

Configuración de firewall y NAT

CC50P

Sebastián Castro A.

Primavera 2006

Firewall

- ◆ Cómo funcionan
 - Filtros de paquetes
 - ACL
- ◆ Para que se usan
 - Seguridad
 - Control de uso de recursos (traffic shapping)
- ◆ Configuración típica

Firewall

◆ Cómo funcionan

- Es una función que puede estar albergada en un switch L3, Router, Host o equipo especializado.
- Mediante el análisis de los paquetes IP que pasan a través de él, se determina si cada paquete debe pasar, devolverse o simplemente descartarse.
- En las versiones preliminares el análisis de los paquetes era uno a uno y no se generaban asociaciones entre paquetes: **stateless**.
- Actualmente se ofrecen diferentes niveles de asociación entre paquetes, esquema llamado **stateful**.
- Existen además esquemas más avanzados que entienden protocolos de capas superiores a la IP: **full inspection**.

Firewall

◆ Cómo funcionan

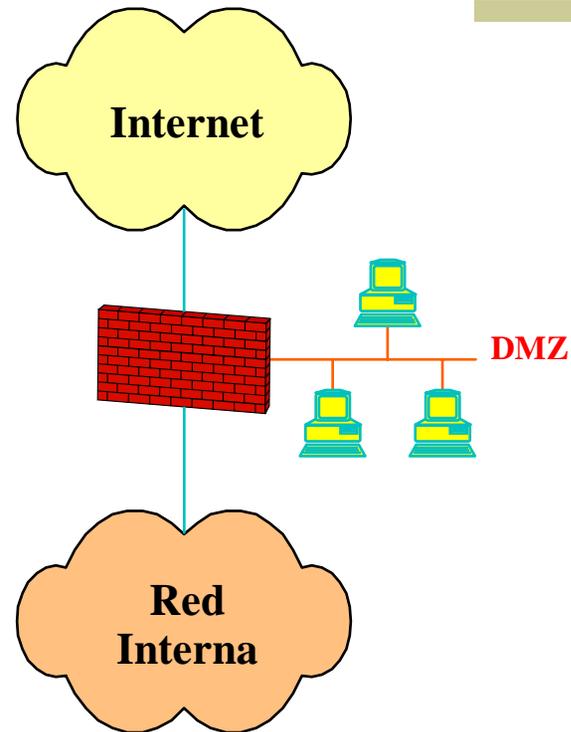
- Los paquetes se analizan y se verifican contra un conjunto de reglas, conocidas como ACL: Access Control List.
 - Las reglas más básicas permiten verificar por las direcciones de origen/destino, los puertos de origen/destino, protocolo, flags específicos de cada protocolo.
 - En la actualidad éstos aparatos pueden reconocer protocolos de nivel mayor.
- A cada regla se asocia una acción, que pueden ser: Permitir (ALLOW, ACCEPT), Rechazar (REJECT), Descartar (DROP).

Firewall

- ◆ Para que se usan
 - Seguridad
 - Restringir tráfico
 - ◆ Indeseable
 - ◆ Innecesario
 - ◆ No permitido
 - Control de uso de recursos
 - Limitando la tasa de tráfico
 - Limitando el tipo de tráfico

Firewall

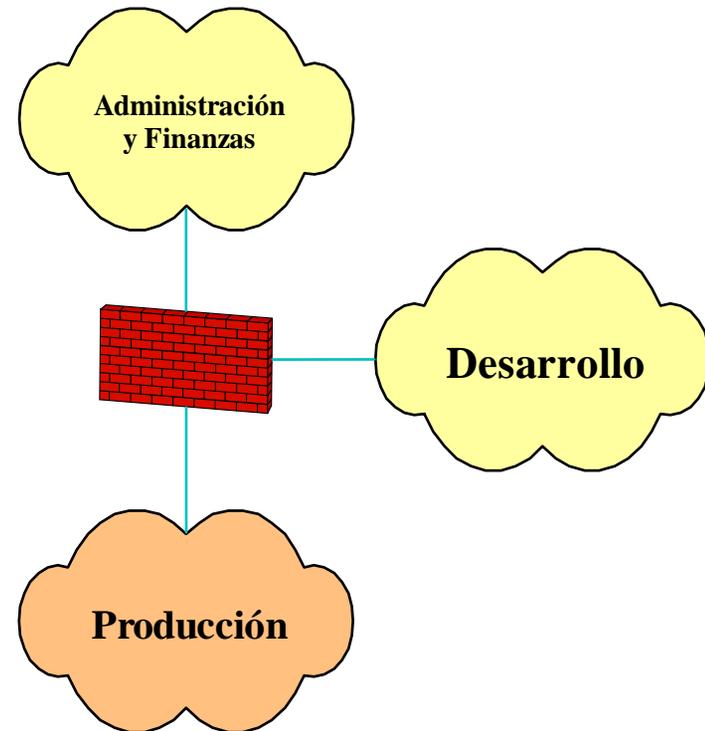
- ◆ Configuración Típica
 - Tres interfaces de red: una para Internet, una para la DMZ y otra para la red interna.



Firewall

◆ Configuración Típica

- Tres o más interfaces de red, para separar áreas que tienen políticas de seguridad diferentes.
- ¿Se puede implementar algo parecido usando otro método?



NAT

- ◆ Qué es
- ◆ Cómo funciona
- ◆ Para que sirve
- ◆ Tipos: NAT y NAPT
- ◆ Usos típicos

NAT

- ◆ Qué es
 - Network Address Translation
 - Consiste en cambiar la dirección de origen y/o destino de un datagrama IP, al pasar por un elemento de red (router, firewall, switch, host, etc).
 - La motivación para el cambio viene de diversas fuentes: necesidad, diseño y/o seguridad.

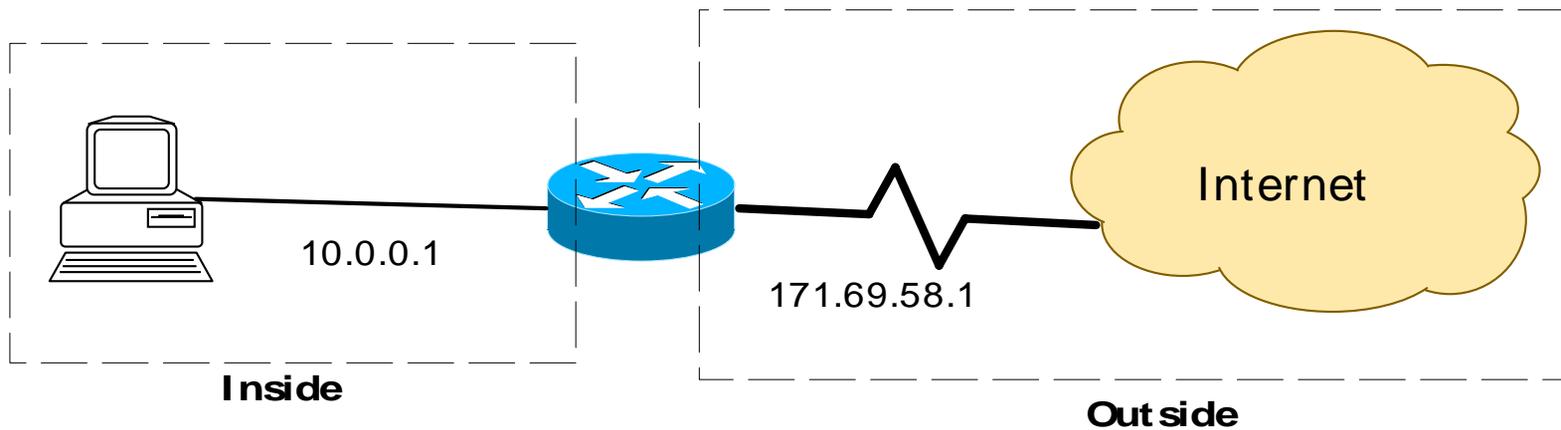
NAT

◆ Cómo funciona

- Existen tres tipos de NAT, dependiendo del mecanismo para decidir la dirección a usar para el cambio:
 - Estático, una dirección tiene asociada otra dirección fija que siempre se usará al cambiar.
 - Dinámica, una dirección no tiene asociación fija, por tanto será reemplazada por otra elegida dinámicamente.
 - Masquerading, también conocido como NAPT, se asigna dinámicamente una dirección y puerto para realizar el cambio.

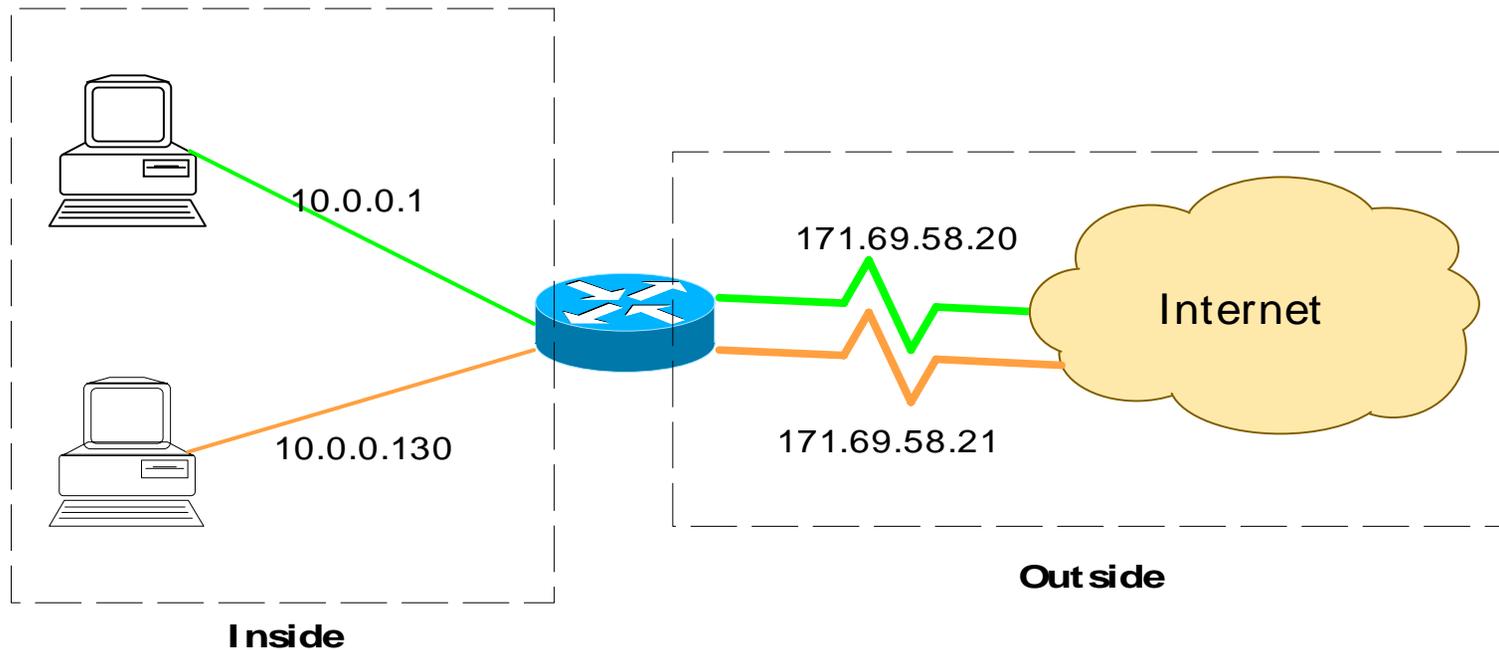
NAT

- ◆ Cómo funciona
 - NAT Estático



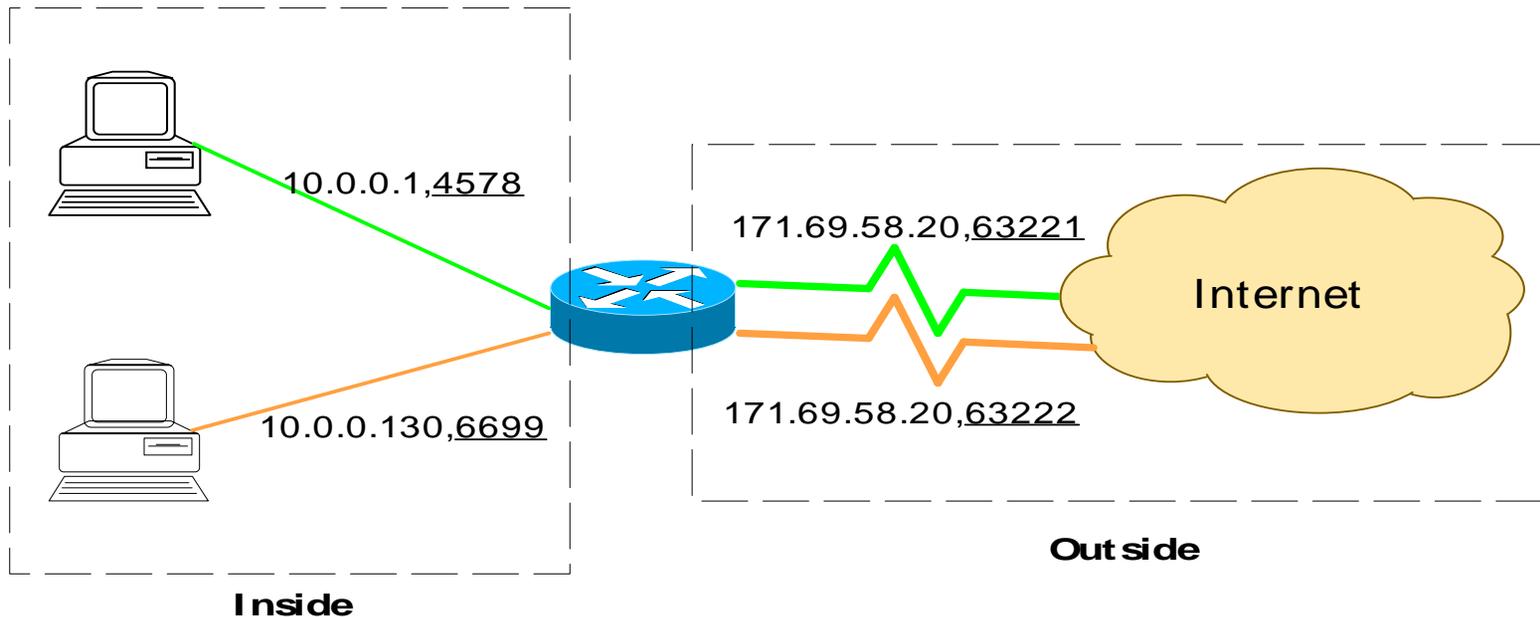
NAT

- ◆ Cómo funciona
 - NAT Dinámico



NAT

- ◆ Cómo funciona
 - NAT o Masquerading



NAT

◆ Para qué sirve

- Necesidad: Se tiene un número insuficiente de direcciones IP públicas, por lo que se deben asociar un varias direcciones privadas a unas pocas direcciones públicas.
- Diseño: Se quiere ocultar la topología local de una red, enmascarándola a través de NAT.
- Seguridad: Mediante el uso de NAT se puede tener un control implícito de que tráfico de “entrada” está permitido.

NAT

◆ Cómo funciona

- Dependiendo de los campos que se cambien, NAT se divide en SNAT (cambia el origen) o DNAT (cambia el destino).
- La diferencia es muy sutil y depende del punto de vista. Según los ejemplos anteriores, se hace DNAT, pero cuando los paquetes de “respuesta” vienen de vuelta, se hace SNAT.
- En otros casos se produce el fenómeno inverso.

Statefull Firewall

◆ Descripción

- Nueva generación de firewalls que permiten controlar aspectos que las primeras generaciones no podían
 - Tasa de entrada/salida de datos (por flujo/conexión o global)
 - Número de conexiones entrantes/salientes (por unidad de tiempo).
- Incluso algunos entienden capas superiores a la 3, pudiendo analizar el “contenido” a nivel de aplicación de un paquete de datos.

Experiencia en laboratorio

- ◆ Vamos a experimentar en el laboratorio los diversos conceptos asociados a firewall, utilizando IPTables en Linux.
- ◆ Revisaremos
 - Filtros de entrada
 - Filtros de salida
 - Control de uso de recursos
 - NAT, DNAT, SNAT.

IPTables

◆ Definición

- Según lo publicado en la página Web <http://www.netfilter.org>
- IPTables es una estructura genérica de tabla para la definición de reglas. Cada regla dentro de una tabla IP consiste de un número de clasificadores (matches) y una acción (objetivo)

◆ Características

- Filtro de paquetes tipo stateless para IPv4 e IPv6
- Filtro de paquetes tipo stateful para IPv4
- Todo tipo de NAT y NAPT
- Infraestructura flexible y extensible
- Etc.

IPTables

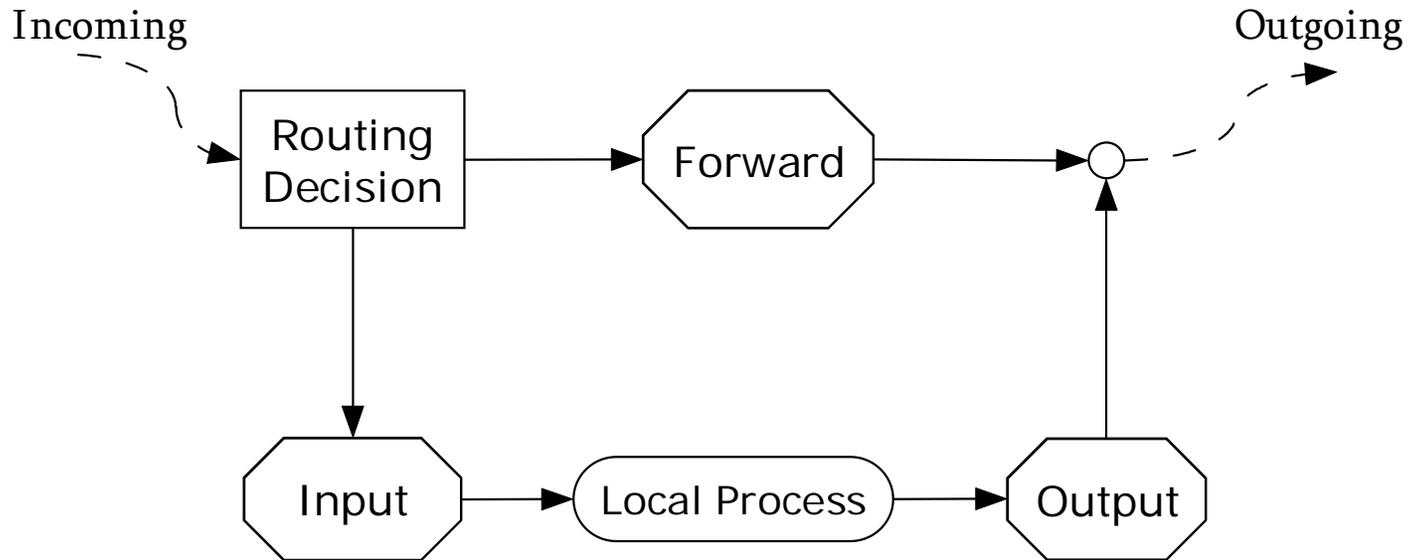
- ◆ La definición de reglas para iptables está definida usando comandos.
 - Estructura general de cada comando:
 - Tabla
 - Comando
 - Cadena
 - Regla

IPTables

- ◆ Tablas:
 - Filter
 - INPUT, OUTPUT, FORWARD.
 - NAT
 - PREROUTING, OUTPUT, POSTROUTING.
 - Mangle
 - PREROUTING, OUTPUT (2.4.17).
 - INPUT, FORWARD, POSTROUTING (2.4.18)

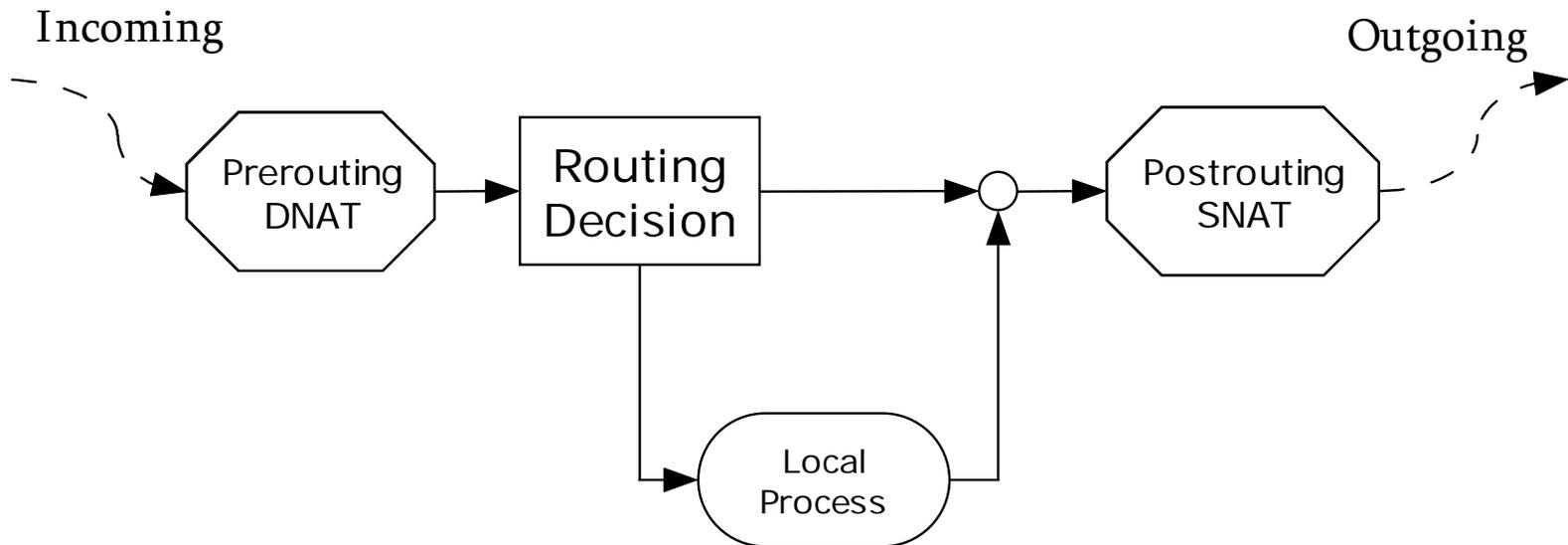
IPTables

- ◆ Cómo se examinan las cadenas [Tabla Filter]



IPTables

- ◆ Cómo se examinan las cadenas [Tabla NAT]



IPTables

- ◆ Comandos:
 - -A, append
 - -D, delete
 - -I, insert
 - -L, list
 - -F, flush

IPTables

◆ Reglas

- Algunas de las opciones para definir reglas son:
 - `!`, negar una condición
 - `-p`, protocol: verifica el campo “protocol” del datagrama. Puede ser *udp*, *tcp*, *icmp* o *all*.
 - `-s`, source: Dirección origen del datagrama.
 - `-d`, destination: Dirección de destino del datagrama.
 - `-i`, in-interface: Nombre de la interface de entrada del datagrama.
 - `-o`, out-interface: Nombre de la interface de salida del datagrama.

IPTables

◆ Sintaxis

- La sintaxis general (muy simplificada para nuestro ejemplo es)
 - `Iptables -t tabla comando cadena definicion_regla -j target`
 - *Tabla* puede ser **nat**, **filter** o **mangle**; *comando* puede ser `-A`, `-D`, `-L`, `-F`, `-I`; *cadena* puede ser INPUT, OUTPUT, FORWARD; *target* puede ser ACCEPT o DROP.

IPTables

◆ Ejemplos de ACL en un router

```
#
# Reglas que autoriza entrada de cualquier paquete que se origine
# como repuestas a una comunicaci3n interior
#
iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
#
# Autorizo el ICMP a la red
iptables -A FORWARD -d 200.27.115.0/24 -p icmp -j ACCEPT
# Autorizo accesos a whois.nic.cl TCP (43) UDP (43)
iptables -A FORWARD -d 200.1.123.2 -p tcp -m tcp --dport 43 -j ACCEPT

# Autorizo accesos a www.nic.cl TCP (80,443,5555)
iptables -A FORWARD -d 200.1.123.3 -p tcp -m tcp --dport 80 -j ACCEPT
iptables -A FORWARD -d 200.1.123.3 -p tcp -m tcp --dport 443 -j ACCEPT
iptables -A FORWARD -d 200.1.123.3 -p tcp -m tcp --dport 5555 -j ACCEPT

# Rechazo algunos accesos
iptables -A FORWARD -p tcp -m tcp --dport 1:21 -j DROP
iptables -A FORWARD -p tcp -m tcp --dport 23:24 -j DROP
iptables -A FORWARD -p tcp -m tcp --dport 26:42 -j DROP
```

IPTables

◆ Ejemplos en un host final

```
# Evitamos que los mails salgan
iptables -A OUTPUT -o eth0 -p tcp --dport 25 -j DROP
# Aceptamos los ICMP
iptables -A INPUT -p icmp --icmp-type any -j ACCEPT
# Aceptamos los accesos al sitio Web
iptables -A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
# Rechazamos todo el resto
iptables -A INPUT -j REJECT --reject-with icmp-host-prohibited
```

IPTables

◆ Ejemplos de NAT

```
#
# Configuracion DNAT
#
iptables -t nat -A PREROUTING -d 200.4.121.6 -p tcp -m tcp --dport 6666 -j DNAT
--to-destination 192.168.10.1
iptables -t nat -A PREROUTING -d 200.4.121.7 -p tcp -m tcp --dport 6667 -j DNAT
--to-destination 192.168.10.11

#
# Configuro SNAT para direcciones locales como 200.1.123.1
#
iptables -t nat -A POSTROUTING -s 172.30.10.0/255.255.255.0 -d 200.27.2.2/255.255.255.255
-o eth1 -j SNAT --to-source 192.80.24.4
# eth0
iptables -t nat -A POSTROUTING -s 172.30.10.0/255.255.255.0 -d ! 172.30.12.0/255.255.255.0
-o eth0 -j SNAT --to-source 192.80.23.3
```