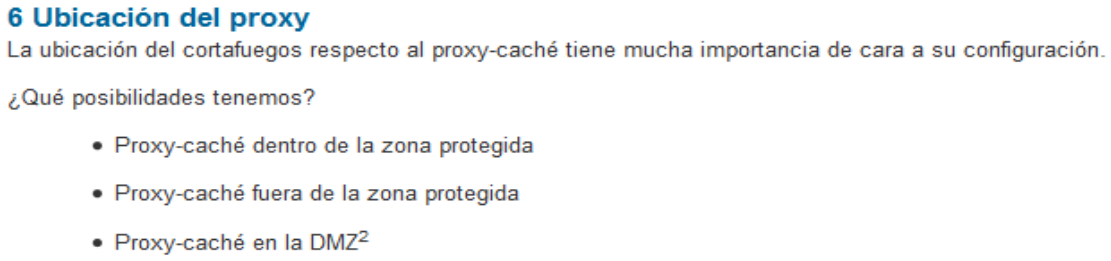
1.1.3 CORTAFUEGOS HIBRIDOS – GSP – PROXY DE SERVICIOS GENÉRICOS

Squid ***WinGate***

El proxy de servicios genéricos (GSP, Generic Services Proxy) fue creado para permitir que los proxies de capa de aplicación manejaran otros protocolos necesarios para los administradores de redes y seguridad.

Lo que GSP hizo en realidad fue crear una forma para que los cortafuegos de capa de aplicación actuasen como cortafuegos de filtrado de paquetes.

Ahora tenemos muchos cortafuegos híbridos en el mercado.



1.1.4 WAF - CORTAFUEGOS DE APLICACIONES WEB – WEB APPLICATION FIREWALLS

ModScurity

Los ***cortafuegos de aplicaciones web o WAF*** permiten proteger sus aplicaciones web de ataques sin necesidad de modificar la aplicación ni tocar una sola línea de código.

Medida de seguridad sobre los flujos HTTP, HTTPS o FTP.

Nos defiende de ataques de ***inyección de SQL, manipulación de parámetros, envenenamiento de cookies, cross-site scripting, desbordamiento de búfer***,…

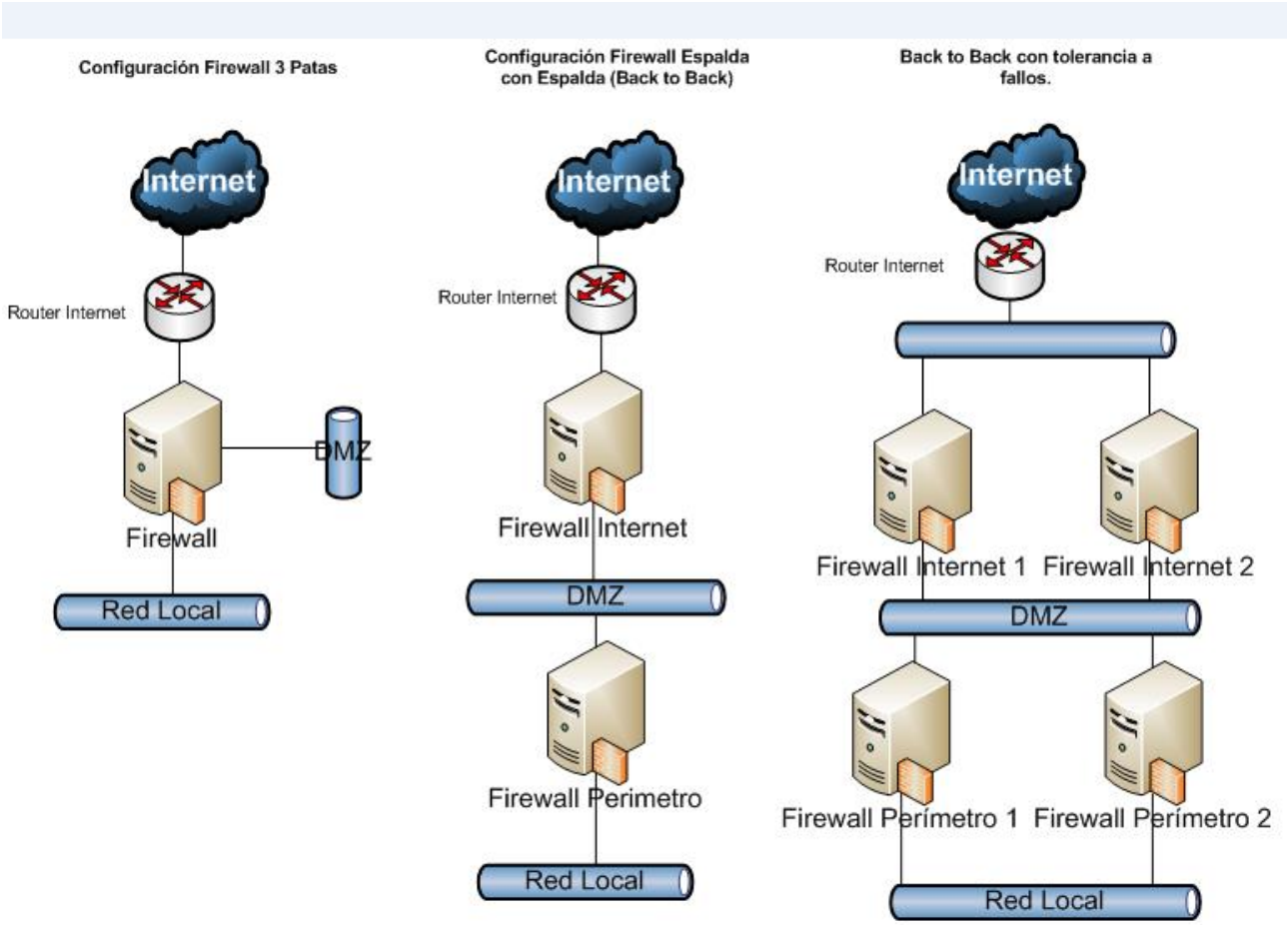
Los WAF pueden bloquear los ataques web antes de que lleguen al servidor.
Pueden utilizarse como intermediarios virtuales (transparentes) para proteger servidores vulnerables.

ModSecurity es uno de los WAF más utilizados.

Ejercicio: instalar una solución WAF basado en Apache y ModSecurity.

Introducción a los Web Application Firewalls (WAF)
Cortafuegos de aplicaciones web (WAF) Cómo proteger sus aplicaciones web

1.2 DISEÑO DE CORTAFUEGOS



Un buen diseño del conjunto de reglas puede ser tan importante para un cortafuegos como un buen hardware.

La mayor parte de de los cortafuegos trabajan con base en un "primer encuentro" cuando deciden si aceptan o rechazan un paquete.

Cuando se diseña un conjunto de ***reglas***, el ***algoritmo de "primer encuentro"*** dicta que las reglas mas específicas sean colocadas en la parte superior del conjunto de reglas, y que las reglas menos específicas o más generales sean colocadas en la parte inferior.

Esta ubicación garantiza que las reglas más generales no enmascararán a las reglas más específicas.

Ejemplo de ***acciones*** que puede incluir una regla:

las acciones que se pueden realizar en la tabla FILTER son:

- -j ACCEPT. Acepta el tráfico.
- -j DROP. Elimina el tráfico.
- -j REJECT. Rechaza el tráfico e informa al equipo de origen.
- -j LOG –log-prefix "IPTABLES_L". Registra el tráfico que cumple los criterios en /var/log.

Iptables – guía rápida

Supuesto ejemplo:

Supondremos que existen los siguientes sistemas en la organización y que ésta desea recibir conexiones desde Internet.

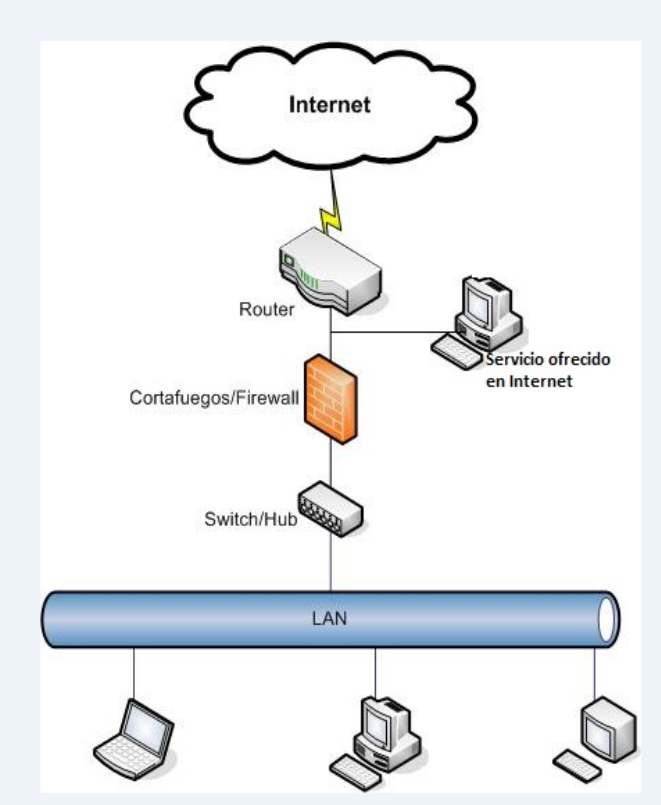
- Un servidor web ofreciendo servicio solamente en el puerto 80.
- Servidor de correo ofreciendo servicio en el puerto 25. Este sistema acepta todo el correo entrante y envía todo el correo saliente. El servidor de correo interno hace contacto con este sistema de manera periódica, para obtener el correo entrante y enviar el tráfico saliente.
- Existe un sistema DNS interno que debe interrogar a sistemas de Internet para resolver nombres de direcciones, pero la organización no presenta su propio DNS externo principal.
- La política de Internet para la organización permite que los usuarios internos utilicen los servicios siguientes: HTTP, HTTPS, FTP, Telnet, SSH

Con base en esta política, podemos construir las reglas de la política para diversas arquitecturas.

1.2.1 SISTEMAS ACCESIBLES A INTERNET AL EXTERIOR DEL CORTAFUEGOS

Los servicios que ofrecemos en Internet no están protegidos por el cortafuegos.

El cortafuegos no permite el tráfico desde el exterior.



Supuesto solución: El cortafuegos solo protege la red interna.

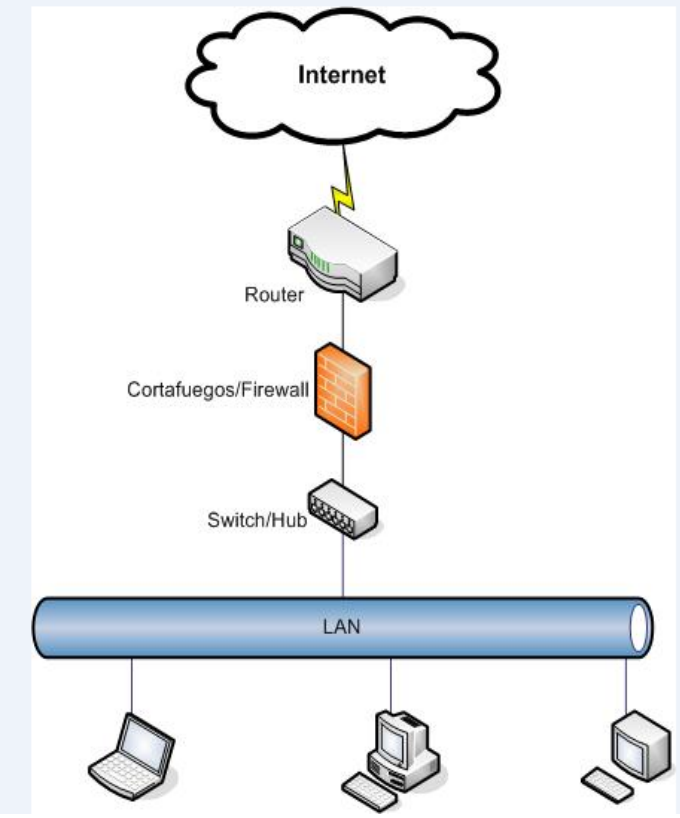
Número de regla	IP origen	IP destino	Servicio	Acción
1	Servidor de correo interno	Servidor de correo	SMTP	ACCEPT
2	Red interna	Cualquiera	HTTP, HTTPS, FTP, Telnet, SSH	ACCEPT
3	DNS interno	Cualquiera	DNS	ACCEPT
4	Cualquiera	Cualquiera	Cualquiera	DROP

El filtrado de paquete debe ser colocado en el router para permitir únicamente la entrada de HTTP desde el exterior hacia el servidor web, y la entrada de SMTP desde el exterior hacia el servidor de correo. (Ej. Mediante ACL en el router).

1.2.2 CORTAFUEGOS SIMPLE

La empresa no ofrece servicios en Internet.

El cortafuegos no permite el tráfico desde el exterior.



Supuesto solución: El cortafuegos permite el acceso a determinados servicios desde Internet

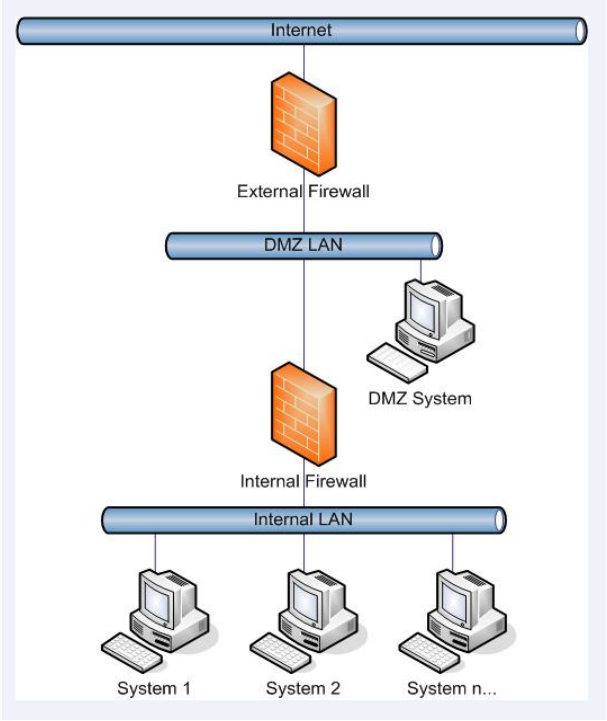
Número de regla	IP origen	IP destino	Servicio	Acción
1	Cualquiera	Servidor web	HTTP	ACCEPT
2	Cualquiera	Servidor de correo	SMTP	ACCEPT
3	Servidor de correo	Cualquiera	SMTP	ACCEPT
4	Red interna	Cualquiera	HTTP, HTTPS, FTP, Telnet, SSH	ACCEPT
5	DNS interno	Cualquiera	DNS	ACCEPT
6	Cualquiera	Cualquiera	Cualquiera	DROP

El cortafuegos agrega las reglas manejadas por el router en la arquitectura anterior.

1.2.3 CORTAFUEGOS DOBLE

Los servicios que ofrecemos en Internet se colocan en la DMZ que es la red que hay entre los dos cortafuegos

El cortafuegos interno no permite el tráfico desde el exterior, el cortafuegos externo permite el tráfico a los servicios que ofrece la DMZ.



Supuesto solución: DMZ entre dos cortafuegos.

FIREWALL EXTERIOR:

Número de regla	IP origen	IP destino	Servicio	Acción
1	Cualquiera	Servidor web	HTTP	ACCEPT
2	Cualquiera	Servidor de correo	SMTP	ACCEPT
3	Servidor de correo	Cualquiera	SMTP	ACCEPT
4	Red interna	Cualquiera	HTTP, HTTPS, FTP, Telnet, SSH	ACCEPT
5	DNS interno	Cualquiera	DNS	ACCEPT
6	Cualquiera	Cualquiera	Cualquiera	DROP

FIREWALL INTERIOR:

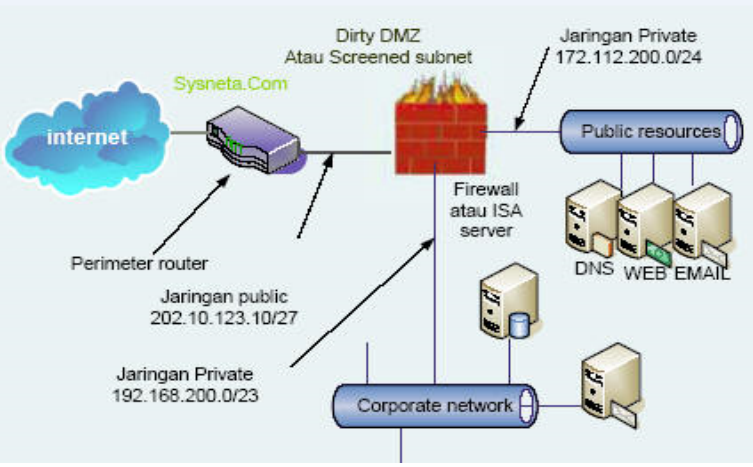
Número de regla	IP origen	IP destino	Servicio	Acción
1	Servidor de correo interno	Servidor de correo	SMTP	ACCEPT
2	Red interna	Cualquiera	HTTP, HTTPS, FTP, Telnet, SSH	ACCEPT
3	DNS interno	Cualquiera	DNS	ACCEPT
4	Cualquiera	Cualquiera	Cualquiera	DROP

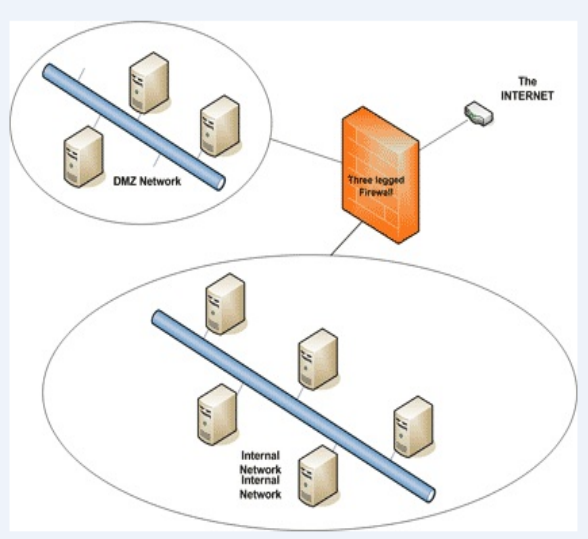
Nota: Estos ejemplos son muy simples, pero nos sirven para ver como trabajan los cortafuegos a fin de permitir solamente el tráfico apropiado.

1.2.4 CORTAFUEGOS DE TRES VÍAS

Utilizamos un solo cortafuegos con tres interfaces: una conectada a Internet, otra a la LAN protegida y otra a la DMZ

Los servicios que ofrecemos en Internet se colocan en la DMZ que es la red que hay conectada a una de las tres interfaces del cortafuegos; según las reglas, la única que es accesible desde Internet



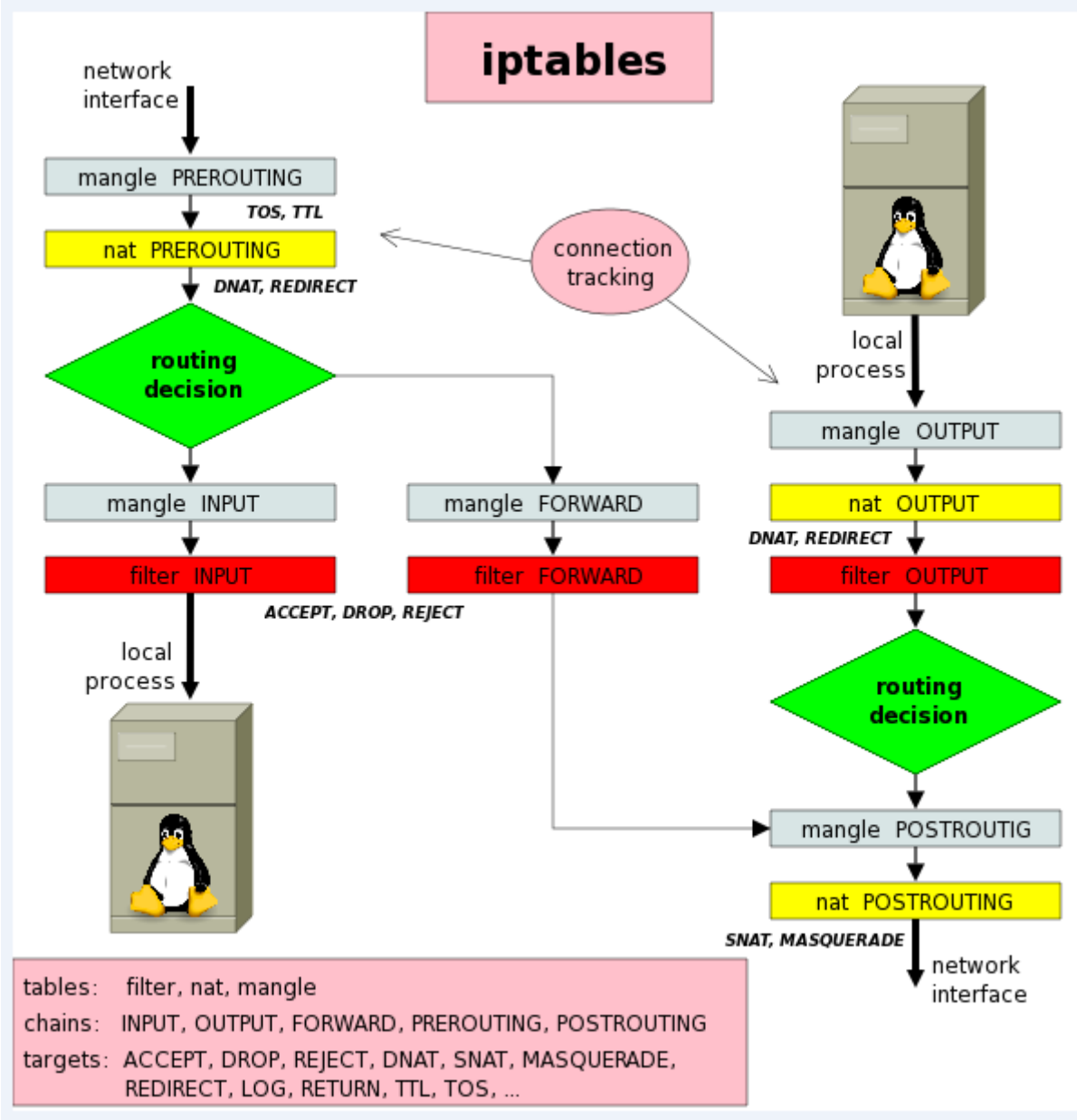


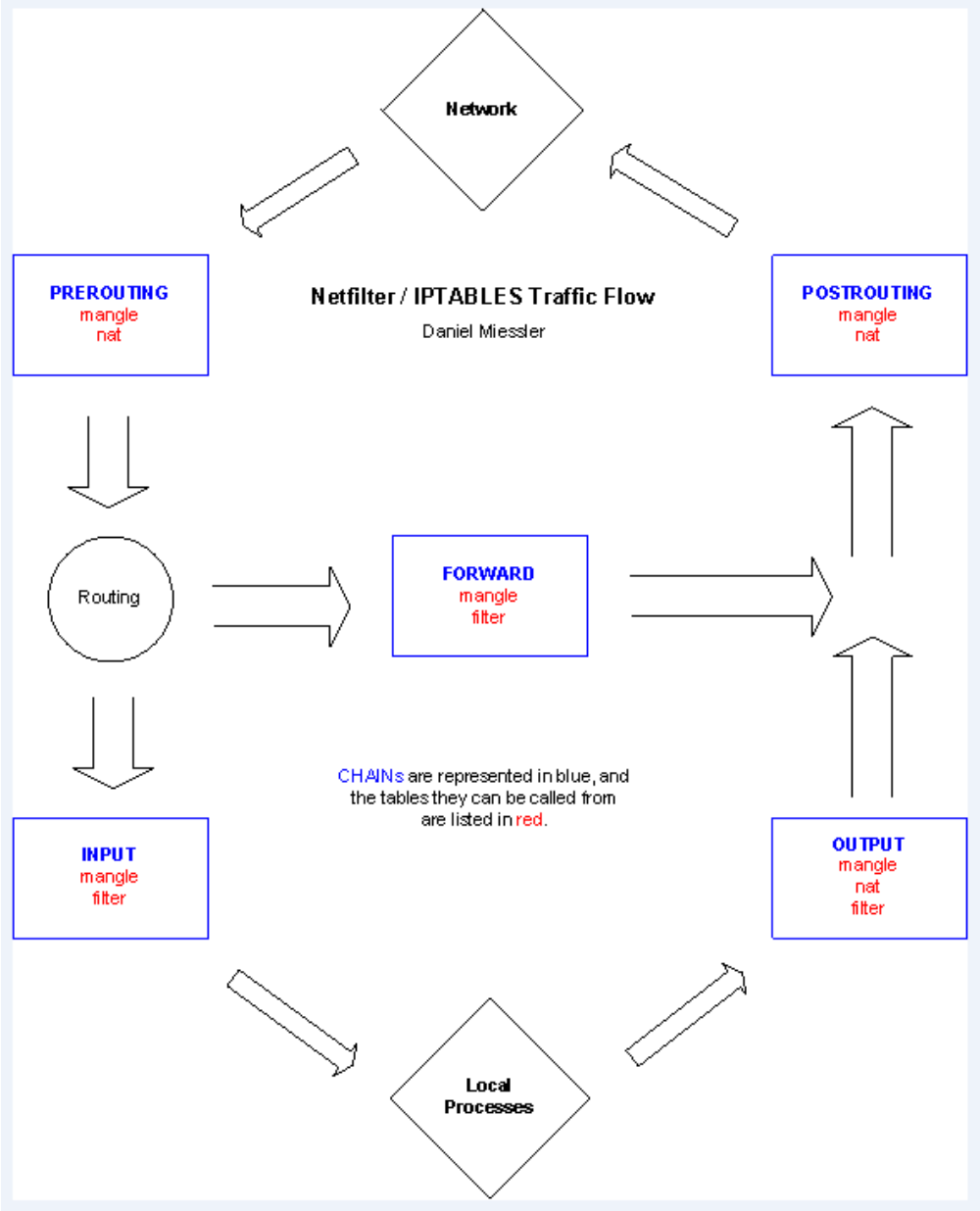
En este caso, tenemos un solo cortafuegos con tres tarjetas de red; las reglas del cortafuegos permitirán el acceso desde Internet a los servicios ofrecidos por la empresa en una tarjeta de red (DMZ) y no permitirán el acceso desde Internet a los equipos conectados en la otra tarjeta de red (LAN).

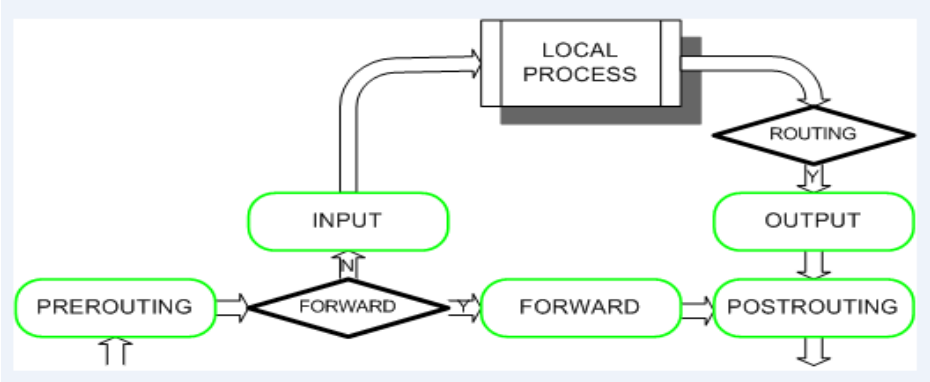
2. IPTABLES

www.pello.info *netfilter* *Netfilter/iptables*

2.1 MODELOS GENERALES




Network
PREROUTING
mangle
nat
Netfilter / IPTABLES Traffic Flow
Daniel Miessler
POSTROUTING
mangle
nat
Routing
FORWARD
mangle
filter
CHAINs are represented in blue, and
the tables they can be called from
are listed in red.
INPUT
mangle
filter
OUTPUT
mangle
nat
filter
Local
Processes


LOCAL
PROCESS
ROUTING
Y
INPUT
OUTPUT
N
PREROUTING
FORWARD
Y
FORWARD
POSTROUTING

2.1.1 TABLAS Y CADENAS DE REGLAS

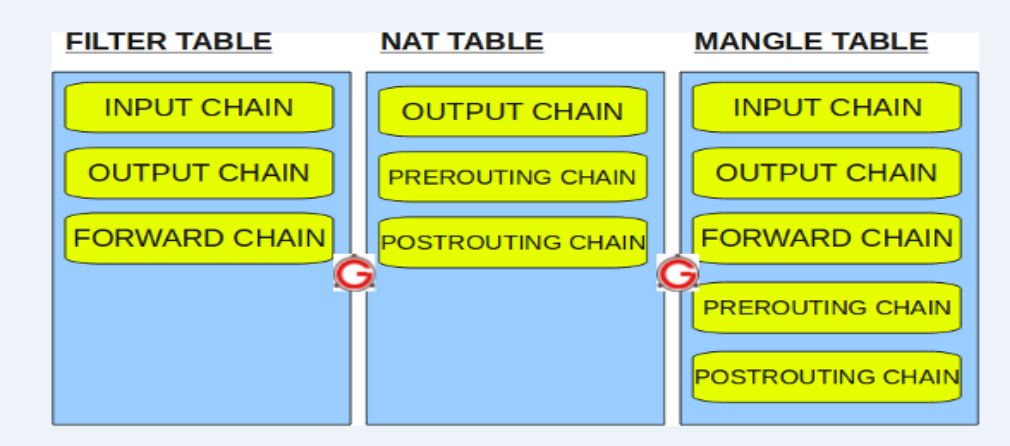


tabla < cadena < regla-acción

Cada tabla contiene cadenas de reglas que afectan al funcionamiento del cortafuegos definiendo la acción que realiza sobre un paquete
El orden en el que se escriben las reglas en una cadena es muy importante
Si el paquete no cumple una regla pasa a evaluar la siguiente regla
Si el paquete cumple la regla se ejecuta esa acción y no evalúa las siguientes reglas de esa cadena
Al final de una cadena podemos colocar una regla que se aplicaría por defecto (en el caso de que el paquete no se hubiese ajustado a ninguna regla de esa cadena).

FILTER: Filtrador de paquetes
Tabla por defecto; la más básica.
Utilizada para filtrar paquetes de entrada, salida y reenvío
Utilizaremos las cadenas incluidas en esta tabla para colocar las reglas que nos permiten aceptar o denegar el tráfico que atraviesa, genera o recibe el equipo en función de la política que debamos implementar.

INPUT:
Paquetes entrantes: cadena de reglas que evalúa los paquetes dirigidos a la máquina.
Filtra paquetes dirigidos a la máquina local a través de un dispositivo de red.

FORWARD:
Paquetes redirigidos: cadena de reglas que evalúa los paquetes que llegan a la máquina destinados a otra máquina; filtra los paquetes que son reenviados.
Filtra paquetes recibidos desde un dispositivo de red y enviados por otro dispositivo de red en la misma máquina.

OUTPUT:
Paquetes salientes: cadena de reglas que evalúa los paquetes generados por la máquina.
Filtra paquetes enviados desde nuestra máquina local a través de un dispositivo de red.

NAT: Traductor de direcciones de red

Tabla que utiliza el protocolo de enmascaramiento para que otras máquinas se conecten a una serie de servicios a través de la IP del cortafuegos. Para ello se modifica la cabecera de los paquetes. Ocultando la IP de la máquina que realmente ofrece el servicio.

Utilizaremos las cadenas incluidas en esta tabla para manipular los paquetes antes y después de tomar decisiones sobre su enrutamiento.

PREROUTING:

Paquetes preenrutados: afecta a los paquetes que llegan a la máquina antes de tomar una decisión sobre su enrutamiento.

Modifica paquetes recibidos a través de un dispositivo de red antes de enrutar.

OUTPUT:

Paquetes salientes: afecta a los paquetes generados por la máquina justo antes de ser enviados

Modifica paquetes generados en la propia máquina antes de que sean dirigidos a través de un dispositivo de red.

POSTROUTING:

Paquetes postrutados: afecta a los paquetes reenviados por la máquina justo antes de ser enviados.

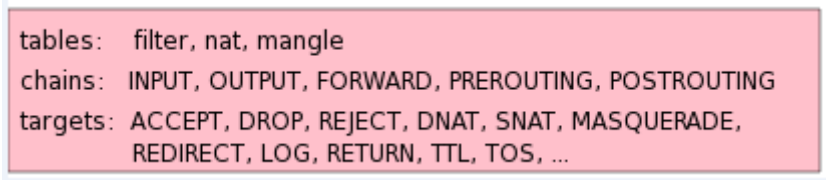
Cadena que modifica paquetes antes de que sean enviados (enrutados) a través de un dispositivo de red.

MANGLE: Alteración de paquetes

Manipula el estado de un paquete

Utilizaremos las cadenas incluidas en esta tabla para incluir reglas que manipulen el estado de los paquetes.

2.1.2 REGLAS



REGLAS

ACCIÓN A EJETUTAR

Acción que realizamos con los paquetes que cumplen la regla correspondiente.

ACCEPT	aceptar	permite el paso
DROP	denegar	no permite el paso y no notifica nada
REJECT	rechazar	no permite el paso y notifica “destino inalcanzable”
DENY	denegar	no permite el paso y no notifica nada
MASQUERADE	enmascarar	(solo en cadenas POSTROUTING con NAT)
REDIRECT	redirigir	(solo en cadenas PREROUTING y OUTPUT con NAT)

Ejemplos de comandos iptables:

Limpia las reglas del cortafuegos (borra - inicializa)

iptables –F

Acepta todos los paquetes de entrada, salida y reenvio…enruta todos con todos. Establecemos acción por defecto (Tabla FILTER)

iptables –P INPUT ACCEPT
iptables –P OUTPUT ACCEPT
iptables –P FORWARD ACCEPT

Desde la propia máquina debe poder acceder a sus propios servicios (interfaz loopback). Permite la salida y la entrada de todos los paquetes por la interfaz loopback.

iptables –A INPUT –i lo –j ACCEPT
iptables –A OUTPUT –o lo –j ACCEPT

2.2 EJEMPLOS

Configurar nombre y tarjetas de red de las máquinas
Configurar el servicio de enrutamiento en el router US01
Crear un directorio para alojar los scripts: **/script/**
Crear un script

Creamos un fichero.sh
Escribimos el script…
#!/bin/sh

Permisos de ejecución

¿Como se ejecuta?

sh nombrescript

Donde se ejecuta si queremos que se ejecute cuando reiniciamos la máquina o cuando reiniciamos el servicio de red.

/etc/rc.local	cuando la máquina se reinicia
/etc/init.d/networking	cuando se reinicia el servicio de red y cuando se reinicia la máquina

2.2.1 LIMPIAR REGLAS

Script: ***LimpiaReglas.sh***

```
echo "Deteniendo el corta fuegos y permitiendo todo el tráfico..."
## FLUSH de reglas
iptables -F
iptables -X
iptables -t nat -F
iptables -t nat -X
iptables -t mangle -F
iptables -t mangle -X
## Permitir todo el tráfico por defecto
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
```

2.2.2 MOSTRAR REGLAS

Script: ***MuestraReglas . sh***

```
## MOSTRA de reglas
# CADENAS Y REGLAS DE LA TABLA FILTER
iptables -L
# CADENAS Y REGLAS DE LA TABLA NAT
iptables –t nat-L
# CADENAS Y REGLAS DE LA TABLA MANGLE
iptables –t mangle -L
```

2.2.3 CORTAFUEGOS US01

Script: ***CortafuegosUS01.sh***

```
# Red local (LAN): 192.168.0.0/24
# IP privada del router : eth0: 192.168.0.1/24      (Esta máquina también es el proxy web)
# IP pública del router: eth1: 5.5.5.5
# IP del servidor web que ofrece servicio en Internet: 192.168.0.12/24
## LIMPIAR de reglas
iptables -F
iptables -X
iptables -Z
iptables -t nat -F
iptables -t nat -X

## ESTABLECER POLÍTICA POR DEFECTO
iptables -P FORWARD DROP
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -t nat -P PREROUTING ACCEPT
iptables -t nat -P POSTROUTING ACCEPT

## REGLAS DE LA TABLA FILTER
# CADENA FORWARD
# Permitimos el tráfico desde la red 192.168.0.0/24 y las respuestas a estas solicitudes
iptables -A FORWARD -s 192.168.0.0/24 -j ACCEPT
iptables -A FORWARD -d 192.168.0.0/24 -m state --state ESTABLISHED,RELATED -j ACCEPT

# Tráfico enrutado hacia un servirdor Web
# Permitir tráfico  que atraviesa el cortafuegos hacia el servidor web
iptables -A FORWARD -p tcp --dport 80 -j ACCEPT

# Tráfico enrutado hacia un servidor VPN
```

Permitimos que el tráfico tcp hacia el puerto 1723 (Escucha de la VPN) que entre por la tarjeta externa atraviese el router
iptables -A FORWARD -i WAN -p tcp --dport 1723 -j ACCEPT
Permitimos que el tráfico gre (Encapsulamiento de la VPN) que entre por la tarjeta externa atraviese el router
iptables -A FORWARD -i WAN -p gre -j ACCEPT

 # CADENA INPUT
Solo podremos conectarnos al cortafuegos desde la máquina 192.168.0.50. Establecer una conexón nueva con nuestro equipo.
iptables -A INPUT -s 192.168.0.50/32 -j ACCEPT
Permitimos los paquetes entrantes referentes a conexiones estableciadas
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
Permitimos la ip del router para que devuelva el ping a si mismo en el propio router
iptables -A INPUT -s 192.168.0.1/32 -j ACCEPT
iptables -A INPUT -s 5.5.5.5/32 -j ACCEPT

 # CADENA OUTPUT
Pendiente estudiar los filtros de la cadena OUTPUT,
En principio recomendamos no filtral el OUTPUT
Si decidimos poner un filtro, tener pendientes los servicios que la máquina puede necesitar desde otros equipos situados dentro o fuera de nuestra LAN

REGLAS DE LA TABLA NAT
 # CADENA PREROUTING
El equipo 10 no pasa por el proxy
iptables -t nat -A PREROUTING -s 192.168.0.10 -j ACCEPT
El equipo 11 no pasa por el proxy
iptables -t nat -A PREROUTING -s 192.168.0.11 -j ACCEPT
El equipo 12 no pasa por el proxy
iptables -t nat -A PREROUTING -s 192.168.0.12 -j ACCEPT

Tráfico saliente hacia cualquier servidor Web
Redirigimos del tráfico originado en la LAN hacia el puerto 80 (HTTP) hacia nuestro servidor proxy ubicado en la misma máquina que el firewall pero en el puerto 3128
iptables -t nat -A PREROUTING -s 192.168.0.0/24 -p tcp --dport 80 -j DNAT --to 192.168.0.1:3128

Tráfico entrante hacia el servidor Web
Redirigimos el tráfico entrante desde Internet en el pueto 80 para el servidor web interno
iptables -t nat -A PREROUTING -d 5.5.5.5 -i eth1 -p tcp --dport 80 -j DNAT --to 192.168.0.12:80

Tráfico entrante hacia el servidor VPN

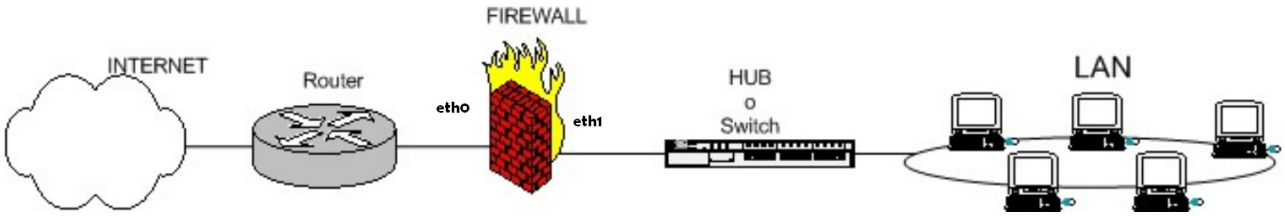
Redirigimos el tráfico del puerto tcp 1723 (Escucha VPN) entrante por la interfaz externa hacia DC10 en el puerto 1723
iptables -t nat -A PREROUTING -i eth1 -p tcp --dport 1723 -j DNAT --to 192.168.0.10:1723
Redirigimos el tráfico entrante del protocolo gre (Encapsulamiento de la VPN) entrante por la interfaz externa hacia DC10
iptables -t nat -A PREROUTING -i eth1 -p gre -j DNAT --to 192.168.0.10

 # CADENA POSTROUTING
#NAT en el tráfico saliente
iptables -t nat -A POSTROUTING -o ext -j MASQUERADE

MOSTRAR de reglas
CADENAS Y REGLAS DE LA TABLA FILTER
iptables -L
CADENAS Y REGLAS DE LA TABLA NAT
iptables –t nat-L
CADENAS Y REGLAS DE LA TABLA MANGLE
iptables –t mangle -L

2.2.4 FIREWALL DE UNA LAN CON SALIDA A INTERNET (Pello)

www.pello.info



Esquema de firewall típico entre red local 192.168.10.0/24 e internet

Ahora vamos a ver una configuración de firewall iptables para el típico caso de red local que necesita salida a internet.
¿Qué es lo que hace falta?
Obviamente, una regla que haga NAT hacia fuera (enmascaramiento en iptables), con lo que se haría dos veces NAT en el firewall y en el router.

Entre el router y el firewall lo normal es que haya una red privada (192.168.1.1 y 192.168.1.2 por ejemplo), aunque dependiendo de las necesidades puede que los dos tengan IP pública.

El router se supone que hace un NAT completo hacia dentro (quizá salvo puerto 23), o sea que desde el exterior no se llega al router si no que de forma transparente se "choca" contra el firewall.

Lo normal en este tipo de firewalls es poner la política por defecto de FORWARD en denegar (DROP), pero eso lo vemos más adelante.
Veamos cómo sería este ***firewall−gateway***:

```
#!/bin/sh
## SCRIPT de IPTABLES − ejemplo del manual de iptables
## Ejemplo de script para firewall entre red−local e internet
##
## Pello Xabier Altadill Izura
## www.pello.info − pello@pello.info

echo −n Aplicando Reglas de Firewall...

## FLUSH de reglas

iptables −F
iptables −X
iptables −Z
iptables −t nat −F

## Establecemos politica por defecto
```

```
iptables −P INPUT ACCEPT
iptables −P OUTPUT ACCEPT
iptables −P FORWARD ACCEPT
iptables −t nat −P PREROUTING ACCEPT
iptables −t nat −P POSTROUTING ACCEPT

## Empezamos a filtrar
## Nota: eth0 es el interfaz conectado al router y eth1 a la LAN
# El localhost se deja (por ejemplo conexiones locales a mysql)

/sbin/iptables −A INPUT −i lo −j ACCEPT

# Al firewall tenemos acceso desde la red local

iptables −A INPUT −s 192.168.10.0/24 −i eth1 −j ACCEPT

# Ahora hacemos enmascaramiento de la red local
# y activamos el BIT DE FORWARDING (imprescindible!!!!!)

iptables −t nat −A POSTROUTING −s 192.168.10.0/24 −o eth0 −j MASQUERADE

# Con esto permitimos hacer forward de paquetes en el firewall, o sea
# que otras máquinas puedan salir a traves del firewall.

echo 1 > /proc/sys/net/ipv4/ip_forward

## Y ahora cerramos los accesos indeseados del exterior:
# Nota: 0.0.0.0/0 significa: cualquier red
# Cerramos el rango de puerto bien conocido

iptables −A INPUT −s 0.0.0.0/0 −p tcp −dport 1:1024 −j DROP
iptables −A INPUT −s 0.0.0.0/0 −p udp −dport 1:1024 −j DROP

# Cerramos un puerto de gestión: webmin

iptables −A INPUT −s 0.0.0.0/0 −p tcp −dport 10000 −j DROP
echo " OK . Verifique que lo que se aplica con: iptables −L −n"

# Fin del script
```

2.3 MANTENIMIENTO Y MONITORIZACIÓN

ENRUTAMIENTO

/proc/sys/net/ipv4/ip_fordwad fichero que contiene 0 si el servicio de enrutamiento está desactivado y 1 si está activo.

echo 1> /proc/sys/net/ipv4/ip_forwardactiva el servicio de enrutamiento.

/etc/sysctl.conf

sysctl –w net.ipv4.ip_forward=1 (descomentar)

sysctl –a verifica el estado de enrutamiento

IPTABLES

/etc/sysconfig/iptables fichero necesario para activar iptables; debe existir cuando arranca la máquina.

iptables-save crea el fichero /etc/sysconfig/iptables
service iptables save crea el fichero /etc/sysconfig/iptables
service iptables restart reinicia el servicio iptables

Ejercicio:

Mostrar reglas cargadas
Limpiar reglas (bajar el cortafuegos)
Cambiar el cortafuegos y recargar iptables sin reiniciar la máquina
Proponer e implementar un cortafuegos para una máquina que no es router US12: iptables como cortafuegos local.

2.4 ALTERNATIVAS

2.4.1 M0N0WALL

M0n0wall **m0n0wall (Wikipedia)**

Articulo sobre m0n0wall (implementaciones)

3. SQUID

Squid **Squid (wikipedia)**

Squid: servidor proxy-caché – Observatorio tecnológico

Manual de instalación – Jorge Armando y Alejandro Gabriel

Cómo configurar squid – Joel Barrios

Squid en Ubuntu

Manuales alcance libre: Configuración de Squid **Opciones básicas**

- Squid
  - Configuración de Squid: Opciones básicas para servidor de intermediación (Proxy).
  - Configuración de Squid: Cachés en jerarquía.
  - Configuración de Squid: Acceso por Autenticación.
  - Configuración de Squid: Restricción de acceso a Sitios de Red.
  - Configuración de Squid: Restricción de acceso a contenido por extensión.
  - Configuración de Squid: Restricción de acceso por horarios.
  - Cómo incluir supervisión contra virus en Squid con SquidClamAV Redirector.
  - Configuración de Squid: Como configurar el administrador de cache.
  - Apéndice: Listas y reglas de control de acceso para Squid.
  - Cómo configurar squid con soporte para direcciones MAC.
  - Cómo instalar y configurar la herramienta de reportes Sarg.
  - Configuración de WPAD.
  - Ejercicio: Servidor DNS dinámico, servidor DHCP, Servidor Intermediario (Proxy) y Shorewall..
    1. Servidor DNS Dinámico y Servidor DHCP.
    2. Ejercicio: Servidor Intermediario (Proxy) y cortafuegos con Shorewall.
    3. Cómo instalar y configurar la herramienta de reportes Sarg.

3.1 CONFIGURACIÓN

Proxy caché
Los Servidores Proxy para contenido de Red (Web Proxies) pueden actuar como: Proxy caché

Filtro del contenido servido
Aplicando políticas de censura de acuerdo a criterios arbitrarios.

Entre otras cosas, **Squid** puede funcionar como **Servidor Intermediario** y **caché de contenido de Red** para los protocolos **HTTP**, **FTP**, **GOPHER** y **WAIS**, **Proxy de SSL**, **caché transparente**, **WWCP**, **aceleración HTTP**, **caché de consultas DNS** y otras muchas más como **filtro de contenido** y **control de acceso por IP y por usuario**.

3.1.1 UBUNTU: CONFIGURACIÓN BÁSICA DE SQUID

Instalación:
Comenzamos instalando la herramienta con el siguiente comando
aptitude install squid3

Configuración:
Una vez instalado realizaremos los siguientes cambios en el archivo de configuración de Squid (***squid.conf***) con el comando
nano /etc/squid3/*squid.conf*
En el que buscaremos las siguientes líneas y las configuraremos de forma que más se adapten a nuestras especificaciones:

http_port 3128 (puerto por defecto de escucha del Squid)
http_port 3128 transparent (intercept) (proxy transparente)
cache_mem 256MB (tamaño de la memoria caché, se recomienda que no sobrepase la mitad del tamaño de la memoria RAM)
cache_dir ufs /var/spool/squid3 100 16 256 (siendo
ufs= formato de defecto de squid en el almacenamiento caché
100 tamaño en MB del disco del proxy que ocupara la caché
16 256 estos dos valores indican los directorios primarios y sub directorios que ocupara la cache no es recomendable variarlos)
error_directory /usr/share/squid3/errors/es (esta línea nos permite cambiar el idioma de la página que mostrara nuestro navegador cuando entremos en una página que no está permitida, si se cambia el en por es nos saldrá en español, así como si lo cambiamos por alguno de los otros idiomas que hay en el directorio; utilizado para personalizar los mensajes de error)

Además necesitaremos:

Al menos una Lista de Control de Acceso
Al menos una Regla de Control de Acceso

Configuración de listas de control de acceso (ACL) y reglas de control de acceso:

Lista de Control de Acceso (ACL):

acl [*NombreDeLaLista*] src [*ComponentesDeLaLista*]

o

acl [*NombreDeLaLista*] src [*FicheroQueContieneLaLista*]

Tipos de listas:

src	**srcdom_regex**	**req_mime,**
dts	**dtsdom_regex**	**macaddress**
srcdomain	**time, url_regex**	**password**
dtsdomain	**urlpath_regex**	

Regla de control de acceso:

http_access [deny o allow] [*ListaDeControlDeAcceso*]

Ejemplo:

(Colocamos las nuevas acl al final de las existentes)

(Colocamos las nuevas reglas de control de acceso al principio de las existentes)

…

acl *mired* src *192.168.0.0/24*

acl *nopermitido* url_regex *"/etc/squid3/nopermitido"*

…

http_access deny *nopermitido*

http_access allow *mired*

…

Otros parámetros de interés:

access_log

Especifica en que directorio se realizara el registro de accesos al squid, este parámetro es importante para definir un análisis de estadísticas con webalizer.

access_log */var/log/squid/access.log*

cache_log

Define donde se almacenaran los mensajes del comportamiento de la cache de squid.

cache_log */var/log/squid/cache.log*

Para crear el directorio cache, en modo consola ejecutamos;

sudo /usr/local/squid/sbin/squid -z

3.1.2 EJEMPLOS DE ACL

Reglas ACL Squid (Proxy Squid) Ejemplos ACL

3.2 EJEMPLO DE IMPLEMENTACIÓN

Configuración Inicial de la máquina US12
Instalación de SQUID
Configuración inicial de SQUID
Configuración del espacio y cuota dedicado a la caché.
Configuración del espacio y cuota dedicado al log.
Redirigimos el tráfico del puerto 80 a través del proxy.

OPCIÓN 1: PROXY NO TRANSPARENTE

En el navegador configuramos la navegación a través del proxy poniendo la ip de de la máquina donde hemos instalado squid.

OPCIÓN 2: PROXY TRANSPARENTE

En el cortafuegos redireccionamos del tráfico desde el cortafuegos al servidor proxy.

Debemos modificar el cortafuegos para que redirija el tráfico que queremos que pase por el proxy; esta configuración dependerá de la ubicación de nuestro proxy:

- Proxy ubicado en nuestro router – puerta de enlace
- Proxy ubicado en nuestrta DMZ
- Proxy ubicado en Internet
- Proxy inverso

Configuración de squid como proxy transparente

Algunas formas de hacerlo:

```
# Todo lo que venga por el exterior y vaya al puerto 80 lo redirigimos  a nuestro proxy en una maquina interna, hay varias forma de hacerlo
iptables -t nat -A PREROUTING -s 192.168.0.0/24 ! –d 192.168.0.12 -i int -p tcp --dport 80 -j DNAT --to 192.168.0.12:3128
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to 192.168.0.12:3128
iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-ports 3128
iptables -t nat -A PREROUTING -p tcp -d 123.123.123.123 --dport 80 -j DNAT --to-destination 10.10.10.10:3128
iptables -t nat -A PREROUTING -s 0/0 -d 0/0 -p tcp –dport 80 -j REDIRECT –to-port 3128
iptables -t nat -A PREROUTING -i $IFINTERNA -p tcp --dport 80 -j REDIRECT --to-port 3128
iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 3128
```

Permitir y Denegar tráfico en SQUID: desarrollo de los filtros del proxy
Configuración de las páginas de aviso de tráfico denegado
Monitorización de la caché
Monitorización del log

3.3 MANTENIMIENTO Y MONITORIZACIÓN

Iniciar, reiniciar y añadir el servicio al arranque del sistema.

Para iniciar por primera vez Squid en modo consola:

sudo service squid start

Para reiniciar en modo consola ejecutar:

sudo service squid restart

Para que Squid se inicie de manera automática al inicio el sistema, en modo consola ejecutar:

sudo chkconfig squid on

Cualquier error al inicio de Squid solo significa que hubo errores de sintaxis, errores de teclado o de las rutas hacia los archivos de las Listas de Control de Acceso.

Para realizar el diagnóstico de problemas indicándole a Squid que vuelva a leer configuración, lo cual devuelve los errores que existen en el archivo /etc/squid/squid.conf, en modo consola ejecutar:

sudo service squid reload

sudo /etc/init.d/squid reload

En caso de errores graves que no permiten iniciar el servicio, examinar el contenido del archivo ***/var/log/squid/squid.out*** ejecutando en consola:

less /var/log/squid/squid.out

Ahora solo nos queda crear las ACL o Listas de control de acceso en las que configuraremos las páginas que permitiremos y las que no.

Ejercicio: Test para saber si utilizamos proxy. ¿Cómo podemos saber si navegamos a través de un proxy?

3.3.1 LOG SQUID

Monitorización de peticiones y generación de informes de acceso

Rotación de logs del proxy cache Squid

Ejercicio: Estudiar el log generado por tu servidor proxy. Controlar su tamaño.

3.3.2 CACHÉ SQUID

Ejercicio: Estudiar el espacio ocupado por la caché de tu servidor proxy. Controlar su tamaño.

3.4 ALTERNATIVAS

3.4.1 WINGATE

WinGate

3.4.2 SQUID + DANSGUARDIAN

DansGuardian (wikipedia) DansGuardian

DansGuardian es un software de filtro de contenido, diseñado para controlar el acceso a sitios web. Incluye un filtro de virus, importante en sistemas Windows, es usado principalmente en instituciones de educación, gobierno y empresas. Se caracteriza por su alto grado de flexibilidad y adaptación de la implementación.

3.4.3 SQUID SOBRE ZENTYAL

Zentyal (wikipedia) Zentyal

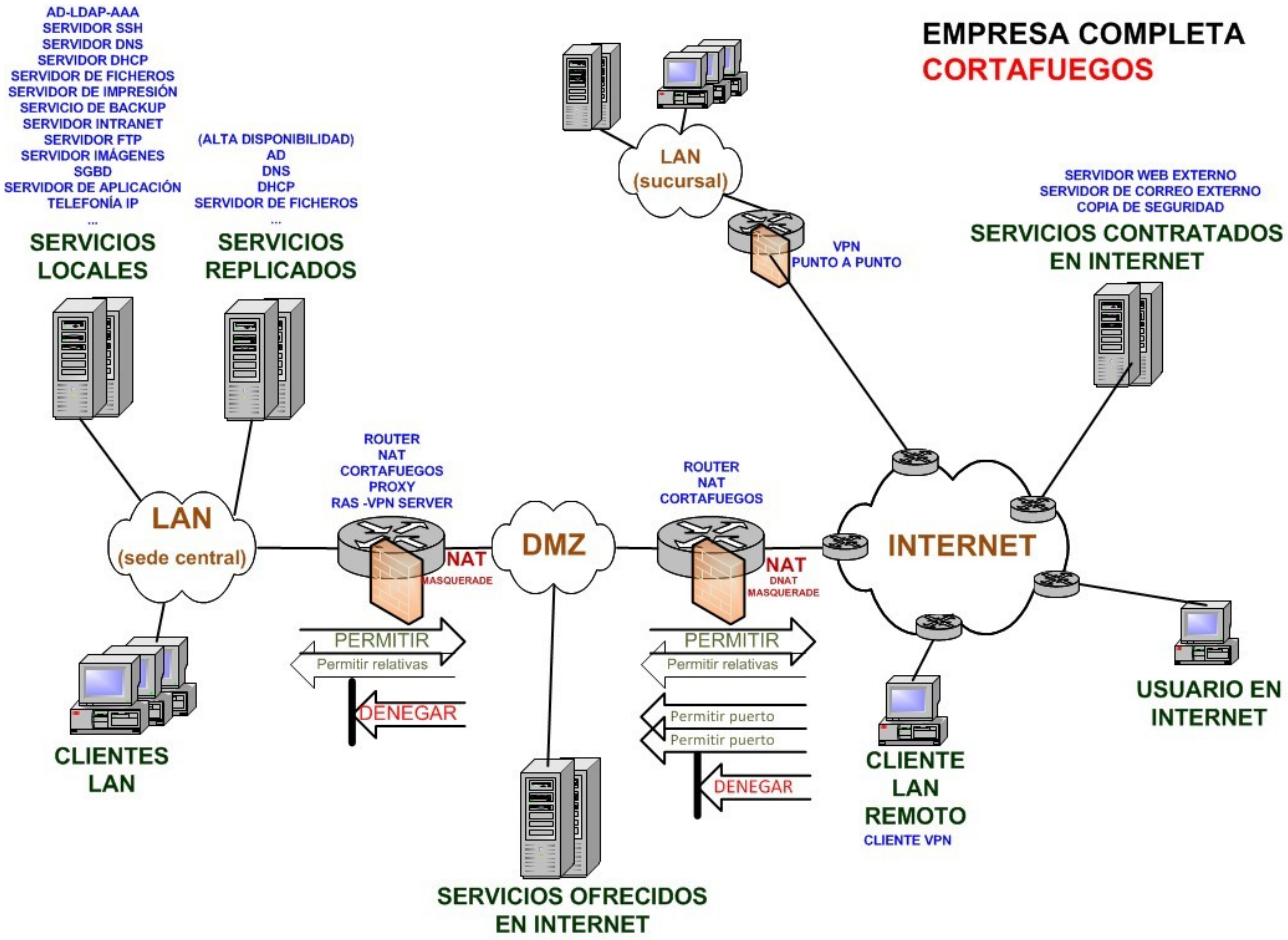
3.4.4 SOCKS

Socks

4. EJERCICIO EMPRESA

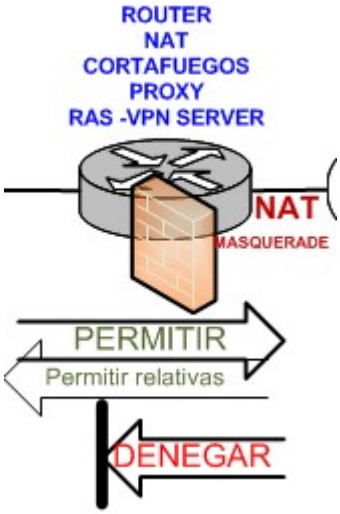
Propuesta del alumno para la empresa que ha estado implementando a lo largo del curso.

E.1 CORTAFUEGOS



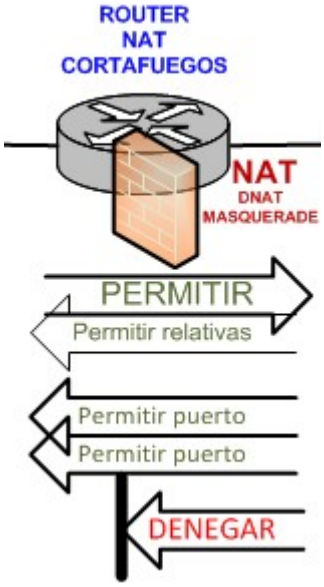
Definir el conjunto de reglas que soporten la política adecuada para nuestro modelo de empresa que permita el correcto funcionamiento de los servicios implementados.

E.1.1 CORTAFUEGOS QUE DEFIENDE LA LAN



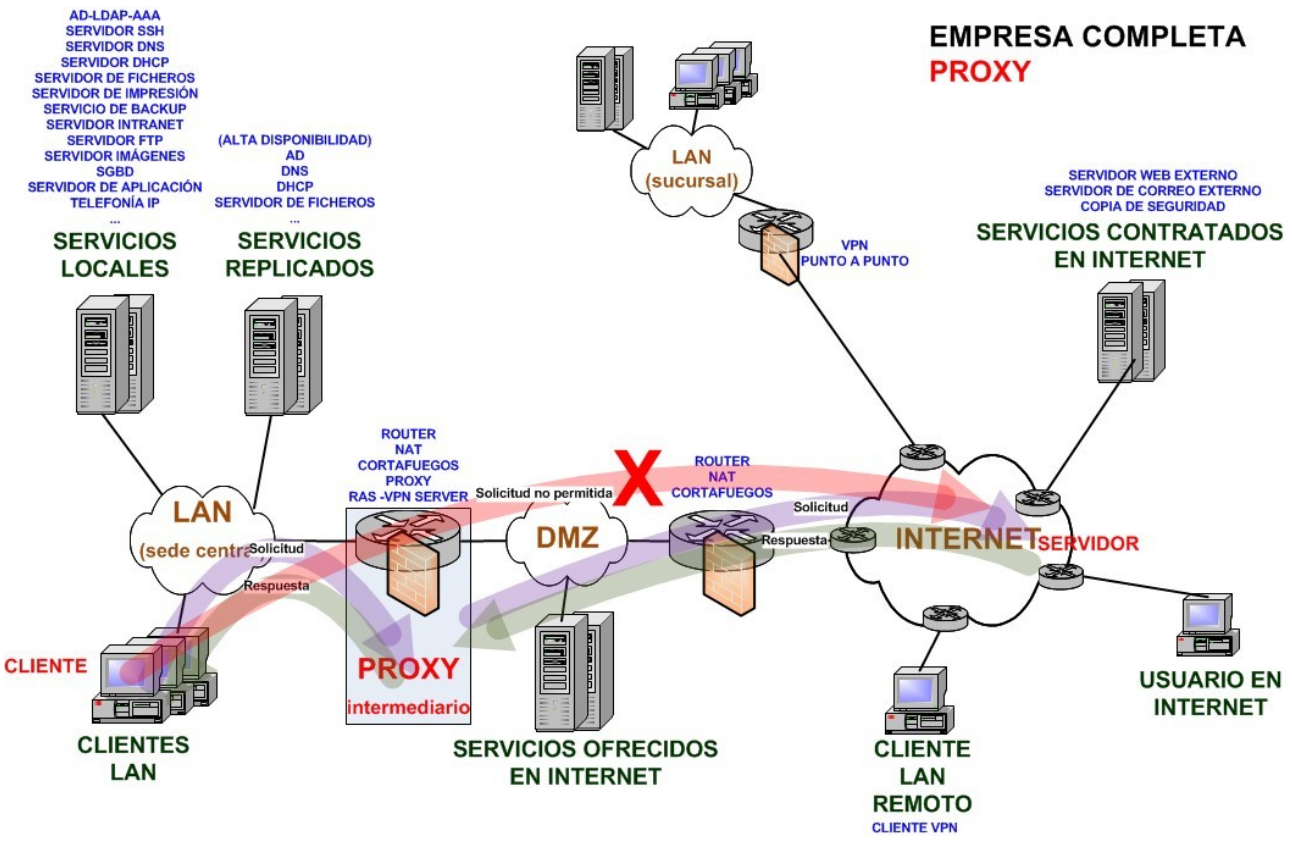
Permitimos el tráfico hacia Internet desde la LAN y la entrada de las respuestas a las solicitudes realizadas desde la LAN.
No permitimos el tráfico desde Internet hacia la LAN

E.1.2 CORTAFUEGOS QUE DEFIENDE LA DMZ



Permitimos el tráfico hacia Internet desde la LAN y la entrada de las respuestas a las solicitudes realizadas desde la LAN.
Permitimos el tráfico hacia determinados puertos de nuestra DMZ desde Internet.
No permitimos resto de tráfico desde Internet hacia la DMZ.

E.2 PROXY



Modelo – propuesta de configuración del proxy de la empresa.

E.2.1 IMPLEMENTACIÓN

Construir un proxy web – caché – transparente para todos los empleados de la empresa.
Implementación de la política de seguridad de la empresa en lo que pueda afectar al proxy.

E.2.2 MANTENIMIENTO Y LOG

Gestión y monitorización del log y la caché del proxy.

E.3 EXPLORAR VULNERABILIDADES

Examinar las diferencias entre las distintas configuraciones de cortafuegos con un explorador de vulnerabilidades.

Limitar a lo necesario el numero de servicios.

E.3.1 AUDITORÍA DE SEGURIDAD

Realizar una auditoría de seguridad interna y externa de nuestra empresa en su configuración definitiva.

ENLACES INTERESANTES - BIBLIOGRAFÍA

Cortafuegos **Firewall** **DMZ** **Proxy**

Netfilter / iptables **Squid** **WinGate**

Application level firewall

WAF **WAF (PCWorld)**

ModSecurity **ModSecurity (wikipedia)**

Encaminamieto de cebolla (Onion routing) **Tor**

"Fundamentos de seguridad de redes" – Eric Maiwald – Editorial Mc Graw Hill – ISBN 970-10-4624-2

"El Tao de la monitorización de seguridad en redes – Richard Bejtlich – Editorial Pearson Educación – ISBN 84-205-4600-3

"Seguridad informática – Jose Fabián Roa Buendía – Editorial Mc Graw Hill – ISBN 978-84-481-8396-7