Seguretat Informàtica (SI)

Tema 4. Seguridad en la red

Davide Careglio

Fuentes: Jordi Nin, "Network Security", Computer Security, 2014

Temario

- Tema 1. Introducción
- Tema 2. Criptografía
- Tema 3. Infraestructura PKI

- Tema 4. Seguridad en la red
- Tema 5. Seguridad en las aplicaciones

- Tema 6. Seguridad en los sistemas operativos
- Tema 7. Análisis forense

Temario

- Tema 1. Introducción
- Tema 2. Criptografía
- Tema 3. Infraestructura PKI

- **Tema 4. Seguridad en la red**
- Tema 5. Seguridad en las aplicaciones

- Tema 6. Seguridad en los sistemas operativos
- Tema 7. Análisis forense

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.1 - Firewall
Definición

- Un firewall es
  - Una parte de un sistema informático o red diseñada para bloquear el acceso no autorizado y permitir comunicaciones autorizadas
  - Un dispositivo o conjunto de dispositivos que está configurado para permitir o denegar transmisiones de red en función de un conjunto de reglas y otros criterios
- Se necesita un firewall cuando
  - Cuando hay que conectar una red segura a una no segura
  - Red segura: red privada, red corporativa, etc.
  - Red no segura: Internet

4.1 - Firewall
Terminología

- Firewall
  - Se puede referir a la política de seguridad y las estrategias de seguridad
- Firewall system
  - Conjunto de hardware y software que implementa un firewall
- Bastion Host
  - Un host seguro expuesto a una red insegura
- Dual-homed host
  - Un ordenador con dos interfaces de red
- Network perimeter o Demilitarized Zone (DMZ)
  - Una red puesta entre la red insegura y la red segura que necesita protección particular

4.1 – Firewall

Que puede hacer

- Proporciona un único punto de defensa, lo que permite un acceso controlado y auditado a los servicios prestados
- Refuerza la seguridad del propio sistema
- Implementa una política de seguridad para acceder a la red segura
- Puede monitorear el tráfico entrante/saliente
- Puede limitar la exposición a una red insegura
- Puede convertirse en el punto donde se toman decisiones de seguridad, ya que todo el tráfico lo atraviesa

4.1 – Firewall

Que NO puede hacer

- No puede proteger la red contra ataques maliciosos desde dentro de la misma red segura
- No puede proteger la red contra el tráfico que no la atraviesa
- No puede proteger la red contra errores/malas configuraciones de los servicios autorizados
- Cualquier dato de aplicación que lo atraviese tiene el potencial de causar problemas (por ejemplo, troyanos)
- Si la política de seguridad no es denegada por defecto, no puede proteger la red contra nuevos ataques

4.1 – Firewall

Tipos

- Filtro de paquetes
  - Inspecciona cada paquete que pasa a través de la red y lo acepta o rechaza según las reglas definidas por el administrador
- Firewall a nivel de circuito
  - Aplica mecanismos de seguridad que se aplica al establecer una conexión TCP/UDP
  - Una vez realizada, los paquetes pueden fluir sin más controles
- Gateway de aplicaciones
  - Aplica mecanismos de seguridad a aplicaciones específicas, como servidores FTP, Telnet y HTTP
- Servidor proxy
  - Intercepta todos los mensajes que ingresan y salen de la red actuando como intermediario entre clientes y servidores
  - El servidor proxy oculta las verdaderas direcciones de red

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.1.1 - Topologías de firewall

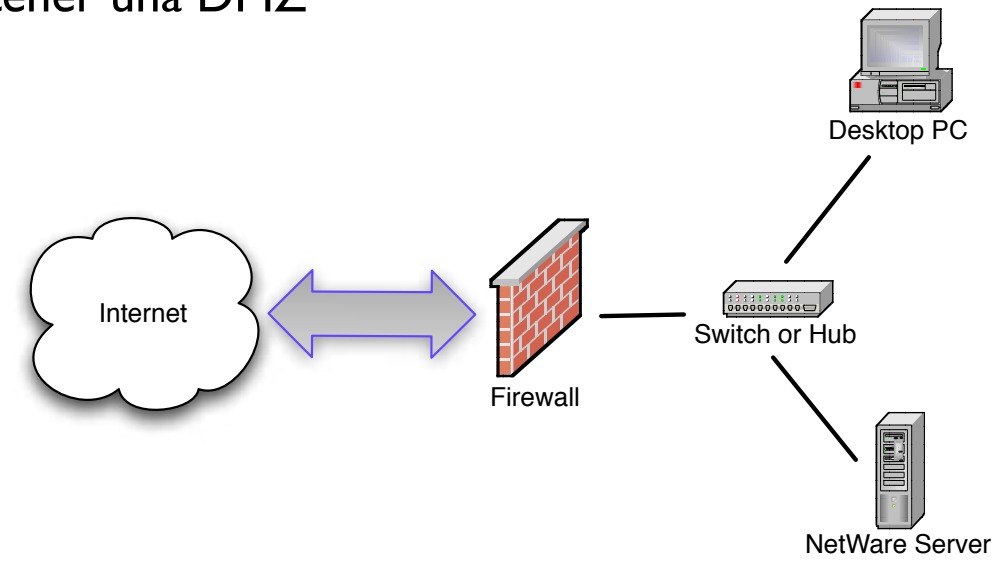
- Las más usadas son

- Dual-homed
- Two-Legged network with a full exposed DMZ
- Restricted DMZ via dialup Firewall
- Three-legged firewall

4.1.1 – Topologías de firewall

Firewall dual-homed

- El Firewall dual-homed (doble referencia) es una de las formas más simples
- Internet entra en la red por el firewall directamente a través de una línea de acceso
- No se puede tener una DMZ

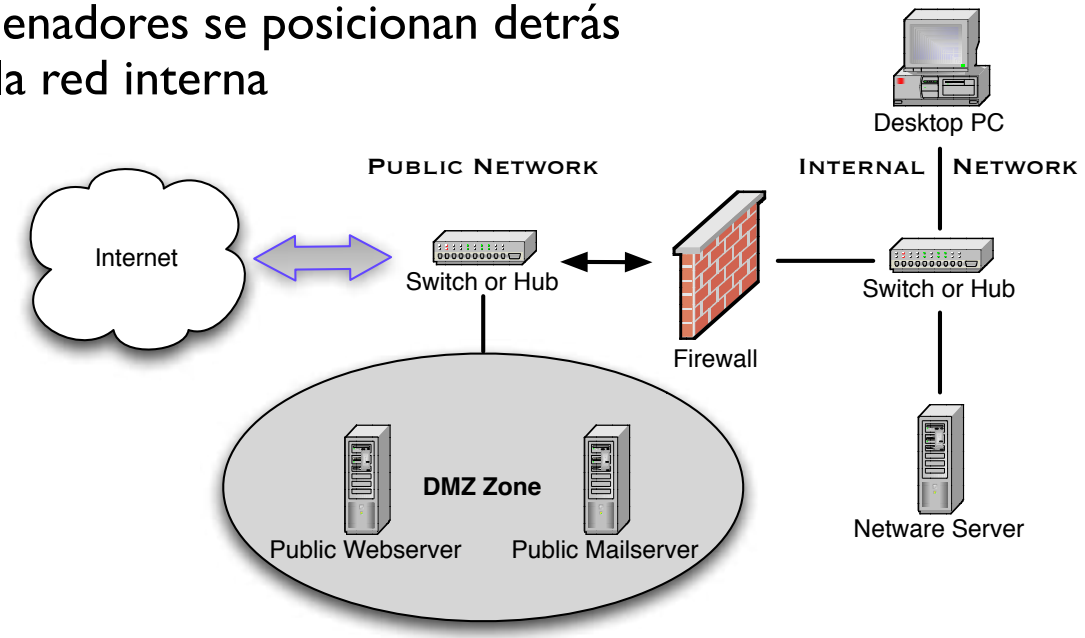


- El firewall se encarga de pasar paquetes que pasan por sus reglas de filtrado entre la red interna e Internet, y viceversa
- Los dos "homes" se refieren a las dos redes de las que forma parte el firewall: una interfaz conectada a la red externa (Internet) y la otra conectada a la red interna

4.1.1 – Topologías de firewall

Two-Legged network with a full exposed DMZ

- Entre el firewall y el router del ISP (Internet) hay un hub o switch
- Se crea una zona DMZ con servidores que tiene acceso directo a Internet (sin el filtrado del firewall)
- El resto de ordenadores se posicionan detrás del firewall en la red interna

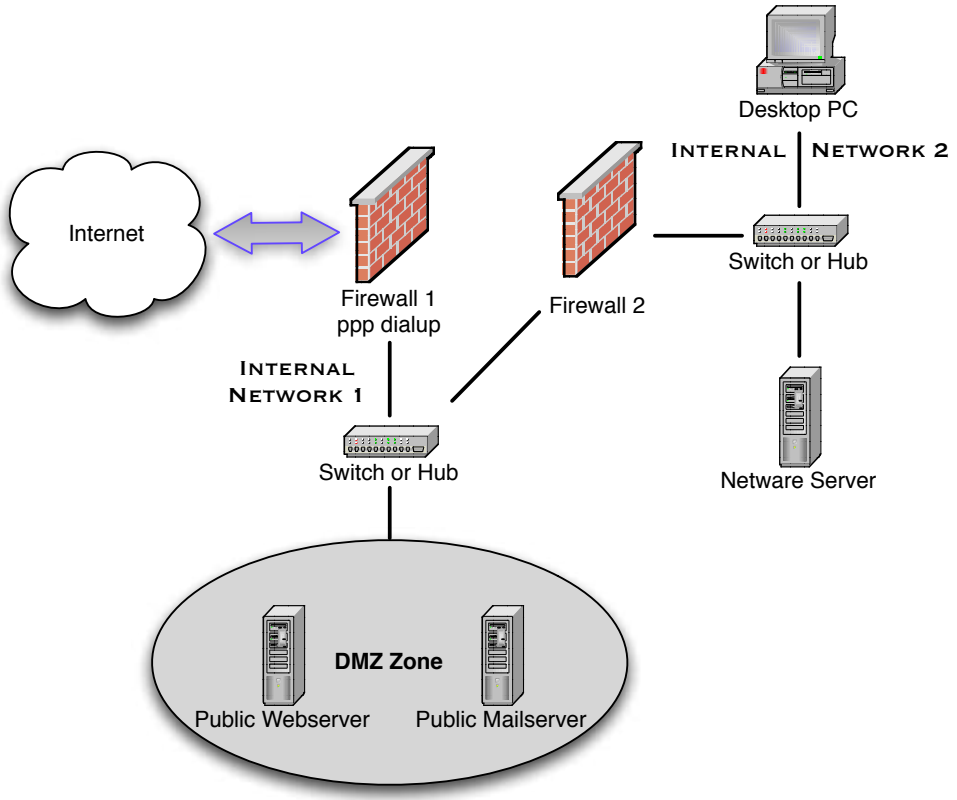


- Ventajas
  - El firewall solo necesita dos interfaces
  - La configuración es muy simple, igual que la anterior
- Desventajas
  - La red DMZ está completamente expuesta a Internet

4.1.1 – Topologías de firewall

Restricted DMZ via Dialup Firewall

- Para proteger la DMZ, una solución es usar un segundo firewall

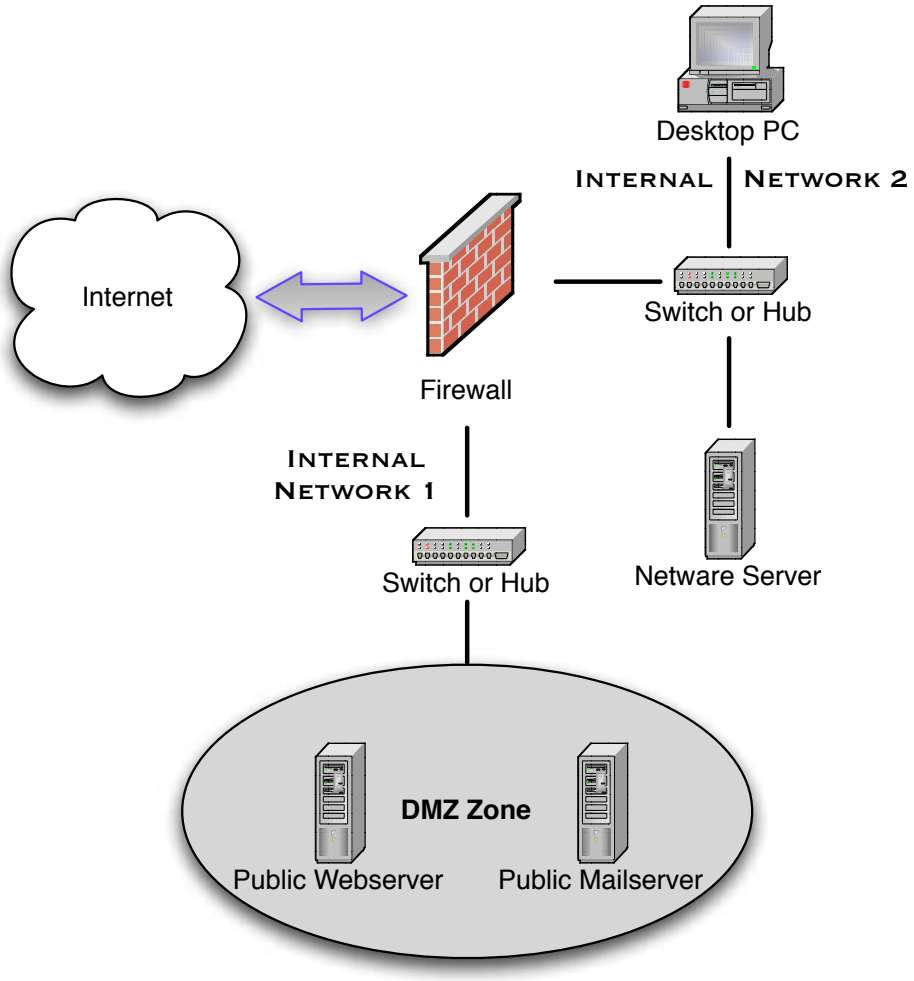


- El primer firewall
  - Protege la DMZ
  - Mantiene la conexión con Internet (ppp dialup)
- El segundo firewall
  - Sigue teniendo dos interfaces (dual-homed)
  - Protege la red interna

4.1.1 – Topologías de firewall

Three-legged firewall

- Para tener un único firewall y una DMZ protegida, se necesita una tercera interfaz



- Ventajas
  - Solo se necesita una @IP pública
  - La red DMZ puede usar @IP privadas
- Desventajas
  - Una interfaz adicional en el firewall
  - Más complejo de configurar (más reglas)

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado e iptables
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.1.2 - Reglas de filtrado

- Las reglas de filtrado son un conjunto de reglas secuenciales para denegar/permitir cierto tráfico de red que contiene un determinado número de puerto, tipo de protocolo, @IP...

- Criterios de filtrado
  - @IP origen y destino
  - Puerto origen y destino
  - Tipo de protocolo (IP, TCP, UDP, ICMP)
  - Estado de la conexión (nueva, respuesta o relacionada)

4.1.2 - Reglas de filtrado

- Para evitar complicar las reglas de filtrado mezclando permisos con prohibiciones, generalmente se usa uno de estos dos enfoques
- En el primero, se crea una lista de condiciones permitidas y se concluyen con una última línea que deniega todo lo que queda
  - Permitir condición_1
  - Permitir condición_2
  - ...
  - Permitir condición_n
  - Prohibir todo
- El segundo enfoque es el contrario del primero: la lista tiene una sería de condiciones prohibidas y se concluyen con una que permite todo
  - Prohibir condición_1
  - Prohibir condición_2
  - ...
  - Prohibir condición_n
  - Permitir todo

4.1.2 - Reglas de filtrado

- En el primero enfoque, la regla por defecto es denegar el acceso excepto si es explícitamente permitido
  - Más seguro ya que puede ser difícil saber que servicios son seguros y cuales no
  - Más restrictivos y menos confortable para los usuario

- En el segundo enfoque, la regla por defecto es aceptar cualquier acceso excepto si es explícitamente denegado
  - Más confortable para los usuarios
  - Más fácil de administrar
  - Menos seguro ya que no puede prevenir ataques desconocidos o errores

4.1.2 - Reglas de filtrado

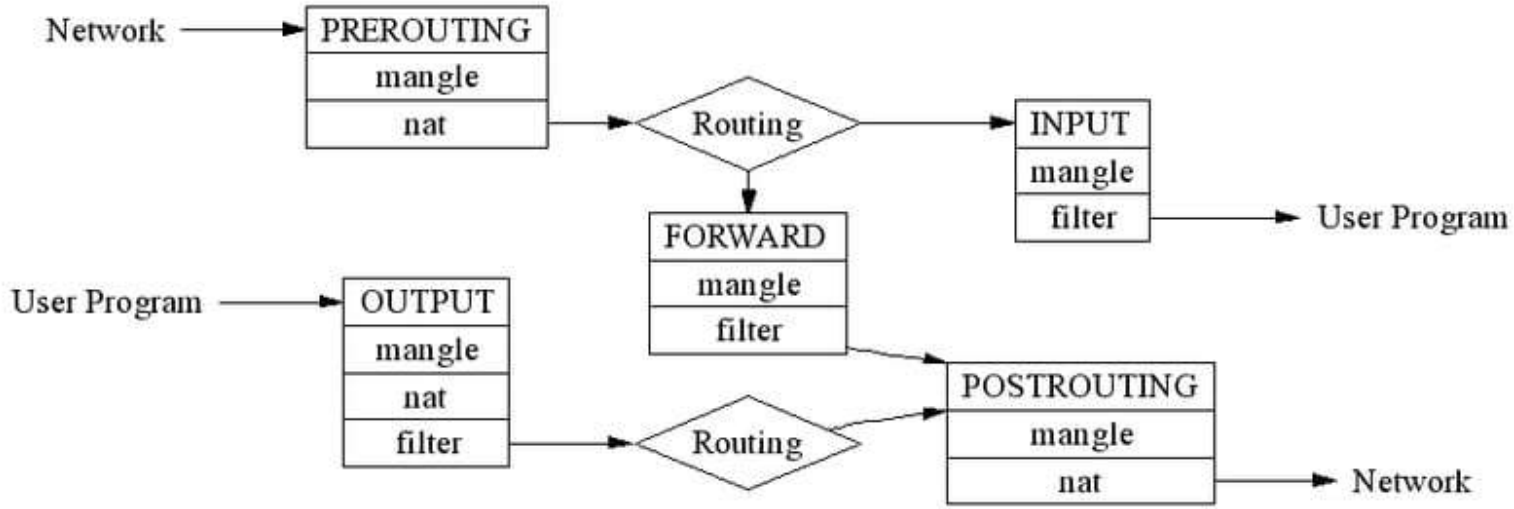
- En XC, se ha visto como crear un firewall usando ACL en Cisco
- En SI, se usará la aplicación **iptables** en Linux

4.1.2 - iptables

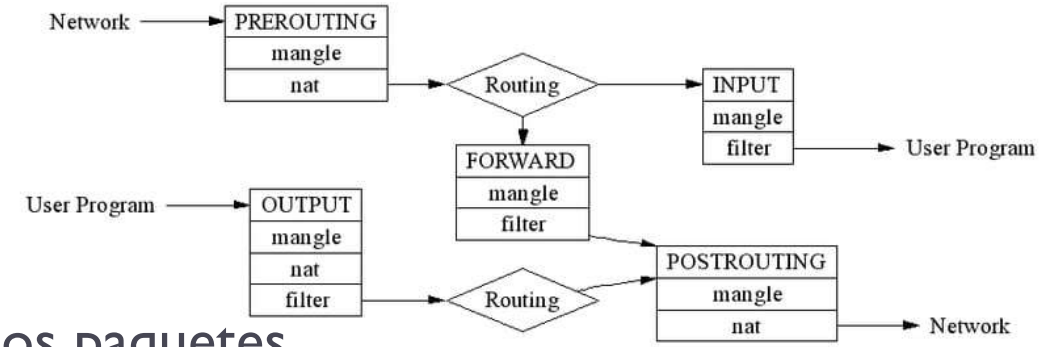
- iptable (Netfilter)
  - Nació en el 1998 y se incorporó en Linux 2.3 en marzo 2000
  - Licencia GPL
  - Es un framework disponible en el kernel de Linux que permite interceptar y manipular paquetes de red
  - Permite filtrar paquetes, realizar traducción de direcciones de red (NAT) o mantener registros de log
- Estructura
  - Las reglas se agrupan en cadenas (**chains**)
    - Cada cadena es una lista ordenada de reglas
  - Las cadenas se agrupan en tablas (**tables**)
    - Cada tabla está asociada con un tipo diferente de procesamiento de paquetes

4.1.2 - iptables

- Chains
  - PREROUTING
  - INPUT
  - OUTPUT
  - FORWARD
  - POSTROUTING
- Tables
  - mangle
  - nat
  - filter



4.1.2 - iptables



- Filter
  - Denegar/permitir determinados paquetes
- NAT
  - Modificar @IP/puertos de los paquetes
- Mangle
  - Modificar algunos otros campos de los paquetes como
  - **Tipo de Servicio**: campo de la cabecera IP usado para dar prioridad a los paquetes (define como tratar los paquetes en los routers)
  - **Tiempo de vida** (TTL): campo de la cabecera IP usado para que un paquete no se quede eternamente en una red si no encuentra una ruta (cada router quita 1 a esta campo y si es 0, lo descarta)
  - **Mark**: se pueden marcar determinados paquetes para que se traten luego de diferente manera (diferente ruta, diferente ancho de banda, etc.)

4.1.2 - iptables

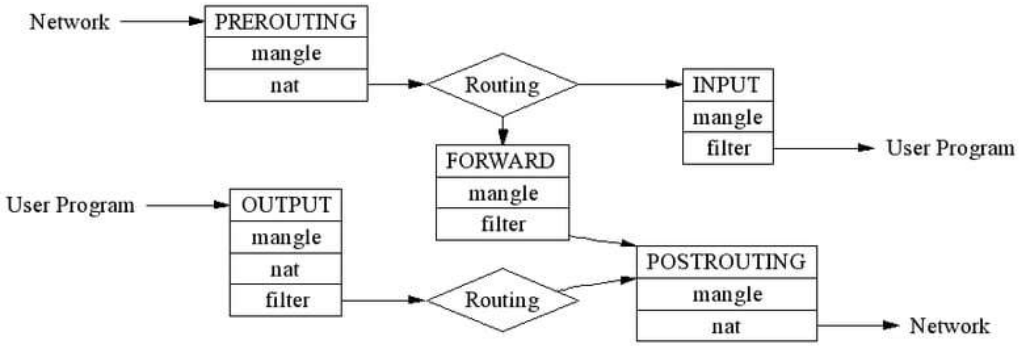


Table	Chain	Chain Function
Filter	FORWARD	Filters packets to servers accessible by another NIC on the firewall
	INPUT	Filters packets destined to the firewall
	OUTPUT	Filters packets originating from the firewall
NAT	PREROUTING	Address translation occurs before routing. Facilitates the transformation of the destination IP to be compatible with the firewall's routing table (DNAT)
	POSTROUTING	Address translation occurs after routing. There is no need to modify the destination IP. Used with NAT of the source IP address using either one-to-one or many-to-one (SNAT)
	OUTPUT	Network address translation for packets generated by the firewall

4.1.2 - iptables

iptables –[t table] action [options] –j type

- action: –{A | D | I} chain n
  - add, delete, insert
- options
  - –i: intefaz de entrada
  - –o: intefaz de salida
  - –p: protocolo {IP | TCP | UDP | ICMP}
  - –s: @IP origen {red+wildcard | host @IP | any}
  - –d: @IP destino {red+wildcard | host @IP | any}
  - –sport: puerto origen
  - –dport: puerto destino
  - –state: estado de la conexión {new | established | related}
- type
  - {ACCEPT | DROP | SNAT | DNAT}

4.1.2 - iptables

- Recordatorio wildcard
- Es como una mascara pero con los bits 0-1 invertidos
- Se compara una @IP de un paquete con una regla de iptables usando solo los bits en correspondencia de un 0 en la wildcard

- Ejemplo

  145.34.5.6 0.0.0.0 → se comparan todos los bits
  también se puede escribir host 145.34.5.6

  145.34.5.6 0.0.0.255 → se comparan solo los bits 145.34.5.0
  sería para todos las @IP de la red 145.34.5.0/24

  145.34.5.6 255.255.255.255 → no se compara ningún bit
  todas serían ciertas, como escribir ANY

4.1.2 - iptables

- Recordatorio: el orden de las reglas es importante

Rule set 1

- iptables -t filter -A INPUT -p ICMP -j DROP
- iptables -t filter -A INPUT -p IP -j ACCEPT

Rule set 2

- iptables -t filter -A INPUT -p IP -j ACCEPT
- iptables -t filter -A INPUT -p ICMP -j DROP

4.1.2 - iptables

- Recordatorio: el orden de las reglas es importante

Rule set 1

- iptables -t filter -A INPUT -p ICMP -j DROP
- iptables -t filter -A INPUT -p IP -j ACCEPT

Rule set 2

- iptables -t filter -A INPUT -p IP -j ACCEPT
- iptables -t filter -A INPUT -p ICMP -j DROP

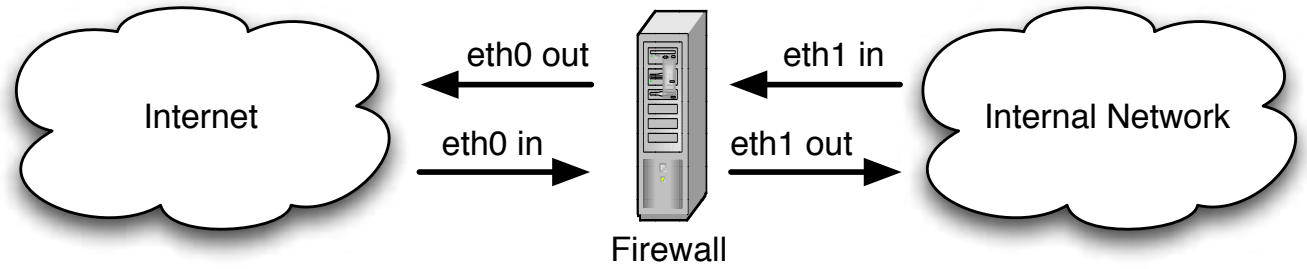
- La primera regla deniega la entrada de paquetes ICMP pero acepta los paquetes IP
- La segunda regla acepta la entrada de paquetes IP y también los ICMP ya que los ICMP son paquetes IP
  - La segunda línea no se comprobaría nunca

4.1.2 - iptables

- Las reglas por defecto en iptables es aceptar
  - Usa por defecto el enfoque 2: lista de reglas de prohibiciones y la última acepta el resto
- Si se quiere modificar

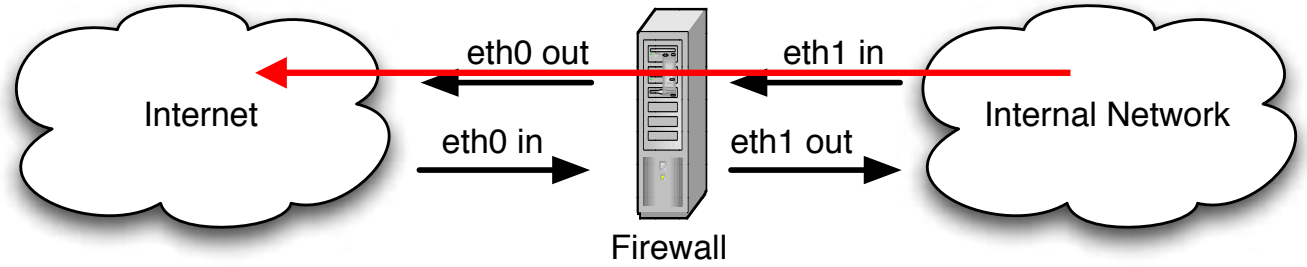
```
iptables –P chain {ACCEPT | DROP}
```

4.1.2 – iptables
Ejemplos

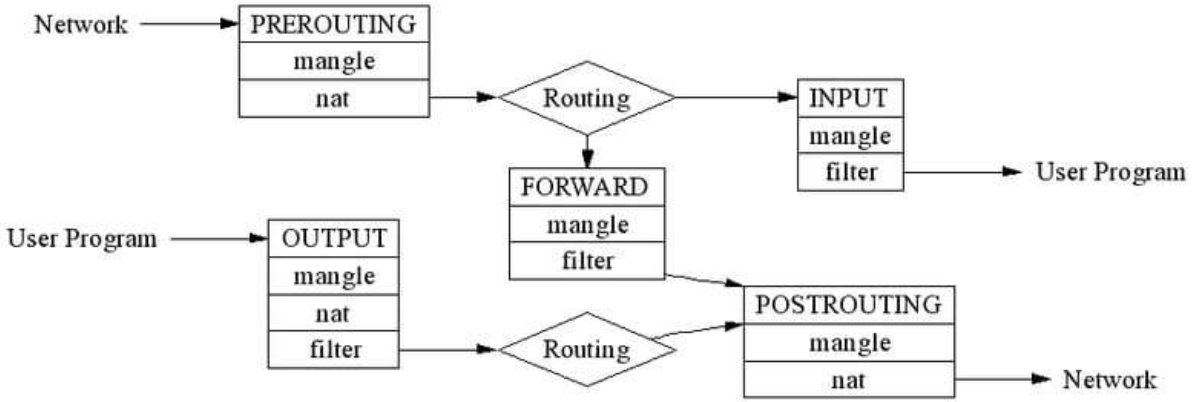


- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet

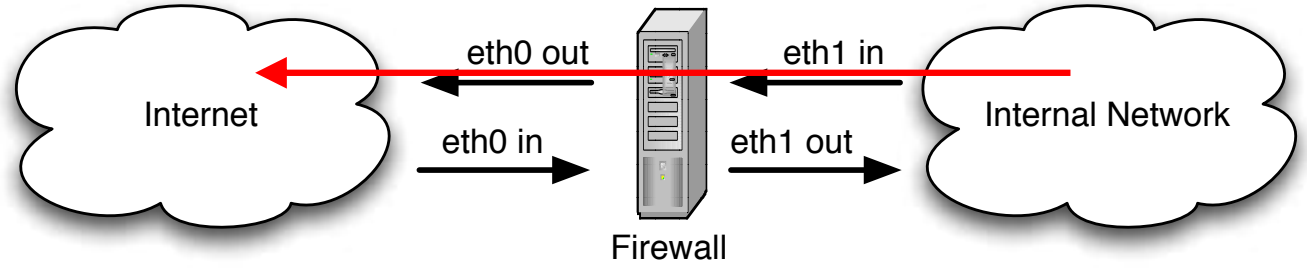
4.1.2 – iptables
Ejemplos



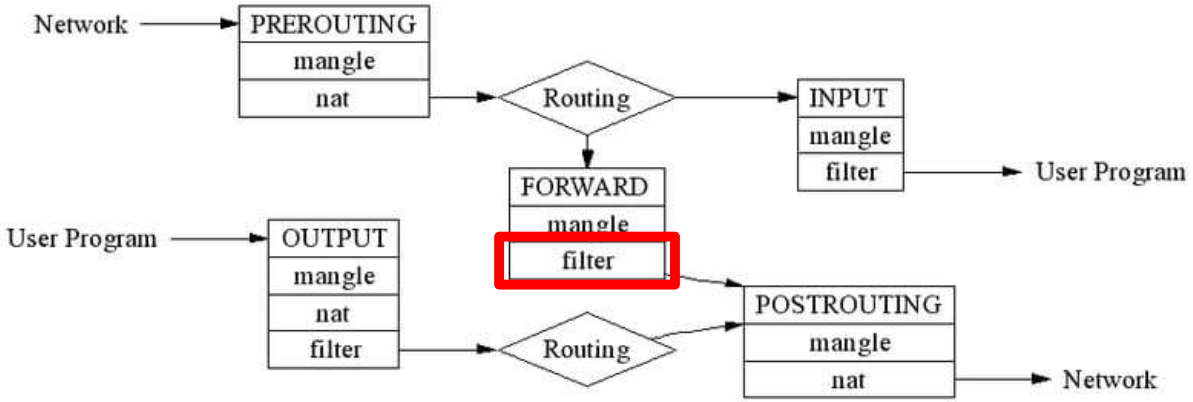
- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet



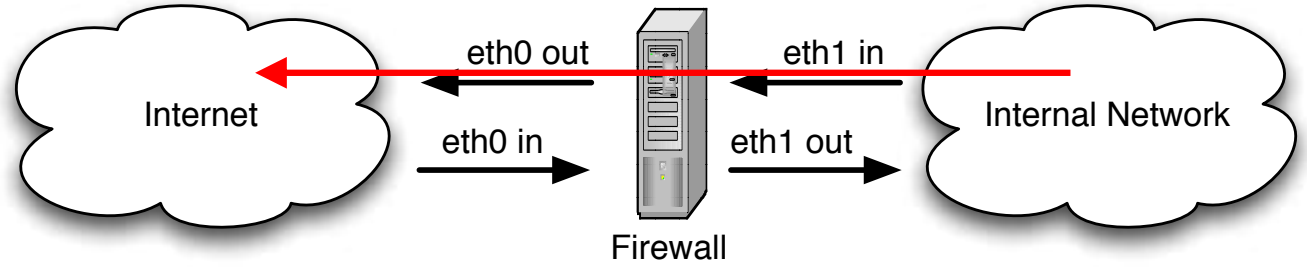
4.1.2 – iptables
Ejemplos



- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet



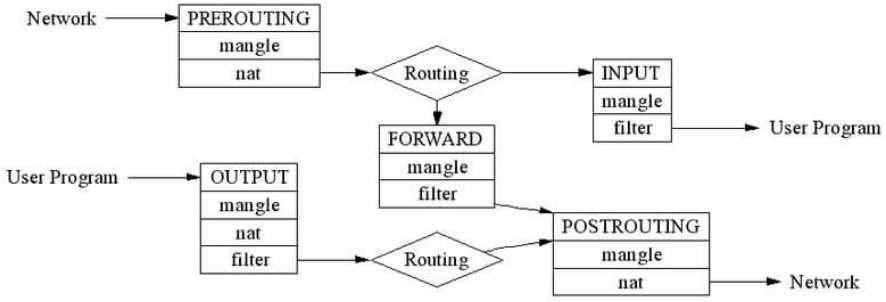
4.1.2 – iptables
Ejemplos



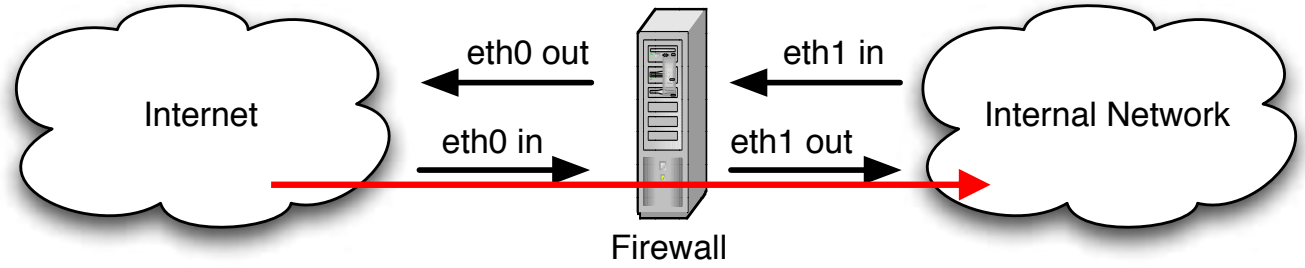
- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet

iptables –t filter –A FORWARD –p TCP –i eth1 –o eth0 –dport 80 –j ACCEPT

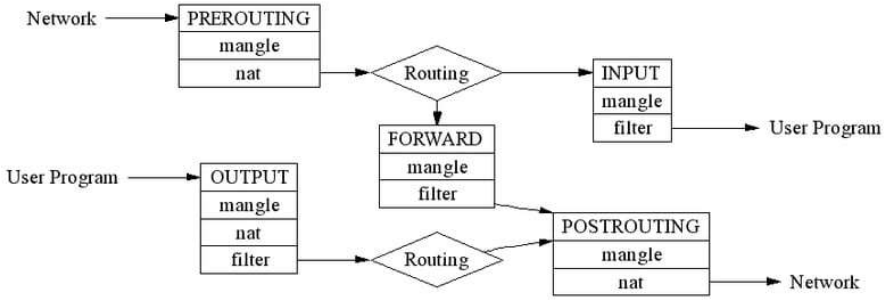
iptables –t filter –A FORWARD –i eth1 –o eth0 –j DROP



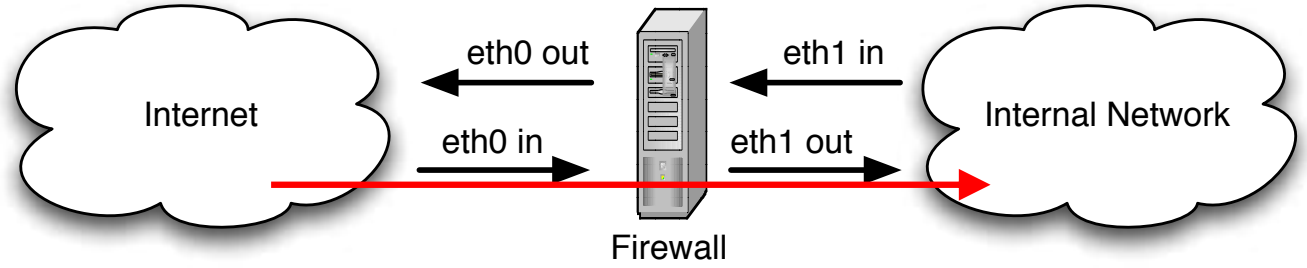
4.1.2 – iptables
Ejemplos



- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet ?



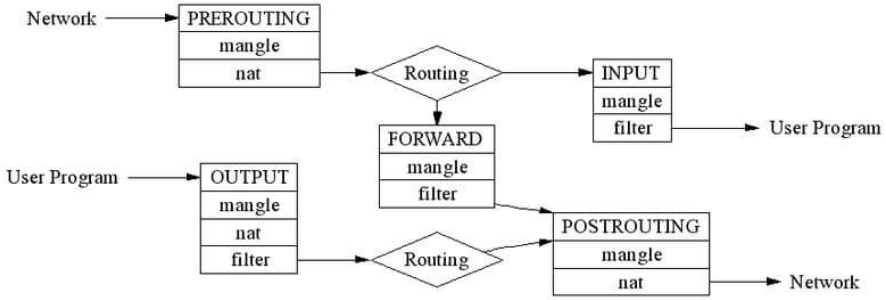
4.1.2 – iptables
Ejemplos



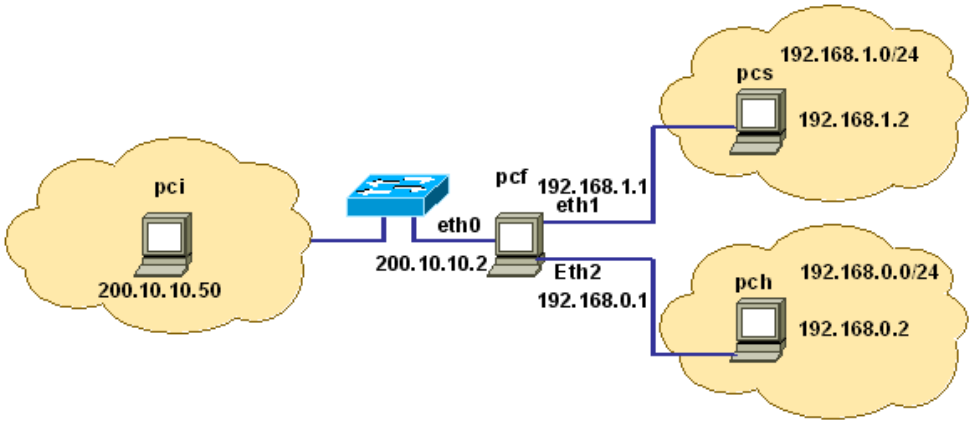
- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet

iptables –t filter –A FORWARD –p TCP –i eth0 –o eth1 –sport 80 –j ACCEPT

iptables –t filter –A FORWARD –i eth0 –o eth1 –j DROP



4.1.2 – iptables
Ejemplos



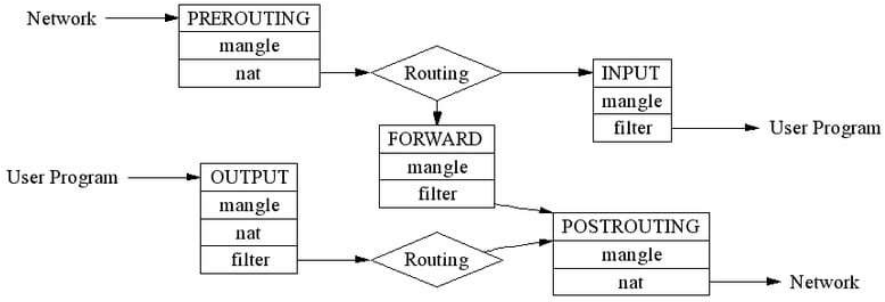
- Suponer que el firewall se ha configurado de esta forma y que todas las reglas por defecto son DROP

  iptables –t filter –A FORWARD –i eth2 –o eth1 –j ACCEPT

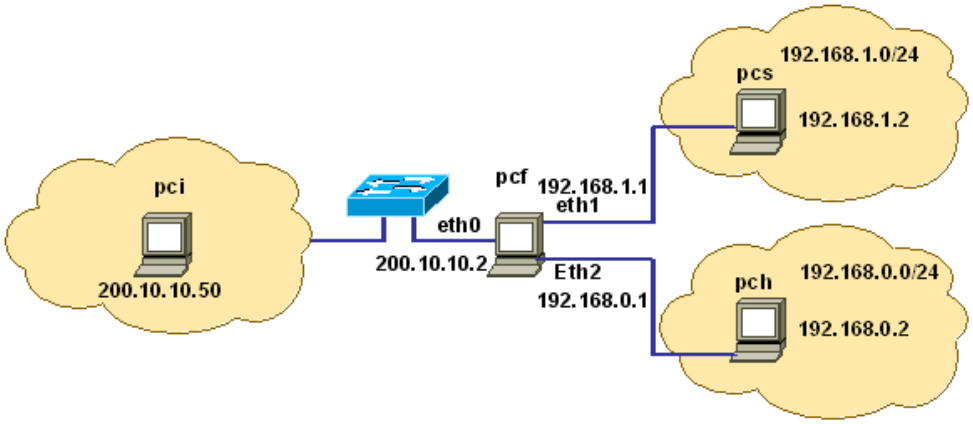
  iptables –t filter –A FORWARD –i eth1 –o eth2 –j ACCEPT

- Indicar la respuestas correctas

1. pci puede transmitir a pcs
2. pch puede transmitir a pcf
3. pch puede transmitir a pcs
4. pcs puede transmitir a pch



4.1.2 – iptables
Ejemplos



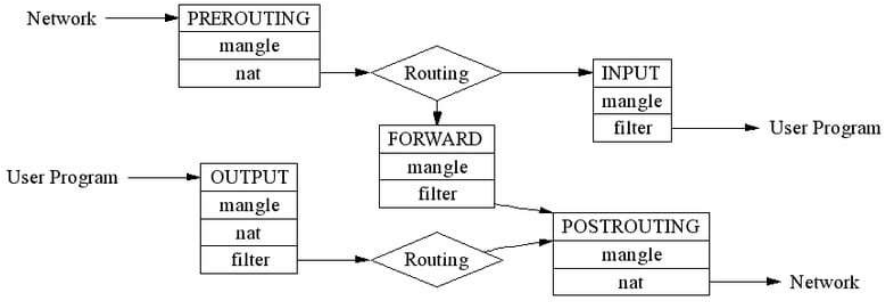
- Suponer que el firewall se ha configurado de esta forma y que todas las reglas por defecto son DROP

  iptables –t filter –A FORWARD –i eth2 –o eth1 –j ACCEPT

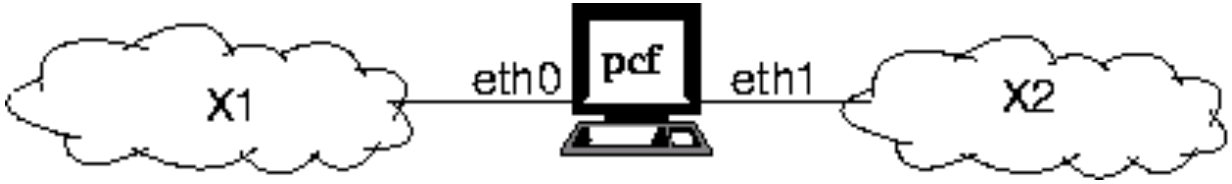
  iptables –t filter –A FORWARD –i eth1 –o eth2 –j ACCEPT

- Indicar la respuestas correctas

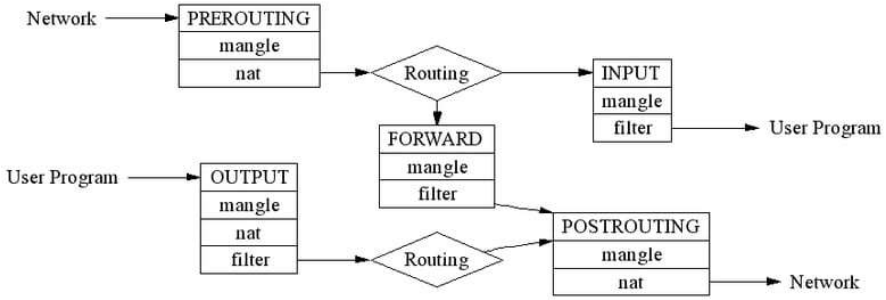
1. pci puede transmitir a pcs
2. pch puede transmitir a pcf
3. pch puede transmitir a pcs
4. pcs puede transmitir a pch



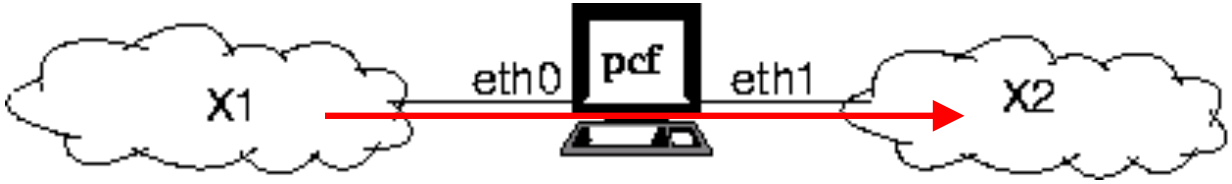
4.1.2 – iptables
Ejemplos



- Se quiere que los hosts de la red X1 puedan empezar una comunicación con los hosts de la red X2 (y estos puedan contestar) pero no viceversa

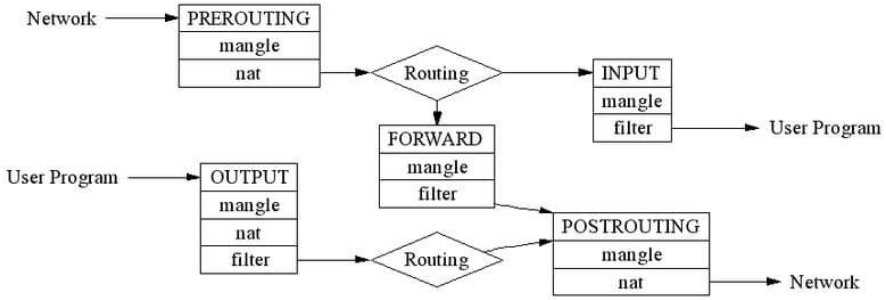


4.1.2 – iptables
Ejemplos

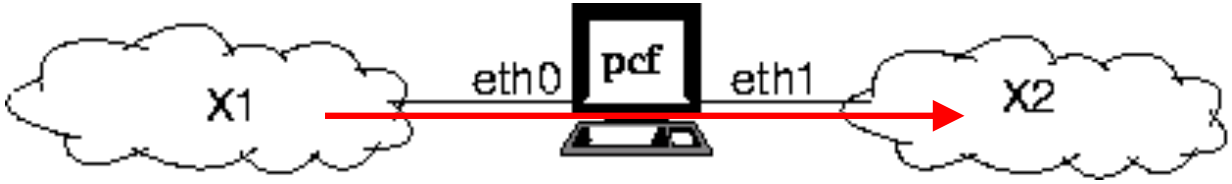


- Se quiere que los hosts de la red X1 puedan empezar una comunicación con los hosts de la red X2 (y estos puedan contestar) pero no viceversa

iptables –t filter –A FORWARD –i eth0 –o eth1 –j ACCEPT



4.1.2 – iptables
Ejemplos

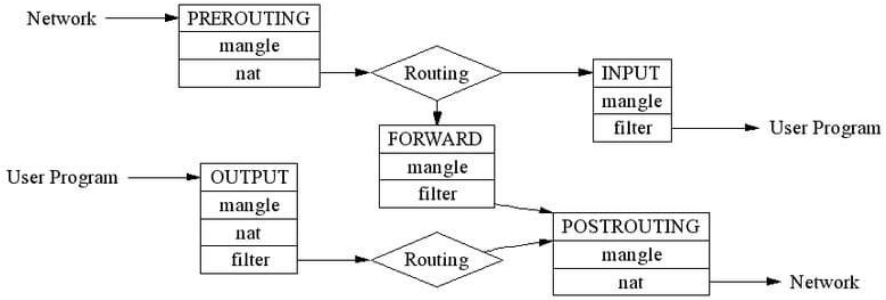


- Se quiere que los hosts de la red X1 puedan empezar una comunicación con los hosts de la red X2 (y estos puedan contestar) pero no viceversa

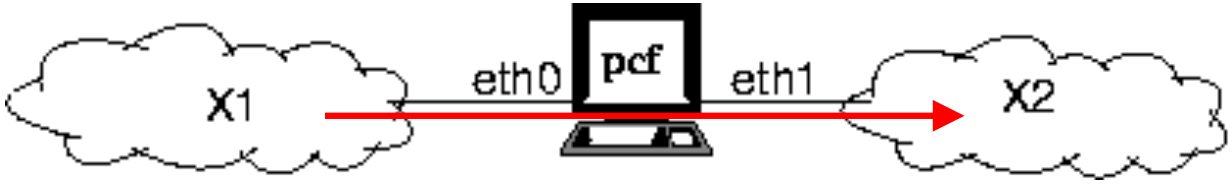
iptables –t filter –A FORWARD –i eth0 –o eth1 –j ACCEPT

iptables –t filter –A FORWARD –i eth1 –o eth0 –j DROP

- Funciona?



4.1.2 – iptables
Ejemplos

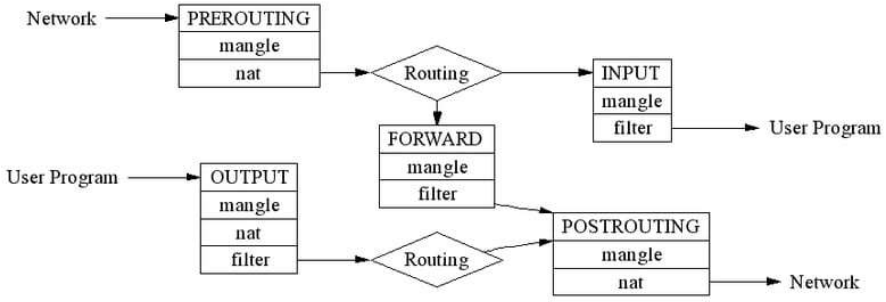


- Se quiere que los hosts de la red X1 puedan empezar una comunicación con los hosts de la red X2 (y estos puedan contestar) pero no viceversa

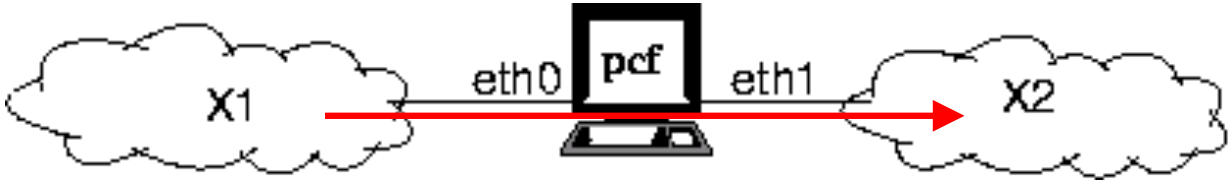
iptables –t filter –A FORWARD –i eth0 –o eth1 –j ACCEPT

iptables –t filter –A FORWARD –i eth1 –o eth0 –j ACCEPT

- Funciona?



4.1.2 – iptables
Ejemplos



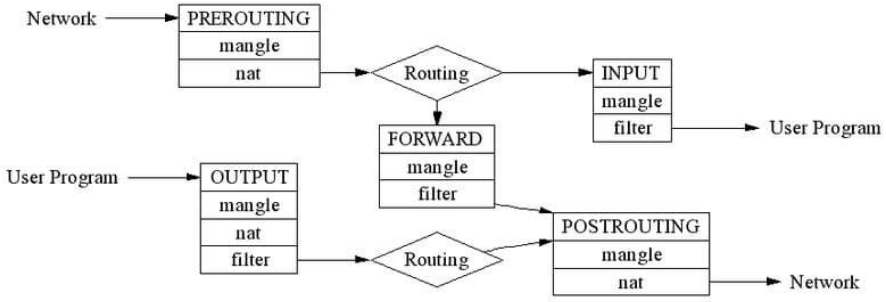
- Se quiere que los hosts de la red X1 puedan empezar una comunicación con los hosts de la red X2 (y estos puedan contestar) pero no viceversa

iptables –t filter –A FORWARD –i eth0 –o eth1 –j ACCEPT

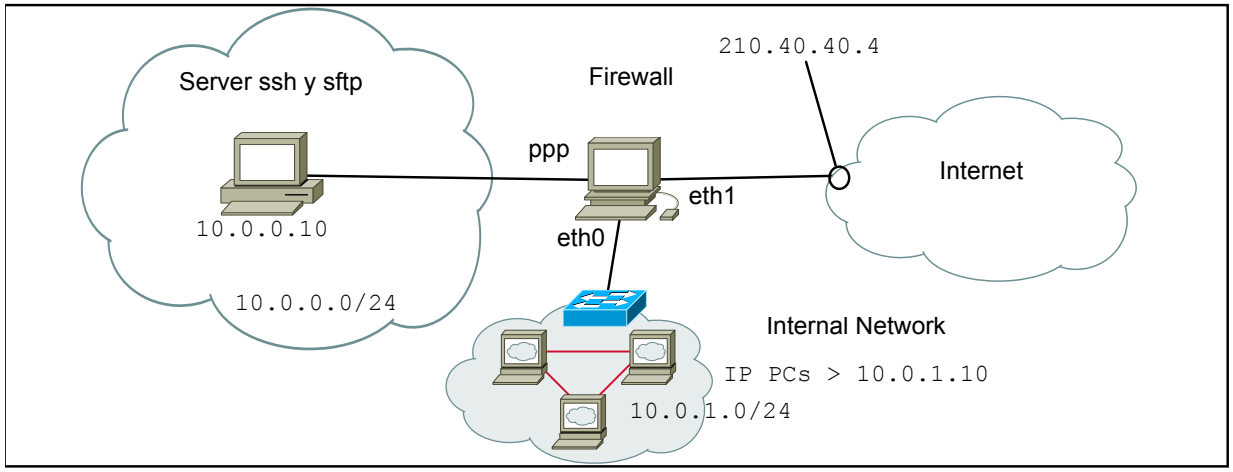
iptables –t filter –A FORWARD –i eth1 –o eth0 –state ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP

- Funciona?

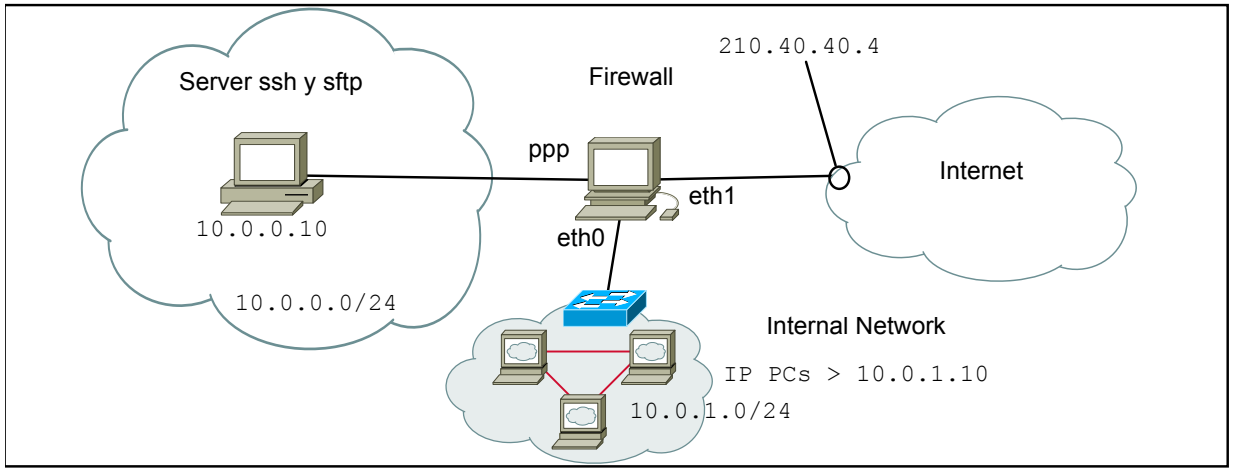


4.1.2 – iptables
Ejemplos

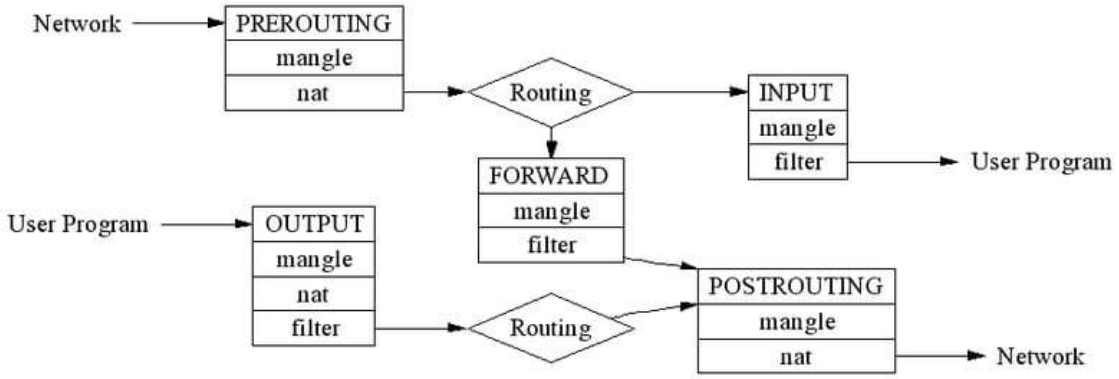


1) Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

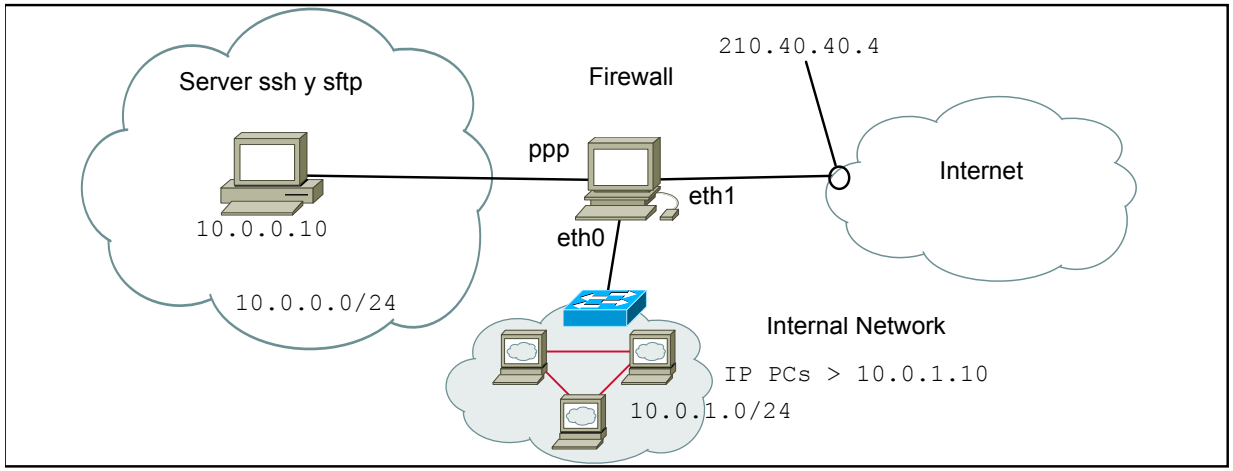
4.1.2 – iptables
Ejemplos



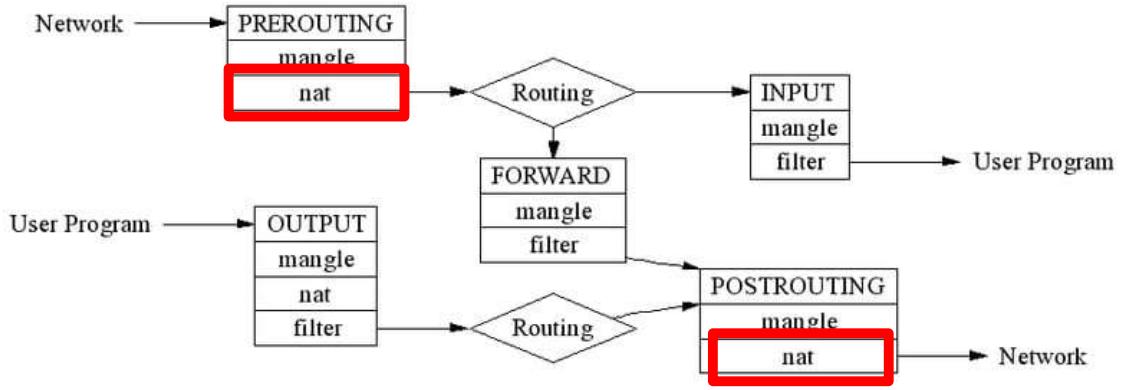
1) Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4



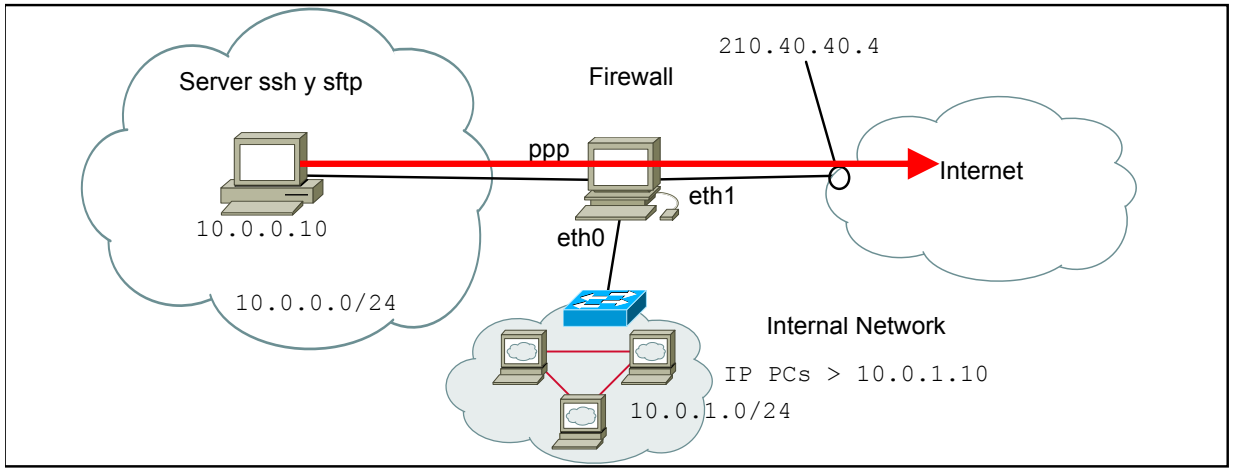
4.1.2 – iptables
Ejemplos



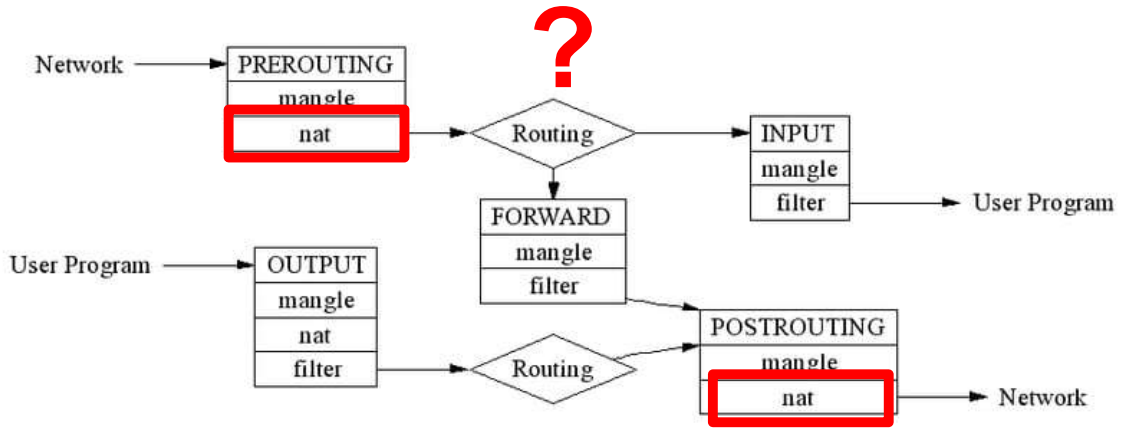
1) Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4



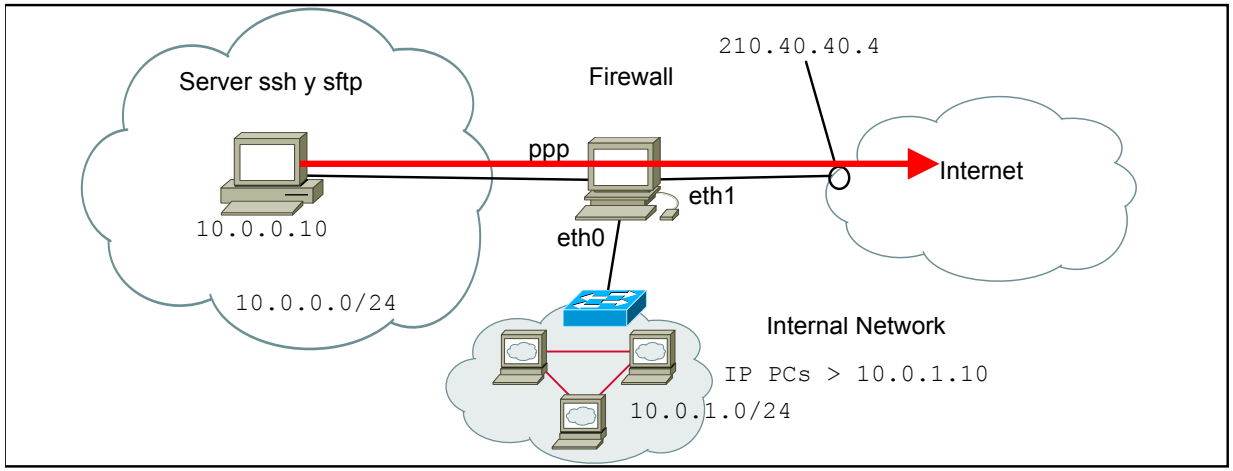
4.1.2 – iptables
Ejemplos



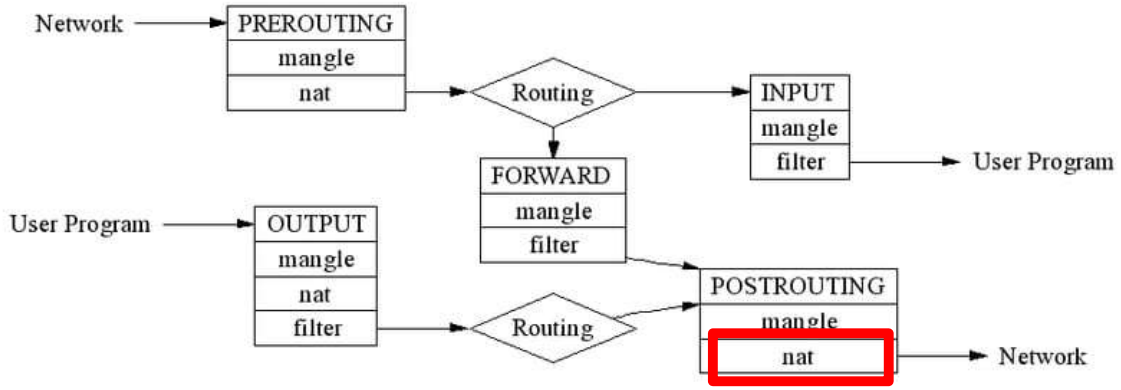
1) Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4



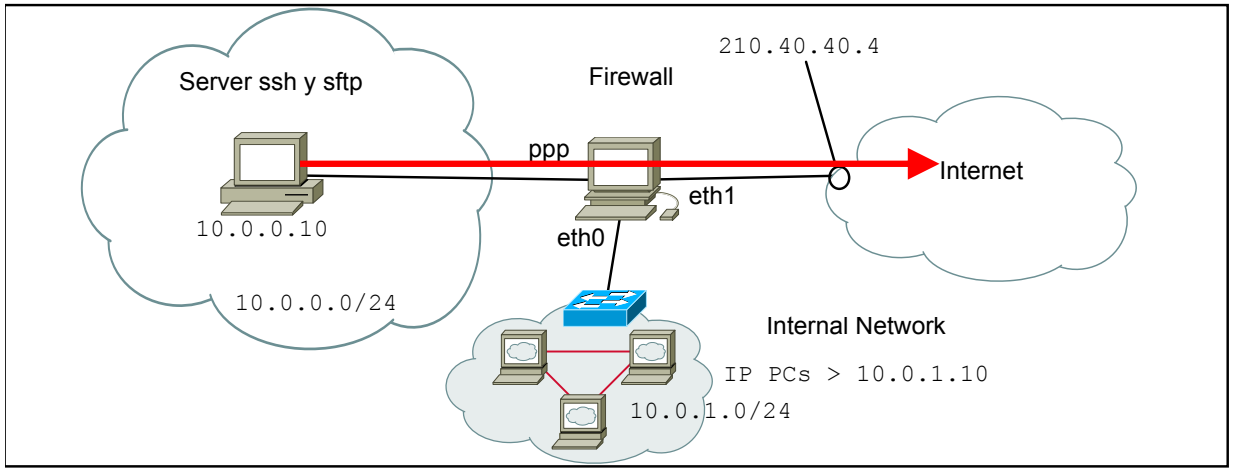
4.1.2 – iptables
Ejemplos



1) Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

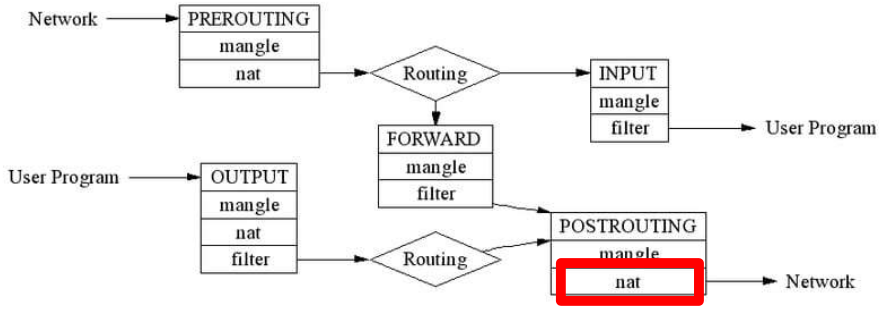


4.1.2 – iptables
Ejemplos

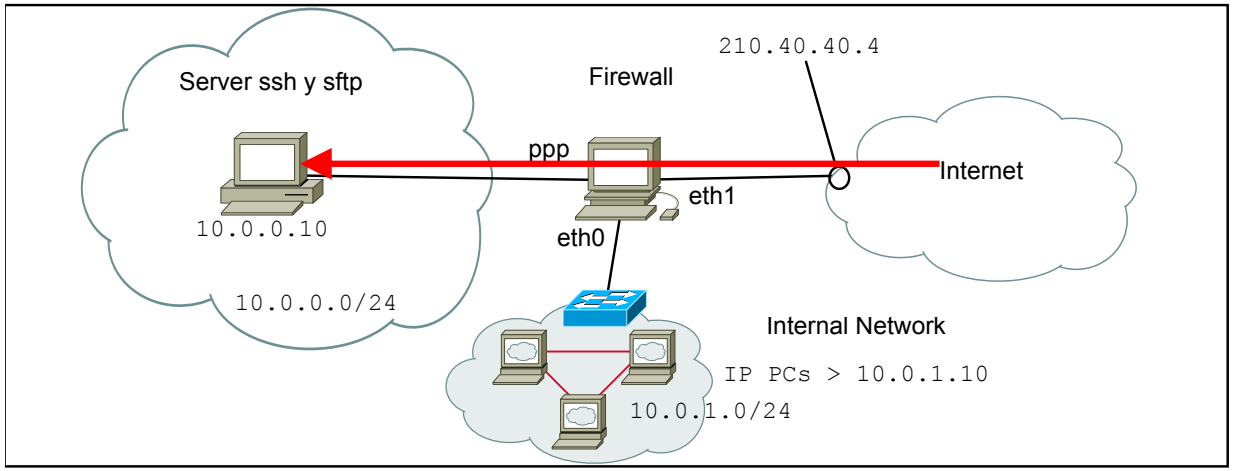


1) Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

iptables –t nat –A POSTROUTING –s 10.0.0.10 –o eth1 –j SNAT --to-source 210.40.40.4

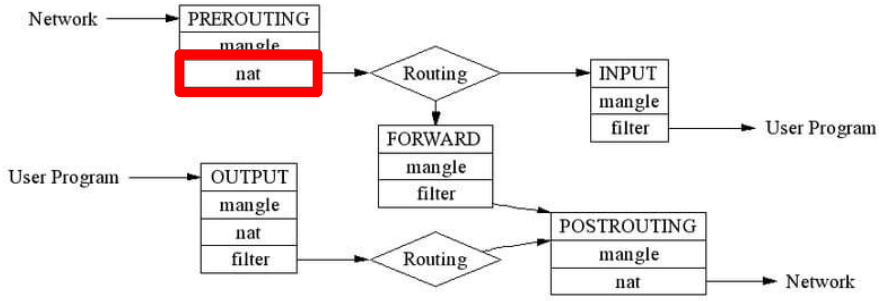


4.1.2 – iptables
Ejemplos

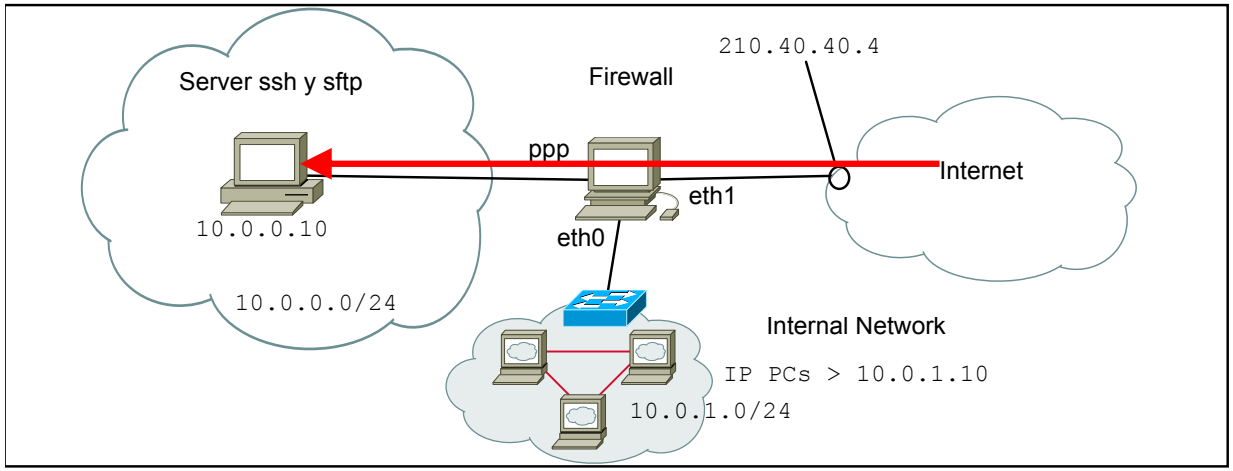


1) Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

iptables –t nat –A POSTROUTING –s 10.0.0.10 –o eth1 –j SNAT --to-source 210.40.40.4



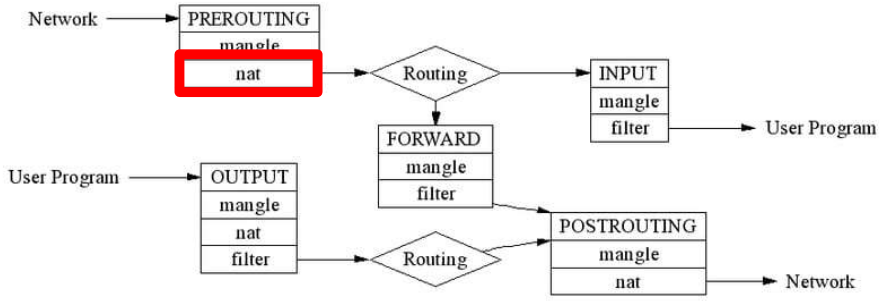
4.1.2 – iptables
Ejemplos



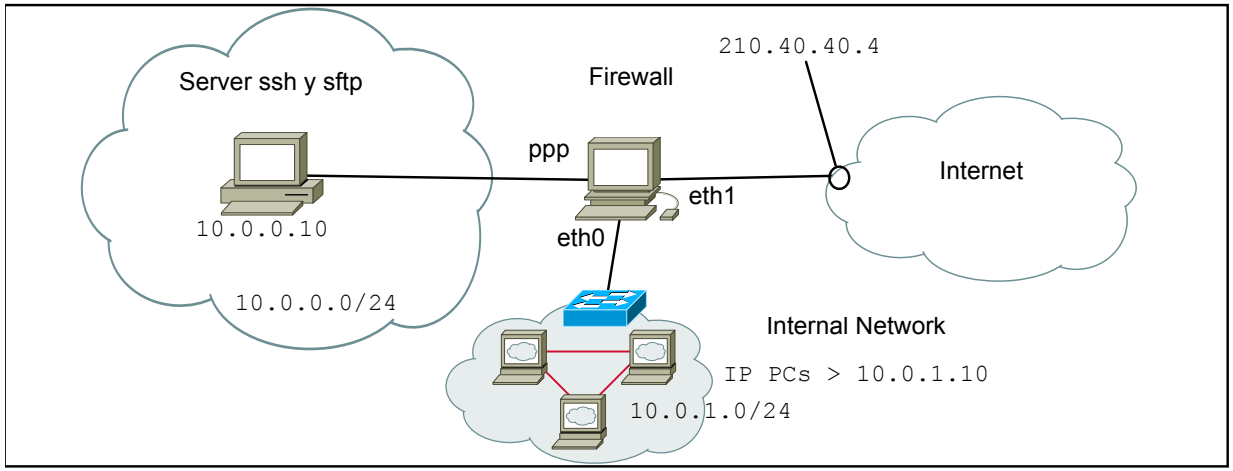
1) Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

iptables –t nat –A POSTROUTING –s 10.0.0.10 –o eth1 –j SNAT --to-source 210.40.40.4

iptables –t nat –A PREROUTING –d 210.40.40.4 –i eth1 –j DNAT --to-destination 10.0.0.10

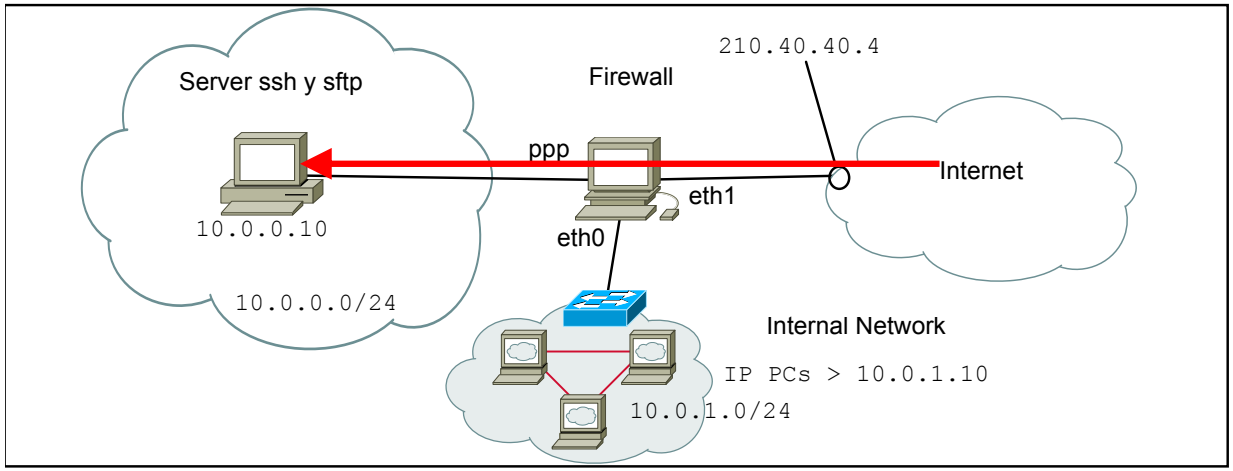


4.1.2 – iptables
Ejemplos

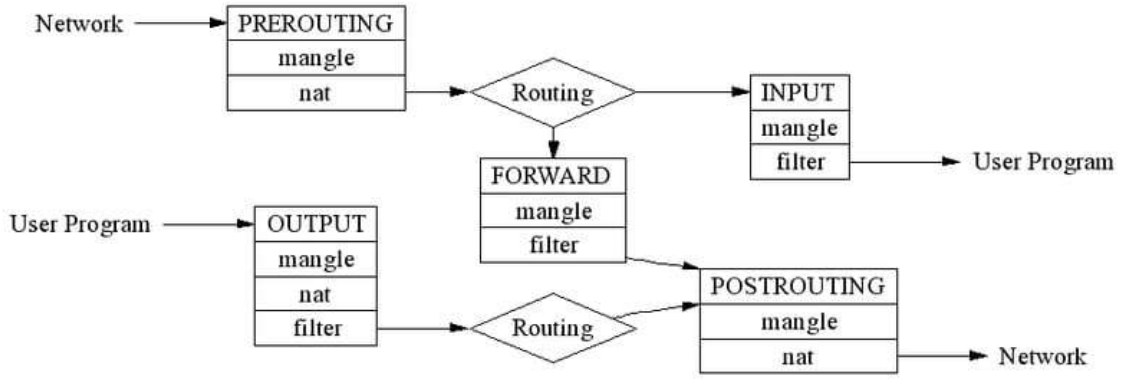


2) Configurar el firewall para que los hosts de Internet solo puedan acceder a los servicios ssh y sftp (puerto 22) del servidor 10.0.0.10

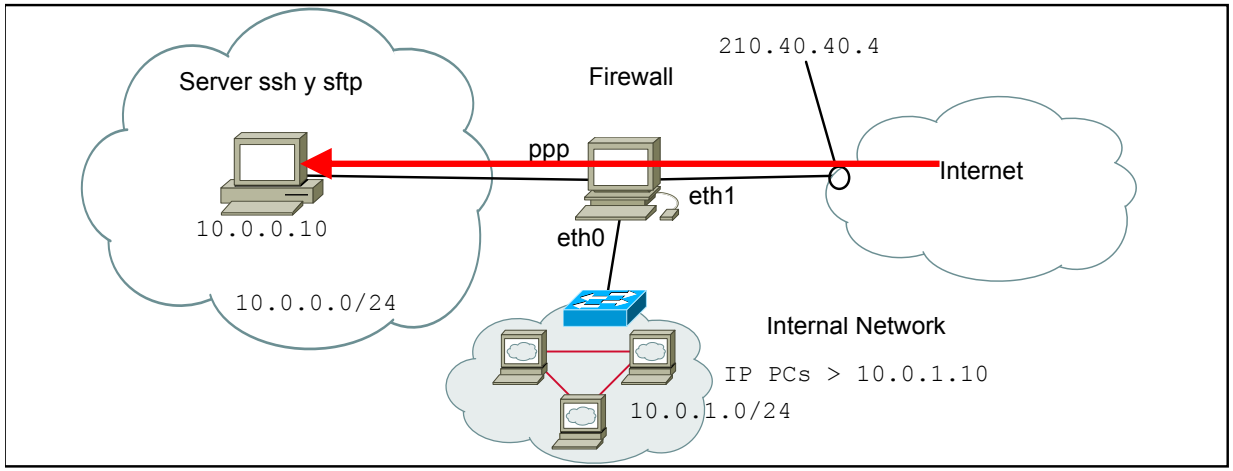
4.1.2 – iptables
Ejemplos



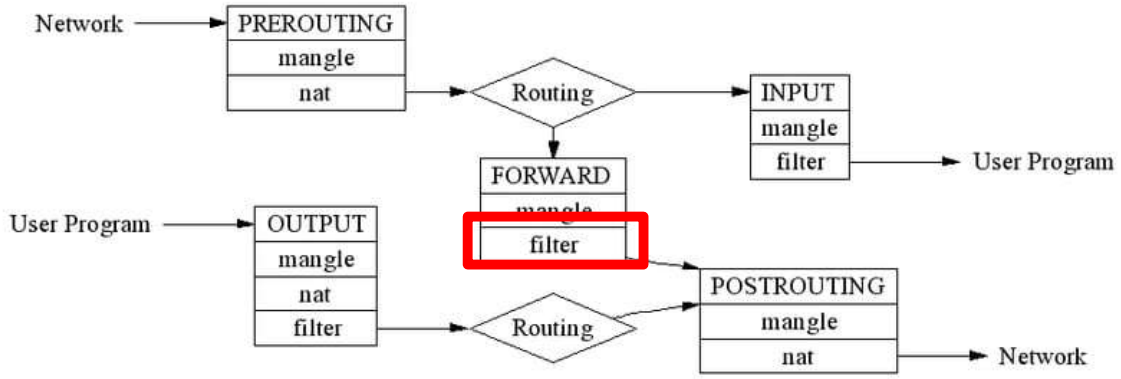
2) Configurar el firewall para que los hosts de Internet solo puedan acceder a los servicios ssh y sftp (puerto 22) del servidor 10.0.0.10



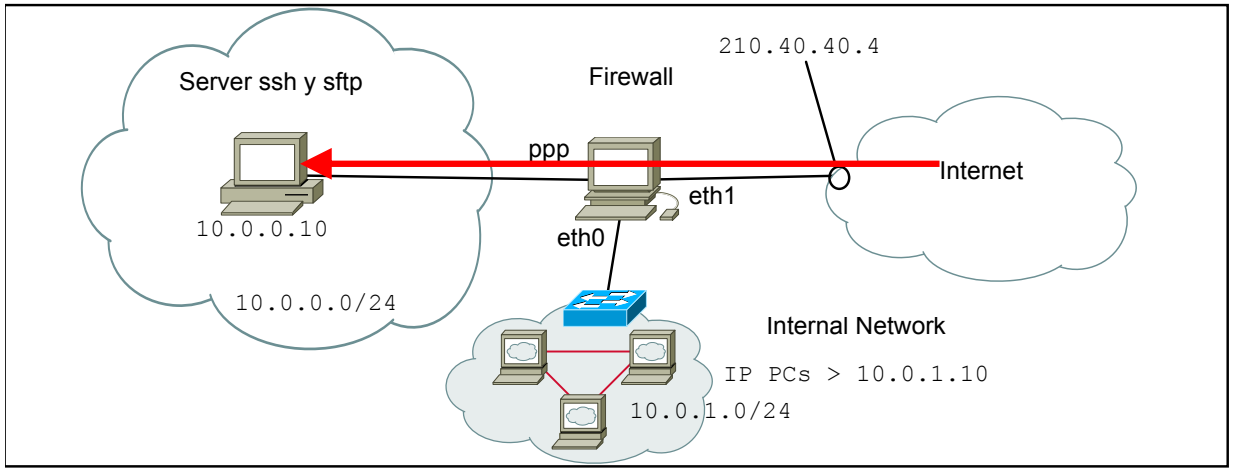
4.1.2 – iptables
Ejemplos



2) Configurar el firewall para que los hosts de Internet solo puedan acceder a los servicios ssh y sftp (puerto 22) del servidor 10.0.0.10



4.1.2 – iptables
Ejemplos

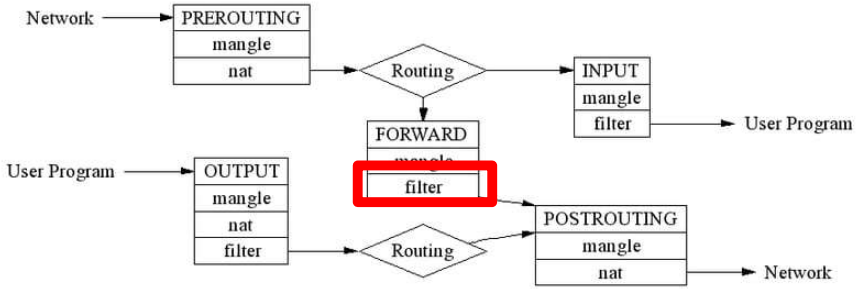


2) Configurar el firewall para que los hosts de Internet solo puedan acceder a los servicios ssh y sftp (puerto 22) del servidor 10.0.0.10

iptables –t filter –A FORWARD –i eth1 –o ppp –d 10.0.0.10 0.0.0.0 –dport 22 –j ACCEPT

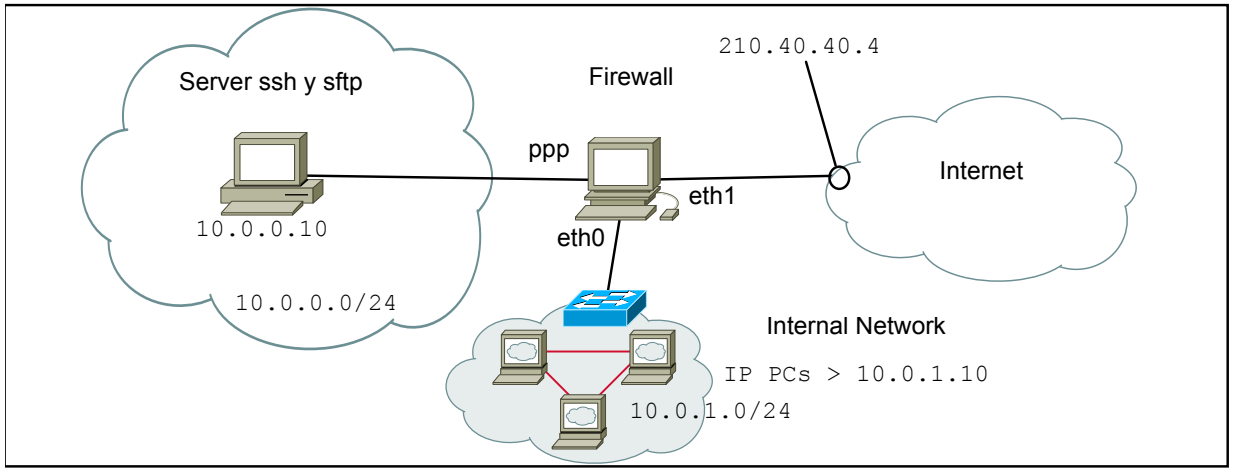
iptables –t filter –A FORWARD –i ppp –o eth1 –s 10.0.0.10 0.0.0.0 –sport 22
–state ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP



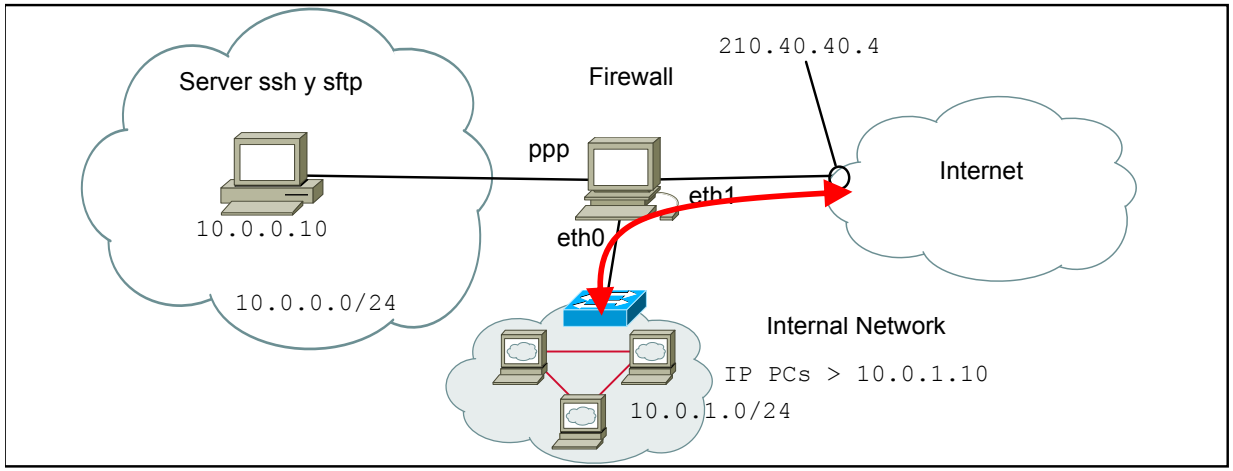
4.1.2 – iptables

Ejemplos

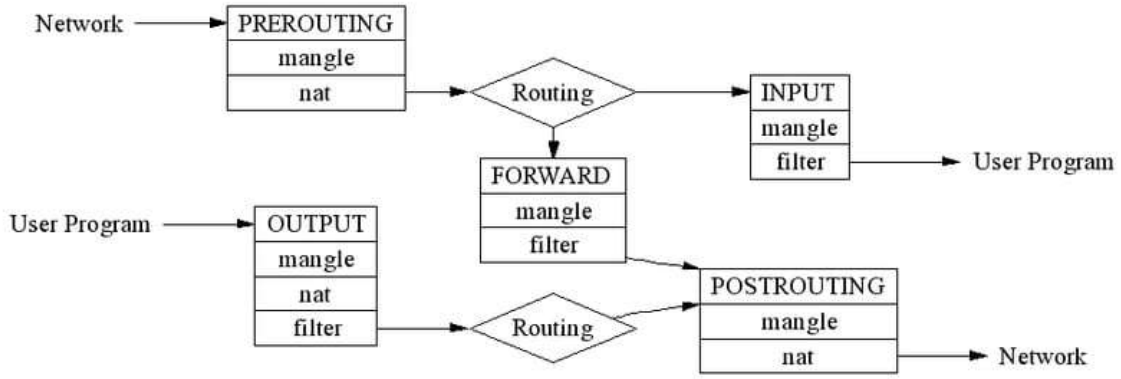


3) Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar

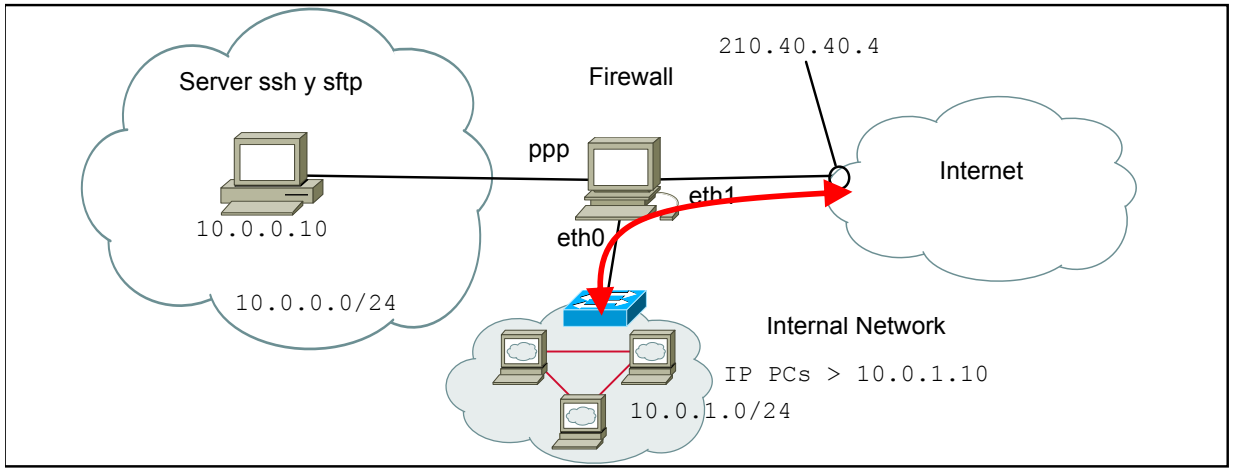
4.1.2 – iptables
Ejemplos



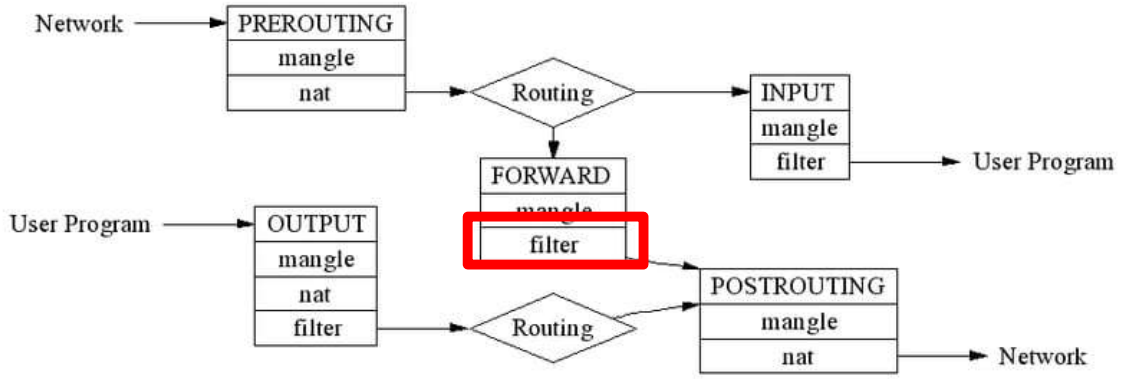
3) Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar



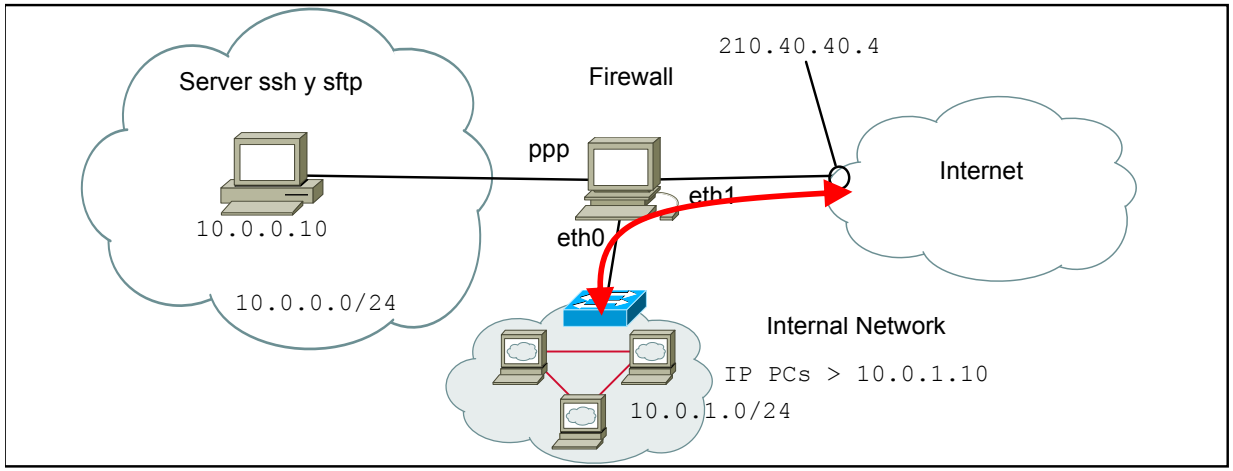
4.1.2 – iptables
Ejemplos



3) Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar



4.1.2 – iptables
Ejemplos

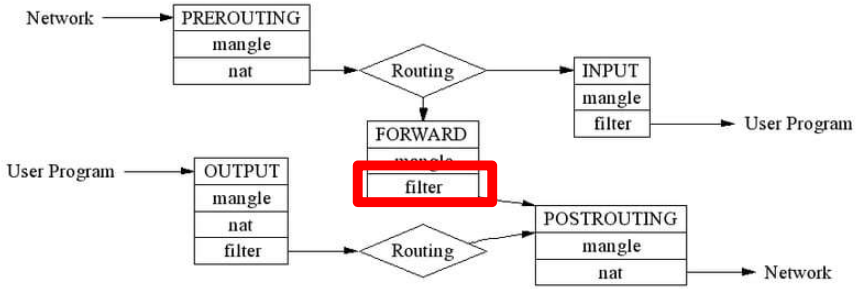


3) Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar

iptables –t filter –A FORWARD –i eth0 –o eth1 –s 10.0.1.0 0.0.0.255 –dport 80 –j ACCEPT

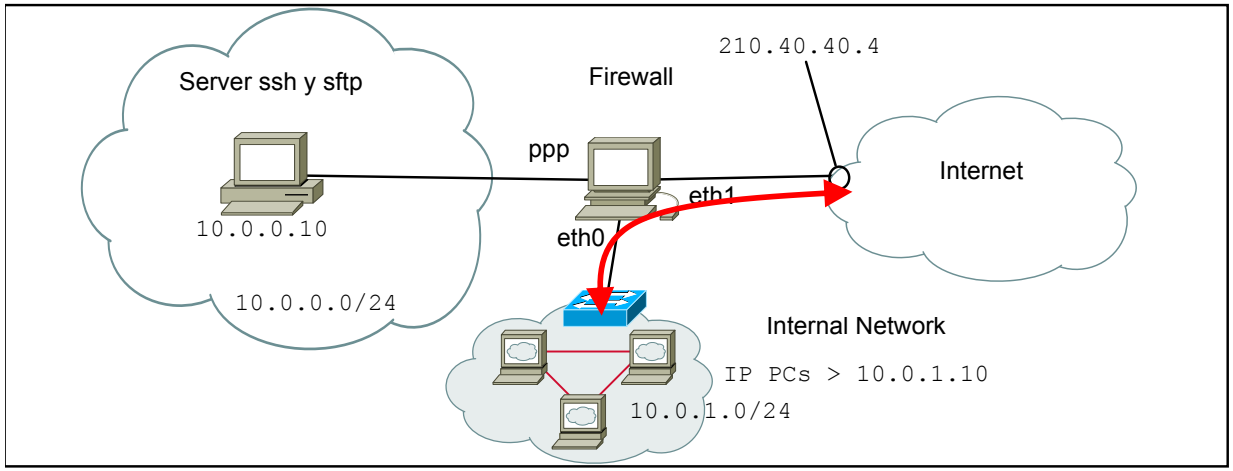
iptables –t filter –A FORWARD –i eth1 –o eth0 –d 10.0.1.0 0.0.0.255 –sport 80 –state ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP



4.1.2 – iptables

Ejemplos



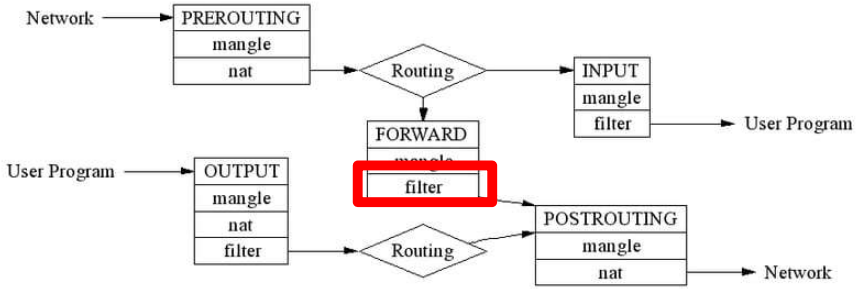
3) Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar

iptables –t filter –A FORWARD –i eth0 –o eth1 –s 10.0.1.0 0.0.0.255 –dport 80 –j ACCEPT

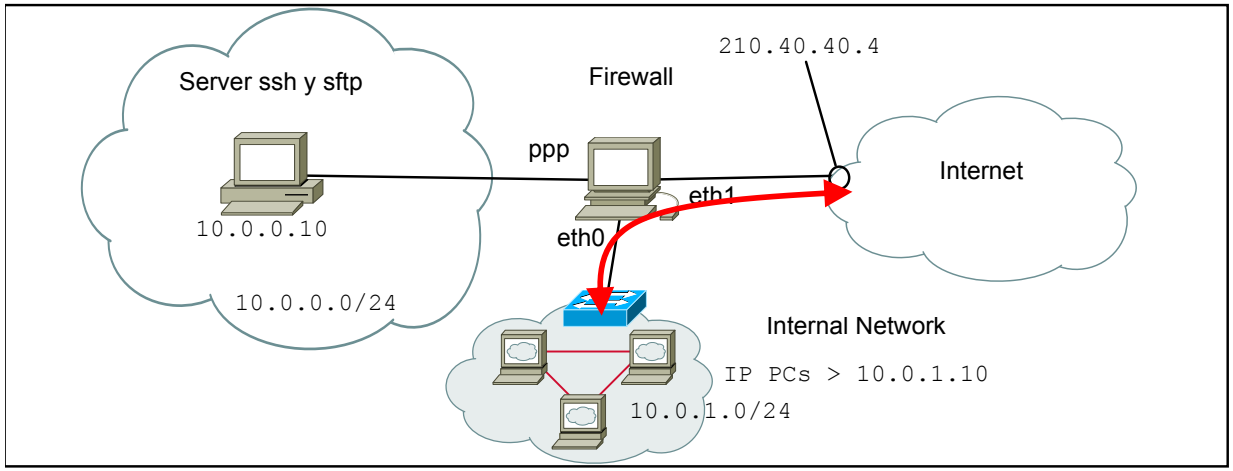
iptables –t filter –A FORWARD –i eth1 –o eth0 –d 10.0.1.0 0.0.0.255 –sport 80 –state ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP

Funciona? Falta algo?



4.1.2 – iptables
Ejemplos

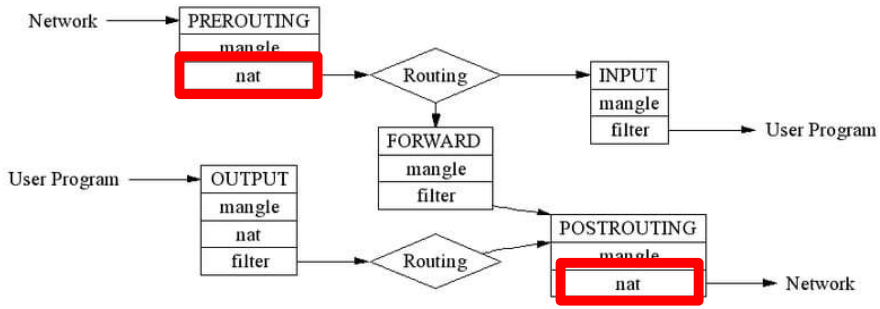


3) Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar

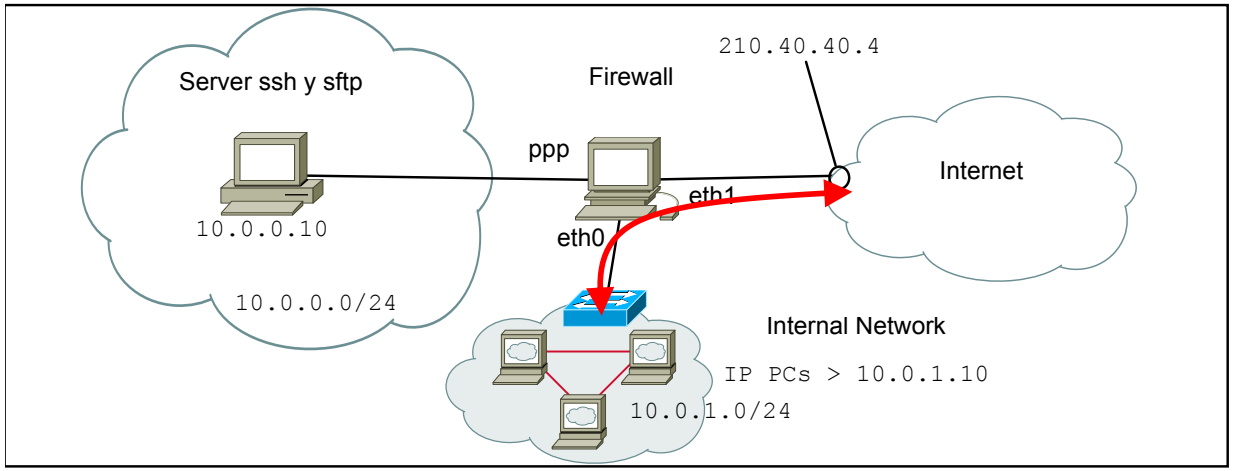
- Falta el NAT dinámico: suponemos se reserva el rango 210.40.40.10-210.40.40.40

iptables –t nat –A POSTROUTING –i eth0 –o eth1 –s 10.0.1.0 0.0.0.255 –j SNAT
--to-source 210.40.40.10-210.40.40.40

- En este caso, no hace falta poner la vuelta ya Internet solo podrá contestar (no puede empezar una comunicación)



4.1.2 – iptables
Ejemplos

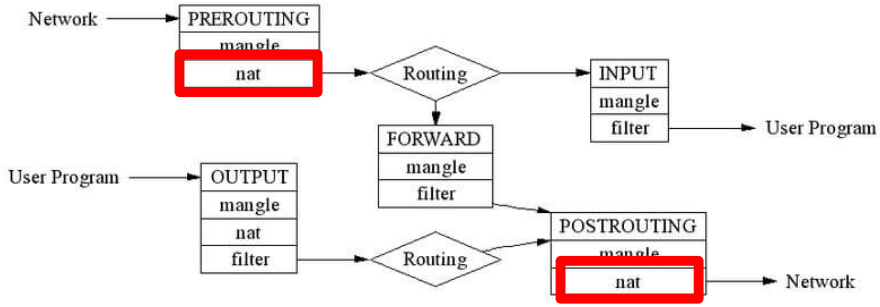


3) Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar

- Si fuera PAT

iptables –t nat –A POSTROUTING –i eth0 –o eth1 –s 10.0.1.0 0.0.0.255 –j MASQUERADE

- MASQUERADE indica la @IP de eth1

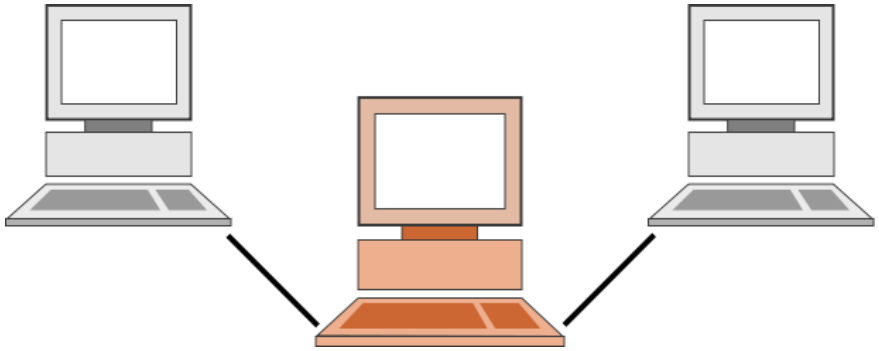


Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.1.3 - Filtrado a nivel de aplicación

- Un servidor proxy es un dispositivo que actúa de intermediario entre clientes que necesitan un servicio y los servidores que proporcionan este servicio



- Proceso
  - Un cliente se conecta al servidor proxy pidiendo un servicio ofrecido por otro servidor
  - El proxy evalúa la petición de acuerdo a sus reglas de filtrado
  - Si la petición es valida, el proxy pide el servicio al servidor y lo reenvía al cliente

4.1.3 - Filtrado a nivel de aplicación

Razones

- Para mantener las máquinas anónimas
- Para acelerar el acceso a los recursos (usando el almacenamiento en caché)
- Para aplicar la política de acceso a los servicios o contenidos de la red, por ejemplo, para bloquear sitios no deseados
- Para registrar / auditar el uso
- Para escanear contenido transmitido en busca de malware antes de la entrega
- Para escanear contenido saliente, por ejemplo para la protección de posibles fugas de datos
- Para eludir las restricciones regionales

4.1.3 - Filtrado a nivel de aplicación

Tipos de proxy y función

- Caching proxy servers
  - Conservan copias locales de los recursos solicitados con más frecuencia
  - Aceleran las solicitudes de servicio al recuperar el contenido guardado de una solicitud anterior

- Web proxy servers
  - Sirven como caché web
  - La mayoría de los programas proxy proporcionan una forma de denegar el acceso a las URL especificadas en una lista negra (filtrado de contenido)
  - Algunos proxies web también reformatean páginas web para un propósito específico o dispositivos particulares , como moviles y tablets

4.1.3 - Filtrado a nivel de aplicación

Tipos de proxy y función

- Anonymous proxy servers: intentan anonimizar el trafico web
  - **Open proxy**
    - No tienen control de acceso
    - El servidor web recibe solicitudes del servidor proxy de anonimato, y por lo tanto no recibe información sobre la dirección del usuario final
    - Tener en cuenta que las solicitudes no son anónimas para el servidor proxy de anonimato
  - **Close proxy**
    - Tienen control de acceso
    - Los usuarios autorizados deben iniciar sesión para obtener acceso a la web
    - El administrador de este proxy puede, por lo tanto, monitorizar el uso y rastrear los usuarios

4.1.3 - Filtrado a nivel de aplicación

Tipos de proxy y función

- Intercepting (transparent) proxy servers
  - Combinan un servidor proxy con un gateway o router (con capacidades NAT)
  - Las conexiones realizadas por los clientes son desviadas por el router a este proxy (sin que los usuarios se enteren, por eso transparente)
  - Se usan comúnmente en las empresas para interceptar el trafico de sus empleados y desviarlos a proxies internos a la empresa en lugar que a Internet
  - De esta forma, la empresa puede monitorizar los empleados y restringir el acceso a determinados sitios

4.1.3 - Filtrado a nivel de aplicación

Tipos de proxy y función

- Reverse proxy servers
  - Son servidores instalados en la vecindad de uno o más servidor web
  - Todo el tráfico proveniente de Internet y con destino a estos servidores pasa por el reverse proxy
  - Existen varias razones para instalar un reverse proxy:
    - Equilibrio de carga: las conexiones se distribuyen entre varios servidores
    - Seguridad: es una capa adicional de defensa y puede proteger contra algunos ataques específicos. Sin embargo, no proporciona ninguna protección contra los ataques a la aplicación web o al servicio en sí.
    - Servicios adicionales: compresión de datos, almacenamiento en caché de contenido estático

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.2 - Sistemas de detección de intrusos

Definiciones

- Intrusión
  - Es el acto de empujar o entrar en un lugar o estado sin invitación, derecho o bienvenida
- Detección de intrusión
  - Se refiere al acto de detectar una intrusión no autorizada en una red
  - Este acceso no autorizado, o intrusión, es un intento de comprometer, dañar, o manipular a otros dispositivos conectados a esta red
- Sistema de detección de intrusos (IDS)
  - Es el equivalente de alta tecnología de una alarma antirrobo
  - Una alarma antirrobo está configurada para monitorear puntos de acceso, actividades hostiles e intrusos

4.2 - Sistemas de detección de intrusos

Definiciones

- Sistema de detección de intrusos (IDS)
  - Herramienta especializada que sabe leer e interpretar el contenido de los archivos de registro (logs) de routers, firewalls, servidores y otros dispositivos de red
- Funcionamiento de un IDS
  - Mantiene una base de datos de de ataque conocidos
  - Compara los patrones de actividad o tráfico que ven en los logs que están monitoreando contra esta base de datos
  - De esta forma, saben reconocer cuándo se produce una coincidencia cercana entre un ataque conocido y un comportamiento actual o reciente en la red

4.2 - Sistemas de detección de intrusos
IDS vs antivirus

- Un IDS activa alarmas o toma varios tipos de acciones automáticas
  - Cierre de enlaces o servidores de Internet
  - Inspección de las trazas pasadas para identificar el patrón
  - Otros intentos activos para identificar a los atacantes y recopilar evidencia de sus actividades
- Por analogía, un IDS hace para una red lo que hace un antivirus para los archivos
  - Inspecciona el contenido del tráfico de la red para buscar, desviar y/o contrarrestar posibles ataques
  - Como un antivirus busca el contenido de archivos entrantes, archivos adjuntos de correo electrónico, etc. para buscar firmas de virus o posibles acciones maliciosas

4.2 - Sistemas de detección de intrusos
IDS vs Firewall

- ¿Puede servir un Firewall como IDS?
  - Se puede configurar un firewall para detectar ciertos tipos de intrusiones y activar una alerta si ocurre
  - Sin embargo, sin una **inspección profunda de paquetes** y reconocimiento de patrones entre paquetes, esto no es suficiente
- Inspección profunda de paquetes (Deep packet inspection, DPI)
  - Es una forma de filtrado de paquetes de red que examina la parte de los datos y posiblemente también de las cabeceras de un paquete cuando pasa por un punto de inspección
  - Se buscan incumplimiento de protocolo, virus, spam, intrusiones o criterios predefinidos
  - Y se decide si el paquete puede pasar, si se necesita encaminarse a un destino diferente, o se usa para recopilar información estadística

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.2.1 - Clasificación por funcionalidad

- Signature-based IDS
- Anomaly-based IDS

4.2.1 - Clasificación por funcionalidad

- Signature-based IDS
  - De manera similar al antivirus, los IDS basados en firmas detectan ataques al comparar la actividad de la red con una base de datos de ataques conocidos
  - Los IDS basados en firmas tienen una base de datos de ataque (por ejemplo, GET /etc/passwd)
  - Si una regla coincide, se activa una alerta. simple y efectivo
  - Problema: nuevos ataques no se pueden reconocer

- Anomaly-based IDS

4.2.1 - Clasificación por funcionalidad

- Signature-based IDS

- Anomaly-based IDS
  - Los IDS basados en anomalías crean un modelo de comportamiento del sistema "normal"
  - Cuando se detecta una desviación del modelo, se envía una alerta
  - Los IDS basados en anomalías clasifican las actividades de red como normales o anómalas
  - Como se puede definir un funcionamiento normal
    - Técnicas de IA: redes neuronales, reconocimiento de patrones, machine learning, sistemas fuzzy, …
    - Técnicas matemáticas: ecuaciones funcionales, análisis estadísticas, …

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.2.2 - Clasificación por arquitectura

- Network IDS
- Host IDS
- Distributed IDS

4.2.2 - Clasificación por arquitectura

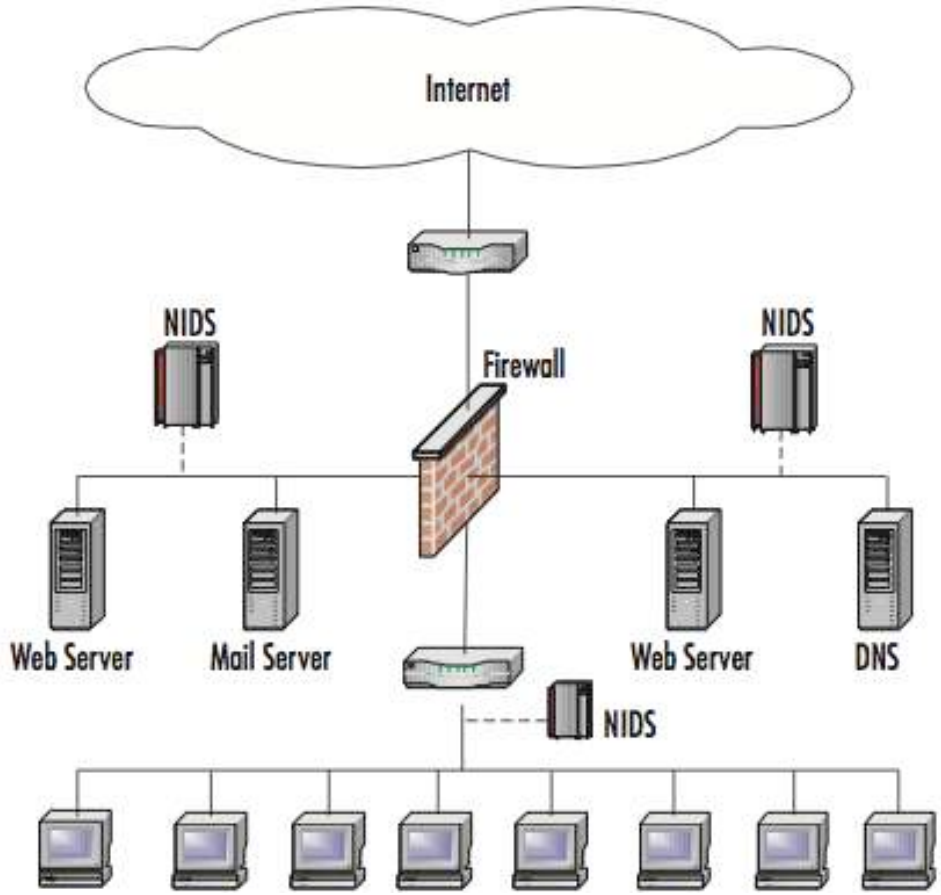
Network IDS

- NIDS monitorea el tráfico de red en segmentos de red o dispositivos de red particulares y analiza los protocolos de red, transporte y aplicación para identificar actividades sospechosas

- Al menos una de las interfaces de red en esta máquina funciona en modo promiscuo, capturando y analizando todos las tramas que pasan a través de él en busca de patrones indicativos de un ataque

4.2.2 - Clasificación por arquitectura

Network IDS - sensores



- Un sensor NIDS monitorea y analiza la actividad de la red en uno o más segmentos de la red.
- Los sensores pueden estar basados en hardware o en software
- Los sensores se pueden implementar en dos modos
  - En línea: el tráfico monitoreado debe pasar por sensor
  - Pasivo: monitorea una copia del tráfico de red real; ningún tráfico pasa a través del sensor. De esta forma, no necesita monitorizar en real-time

4.2.2 - Clasificación por arquitectura

Network IDS - funciones

- **Recopilación de información**: NIDS puede recopilar información sobre hosts y actividad de red para identificar usuarios, sistemas operativos, aplicaciones o características de red
- **Registro**: los NIDS realizan un log extenso de datos relacionados con eventos detectados. Esto puede usarse para confirmar la validez de las alertas, para investigar incidentes, etc.
- **Detección**: los NIDS utilizan una combinación de detección basada en firmas y detección basada en anomalías para realizar un análisis en profundidad
- **Prevención**: una vez que se activa una alerta, se pueden abortar otras conexiones similares

4.2.2 - Clasificación por arquitectura
Host IDS

- HIDS monitorea las características de un solo host y los eventos que ocurren en este host y busca alguna actividad sospechosa
- HIDS inspecciona el tráfico de red para el host, los registros del sistema, los procesos en ejecución, los accesos y modificaciones de archivos, los cambios de configuración del sistema y las aplicaciones, …
- HIDS tiene agentes instalados en los hosts o aplicaciones de interés: cada agente monitorea la actividad y transmite información a los servidores de administración
- Funciones principales: verificadores de integridad del sistema (SIV), monitores de archivos de registro (LFM) y sistemas de trampa (tipo honeypot*)

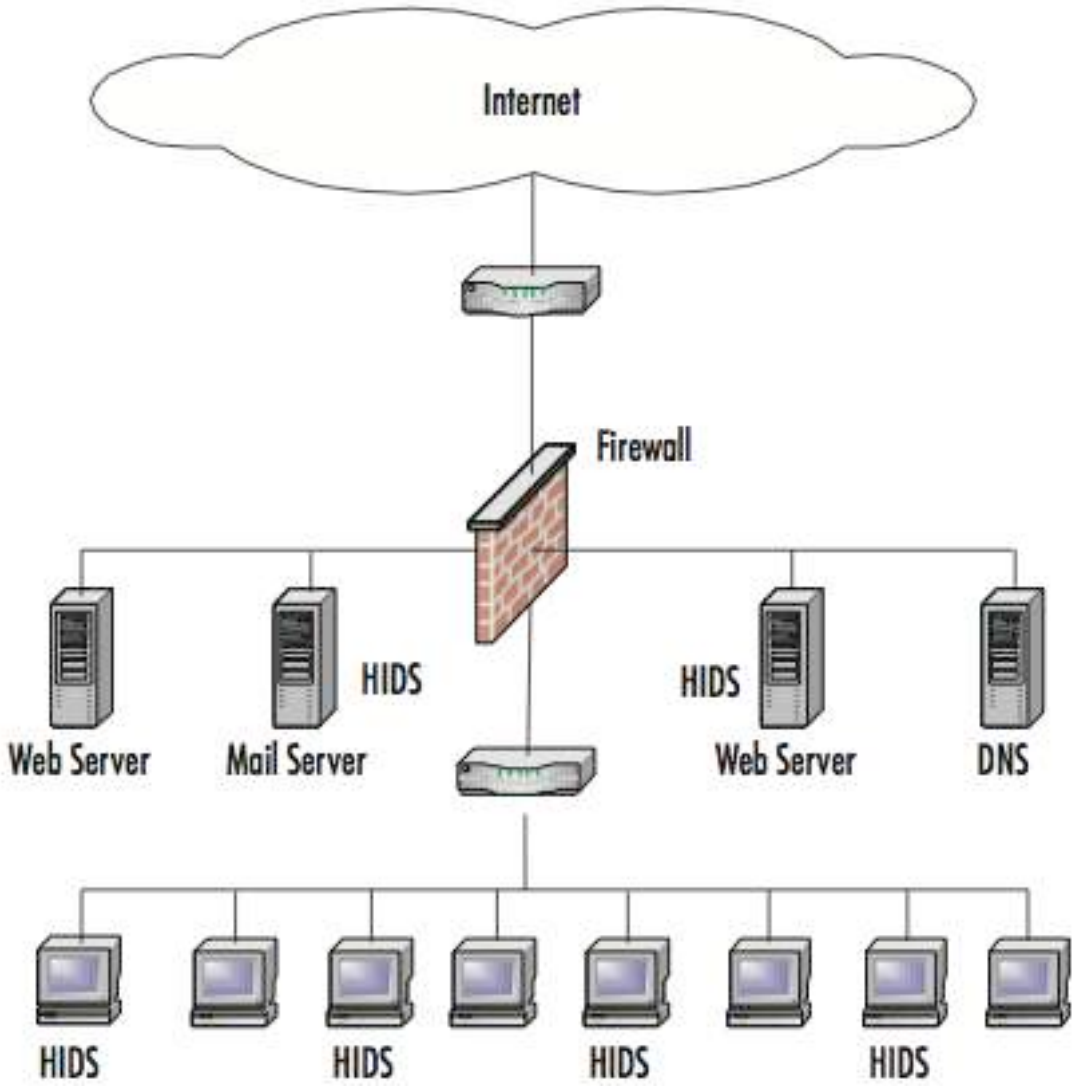
4.2.2 - Clasificación por arquitectura

Honeypot

- Sistema de trampa o señuelo
- Simula una vulnerabilidad de un host o red
- El atacante ataca este señuelo pensando haber encontrado un hueco para acceder a un sistema
- De esta forma
  - Se detecta el ataque antes que afecte a los sistemas reales
  - Se puede obtener información del atacante
  - Se puede ralentizar el ataque a los sistemas críticos
  - …

4.2.2 - Clasificación por arquitectura

Host IDS

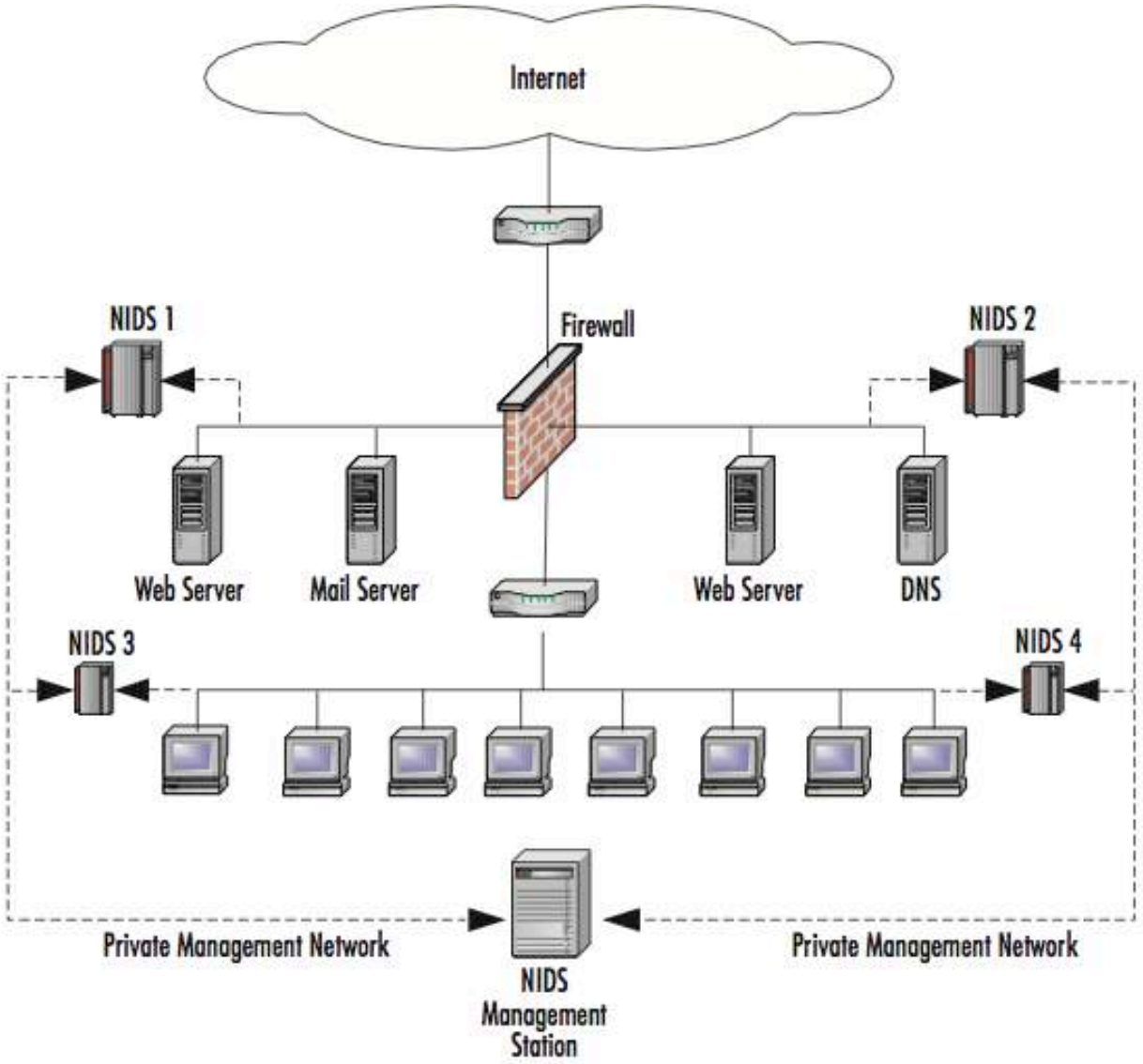


4.2.2 - Clasificación por arquitectura
Distributed IDS

- Los sensores se distribuyen por el sistema e informan a una estación de administración centralizada
- Los registros de ataque se cargan periódica o continuamente en la estación de administración y se pueden almacenar en una base de datos central
- Se pueden descargar nuevas firmas de ataque a los sensores según sea necesario
- Las reglas para cada sensor pueden adaptarse para satisfacer sus necesidades individuales
- Las alertas pueden enviarse a un sistema de mensajería ubicado en la estación de administración y usarse para notificar al administrador de IDS
- En un DIDS, los sensores individuales pueden ser NIDS, HIDS o una combinación de ambos

4.2.2 - Clasificación por arquitectura

Distributed IDS



Tema 4. Índice

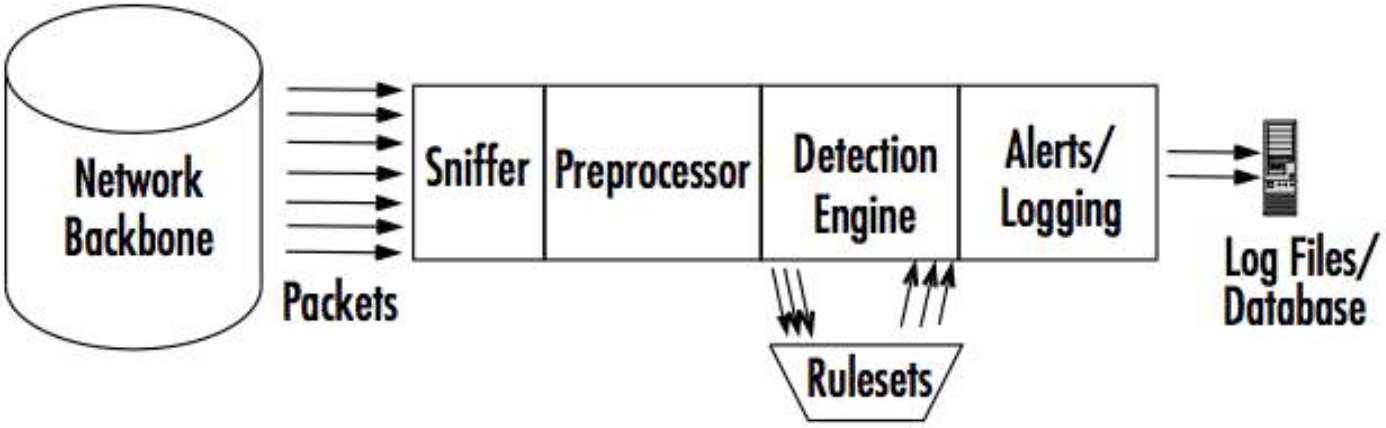
- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.2.3 - Snort

- Snort es una aplicación de packet sniffer / packet logger / NIDS
- El nombre viene porque es una aplicación que hace sniffer y más
- Propiedades principales
  - Usa una arquitectura NIDS y está basada en firmas (signature-based)
  - Está disponibles para múltiples OS
  - Usa un volcado de datos en hexdump
  - Muestra los diferentes tipos de paquetes de red en una misma forma
  - Es gratuito
  - Tiene una base de datos de ataques que se actualiza constantemente a través de internet

4.2.3 – Snort

Arquitectura



- Sniffer: captura los paquetes
- Preprocessor: determina el tipo de paquetes o comportamiento hay que tratar. Se usan plug-ins para detectar tipos (HTTP) o comportamientos específicos
- Detection engine: compara un paquete con las firmas (reglas), se hay una coincidencia, se envía al siguiente bloque
- Alerts/logging: se guardan registros o saltan alarmas según las reglas anteriores. El formato puede ser variado. Se pueden usar GUI para facilitar la lectura de los logs

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.3 - Seguridad punto a punto

- Al principio, Internet era una red académica usada para intercambiarse datos entre personas de confianza
- La seguridad no era un problema

- Hoy en día, la seguridad si es un problema ya que Internet se usa para acceder a servicios bancarios, compras on-line, etc.
- Por lo tanto, ya que la arquitectura TCP/IP no nació con seguridad, hoy en día se necesita algún método seguro para proteger la comunicaciones de red

4.3 - Seguridad punto a punto

- Típicamente se usan métodos distintos según el nivel que se quiere proteger
- Capa de aplicación
  - Los datos generados de las aplicaciones se pueden encriptar antes de ser encapsulados en la capa inferior
  - Cada aplicación puede definir el método a usar
- Capa de transporte
  - Se puede añadir TLS (Transport Layer Security) a la conexión entre dos extremos
  - TLS permite el uso de certificados X.509 para identificar los extremos
  - TLS usa criptografía simétrica para encriptar toda la información de esta conexión (la clave secreta se genera exclusivamente para esta sesión y se decide usando algunos de los métodos vistos en el Tema 3)

4.3 - Seguridad punto a punto

- Típicamente se usan métodos distintos según el nivel que se quiere proteger
- Capa de red
  - Todos los datagramas independientemente de la aplicación y de la capa de transporte, deben encaminarse de forma segura en la red
  - Uno de los protocolos más usados es **IPsec**
- Capa de enlace
  - Se puede aplicar una seguridad especifica dependiente de la tecnología de nivel 2
  - Eso puede incluir autentificación antes de establecer la conexión física de un dispositivo a la red
  - Por ejemplo, contraseña para conectarse a una WiFi o para conectarse con el router ADSL al router del ISP

4.3 - Seguridad punto a punto
Internet Protocol Security

- RFC 4301 y 4309
- Es el protocolo de seguridad de la capa de red más utilizado
- Es un conjunto de protocolos cuya función es asegurar las comunicaciones sobre IP autenticando y/o cifrando cada paquete
- Las funciones principales son:
  (1) asegurar el flujo de paquetes
  (2) garantizar la autenticación mutua
  (3) establecer parámetros criptográficos

4.3 - Seguridad punto a punto
Virtual Private Network

- El uso más común de las implementaciones de IPsec es proporcionar servicios de redes privadas virtuales (VPN)
- Una VPN es una red virtual, construida sobre redes físicas existentes, que puede proporcionar un mecanismo de comunicación seguro para los datos entre redes IP
- Generalmente, proporciona un canal seguro entre redes que se comunican usando una red pública e insegura, i.e., Internet

Tema 4. Índice

- Firewalls
  - Topologías de firewall
  - Reglas de filtrado
  - Filtrado a nivel de aplicación
- Sistemas de detección de intrusos
  - Clasificación por funcionalidad
  - Clasificación por arquitectura
  - Snort
- Seguridad punto a punto
  - Arquitecturas VPN
  - Fundamentos de IPsec
  - Ejemplos prácticos

4.3.1 - Virtual Private Network

- Las VPNs pueden usar tanto cifrado simétrico como asimétrico

- Cifrado simétrico: se usa una misma clave para cifrar y descifrar los paquetes. La ventaja es que es relativamente eficiente (por ejemplo, AES)

- Cifrado asimétrico: se usan claves diferentes para cifrar y descifrar paquetes o para firmar digitalmente y verificar la firma. Se usa típicamente para verificar la identidad de los extremos

4.3.1 – VPN

Gateway-to-Gateway architecture

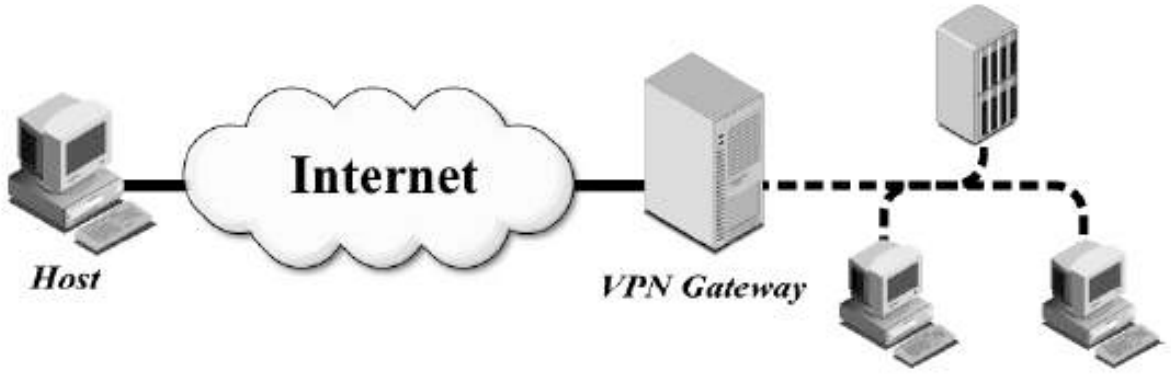
- Esta arquitectura proporciona comunicaciones de red seguras entre dos sistemas mediante el establecimiento de una conexión VPN entre los dos routers (gateways) de acceso de cada sistema

- El encaminamiento en cada sistema está configurado de modo que los paquetes que van de un sistema al otro, se encaminan a través de esta conexión IPsec (túnel IPsec)
- Esta es la VPN más fácil de implementar, ya que todos los paquetes de cualquier hosts de cualquiera de los dos sistema se encaminan igual

4.3.1 – VPN

Host-to-Gateway architecture

- Este modelo se utiliza para proporcionar acceso remoto seguro desde una red externa a los servicios internos de un sistema/empresa
- La empresa implementa un gateway VPN en su sistema; cada usuario establece una conexión VPN entre su host y este gateway

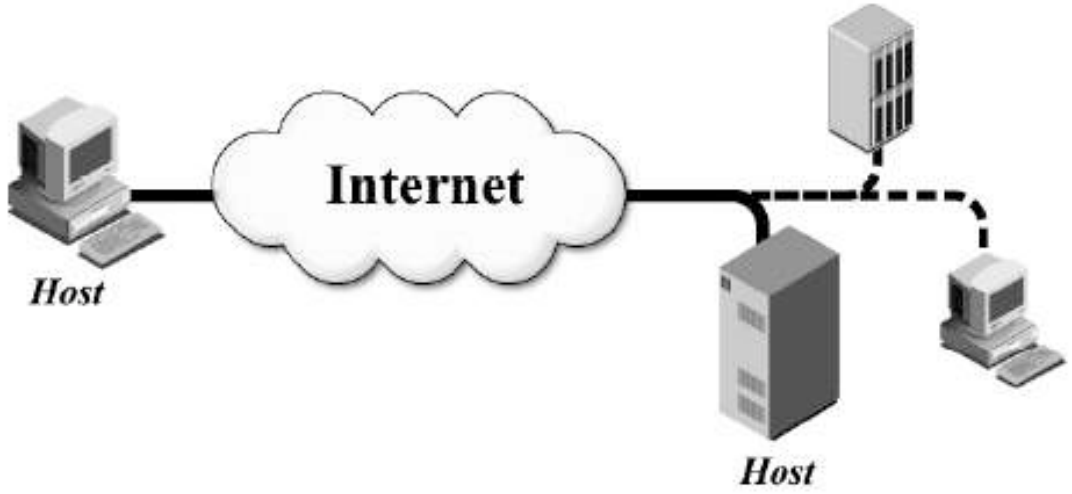


- La conexión VPN la establece el usuario cuando la necesita
- Típicamente el gateway necesita una autentificación del usuario (por ejemplo usuario/contraseña)
- Es un modelo más complejo de gestionar y el gateway puede que necesite mantener un número elevado de conexiones VPN

4.3.1 – VPN

Host-to-Host architecture

- Generalmente, se usa para necesidades especiales, como cuando un administrador de sistema necesite acceder remotamente a un solo servidor



- Este modelo es el único que proporciona seguridad extremo a extremo, i.e. los paquetes se quedan cifrado durante todo el recorrido
- Esto puede ser un problema, ya que los firewalls, IDS y otros dispositivos no pueden inspeccionar los paquetes, lo que puede provocar algo de inseguridad en la red interna
- Este tipo de VPN se suele bloquear y solo se permite a usuario especificos

Tema 4. Índice

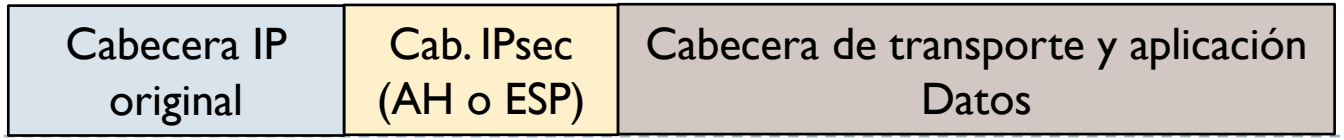
4.3.2 - Fundamentos de IPsec

- IPsec es una **colección de protocolos** que ayudan a proteger las comunicaciones a través de redes IP
- Dependiendo de su implementación, puede proporcionar cualquier combinación de los siguientes tipos de protección:
  - **Confidencialidad**: aseguro que el datagrama no puede ser leído por alguien no autorizado → IPsec usa cifrado y clave secreta
  - **Integridad**: puede determinar si un paquete ha sido modificado durante la transmisión → IPsec usa un checksum criptográfico llamado Message Authentification Code (MAC)
  - **Autentificación**: cada extremo de la comunicación debe identificarse de manera que los paquetes se están enviando entre los extremos correctos
  - **Control de acceso**: los extremos pueden filtrar para asegurar que solo los usuarios autorizados IPsec pueden acceder a recursos particulares de la red
  - **Protección de repetición**: se asegura que los mismos datos no se entregan varias veces y que nadie pueda introducir datos nuevos
  - **Protección de análisis de tráfico**: una persona que monitoriza el tráfico no puede saber qué partes se están comunicando, con qué frecuencia se producen las comunicaciones o cuántos datos se intercambian; pueden contar el número de paquetes

4.3.2 – Fundamentos de IPsec
Modos de funcionamiento

- Tunnel mode
  - Se añade una cabecera IP adicional que se pone delante de la cabecera de cada datagrama
  - Se añade entre las dos cabecera, una cabecera IPsec (el contenido de esta cabecera depende del protocolo usado)
  - Se usa generalmente para VPN Gw-t-Gw

- Transport mode
  - No se añade ninguna cabecera adicional
  - Se añade entre las dos cabecera, una cabecera IPsec (el contenido de esta cabecera depende del protocolo usado)
  - Se usa generalmente para VPN H-t-H



4.3.2 – Fundamentos de IPsec

Protocolos principales

- Authentication Header (AH)
  - Proporciona integridad, autenticación y no repudio si se eligen los algoritmos criptográficos apropiados
- Encapsulating Security Payload (ESP)
  - Proporciona confidencialidad y la opción de autenticación y protección de integridad
- Security Association (SA)
  - Por ejemplo, Internet Key Exchange (IKE)
  - Negocia, crea y gestiona la conexión segura entre extremos
  - Muy flexible ya que emplea una negociación de parámetros de seguridad muy abierta y variada

4.3.2 – Fundamentos de IPsec

Authentication Header (AH)

- AH proporciona protección de integridad para cabeceras IP y datos, así como autenticación de usuario
- Opcionalmente, puede proporcionar protección de reproducción y protección de acceso
- En cambio, no puede cifrar ninguna parte de los paquetes

- Se puede usar en Transport o Tunnel mode

4.3.2 – Fundamentos de IPsec

AH - problemas

- Campos con valores dinámicos
  - Hay campos de la cabecera IP que se alteran durante el camino de un extremo al otro, por ejemplo el TTL o el checksum
  - Por lo tanto, el extremo que recibe este paquete le daría un Hash diferente que el origen
  - Para evitar este problema, el Hash se calcula excluyendo estos campos
- Si se usa un NAT en el medio
  - Las @IP origen y destino hay que incluirlas en el calculo del Hash porque deben hacer parte de la integridad
  - Pero si hay que modificar una @IP debido a un NAT, evaluar la coincidencia del Hash origen y destino daría un error
  - Para estos casos, se debe usar un NAT particular llamado NAT transversal

4.3.2 – Fundamentos de IPsec
Encapsulating Security Payload (ESP)

- Puede proporcionar autenticidad, cifrado e integridad
- Se pueden deshabilitar algunas de estas funciones:
  - Solo cifrado
  - Solo integridad
  - Cifrado e integridad
- Al contrario que con AH, la cabecera del paquete IP no está protegida por ESP
  - Por lo tanto, los problemas anteriores no se presentan con ESP
  - En Tunnel model, la cabecera IP interna pero si que está protegida por el cifrado y la integridad

4.3.2 – Fundamentos de IPsec

Security Association

- Para poder empezar la transmisión de paquetes IPsec, hay que previamente crear una asociación de seguridad
- Los dos extremos que se comunican deben establecer parámetros de seguridad compartidos, como algoritmos y claves
- Una vez establecido, se asocia a esta conexión un identificador Security Parameter Index (SPI) que luego se usará en las cabeceras IPsec de los paquetes para identificar la conexión
- Existen varios protocolos para gestionar esta asociación, como por ejemplo Internet Key Exchange (IKE) y Kerberized Internet Negotiation of Keys (KINK)

4.3.2 – Fundamentos de IPsec
Parámetros SA

- Qué algoritmo se usa para cifrar el paquete IP, por ejemplo AES o DES
- Qué función hash se usa para garantizar la integridad de los datos, como MD5 o SHA
- La vida útil de esta asociación
- La clave secreta, por ejemplo usando Diffie-Helmann
- Qué algoritmo de autentificación usar
  - Una clave previamente conocida por ambos extremos (pre-shared key)
  - Intercambio de mensajes aleatorios cifrados con claves públicas y descifrado con la privada (cifrado asimétrico)
  - Certificados de clave pública emitidos por una CA
- Una vez creada la SA, hay que establecer que se va a usar IPsec y negociar los parámetros
  - Si se usa AH o ESP o ambos
  - Si se usa tunnel o transport mode

Tema 4. Índice

4.3.3 – Ejemplos
ESP en VPN Gw-to-Gw

- Establecer una conexión IPsec entre el gateway G_A y G_B que proporcione cifrado pero no integridad completa sobre todo el datagrama (cabecera externa excluida)
- Al principio, hay que crear la asociación segura (SA) y la conexión IPsec:

1) El usuario H_A envía un datagrama IP al usuario H_B
2) La red A envía el datagrama a G_A
3) G_A recibe el datagrama, altera si necesario (NAT) la @IP origen de H_A
4) G_A inicia una negociación con G_B para definir los parámetros de seguridad (entre otros, autentificación, integridad del payload y cifrado) y crear la SA con identificador SPI
5) G_A usa los parámetros configurados en la SA para negociar la conexión IPsec. En este caso se usa ESP en tunnel mode

4.3.3 – Ejemplos
ESP en VPN Gw-to-Gw

Una vez creada la conexión IPsec, se puede enviar el datagrama

a) G_A cifra el datagrama
b) G_A añade una cabecera ESP con la información de integridad y el identificador SPI
c) G_A añade una nueva cabecera IP (@IP origen G_A e @IP destino G_B)
d) G_A envía el paquete a G_B
e) G_B recibe el paquete y usa el SPI de la cabecera ESP para reconocer la conexión segura
f) G_B quita la cabecera IP adicional, comprueba la integridad del paquete y descifra el paquete original
g) G_B envía el paquete a H_B

4.3.3 – Ejemplos
AH en VPN H-to-H

- Establecer una conexión IPsec entre el host H_A y el host H_B con integridad y autentificación
- Al principio, hay que crear la asociación segura (SA) y la conexión IPsec:

1) El usuario H_A inicia una negociación con H_B para definir los parámetros de seguridad (entre otros, autentificación e integridad) y crear la SA con identificador SPI
2) H_A usa los parámetros configurados en la SA para negociar la conexión IPsec. En este caso se usa AH en transport mode

4.3.3 – Ejemplos
AH en VPN H-to-H

Una vez creada la conexión IPsec, se puede enviar el datagrama

a) H_A añade una cabecera AH al payload que viene de la capa de transporte o de aplicación con la información de integridad y el identificador SPI
b) H_A añade la cabecera IP (@IP origen H_A e @IP destino H_B), no se usa cifrado con AH
c) H_A envía el paquete a H_B
d) H_B recibe el paquete y usa el SPI de la cabecera AH para reconocer la conexión segura
e) H_B comprueba la integridad del paquete
f) H_B quita la cabecera IP y AH y envía el payload a la capa superior

Seguretat Informàtica (SI)

Tema 4. Seguridad en la red

Davide Careglio

Fuentes: Jordi Nin, "Network Security", Computer Security, 2014