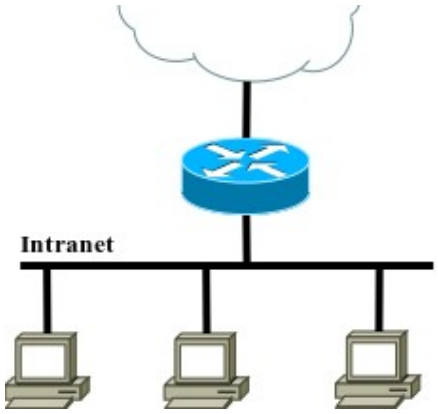
7. Diseños de red

A la hora de introducir un firewall para proteger una red pueden aparecer diferentes situaciones:

a) Router con filtros:

Se trata de la arquitectura más sencilla y consiste en un router que conecta la intranet de la organización con Internet y que se encarga del análisis y filtrado del tráfico. Es muy común en entornos SOHO (Small Office – Home Office) donde no se ofertan servicios en Internet.

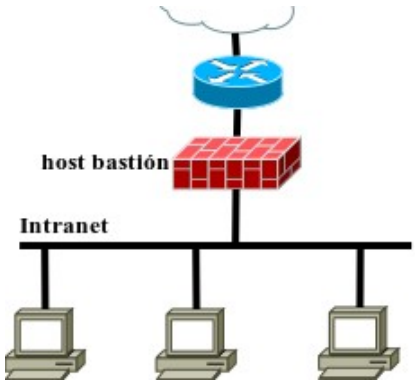


Esta arquitectura tiene como desventajas que:

- Las posibilidades de definir filtros en los routers son limitadas y el rendimiento se resiente si el router se carga con una tarefa de filtrado compleja.
- En el caso de verse comprometido el router, el intruso tiene acceso directo a toda la intranet de la organización.

b) Host bastión:

En esta arquitectura el router se dedica a su función original (enrutamiento) y el control del tráfico recae sobre una máquina especial llamada host bastión. El host bastión implementa filtros más complejos, descargando de esta tarea al router.

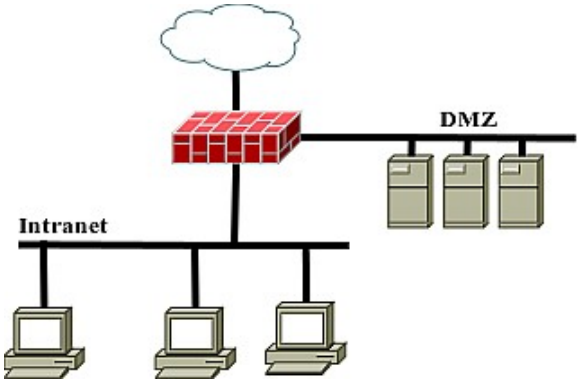


El host bastión debe ser una máquina 'fortificada'; es decir, configurada siguiendo las buenas prácticas de securización del fabricante correspondiente. Si el host bastión se viese comprometido la seguridad de la organización estaría comprometida (p.e. si un hacker logra instalar un sniffer en él se podría capturar todo el tráfico de la red interna al estar directamente conectado a ella).

Es una arquitectura muy presente en entornos SOHO (Small Office – Home Office) donde no se ofertan servicios en Internet o como mucho alguno básico (p.e. acceder a cámaras IP de vigilancia o algún servidor interno).

c) DMZ: Zona Desmilitarizada o zona neutra de seguridad

Una **zona DMZ** es una subred separada de la organización donde se sitúan los servidores que van a ser accesibles desde el exterior. El host bastión está configurado en base a una serie de reglas que permiten el acceso a determinados servicios que se ejecutan en la red DMZ (los que deseamos que sean accesibles desde el exterior) a la vez que protegen a los equipos de la Intranet.



Al ser accesibles los equipos de la DMZ desde Internet, éstos van a estar expuestos a ataques. En caso de que alguno de los servidores se viese comprometido, la DMZ estaría expuesta pero los equipos de la Intranet seguirían protegidos (al estar situados en una red diferente separada por el host bastión).

Dos puntos importantes en relación al tráfico con origen equipos de la DMZ:

- **Evitar los 'zone pinholes'**: los servidores situados en la DMZ no deberían poder iniciar comunicaciones contra equipos de la intranet. Permitir esas conexiones entre la DMZ y la intranet (zone pinhole) es peligroso; ya que, si el servidor autorizado fuera comprometido, el atacante podría aprovechar el canal de comunicación DMZ→ intranet autorizado para acceder a los equipos de la intranet (tradicionalmente menos protegidos y monitorizados que los servidores públicos).
- **Sneaky Rule**: hay que denegar el tráfico de salida de los servidores de la DMZ (conexiones iniciadas por los servidores). La misión de los servidores es recibir peticiones de los clientes y responder con el resultado; es decir, la conexión es iniciada por un cliente y autorizada por el firewall, por lo que las respuestas de los servidores estarán autorizadas para salir (stateful firewall). Lo que se está impidiendo con esta regla es que los servidores inicien conexiones hacia fuera, que normalmente no son necesarias para desempeñar a sus funciones. En el caso de ser necesario, deberían permitirse únicamente las conexiones creando las reglas correspondientes de la forma más restrictiva posible.

Ejemplo de Sneaky Rule en un firewall de red que registra y bloquea las conexiones iniciadas por los servidores situados en la red DMZ 172.16.0.0/16 en la interfaz eth2:

```
$ sudo iptables -N SNEAKY
$ sudo iptables -A INPUT -i eth2 -s 172.16.0.0/16 -j SNEAKY
$ sudo iptables -A FORWARD -i eth2 -s 172.16.0.0/16 -j SNEAKY
$ sudo iptables -A SNEAKY -j LOG --log-prefix "iptables: Sneaky Rule"
$ sudo iptables -A SNEAKY -j DROP
```

d) Diseño Multi-DMZ:

Con frecuencia los servidores de la DMZ externa tienen que comunicarse con servidores de las redes internas. Si un servidor de la DMZ es comprometido, cuenta con una posición ventajosa para atacar a los servidores de las redes internas con los que se comunica. Por eso es una buena idea separarlos de otros equipos críticos para la organización, configurando una red dedicada.

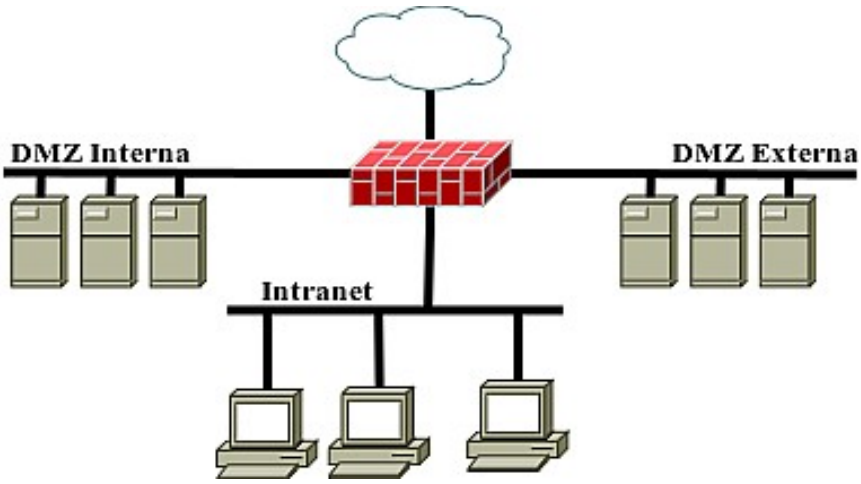


Fig. Arquitectura con varias DMZ

Es recomendable usar varios cortafuegos para separar las distintas zonas de seguridad. Si en una organización existen diferentes niveles de seguridad; por ejemplo, un departamento con sistemas con información crítica, se deben crear zonas de seguridad controladas por cortafuegos.

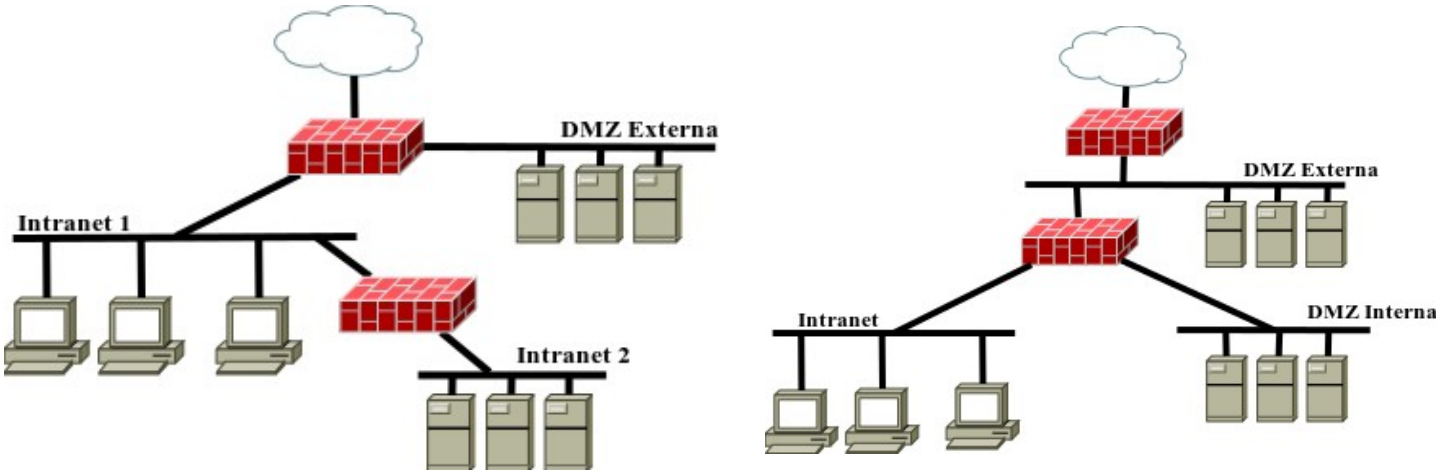


Fig. Ejemplos de diseños con zonas con diferentes niveles de seguridad.

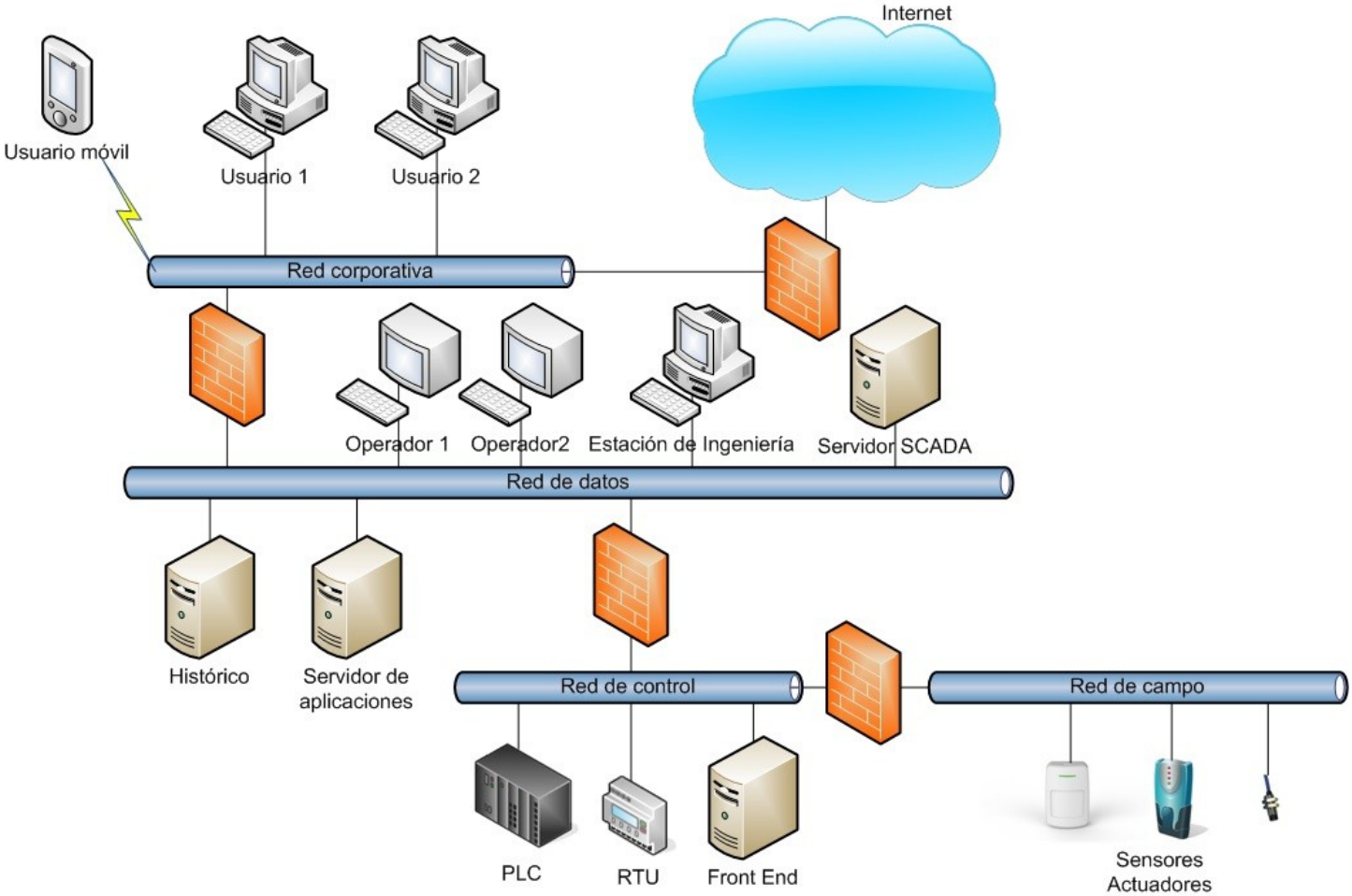


Fig. Ejemplo de diseño con zonas con diferentes niveles de seguridad en un entorno industrial. Fuente: https://www.incibe-cert.es/blog/divide-venceras-segmentacion-rescate

e) Diseño DMZ con VLANs:

Usando un Firewall con soporte de VLANs se pueden aislar los servidores de la DMZ, creando 'múltiples' DMZs con reglas de filtrado específicas para cada VLAN. Debido al funcionamiento de las VLANs, las comunicaciones directas entre equipos situados en VLANs diferentes no son

posibles. Por tanto, ubicar a los servidores en VLANs diferentes, los mantendrá aislados los unos de los otros, añadiendo una capa más a la infraestructura de seguridad perimetral.

A modo de ejemplo, tenemos 3 servidores y cada uno de ellos colocado en una VLAN diferente. Si fuera asaltado con éxito un servidor de la VLAN 3, el atacante quedaría confinado en la VLAN 3 (a la que pertenece el servidor) y no podría acceder a los servidores de las VLAN 1 y VLAN 2 directamente. La única forma de saltar entre VLAN es a través del firewall (si hubiese reglas que lo permitan).

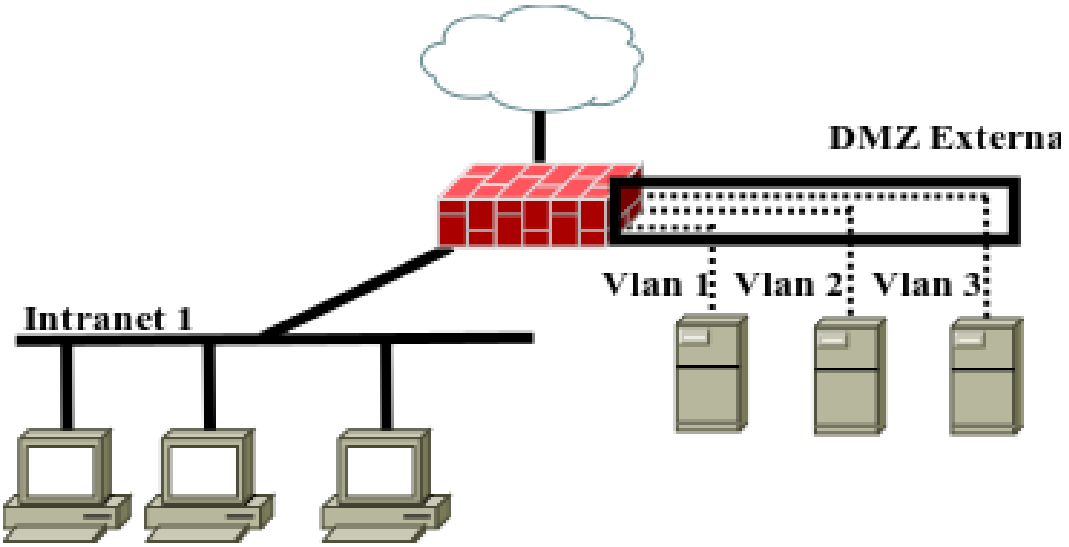


Fig. DMZ con VLANs

Si el firewall soporta IEEE 802.1Q se puede usar un enlace troncal con el consiguiente ahorro de puertos en el switch y en el propio firewall.[1]

8. NAT

Hace unos años cuando una organización deseaba conectar su red de forma permanente a Internet lo normal era que solicitara al NIC una red adecuada a sus necesidades (clase B o C normalmente) y asignara números IP de dicha red a todas sus máquinas. De esta forma todos los ordenadores tenían acceso directo a Internet con todas las funcionalidades. Debido al crecimiento exponencial de Internet, cada vez resulta más difícil obtener números IP del NIC correspondiente; por otro lado, en muchas ocasiones no se necesita, o incluso no se desea, disponer de un acceso directo a Internet con completa funcionalidad, por razones de seguridad fundamentalmente. Estos dos motivos, la seguridad y la dificultad para conseguir direcciones públicas, han impulsado a las organizaciones a hacer un mayor uso de las redes privadas según los rangos especificados en el **RFC 1918** (10.0.0.0, 172.16.0.0 a 172.31.0.0, y 192.168.0.0 a 192.168.255.0). Estas redes privadas no pueden intercambiar datagramas directamente con el exterior, por lo que han de utilizar un equipo intermedio que les permita comunicarse con el exterior. Una solución a este problema es el uso de proxys, que ejercerían de intermediarios a nivel de aplicación entre el equipo del interior y la red externa.

Otra solución ampliamente utilizada hoy en día es la **traducción de direcciones o NAT (Network Address Translation)**. El NAT puede realizarse en un host (p.e. en Linux la función

[1]Para que las VLANs se puedan expandir a través varios switches sin desperdiciar puertos, se creó el concepto de enlace Trunk o troncal. Por un enlace troncal irán mezcladas tramas de diferentes VLANs por lo que es preciso marcarlas o etiquetarlas de alguna manera a fin de poder entregarlas a la VLAN adecuada en el otro extremo. El marcado se hace añadiendo un campo nuevo en la cabecera de la trama donde se indica el ID de la VLAN a la que pertenece. Aunque existen técnicas de etiquetado (tagging) propietarias; a día de hoy, el estándar IEEE 802.1q permite crear redes complejas con VLANs utilizando equipos de diferentes fabricantes.

NAT se denomina 'IP Masquerade') aunque también se implementa en muchos routers. Dado que generalmente la función de NAT se realiza en la frontera entre una red local y el exterior, la opción del router es bastante popular (p.e. los routers ADSL que se instalan en los hogares).

Entre los motivos existentes para usar NAT destacan:

- Traducción de direccionamiento privada a direccionamiento público.
- Permitir que varios servicios alojados en servidores y puertos distintos sean accesibles a través de una sola dirección IP.
- Balanceo de IP entre distintos servidores y alta disponibilidad.
- Ocultación del direccionamiento interno.
- Conexión de múltiples servidores y/o estaciones a una red externa a través de una sola dirección IP.

La traducción se puede hacer de dos maneras en función de lo que se modifica:

- **NAT básico**: se modifica únicamente la dirección IP.
- **NAPT (Network Address Port Translation) o PAT**: se cambia también el número de puerto TCP o UDP.

En función de cómo se establece la traducción se tiene:

- **NAT Estático**: la traducción de la dirección pública y la privada se realiza de acuerdo con una tabla de equivalencia que se carga en la configuración del dispositivo NAT y que no se modifica dinámicamente.
- **NAT dinámico**: la tabla de equivalencia es gestionada dinámicamente por el dispositivo NAT de forma que las direcciones y/o números de puerto se puedan reutilizar.

Combinando el NAT Básico o el NAPT con las modalidades estática y dinámica obtenemos cuatro combinaciones de NAT:

- **NAT básico estático**: asociación manual de IP privada – IP pública.
- **NAT básico dinámico o Pool NAT**, asociación automática de IP privada – IP pública bajo demanda.
- **NAPT estático**, también conocido como redirección de puertos o port forwarding y que consiste en asociar de forma manual una IP Pública – puerto con una IP Privada – puerto.
- **NAPT dinámico**, también conocido como **NAT de ocultamiento** o **Hide NAT** y que que consiste en asociar de forma automática y bajo demanda una IP Privada – puerto con una IP Pública – puerto.

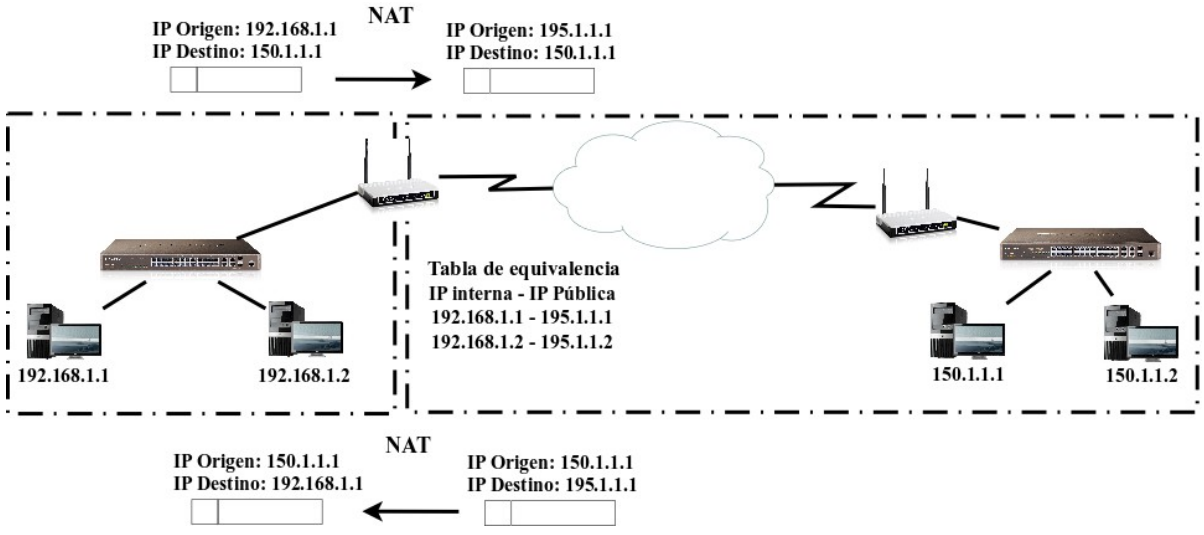


Fig. Ejemplo de NAT Básico Estático

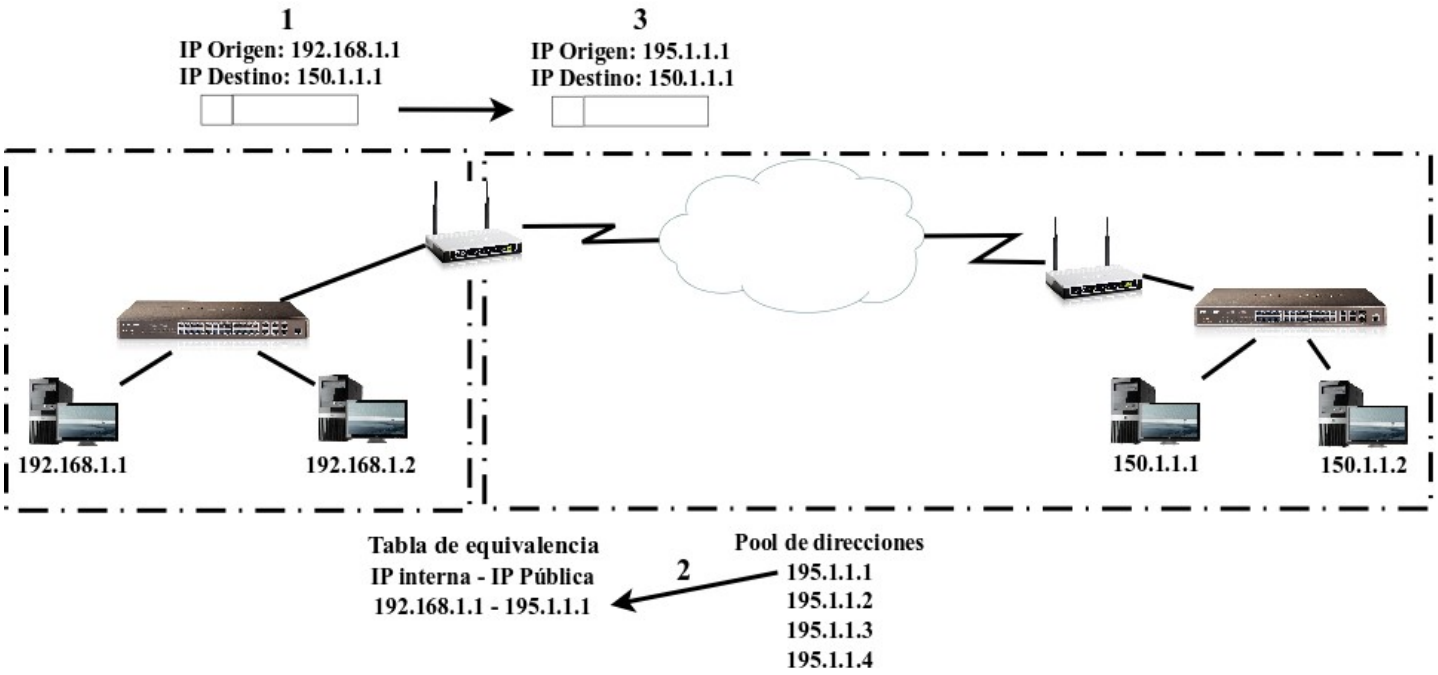


Fig. Ejemplo de NAT Básico Dinámico

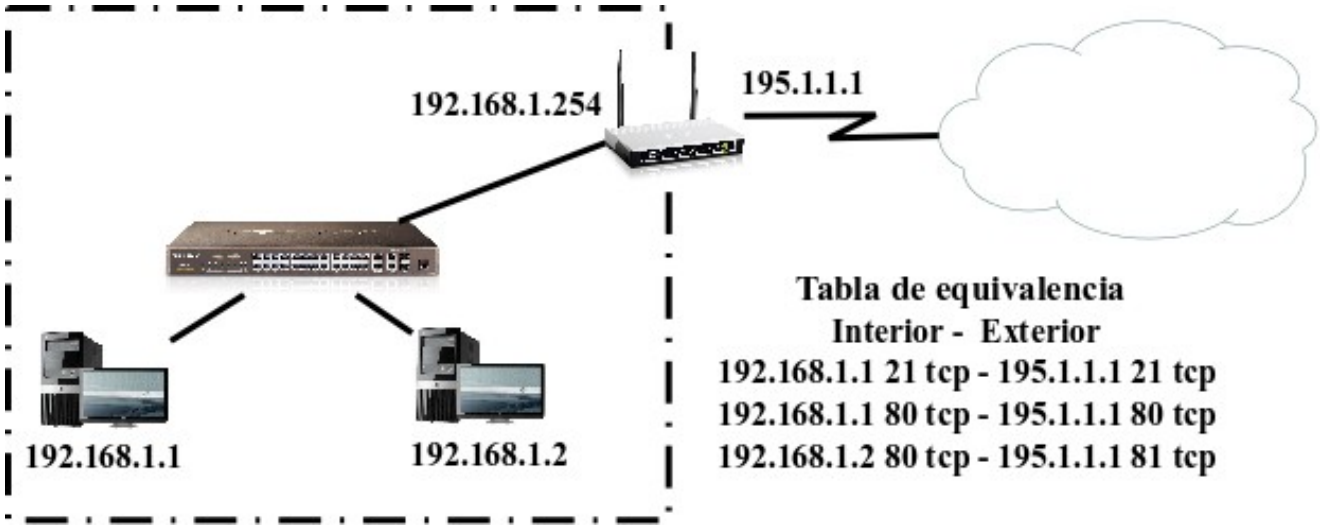


Fig. Ejemplo de NAPT Estático

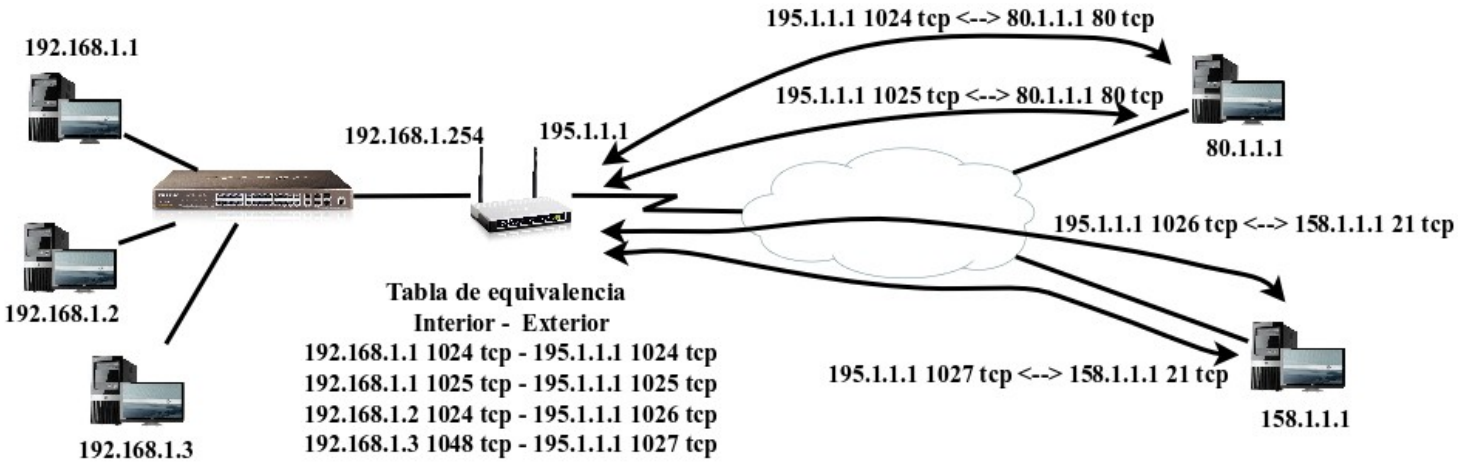


Fig. Ejemplo de NAPT Dinámico

Estas cuatro modalidades de NAT pueden coexistir en una misma red; por ejemplo, utilizando NAPT estático para los servidores que deban ser accesibles desde el exterior y NAPT dinámico para que el resto de los hosts puedan salir al exterior.

9. TRADUCCIÓN DE DIRECCIONES en netfilter/iptables

Netfilter permite hacer NAT y NAPT, siendo la tabla nat la encargada de realizarlo a través de las acciones SNAT e DNAT.

SNAT (Source Network Address Translation)

SNAT consiste en la traducción de las direcciones IP origen de los paquetes que atraviesan el cortafuegos en una o varias direcciones IP públicas. Los sistemas origen consiguen de este modo acceder a internet sin tener una dirección IP pública propia y además consiguen la protección extra de no ser accesibles desde internet.

Masquerade o *Enmascaramiento* es exactamente lo mismo que SNAT, pero usando siempre como dirección pública la dirección pública del firewall.

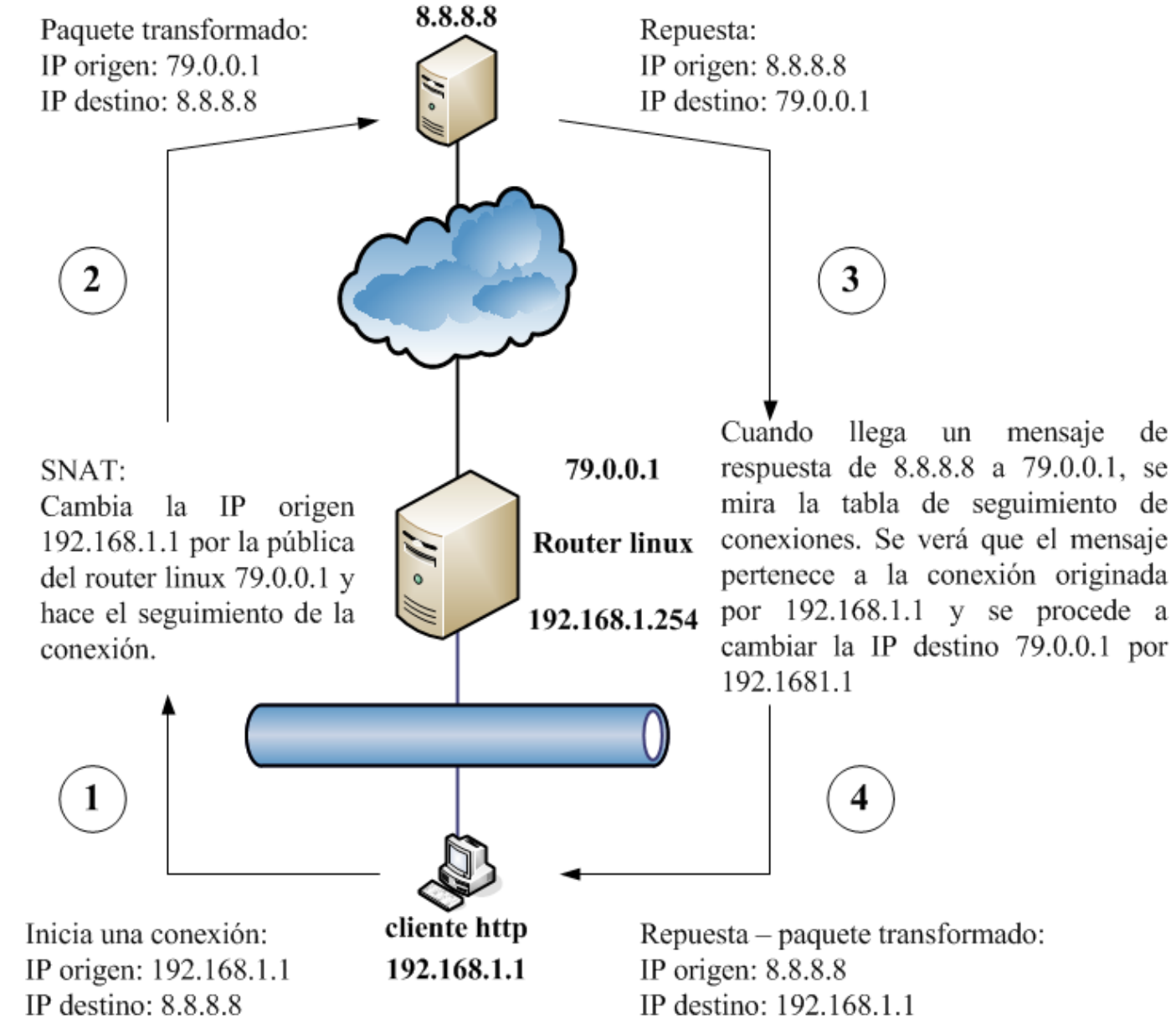


Fig. Ejemplo de SNAT

Para comprender la figura anterior, donde se muestra el funcionamiento del SNAT, hay que tener presente que las direcciones privadas no son válidas para viajar por Internet; por lo que,

cualquier paquete con una IP privada no llegará al destinatario al ser descartado en los routers de Internet.

1. El cliente http tiene la dirección privada 192.168.1.1 y quiere conectarse con la máquina 8.8.8.8, por lo que envía un paquete con IP origen 192.168.1.1 e IP destino 8.8.8.8. Como el equipo destino no está en la misma red encamina el paquete hacia el router.
2. El router recibe el paquete y cambia a IP origen 192.168.1.1 (privada) por la IP 79.0.0.1 (pública), guardando la correspondencia entre IP privada-IP pública en una tabla (esta información será usada para hacer la transformación inversa cuando llegue el mensaje de respuesta). Ahora que el router tiene un paquete con IPs públicas (válidas en Internet) tanto en el campo origen como destino, procede a encaminarlo hacia al destino.
3. El destinatario recibe la petición del cliente y procede a generar un mensaje de respuesta. La IP origen es la del equipo (8.8.8.8) y la IP destino es la que aparecía como origen en el mensaje recibido (la IP pública 79.0.0.1). Una vez creado el mensaje de respuesta procede a enviar el paquete.
4. El paquete es encaminado y llega al router-linux; que tras consultar su tabla de transformaciones, procede a cambiar la IP destino 79.0.0.1 por 192.168.1.1 (la IP del cliente que inició todo el proceso).

Usando SNAT o Masquerade, el equipo 192.168.1.1 puede iniciar una conexión hacia 8.8.8.8. Sin embargo, el equipo 8.8.8.8 no puede iniciar una conexión hacia 192.168.1.1, al ser una dirección IP privada no alcanzable en Internet. No confundir iniciar conexiones con responder mediante paquetes a conexiones iniciadas por el otro equipo.

En netfilter, SNAT o Masquerade se hace en la cadena POSTROUTING y permite especificar que dirección IP origen debe ponerse (también se puede indicar el/los puerto/s):

La IP origen de los paquetes procedentes de equipos de la red 192.168.1.0/24 será reemplazada por 79.0.0.1

```
$ iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j SNAT --to-source 79.0.0.1
```

La IP origen de los paquetes procedentes de equipos de la red 192.168.1.0/24 será reemplazada por la IP pública del router

```
$ iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j MASQUERADE
```

La IP origen de los paquetes procedentes de equipos de la red 192.168.1.0/24 será reemplazada por 79.0.0.1 y el puerto origen tcp/udp se cambiará por uno del rango 100-1100

```
$ iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j SNAT --to-source 79.0.0.1:100-1100
```

La IP origen de los paquetes procedentes de equipos de la red 192.168.1.0/24 será reemplazada por una IP del rango 79.0.0.1-79.0.0.6

```
$ iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j SNAT --to-source 79.0.0.1-79.0.0.6
```

La IP origen de los paquetes procedentes de equipos de la red 192.168.1.0/24 será reemplazada por una IP de los rangos 79.0.0.1-79.0.0.6 y 80.0.0.1

```
$ iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j SNAT --to-source 79.0.0.1-79.0.0.6 --to-source 80.0.0.1
```

Con `SAME` o con `SNAT --persistent` además de SNAT se usa siempre la misma IP pública con la misma IP privada. Es decir, si un sistema abre una conexión que es traducida con una dirección externa, en las siguientes conexiones que abra, el firewall intentará asignarle la misma dirección IP externa:

```
$ iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j SAME --to 79.0.0.1-79.0.0.6
```

```
$ iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j SNAT --to-source 79.0.0.1-79.0.0.6 --persistent
```

DNAT (Destination Network Address Translation)

DNAT consiste en traducir una dirección IP pública en una dirección IP privada, siendo el proceso contrario a SNAT.

Se usa habitualmente cuando se tienen varios servidores detrás del firewall, de modo que una única dirección pública se traduce en varias direcciones privadas en base a diferentes puertos y protocolos (también conocido como Port Forwarding o redirección de puertos).

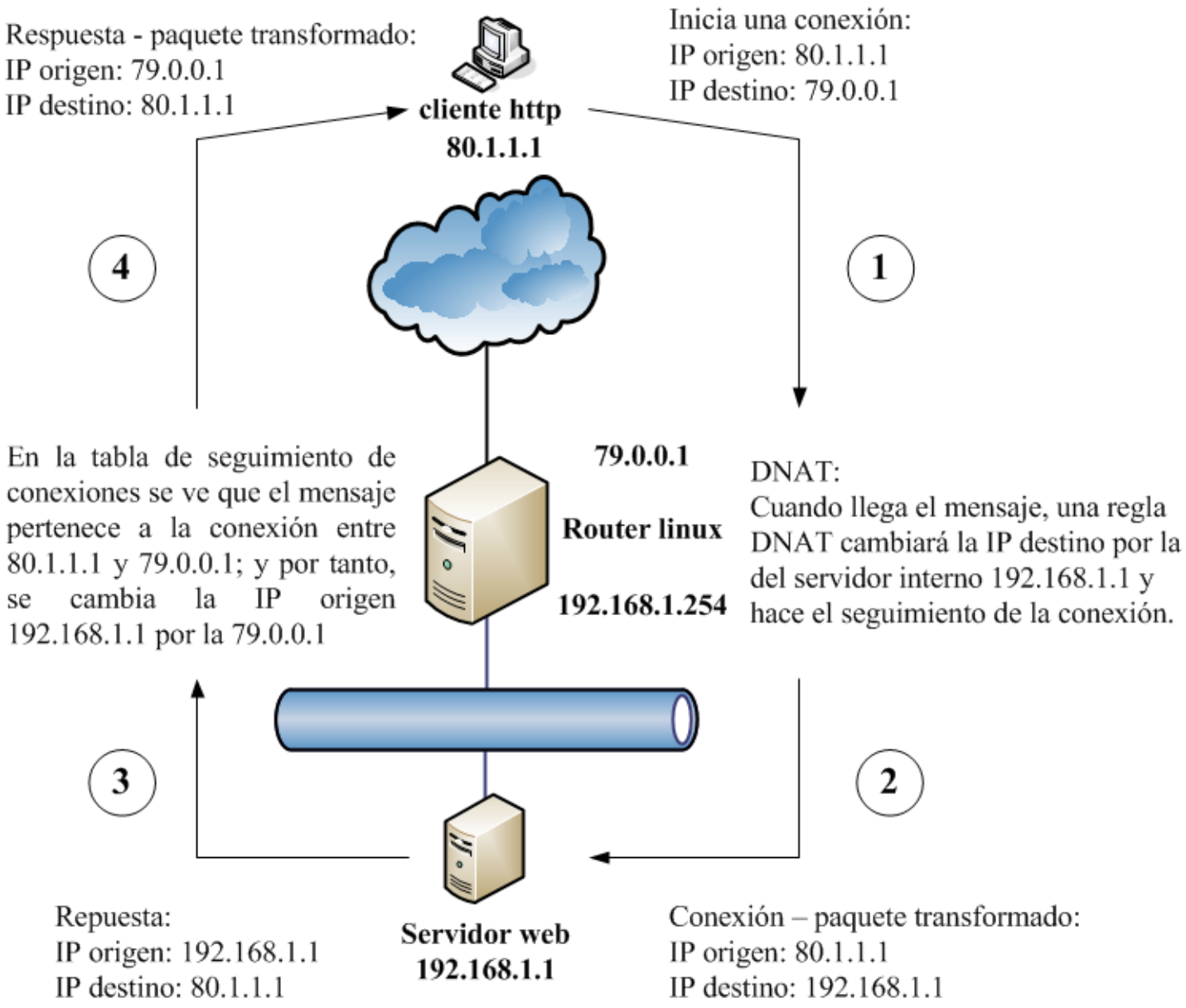


Fig. Ejemplo de DNAT

Si DNAT está configurado, pero SNAT no, el equipo 80.1.1.1 podrá establecer conexiones con 192.168.1.1 usando 79.0.0.1 como dirección IP destino; sin embargo, el equipo 192.168.1.1 no podrá iniciar conexiones a 80.1.1.1.

En netfilter, DNAT se hace en las cadenas PREROUTING y OUTPUT:

La IP destino de los paquetes destinados a 79.0.0.1 será reemplazada por 192.168.1.1

```
$ iptables -t nat -A PREROUTING -d 79.0.0.1 -j DNAT --to-destination 192.168.1.1
```

La IP destino de los paquetes tcp destinados a 79.0.0.1 puerto 80 será reemplazada por 192.168.1.1

```
$ iptables -t nat -A PREROUTING -d 79.0.0.1 -p tcp --dport 80 -j DNAT --to-destination 192.168.1.1
```

La IP destino de los paquetes tcp destinados a 79.0.0.1 y a cualquiera de los puertos 22, 80 o 443 será reemplazada por 192.168.1.1

```
$ iptables -t nat -A PREROUTING -d 79.0.0.1 -p tcp -m multiport --dports 22,80,443 -j DNAT --to-destination 192.168.1.1
```

Los paquetes tcp destinados a 79.0.0.1 puerto 22000 serán transformados para tener IP destino 192.168.1.1 y puerto destino tcp/22

```
$ iptables -t nat -A PREROUTING -d 79.0.0.1 -p tcp --dport 22000 -j DNAT --to-destination 192.168.1.1:22
```

Los paquetes tcp con origen el equipo 80.0.0.1 y destino 79.0.0.1 puerto 22000 serán transformados para tener IP destino 192.168.1.1 y puerto destino tcp/22

```
$ iptables -t nat -A PREROUTING -s 80.0.0.1 -d 79.0.0.1 -p tcp --dport 22000 -j DNAT --to-destination 192.168.1.1:22
```

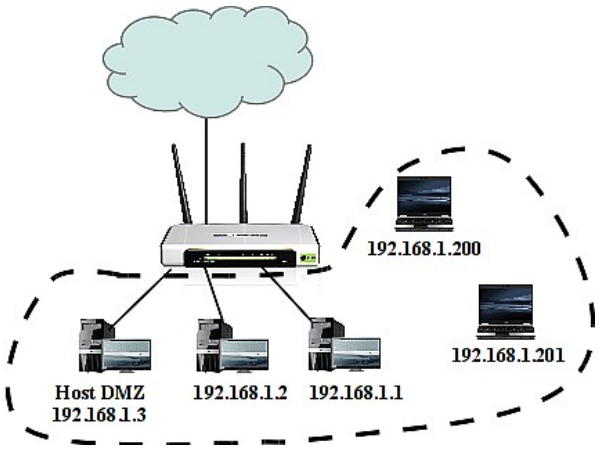
Cuando se hace SNAT de una dirección privada en una pública y DNAT de la misma dirección pública en la misma dirección privada, se produce un mapeo de una dirección privada en una pública en ambas direcciones conocido como traducción total (**Full NAT**) y en iptables se implementa con `NETMAP`:

```
$ iptables -t nat -A PREROUTING -s 192.168.1.0/24 -j NETMAP --to 190.1.1.0/24
```

En muchos routers SOHO se ofrece una funcionalidad denominada DMZ. Esta denominación comercial no se corresponde con el concepto de red DMZ. El configurar un equipo como Host DMZ lo que hace es redirigir todo el tráfico entrante al router desde Internet (que no sea respuesta a peticiones emitidas por otros equipos de la LAN) hacia el 'host dmz'; por tanto, el host dmz es accesible completamente desde Internet. Estos routers usan tanto de DNAT como SNAT para dar servicio a los equipos LAN y exponer al host dmz.

El host dmz está expuesto a Internet, pero no deja de ser un equipo de la LAN. Esto significa que si el host dmz se viese comprometido, el atacante estaría situado en nuestra red local con acceso directo al resto de los equipos.

En contraposición, una red DMZ es una red separada de la LAN; por lo que si uno de los equipos situados en la DMZ se viese comprometido, los equipos de la LAN no serían directamente accesibles al estar separados por el firewall. Es más, esas comunicaciones nunca deberían permitirse (Sneaky Rule).



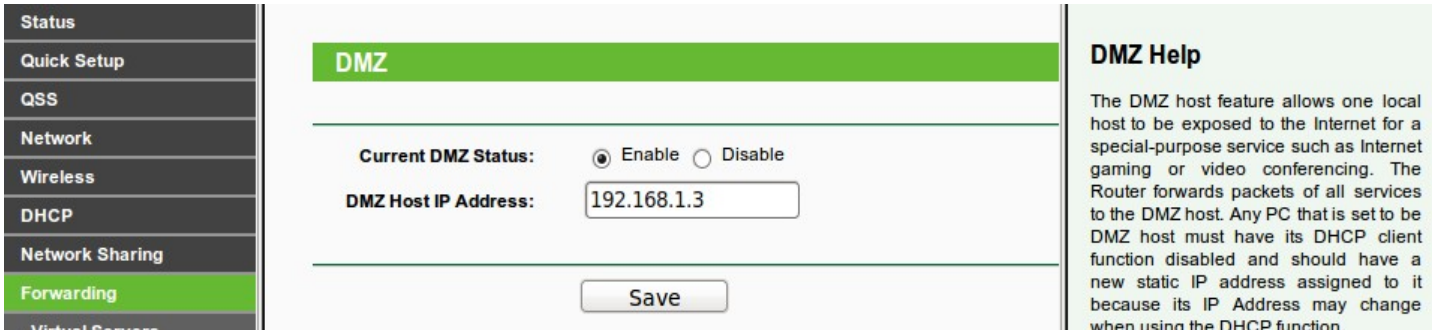
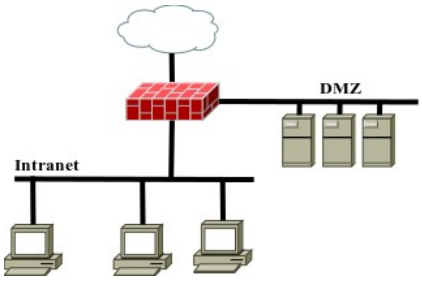


Fig. Ejemplo host DMZ en un router TP-Link

REGLAS 'No NAT'

En ocasiones un firewall que está haciendo NAT debe evitar hacerlo para algunas direcciones origen/destino. Por ejemplo, en una organización con una intranet y una red DMZ con servidores a los que se les permite el acceso tanto desde Internet como desde la intranet, si se analiza el tráfico se puede concluir que no siempre es necesario hacer NAT:



Interfaz	Dirección IP	Zona
eth0	192.0.2.1 /24	WAN (interfaz externa)
eth1	192.168.10.1 /24	LAN (intranet)
eth2	192.168.2.1 /24	DMZ

- El firewall debe hacer SNAT para permitir la salida a Internet de los equipos de la LAN.
- El firewall debe hacer SNAT para permitir la salida a Internet de los equipos de la DMZ.
- Para hacer accesibles los servidores de la DMZ desde Internet hay que hacer DNAT.
- Para la comunicación entre los equipos de la LAN y la DMZ no es necesario hacer NAT.

A continuación se pueden ver las reglas en la tabla nat para que en las comunicaciones entre LAN y DMZ no se haga NAT:

```
$ iptables -t nat -A PREROUTING -s 192.168.10.0/24 -d 192.168.2.0/24 -j ACCEPT
$ iptables -t nat -A POSTROUTING -s 192.168.10.0/24 -d 192.168.2.0/24 -j ACCEPT
```

Estas reglas deben colocarse antes de las reglas SNAT y DNAT.

10. Buenas prácticas al desplegar cortafuegos

A continuación se indican una serie de buenas prácticas a la hora de configurar un cortafuegos:

Sencillez y claridad:

- Aplicar el principio KISS para elegir la solución más sencilla de entre las que cumplan los requisitos necesarios; evitando así, estructuras complejas difíciles de analizar.
- Separar grupos complejos de reglas en cadenas separadas.
- Usar alias o variables para facilitar la comprensión y el mantenimiento de las reglas.

Denegar por defecto (CleanUp Rule):

- Permitir únicamente el tráfico requerido por las necesidades de la red y rechazar el resto.
- Aunque la opción de 'Permitir por defecto' es más cómoda y viene activada por defecto en muchos routers y cortafuegos, no es la más segura.

Denegar la entrada y salida de direcciones marcianas (martians):

- Las direcciones marcianas son todas aquellas que por su asignación no pueden generar tráfico Internet válido (y por tanto es casi seguro que enmascaran tráfico ilícito), incluyendo:
  - Direcciones reservadas por IANA. (0.0.0.0/8, 10.0.0.0/8, 169.254.0.0/16, 172.16.0.0/12, 192.0.2.0/24, 192.88.99.0/24, 198.18.0.0/15, 192.168.0.0/16, 224.0.0.0/4, 240.0.0.0/4).
  - RFC 3232: este documento y establece una base de datos online y en continuo cambio como referencia obligada para la asignación de los rangos de direcciones IP por IANA.

- Denegar la entrada y salida de tráfico originado en direcciones válidas pero que no están en uso (no asignadas o unallocated).

Antispoofing rules:

- Denegar la entrada de tráfico con tus direcciones y la salida de tráfico no originado con tus direcciones; es decir, tráfico cuya IP de origen no se corresponde con el segmento de red del que procede.

Anti-Lockdown y Stealth Rules:

- Prohíbir el acceso no autorizado al cortafuegos:
  - En primer lugar, garantizar el acceso desde equipos autorizados (**Anti-Lockdown Rule**).
  - En segundo lugar, prohibir el acceso al resto (**Lockdown Rule o Stealth Rule**).

Sneaky Rule:

- Denegar el tráfico de salida a los servidores de la DMZ (Sneaky Rule).

Registro y detección:

- Registrar el tráfico denegado (**Log Denied Rule**).
- No registrar el tráfico ruidoso pero necesario como el broadcast (**No Logging Rule**).
- Alertar en caso de intentos de administración desde equipos no autorizados.
- Alertar en caso de tráfico desde una zona desmilitarizada.
- Centralización y análisis automático de registros.

Permitir únicamente el tráfico deseado:

- Permitir únicamente el tráfico deseado aplicando las mayores restricciones posibles.

Si se quiere permitir el hacer ping la forma correcta es la segunda:

```
$ iptables -A OUTPUT -p icmp -m conntrack --ctstate NEW -j ACCEPT
$ iptables -A OUTPUT -p icmp --icmp-type 8 -m conntrack --ctstate NEW -j ACCEPT
```

- Permitir el tráfico de gestión y monitorización necesario.
- Permitir el tráfico de los protocolos de enrutamiento y de red necesarios.

Revisar y verificar el funcionamiento del cortafuegos:

- Revisar los puertos abiertos en el cortafuegos y el tráfico permitido a través de él.
- Usar REJECT durante las pruebas y una vez verificadas cambiar a DROP.
- Usar analizadores de tráfico y exploradores de puertos, guardando los resultados para compararlos con futuras revisiones.

Documentar la configuración:

- En entornos grandes se debe mantener una documentación detallada describiendo la configuración. Debido a lo sensible de esta documentación, debe estar protegida.
- Comentar las reglas que tengan un significado especial o que otros administradores puedan no entender.

```
$ iptables -A INPUT -i eth1 -m comment --comment "my local LAN"
```

División en subredes por niveles de seguridad:

- Bien físicamente o con VLANs separar creando:
  - DMZs externa e interna.
  - Red para dispositivos móviles/wireless.

- Una o varias intranets.

- En redes complejas con múltiples cortafuegos, usar equipos de diferentes fabricantes.

11. Filtrado y reglas

Los firewalls a la hora de filtrar paquetes pueden combinar distintas técnicas:

- **Juego de reglas (ruleset):** también llamado ACL (listas de control de acceso) en muchos dispositivos.
- **Filtrado de camino inverso (Reverse Path Filtering – RPF):** permite verificar todo el tráfico de entrada de una interfaz, comprobando que las direcciones de origen son conocidas mediante la tabla de enrutamiento (Routing Information Base - RIB). Se chequea cada paquete con la tabla de rutas y si la conexión procede de una IP e interfaz donde se sabe que no reside la red, el paquete se descarta directamente (p.e. un paquete procedente de la WAN con IP origen de nuestra organización). Es un filtrado basado en direcciones IP de origen. En `/etc/sysctl.conf` encontramos:

```
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
net.ipv4.conf.default.rp_filter=1
net.ipv4.conf.all.rp_filter=1
```

- **Rutas de descarte (null routes)**: permite enviar el tráfico especificado a una interfaz virtual de descarte modificando la tabla de enrutamiento. Es un filtrado basado en la dirección IP de destino.

```
manuel@x99:~$ sudo ip route add blackhole 192.168.2.100
manuel@x99:~$ ip route
default via 10.221.0.254 dev wlp1s0 proto dhcp metric 600
10.137.247.0/24 dev lxdbr0 proto kernel scope link src 10.137.247.1 linkdown
10.221.0.0/16 dev wlp1s0 proto kernel scope link src 10.221.0.58 metric 600
169.254.0.0/16 dev wlp1s0 scope link metric 1000
blackhole 192.168.2.100
```

Se pueden definir varias tipos de rutas de descarte:

```
manuel@x99:~$ man ip-route
…
unreachable - these destinations are unreachable. Packets are discarded and the
ICMP message host unreachable is generated.

blackhole - these destinations are unreachable. Packets are discarded silently.

prohibit - these destinations are unreachable. Packets are discarded and the ICMP
message communication administratively prohibited is generated.
...
```