UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

FIB

Tecnologies de Xarxes de Computadors

Tema 2. Seguridad en las redes

Davide Careglio

Temario

- Tema 1. Introducción
- Tema 2. Seguridad en redes
- Tema 3. Redes troncales
- Tema 4. QoS
- Tema 5. Redes de acceso cableadas

Temario

- Tema 1. Introducción
- Tema 2. Seguridad en redes
- Tema 3. Redes troncales
- Tema 4. QoS
- Tema 5. Redes de acceso cableadas

Tema 2. Seguridad en las redes

FIB

Índice

- Introducción
- Firewalls
  - Arquitecturas
  - Tecnologías
  - Reglas de filtrado
- Seguridad en IP
  - Introducción y usos
  - Arquitecturas VPN
  - Familia IPsec
- Sistemas de detección de intrusos (IDS)
  - Funcionalidades y arquitecturas
  - Tecnologías

Tema 2. Seguridad en las redes

FIB

Índice

- Introducción
- Firewalls
  - Arquitecturas
  - Tecnologías
  - Reglas de filtrado
- Seguridad en IP
  - Introducción y usos
  - Arquitecturas VPN
  - Familia IPsec
- Sistemas de detección de intrusos (IDS)
  - Funcionalidades y arquitecturas
  - Tecnologías

2.1 – Introducción

FIB

Seguridad en Internet

- 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, información personal, etc.
- Por lo tanto, ya que la arquitectura TCP/IP no nació con seguridad, hoy en día se necesita algún método seguro para proteger las comunicaciones

2.1 – Introducción

FIB

- Típicamente se usan métodos distintos según el nivel que se quiere proteger
- Capa de aplicación
  - Las aplicaciones pueden por ejemplo
    - cifrar los datos antes de ser encapsulados en la capa inferior
    - autenticar los usuarios
    - controlar la integridad de los datos
  - 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 usa criptografía hibrida
  - TLS permite el uso de certificados para identificar los extremos

2.1 – Introducción

FIB

Seguridad en Internet

- 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
  - Se pueden usar protocolos y dispositivos específicos (hardware y/o software) de protección y control de acceso (**por ejemplo?**)
- Capa de enlace
  - Se puede aplicar una seguridad especifica dependiente de la tecnología de nivel 2
  - Eso puede incluir por ejemplo
    - Autenticación antes de aceptar un nuevo dispositivo en una red (p. e., contraseña para conectarse a una WiFi o del router domestico al ISP)
    - Aplicar una whitelist de direcciones MAC

2.1 – Introducción

FIB

- 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
  - Se pueden usar protocolos y dispositivos específicos (hardware y/o software) de protección y control de acceso (**por ejemplo?**)
- Capa de enlace
  - Se puede aplicar una seguridad especifica dependiente de la tecnología de nivel 2
  - Eso puede incluir por ejemplo
    - Autenticación antes de aceptar un nuevo dispositivo en una red (p. e., contraseña para conectarse a una WiFi o del router domestico al ISP)
    - Aplicar una whitelist de direcciones MAC

Tema 2. Seguridad en las redes

FIB

Índice

- Introducción
- Firewalls
  - Arquitecturas
  - Tecnologías
  - Reglas de filtrado
- Seguridad en IP
  - Introducción y usos
  - Arquitecturas VPN
  - Familia IPsec
- Sistemas de detección de intrusos (IDS)
  - Arquitecturas
  - Tecnologías

2.1.1 – Introducción a Firewall

FIB

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
  - Cuando hay que añadir un segundo nivel de control interno

2.1.1 – Introducción a Firewall

FIB

Terminología

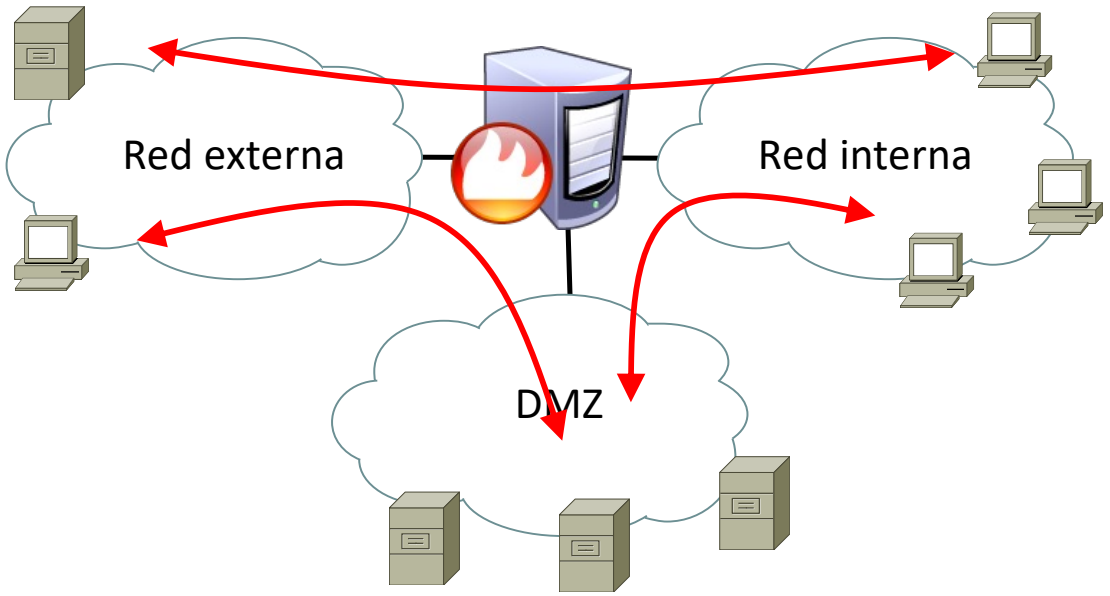
- DeMilitarised Zone (DMZ)
  - Parte de una infraestructura interna a una organización donde se instalan los servidores accesibles desde Internet, es decir es una zona de acceso público
  - Típicamente servidores de correo, DNS, web, de almacenamiento, etc.
- Red interna
  - Zona idealmente segura
  - Se instalan todas aquellas maquinas de acceso privado
- Red externa
  - Zona externa a la organización
  - Generalmente Internet

2.1.1 – Introducción a Firewall

FIB

Esquema

- Es una barrera que proporciona un **control de acceso** para toda la información que cruza este Firewall en cualquier dirección



2.1.1 – Introducción a Firewall

FIB

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

2.1.1 – Introducción a Firewall

FIB

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
- Si no se controla, 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

2.1.1 – Introducción a Firewall

FIB

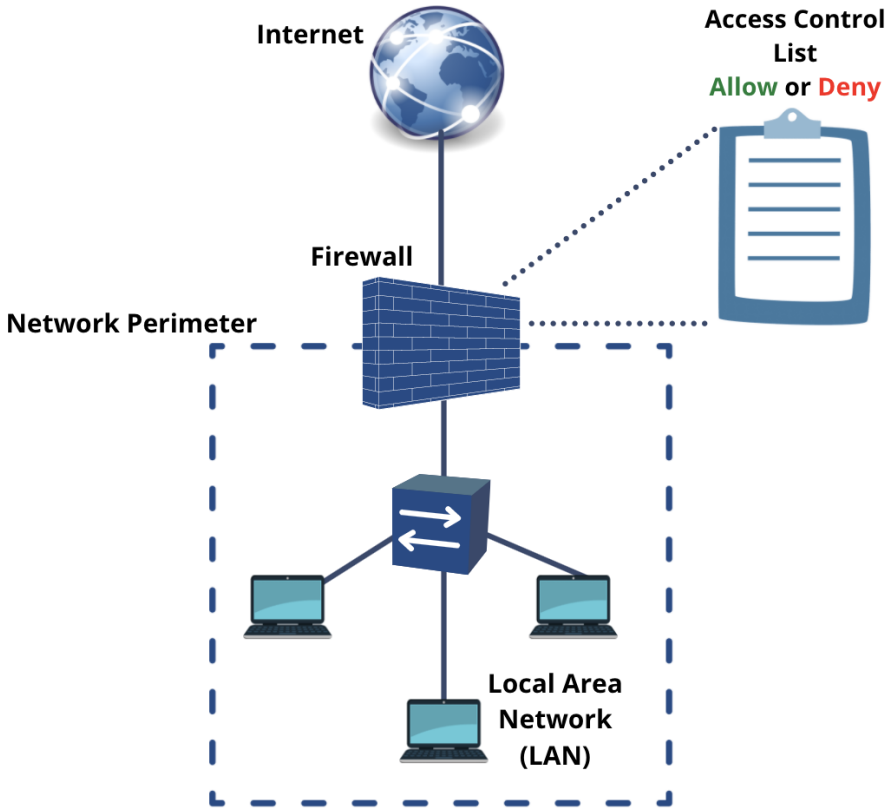
Diferentes tipos

- Firewall a nivel de paquetes (packet-filtering Firewall)
- Firewall a nivel de circuito (circuit-level Gateway)
- Firewall a nivel de aplicación (application-level Gateway)
- Firewall con inspección de estados (Stateful inspection Firewall)
- Firewall de próxima generación (Next generation Firewall)

2.1.1 – Introducción a Firewall

Firewall a nivel de paquetes

- Compara cada paquete recibido con un conjunto de criterios establecidos (p.e., lista de acceso), como
  - Direcciones IP
  - El tipo de paquete
  - Los puertos
  - Otros aspectos de las cabeceras de los protocolos 3 y 4
- Si no cumple con todos los criterios, el paquete es rechazado



Fuente imagen: https://digital.com/best-vpn-services/what-are-the-type-of-firewalls/

2.1.1 – Introducción a Firewall

FIB

Firewall a nivel de paquetes

- Ventajas
  - Un solo dispositivo puede filtrar el tráfico de toda la red
  - Extremadamente rápido y eficiente en el escaneo del tráfico
  - Económico y uso muy limitado de recursos
  - Las reglas pueden ser extremadamente intuitivas y fáciles de configurar
  - Más transparentes para los usuarios legítimos ya que los usuarios obtienen el acceso requerido con poca interferencia del firewall
- Desventajas
  - No controla el contenido (datos) de los paquetes
  - Puede ser inviable configurar reglas complejas
  - Un atacante podría descubrir las reglas de filtrado y hacerse pasar por un usuario legitimo modificando los campos de sus paquetes

2.1.1 – Introducción a Firewall

FIB

Firewall a nivel de circuito

- Se implementan en la capa de sesión del modelo OSI y monitorean sesiones como el three-way handshaking del TCP para ver si una conexión solicitada es legítima o no
  - Inspecciona si el host remoto se considera confiable
  - No inspeccionan pero los paquetes
- El Gateway no permite conexiones de un extremo a otro, en cambio configura dos conexiones
  - Una entre el Gateway y el host remoto
  - La otra entre el Gateway y el host interno
- Una vez establecidas las dos conexiones, el Gateway transmite paquetes de una conexión a otra sin examinarlos

2.1.1 – Introducción a Firewall

FIB

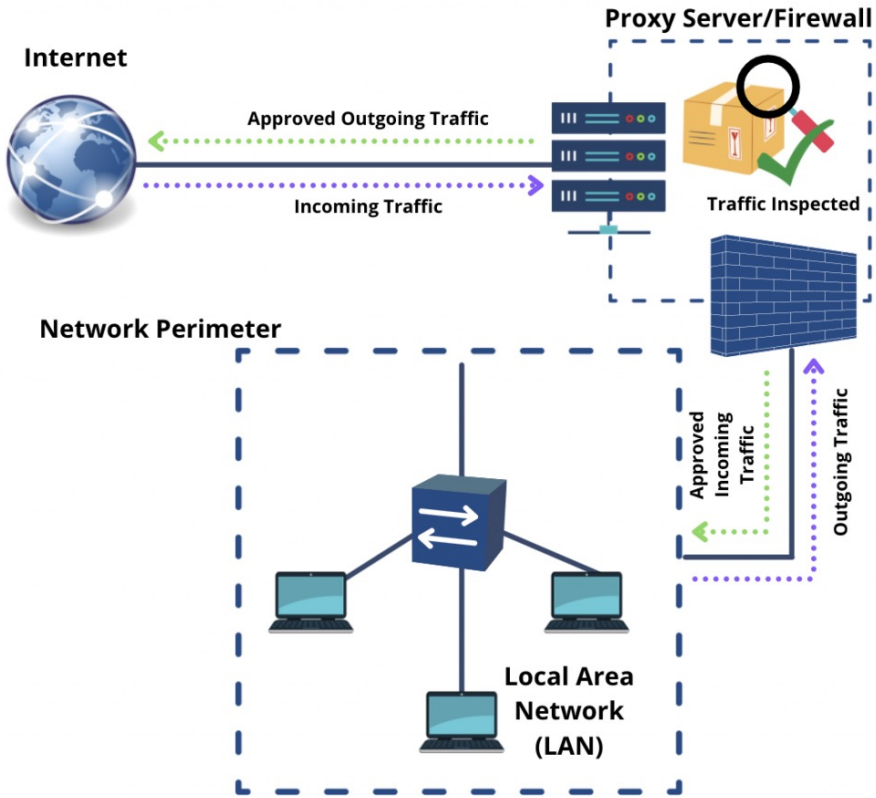
Firewall a nivel de circuito

- Ventajas
  - Sólo procesa las transacciones solicitadas
  - Fácil de configurar y administrar
  - Bajo costo y mínimo impacto en la experiencia del usuario final
- Desventajas
  - Se necesita usar en combinación con otros firewall para tener una protección completa
  - No se inspecciona el contenido de los paquetes
  - No hay conexión extremo-a-extremo segura

2.1.1 – Introducción a Firewall

Firewall a nivel de aplicación

- Funciona en la capa de aplicación del modelo OSI y proporciona protección para los protocolos de esta capa
- Se suelen llamar también Proxy Firewall
- Su función es bloquear o reenviar paquetes en función de la información de las capas de aplicación
  - Actúa como un retransmisor del tráfico a nivel de aplicación
  - Generalmente se añade una fase de autenticación



Fuente imagen: https://digital.com/best-vpn-services/what-are-the-type-of-firewalls/

2.1.1 – Introducción a Firewall

FIB

Firewall a nivel de aplicación

- Ventajas
  - Examina todas las comunicaciones entre hosts externas e internos, verificando no solo la dirección, el puerto y la información del encabezado TCP, sino también el contenido en sí antes de permitir que el tráfico pase
  - Proporciona un control muy fine sobre cualquier información (p.e., puede bloquear el acceso a determinadas páginas de un servidor web)
  - Puede proporcionar anonimato
- Desventajas
  - Puede ser muy costoso económicamente
  - Puede no funcionar para todas las posibles aplicaciones
  - Las prestaciones se ven muy afectadas
  - Su configuración puede resultar muy compleja

2.1.1 – Introducción a Firewall

FIB

Firewall con inspección de estados

- Examina cada paquete
- Y también realiza un seguimiento de si ese paquete es parte o no de una sesión TCP u otra sesión de red permitida y establecida
  - Inspecciona el tráfico de paquetes
  - Registra los datos relevantes (dirección de origen, tipo de paquete, destino, etc.)
  - Compara el tráfico futuro con ese registro para validarlo
- Esto ofrece más seguridad que el filtrado de paquetes o circuitos

2.1.1 – Introducción a Firewall

FIB

Firewall con inspección de estados

- Ventajas
  - Supervisa toda la sesión para conocer el estado de la conexión
  - Ofrece un alto grado de control sobre qué contenido entra o sale de la red
  - Ofrece capacidades de registro considerables
  - Pueden ser efectivos contra ataques de tipo DDoS
- Desventajas
  - Consuma muchos recursos
  - Interfiere con la velocidad de las comunicaciones
  - Más caro que otras opciones de firewall
  - No suele proporcionar autenticación para validar que los hosts orígenes no sean falsificadas

2.1.1 – Introducción a Firewall

FIB

Firewall de próxima generación

- Combina
  - Filtrado de paquetes
  - Inspección de estado
  - Inspección profunda de paquetes (DPI)
  - Y otros sistemas de seguridad de red, como IDS/IPS, filtrado de malware y antivirus

2.1.1 – Introducción a Firewall

FIB

Firewall de próxima generación

- Ventajas
  - Combina DPI con filtrado de malware y otros controles para proporcionar un nivel óptimo de filtrado
  - Realiza un seguimiento de todo el tráfico desde la capa de enlace hasta la aplicación para obtener información más precisa que otros métodos
  - Se puede actualizar automáticamente para proporcionar el contexto actual
- Desventajas
  - Para obtener el mayor beneficio, las organizaciones deben integrarlos con otros sistemas de seguridad, lo que puede ser un proceso complejo
  - Más costoso que otros tipos de firewall

2.1.2 – Arquitecturas de Firewall

FIB

Arquitecturas más comunes

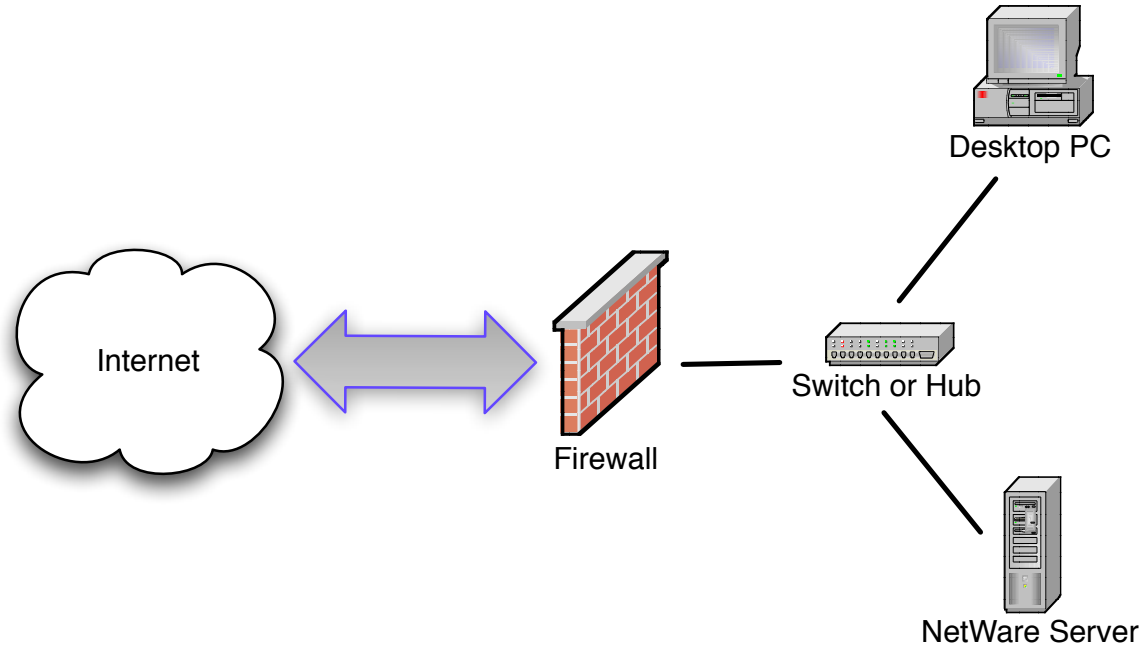
- Existen 4 arquitecturas base
  - Dual-homed
  - Two-Legged network with a full exposed DMZ
  - Restricted DMZ via dialup Firewall
  - Three-legged firewall
- A partir de estas 4, se pueden crear otras mezclándolas

2.1.2 – Arquitecturas de Firewall

FIB

Dual-homed

- El Firewall dual-homed es la arquitectura más simple
  - 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 a la red interna
  - El firewall se encarga de pasar paquetes que pasan por sus reglas de filtrado entre la red interna e Internet, y viceversa



2.1.2 – Arquitecturas de Firewall

FIB

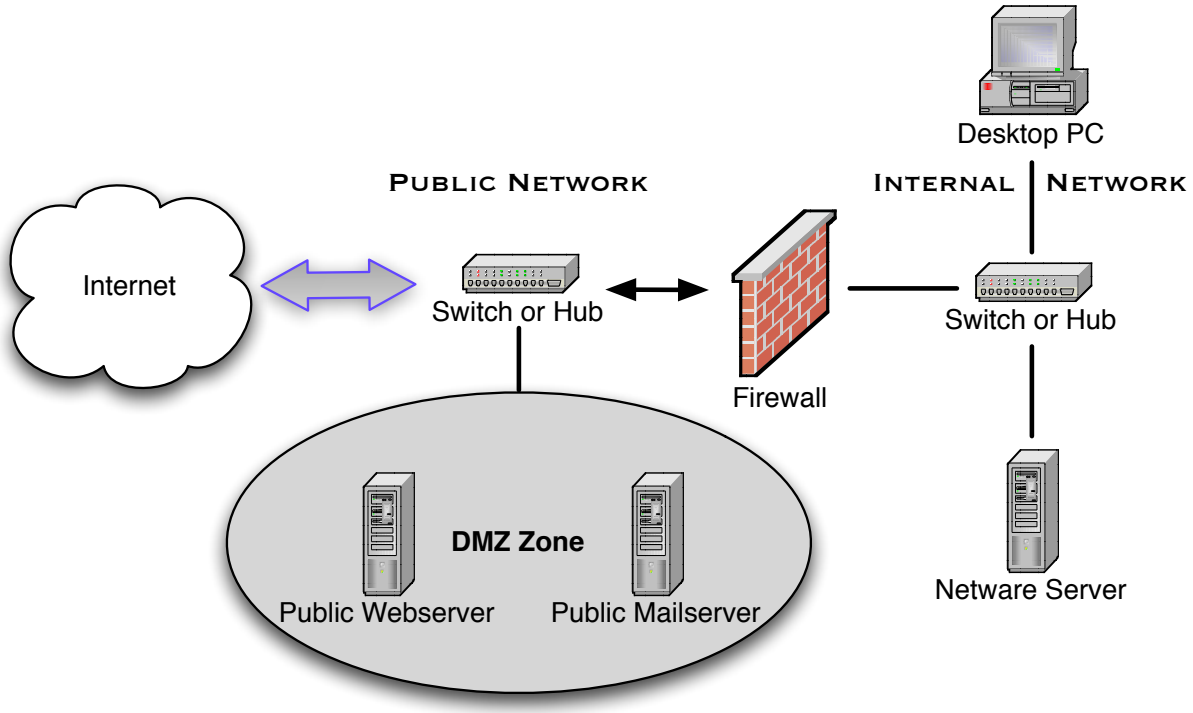
Dual-homed

- Ventajas
  - Muy fácil de configurar
  - Toda la red interna está protegida
  - Es suficiente un único firewall con solo 2 interfaces
  - La red interna puede usar direccionamiento privado
- Desventajas
  - No hay DMZ
  - O bien no se usan servidores públicos
  - O bien estos se ponen en la red interna con reglas especificas en el firewall. Esto puede ser critico ya que son accesibles desde la red externa y los atacantes pueden saltar de un servidor comprometido a cualquier equipo interno sin control

2.1.2 – Arquitecturas 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 públicos accesibles directamente desde Internet (sin pasar por el firewall)
- El resto de maquinas está en la red interna, detrás del firewall



2.1.2 – Arquitecturas de Firewall

FIB

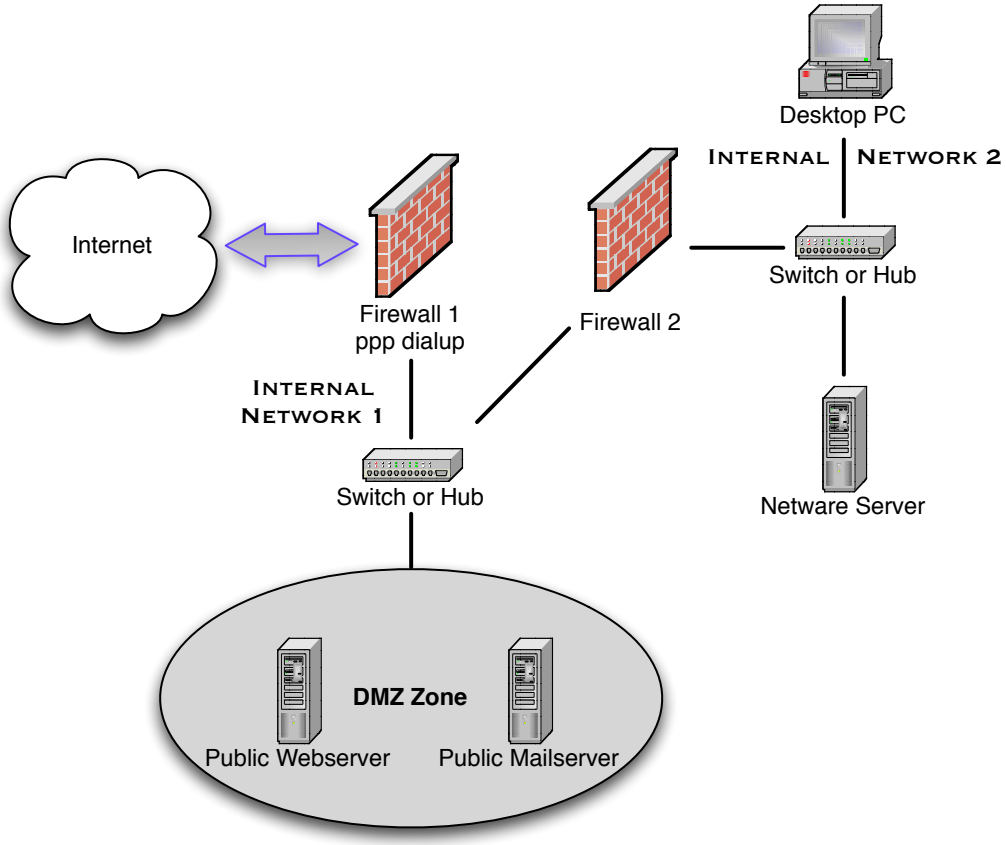
Two-Legged network with a full exposed DMZ

- Ventajas
  - Sigue siendo fácil de configurar
  - Toda la red interna está protegida
  - Es suficiente un único firewall con solo 2 interfaces
  - La red interna puede usar direccionamiento privado
- Desventajas
  - La zona DMZ está completamente expuesta
  - Puede ser un riesgo incluido para la red interna. Si un servidor público está comprometido, el problema (p.e., un malware) podría saltar a la red interna ya que los equipos internos también usan estos servidores públicos

2.1.2 – Arquitecturas de Firewall

Restricted DMZ via Dialup Firewall

- Se usan 2 firewalls
  - El Firewall 1 protege la DMZ y mantiene la conexión con Internet
  - El Firewall 2 protege la red interna



2.1.2 – Arquitecturas de Firewall

FIB

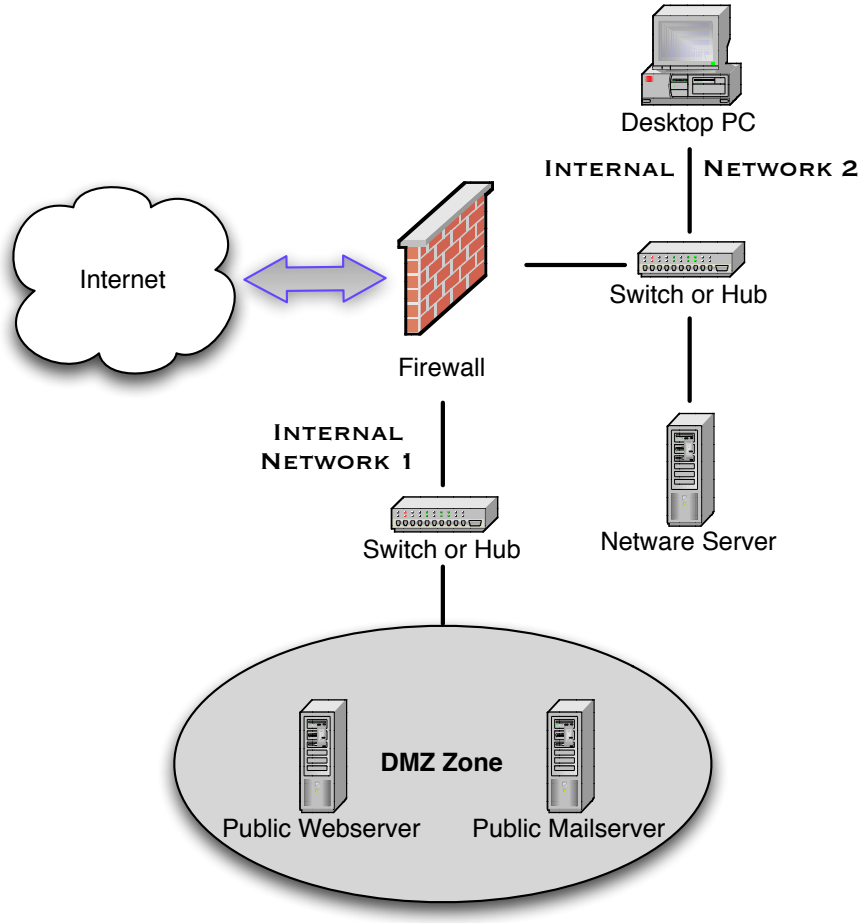
Restricted DMZ via Dialup Firewall

- Ventajas
  - Relativamente fácil de configurar
  - Toda la red interna está protegida
  - La zona DMZ también está protegida
  - La red interna y la DMZ pueden usar direccionamiento privado
- Desventajas
  - Se necesitan 2 Firewalls
  - Se necesita mantener la configuración de los 2 Firewalls coherente

2.1.2 – Arquitecturas de Firewall

Three-legged firewall

- Para tener un único firewall y una DMZ protegida, se necesita un Firewall de tres patas (tres interfaces)



2.1.2 – Arquitecturas de Firewall

FIB

Three-legged firewall

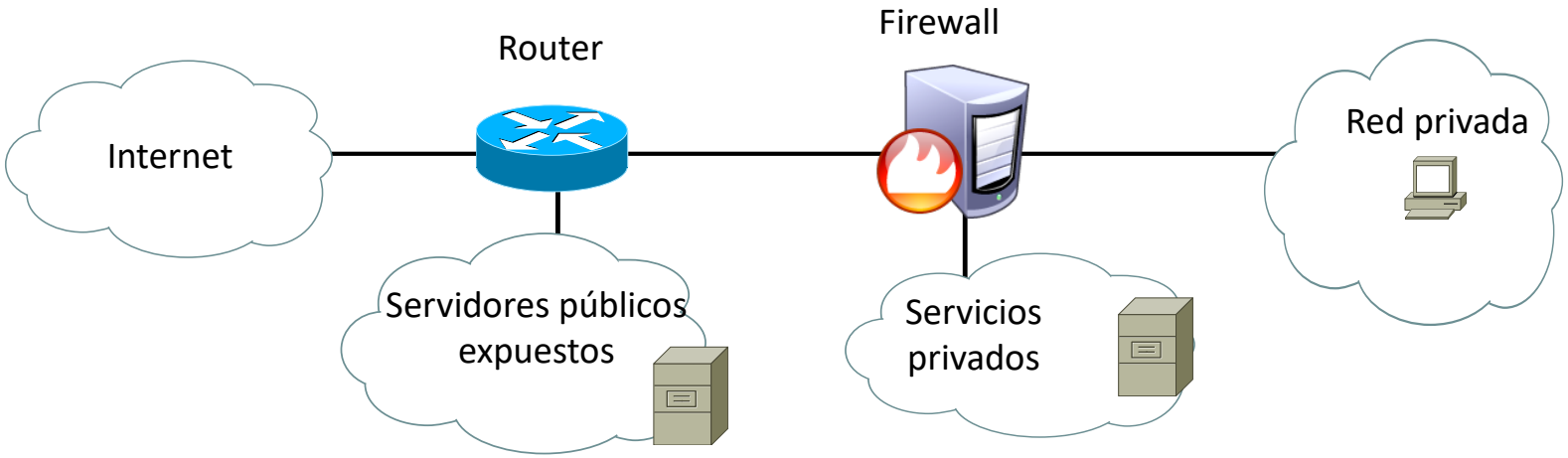
- Ventajas
  - Se necesita un único Firewall
  - Toda la red interna está protegida
  - La zona DMZ también está protegida
  - La red interna y la DMZ pueden usar direccionamiento privado
- Desventajas
  - Se necesita un Firewalls con 3 interfaces (coste)
  - Más complejo de configurar (más reglas)

2.1.2 – Arquitecturas de Firewall

FIB

Más arquitecturas

- Estas cuatros arquitecturas no son las únicas
- Por ejemplo, se pueden combinar la segunda y la cuarta
  - Un único Firewall
  - Servidores públicos expuestos
  - Servidores privados
  - Red privada



2.1.3 - Tecnologías

FIB

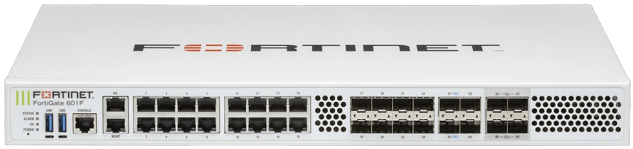
Diferentes soluciones

- Dispositivos hardware
- Host-based firewall
- Cloud-based firewall

2.1.3 - Tecnologías

Dispositivos hardware

- Son dispositivos físicos que se instalan en las infraestructura de la organización
- Pueden ser dispositivos con tecnología propietaria (Cisco, Juniper, Palo Alto, Fortinet, etc.)



- O pueden usar tecnología de código abierto (iptables, firewalld, nftables, pfSense, etc.)

Fuente imágenes: https://www.paloaltonetworks.com/, https://www.fortinet.com

2.1.3 - Tecnologías

FIB

Host-based firewall

- Son Firewall software que se instalan directamente en los hosts que se quieren proteger
- Típicamente son firewall a nivel de paquetes
- Por ejemplo
  - Microsoft defender
  - Apple firewall
  - Firewalld

2.1.3 - Tecnologías

FIB

Cloud-based firewall

- Muchos servicios se están actualmente moviendo a la nube
- Lo mismo está pasando con los firewall y ahora se ofrece Firewalls como servicio (FWaaS)
- Las razones son
  - Costes reducidos
  - Se pueden combinar fácilmente con otros servicios de seguridad (arquitecturas de confianza cero, antivirus, etc.)
  - Fácilmente actualizable a nuevas tecnologías
  - Fácilmente adaptables a nuevas amenazas

2.1.4 – Firewall software

FIB

Reglas de filtrado

- Las reglas de filtrado son un conjunto de reglas secuenciales para denegar/permitir cierto tráfico de red según algunos criterios
- Los criterios de filtrado son
  - @IP origen y destino
  - Puerto origen y destino
  - Tipo de protocolo (IP, TCP, UDP, ICMP)
  - Estado de la conexión (nueva, respuesta o relacionada)

2.1.4 – Firewall software

FIB

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

2.1.4 – Firewall software

FIB

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

2.1.4 – iptables

FIB

Introducción

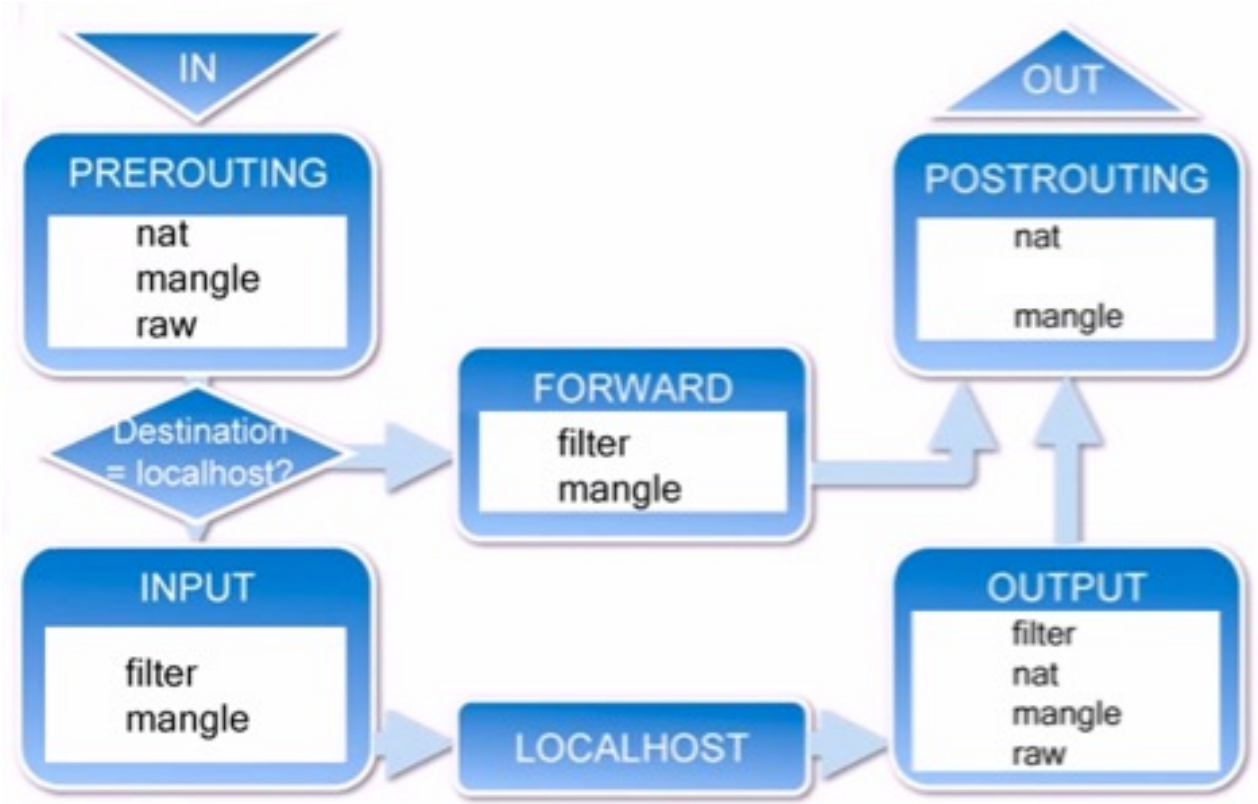
- Nació en el 1998 y se incorporó en Linux 2.3 en marzo 2000
- Es un framework disponible en el kernel de Linux que permite interceptar y manipular paquetes de red
  - Filtrar paquetes
  - Realizar traducción de direcciones de red (NAT)
  - Mantener registros de log
- Se está lentamente reemplazando por otras soluciones como firewalld y nftables pero se sigue usando ampliamente
- 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

2.1.4 – iptables

FIB

Estructura

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



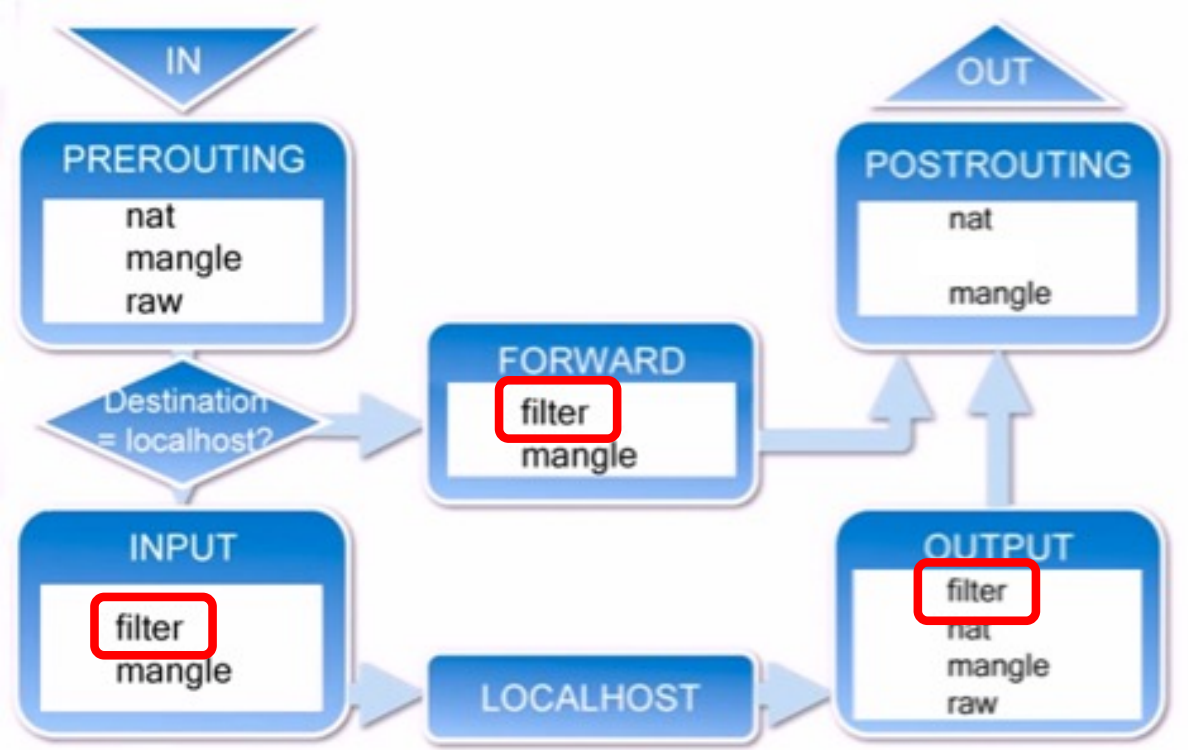
Fuente imagen: https://ranxing.files.wordpress.com/2014/11/untitled.png

2.1.4 – iptables

FIB

Tablas

- Filter
  - Denegar/permitir determinados paquetes

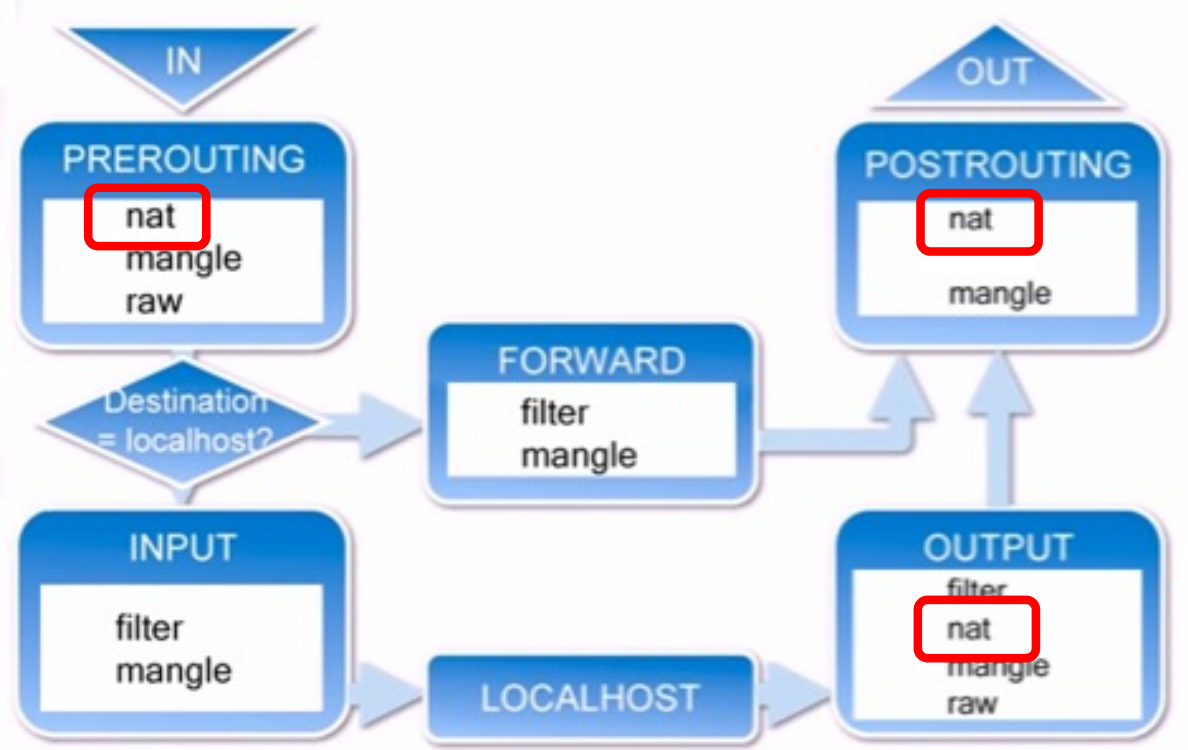


2.1.4 – iptables

FIB

Tablas

- NAT
  - Modificar @IP/puertos de los paquetes

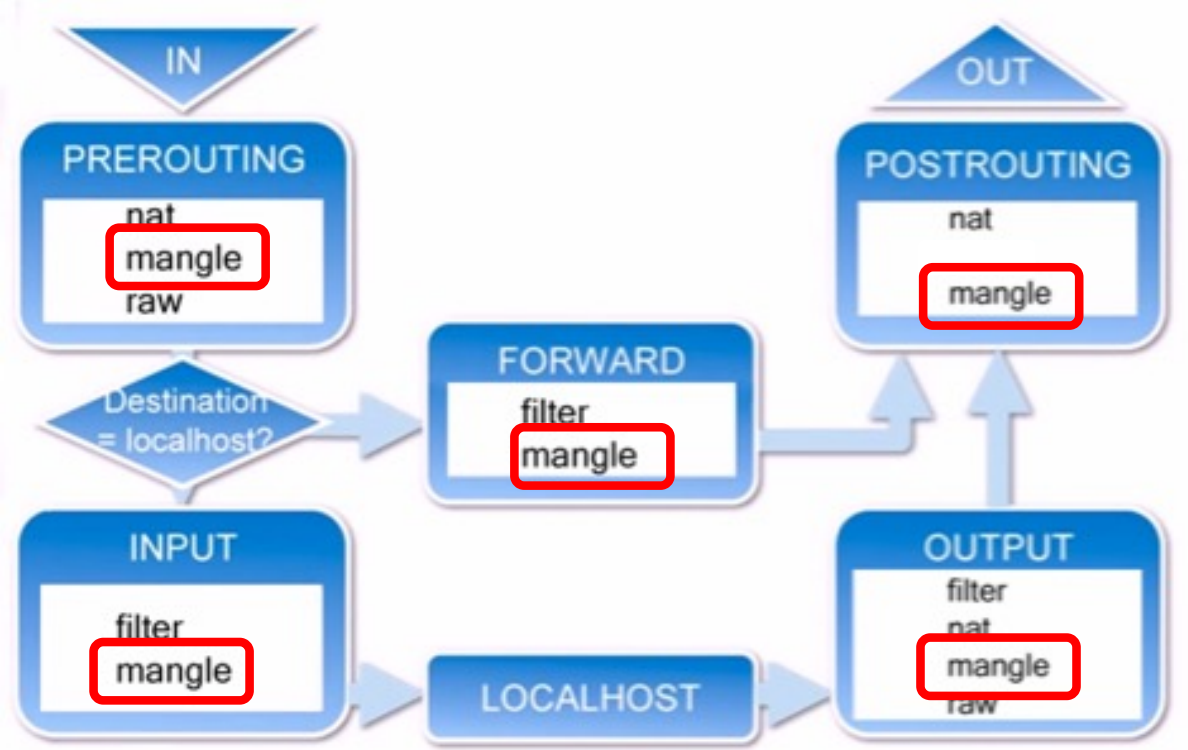


2.1.4 – iptables

FIB

Tablas

- Mangle
  - Modificar algunos otros campos de los paquetes como



2.1.4 – iptables

FIB

Tablas

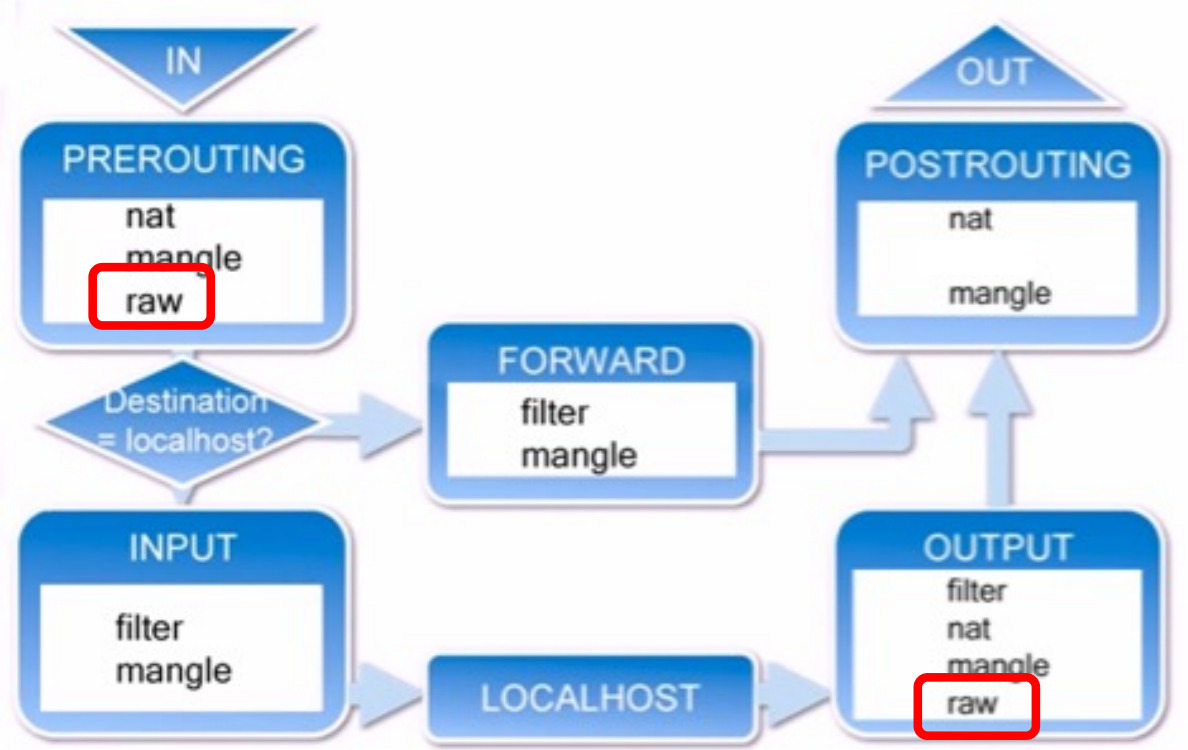
- Mangle
  - Modificar algunos otros campos de los paquetes como
  - **Tipo de Servicio**: campo de la cabecera IP usado para dar prioridad a los paquetes (define como tratar los paquetes en los routers)
  - **Tiempo de vida** (TTL): campo de la cabecera IP usado para que un paquete no se quede eternamente 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.)

2.1.4 – iptables

FIB

Tablas

- Raw
  - Nueva tabla desde Linux Kernel 2.6
  - Se consulta antes de cualquier otra



2.1.4 – iptables

FIB

Tablas

- Raw
  - Nueva tabla desde Linux Kernel 2.6
  - Se consulta antes de cualquier otra
  - iptables usa estados para reconocer paquetes relacionados entre ellos
    - Por ejemplo, paquetes de una misma conexión TCP
  - Para mantener estos estados, usa un modulo llamado CONNTRACK
  - Aunque tenga otras aplicaciones, la tabla RAW se usa principalmente para marcar paquetes con la opción NOTRACK para que iptables no guarde el estado de estos paquetes en CONNTRACK
  - Veremos luego ejemplo de aplicación de CONNTRACK

2.1.4 – iptables

FIB

Tipo de firewall?

- Firewall a nivel de paquetes
- Firewall a nivel de circuito
- Firewall a nivel de aplicación
- Firewall con inspección de estados
- Firewall de próxima generación

2.1.4 – iptables

FIB

Tipo de firewall?

- Firewall a nivel de paquetes
- Firewall a nivel de circuito
- Firewall a nivel de aplicación
- Firewall con inspección de estados
- Firewall de próxima generación

2.1.4 – iptables

FIB

Comandos

```
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}

2.1.4 – iptables

FIB

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

2.1.4 – iptables

FIB

Regla por defecto

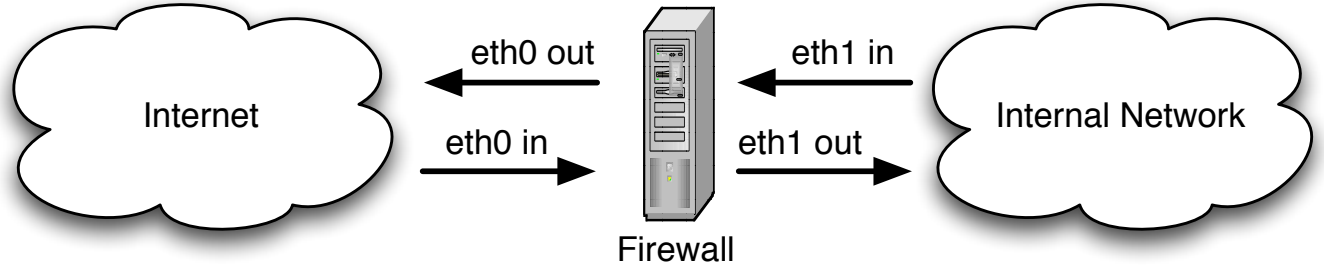
- Las reglas por defecto en iptables es aceptar
  - Usa por defecto el enfoque 2: lista de reglas de prohibiciones y la última acepta el resto
- Si se quiere modificar

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

2.1.4 – iptables

FIB

Ejemplos

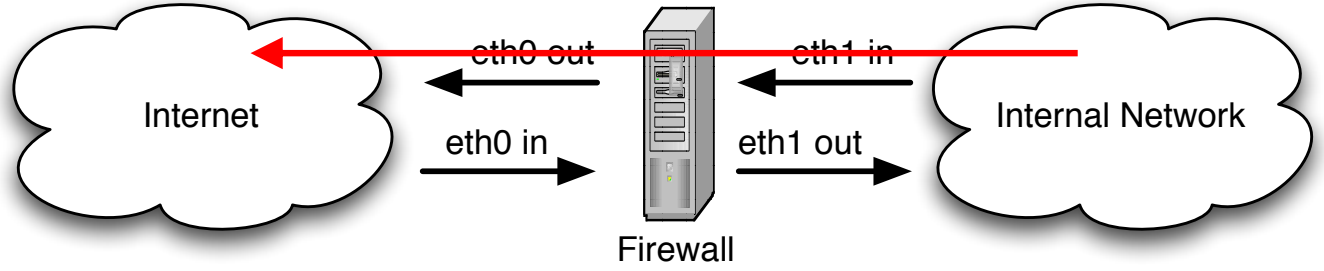


- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet

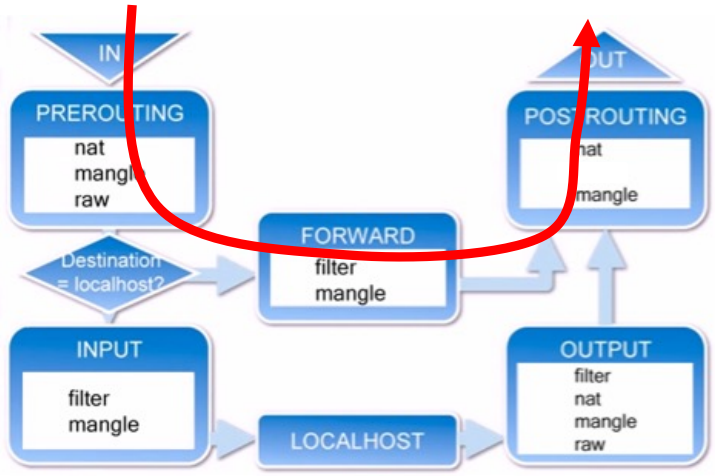
2.1.4 – iptables

FIB

Ejemplos



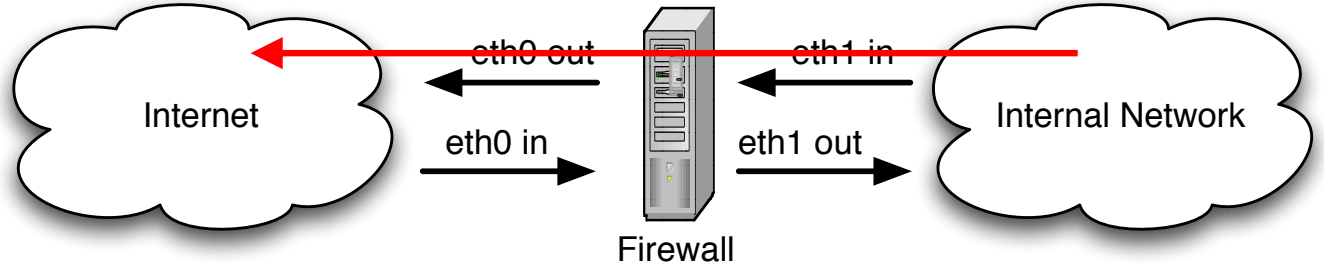
- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet



2.1.4 – iptables

FIB

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

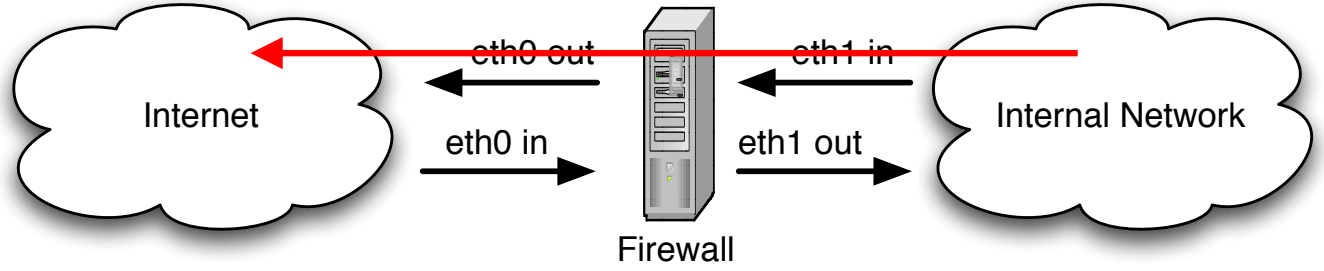
iptables –t filter –P FORWARD DROP

- Funciona?

2.1.4 – iptables

FIB

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

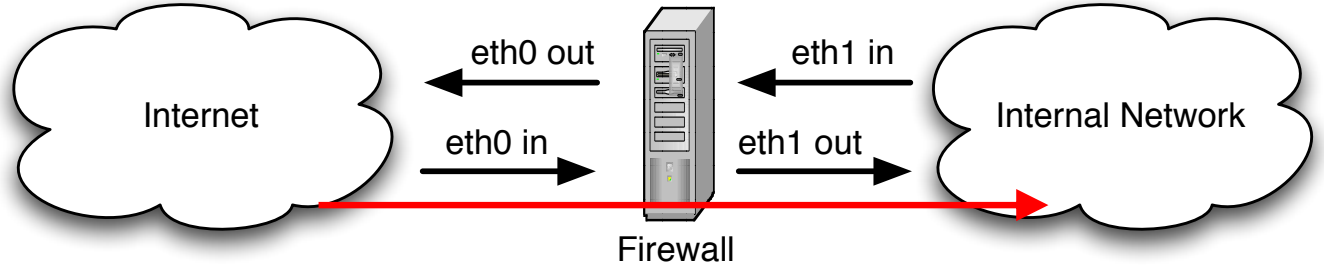
iptables –t filter –P FORWARD DROP

- Funciona? NO, la comunicación suele ser bidireccional

2.1.4 – iptables

FIB

Ejemplos



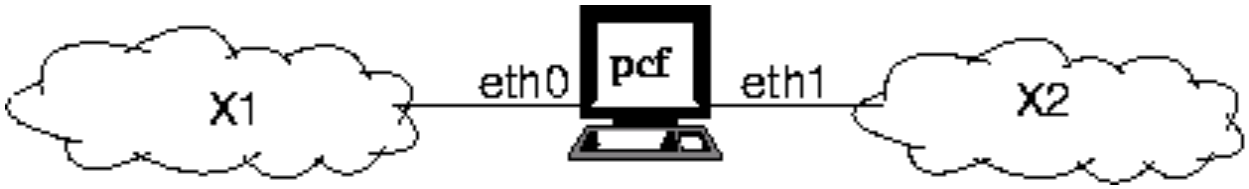
- Los hosts de la red interna solo pueden acceder a servicio HTTP de Internet

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

2.1.4 – iptables

FIB

Ejemplos

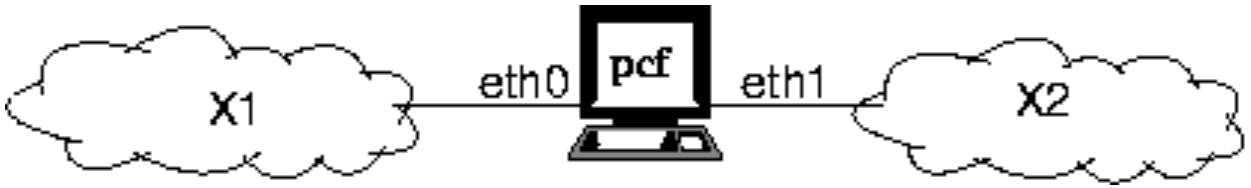


- Se quiere que los hosts de la red X1 puedan empezar una comunicación con los hosts de la red X2 (y estos puedan contestar) pero no viceversa

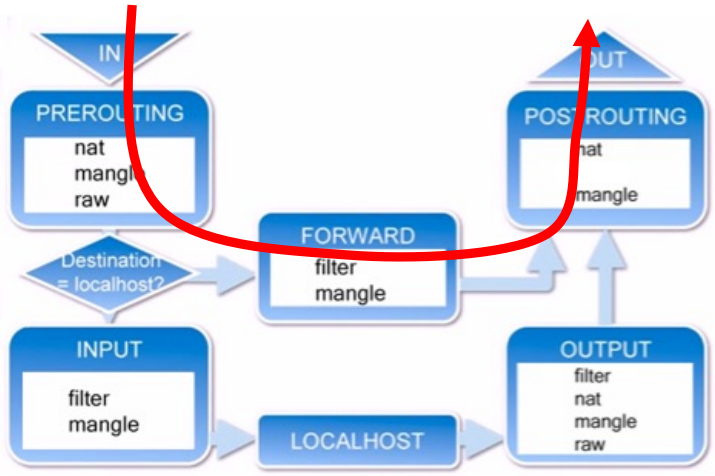
2.1.4 – iptables

FIB

Ejemplos



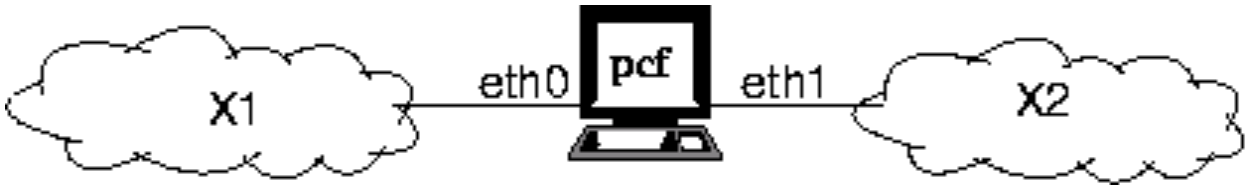
- Se quiere que los hosts de la red X1 puedan empezar una comunicación con los hosts de la red X2 (y estos puedan contestar) pero no viceversa



2.1.4 – iptables

FIB

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

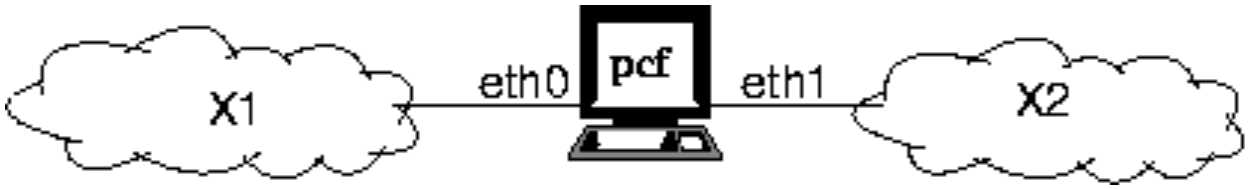
iptables –t filter –A FORWARD –i eth0 –o eth1 –j ACCEPT

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

2.1.4 – iptables

FIB

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

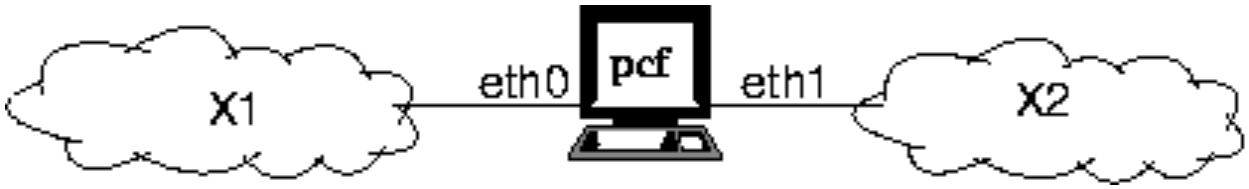
iptables –t filter –A FORWARD –i eth1 –o eth0 –j DROP

- Funciona?

2.1.4 – iptables

FIB

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

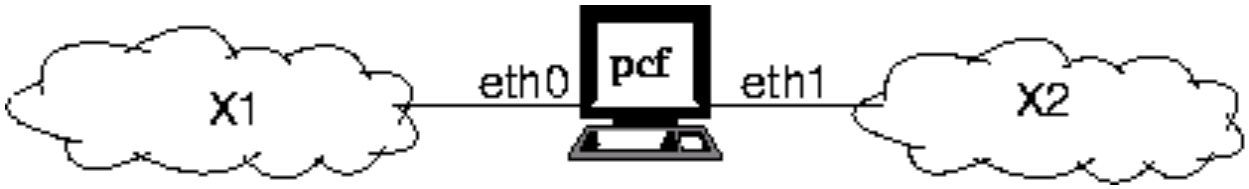
iptables –t filter –A FORWARD –i eth1 –o eth0 –j DROP

- Funciona? No, X2 no puede contestar

2.1.4 – iptables

FIB

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

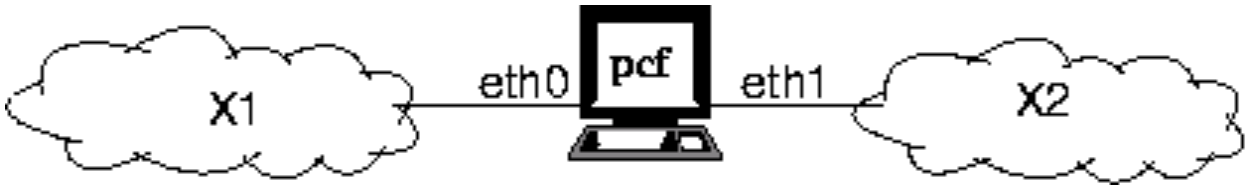
iptables –t filter –A FORWARD –i eth1 –o eth0 –j ACCEPT

- Ahora funciona?

2.1.4 – iptables

FIB

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

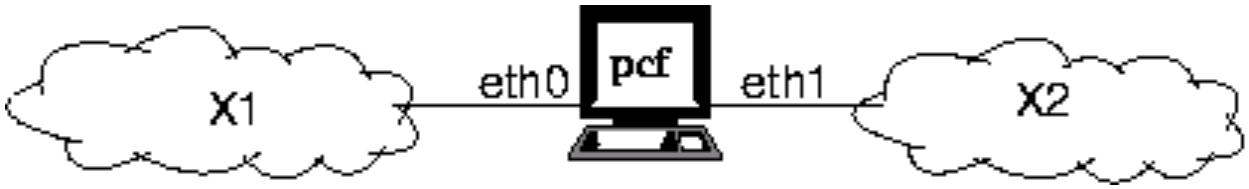
iptables –t filter –A FORWARD –i eth1 –o eth0 –j ACCEPT

- Ahora funciona? Tampoco, ya que ahora X2 puede empezar una comunicación con X1

2.1.4 – iptables

FIB

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 –m conntrack –cstate ESTABLISHED –j ACCEPT

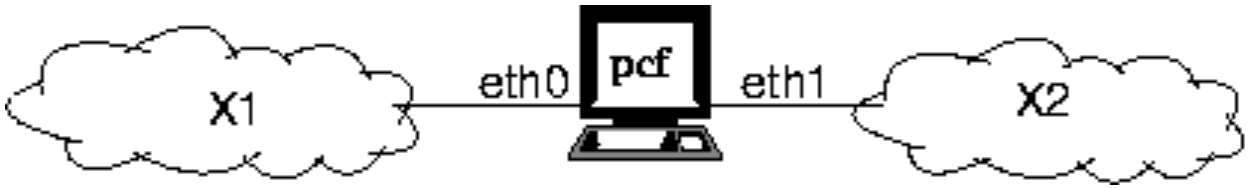
iptables –P FORWARD DROP

- Ahora funciona?

2.1.4 – iptables

FIB

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

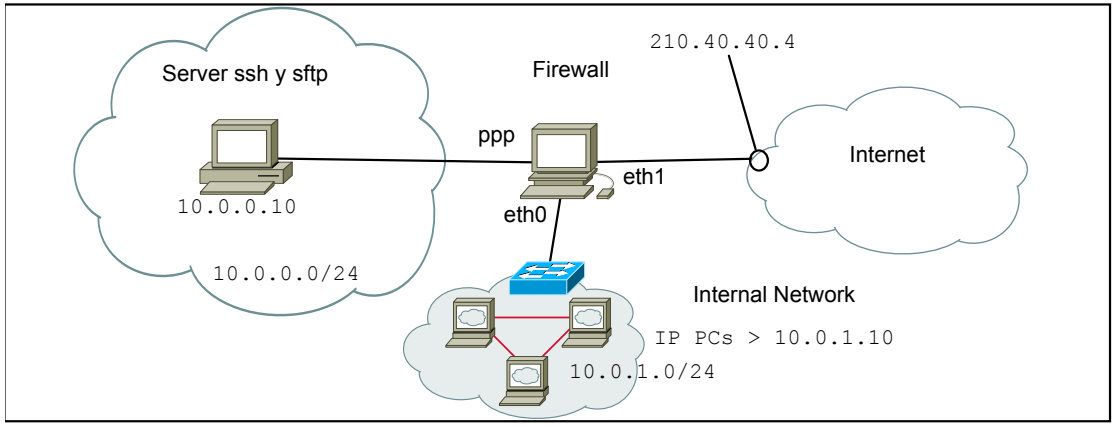
iptables –t filter –A FORWARD –i eth1 –o eth0 –m conntrack –cstate ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP

- Ahora funciona? Ahora si
- Se consulta el estado en el modulo CONNTRACK en el sentido X2→X1 y solo deja pasar las respuestas
- Es decir paquetes de una conexión ya establecida en el sentido contrario X1→X2 y guardada en CONNTRACK

2.1.4 – iptables

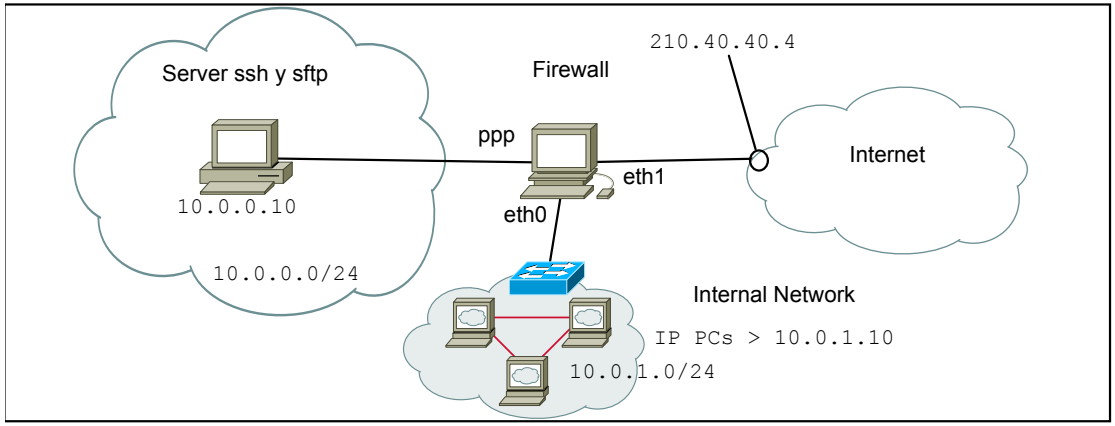
Ejemplos



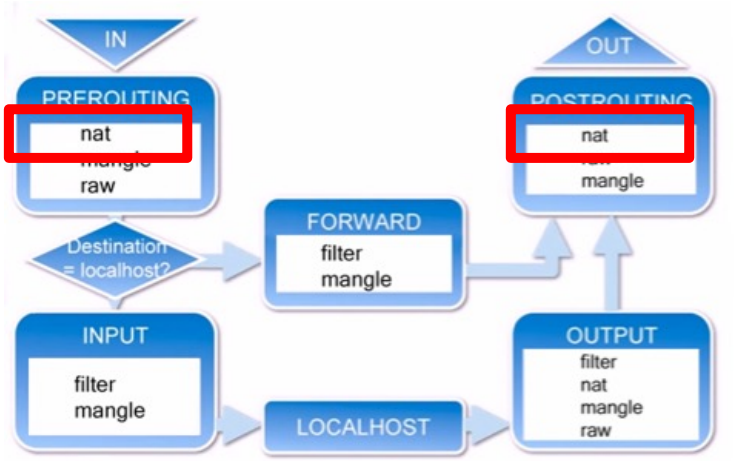
- Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

2.1.4 – iptables

Ejemplos



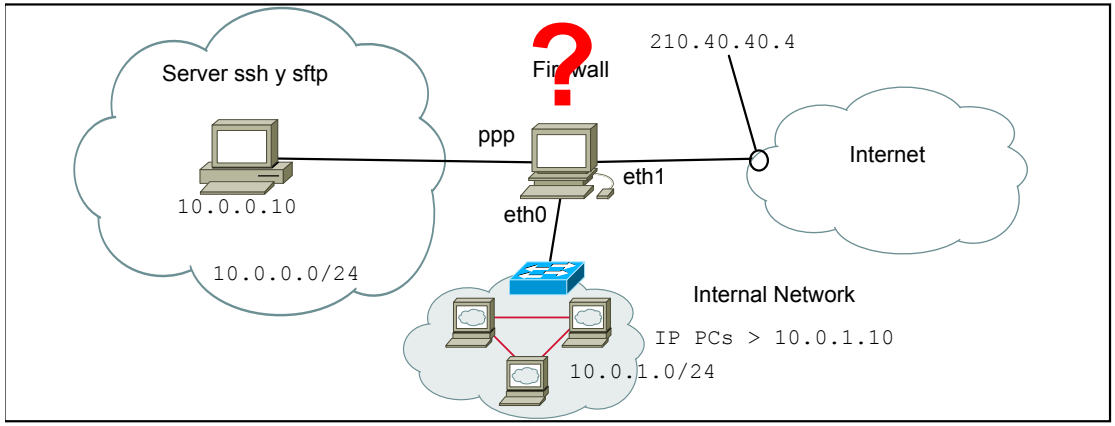
- Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4



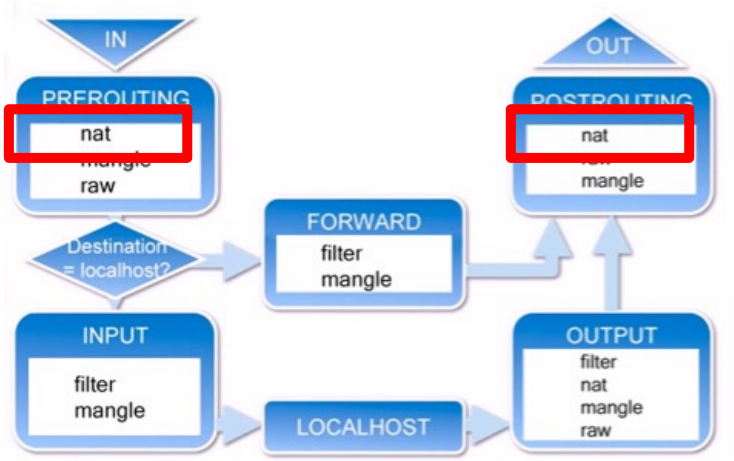
2.1.4 – iptables

FIB

Ejemplos

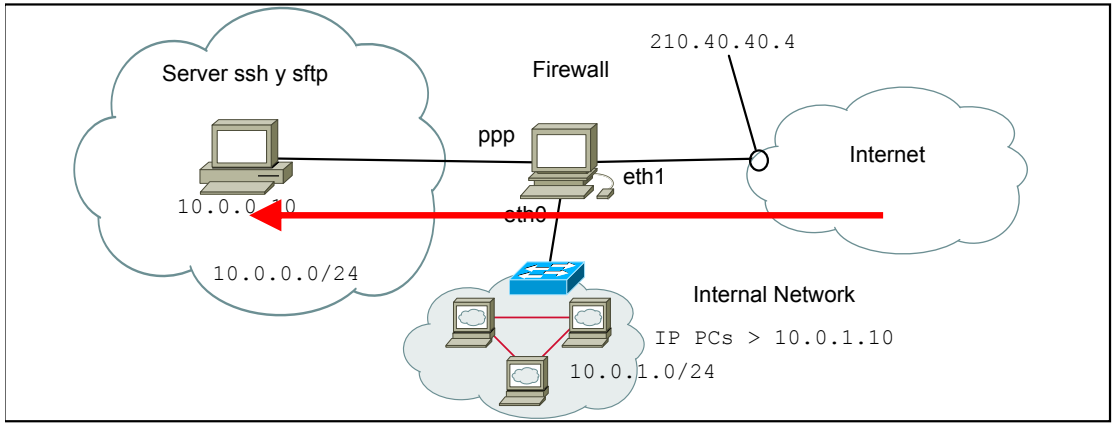


- Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4



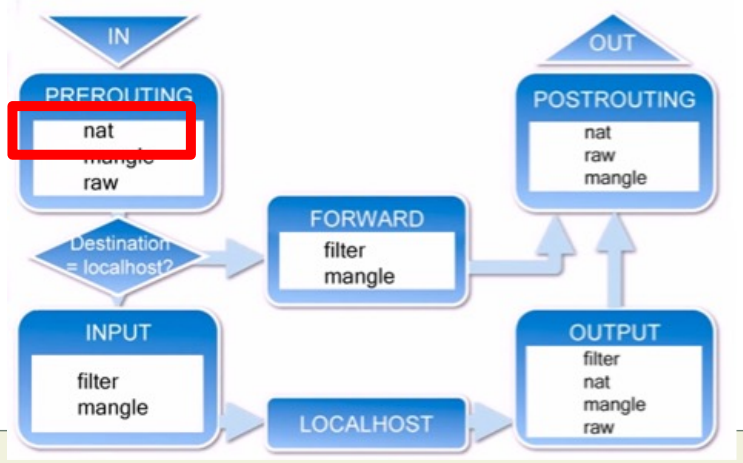
2.1.4 – iptables

Ejemplos



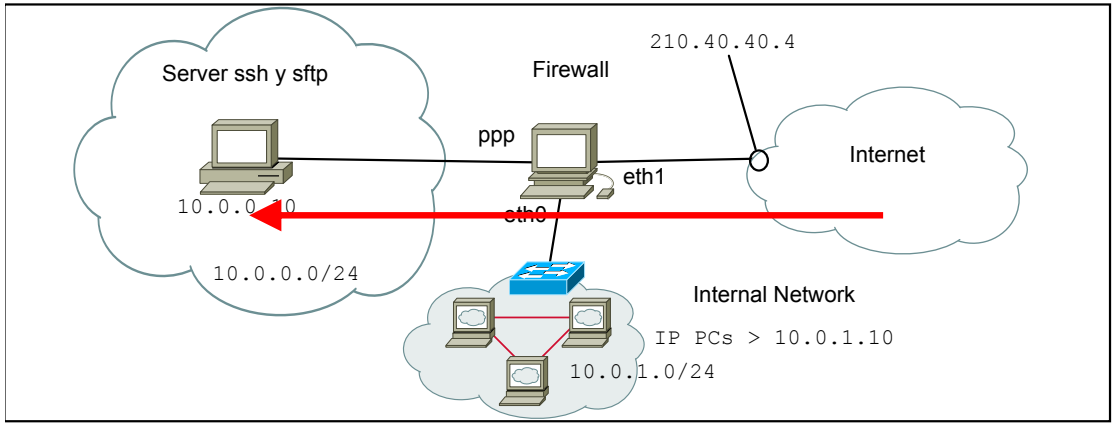
- Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

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



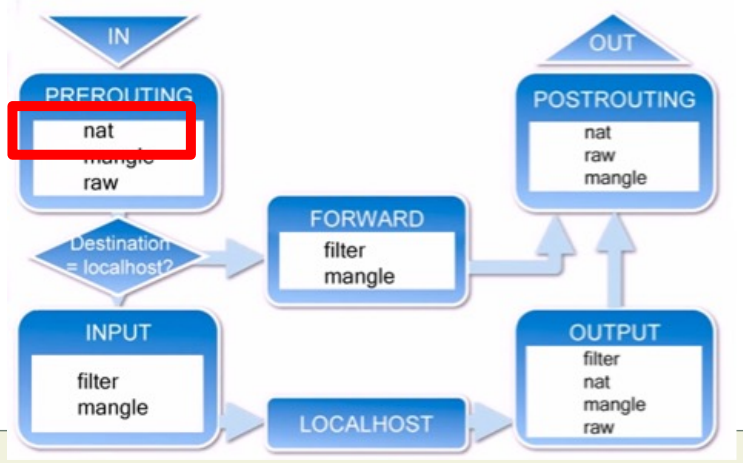
2.1.4 – iptables

Ejemplos



- 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 PREROUTING –d 210.40.40.4 –i eth1 –j DNAT --to-destination 10.0.0.10

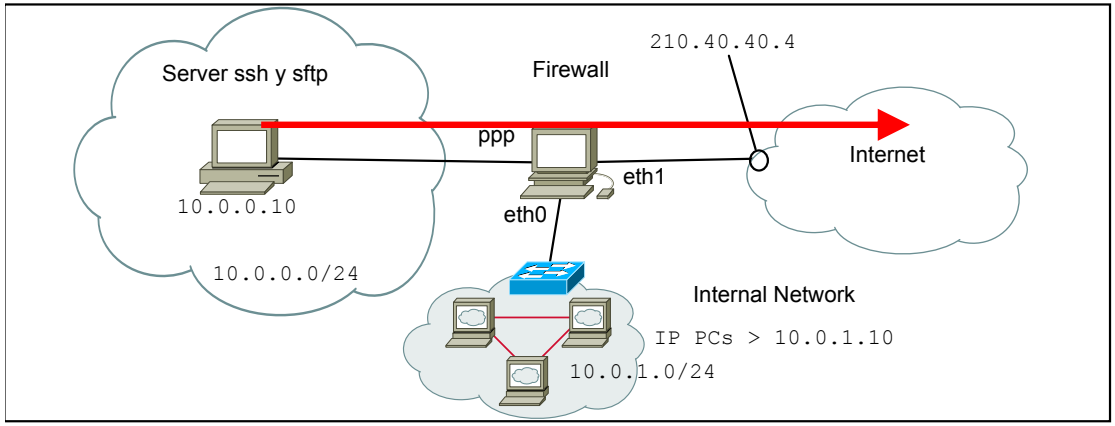


Hay que modificar la @IP destino antes de tomar decisiones sobre el forwarding del paquete
Si no se hace, la decisión sería sobre la @IP pública y no la real del destino

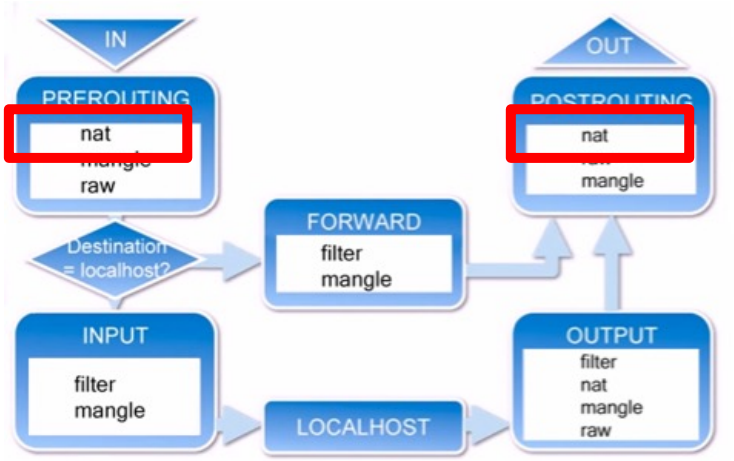
2.1.4 – iptables

FIB

Ejemplos



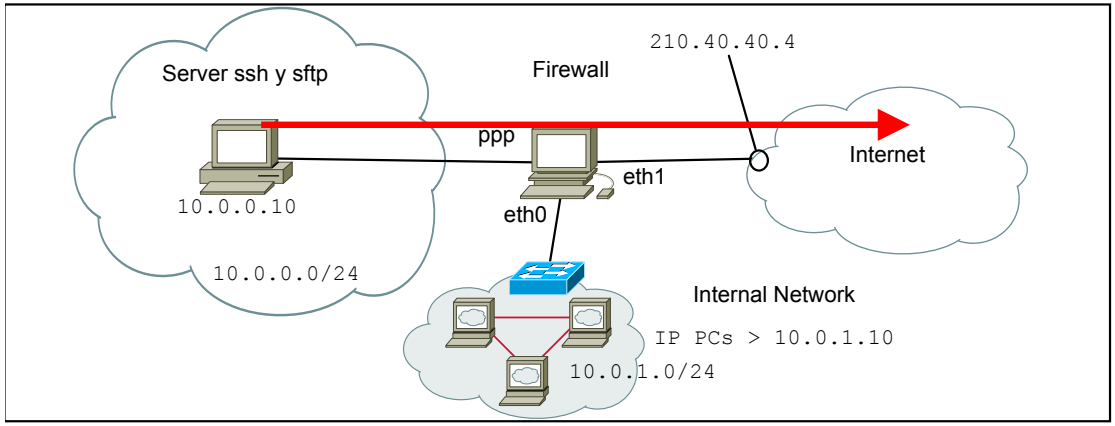
- Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4



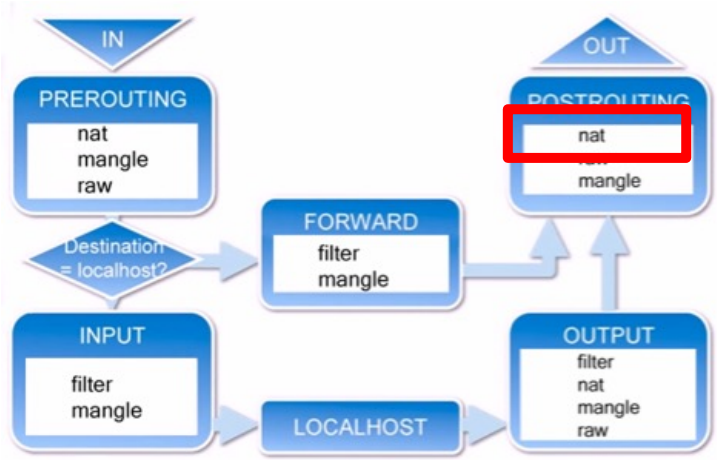
?

2.1.4 – iptables

Ejemplos



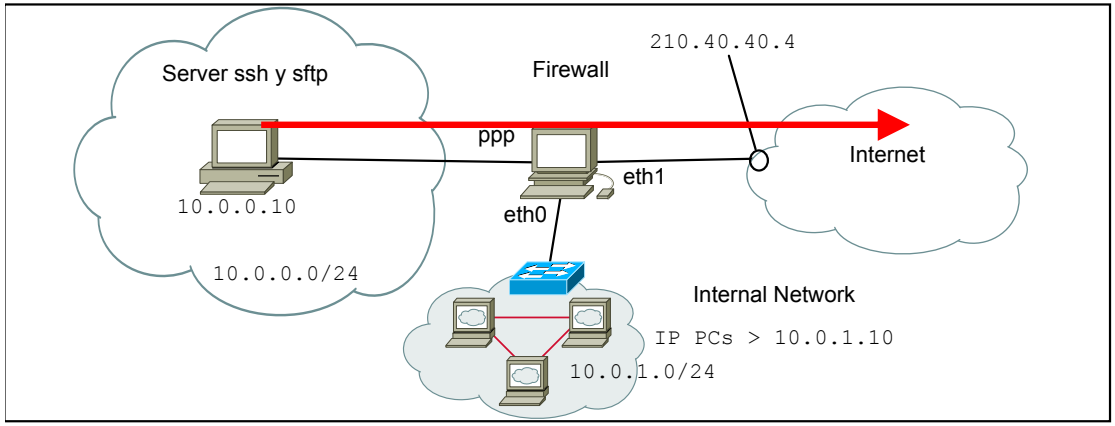
- Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4



Hay que modificar la @IP origen después de tomar decisiones sobre el filtrado del paquete en el FORWARD
Si no se hace, el filtrado sería sobre la @IP pública y no la real del destino

2.1.4 – iptables

Ejemplos

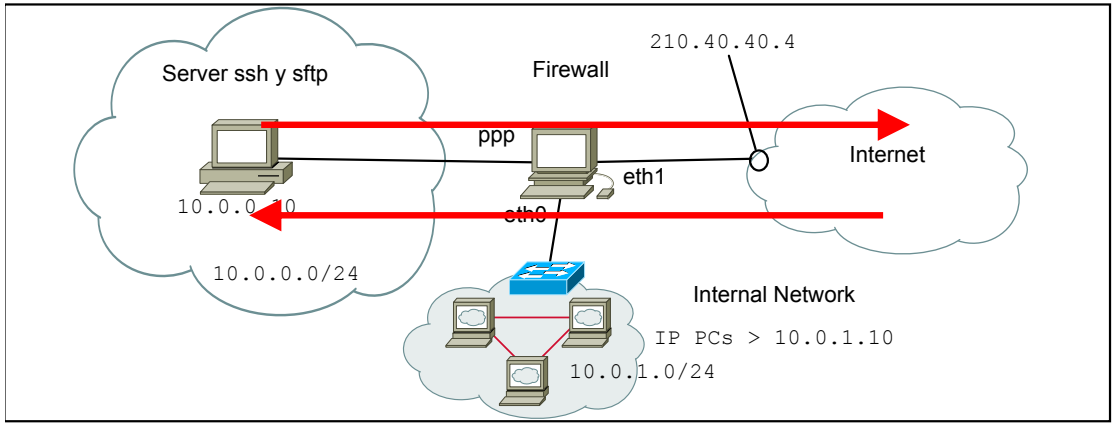


- Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

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

2.1.4 – iptables

Ejemplos



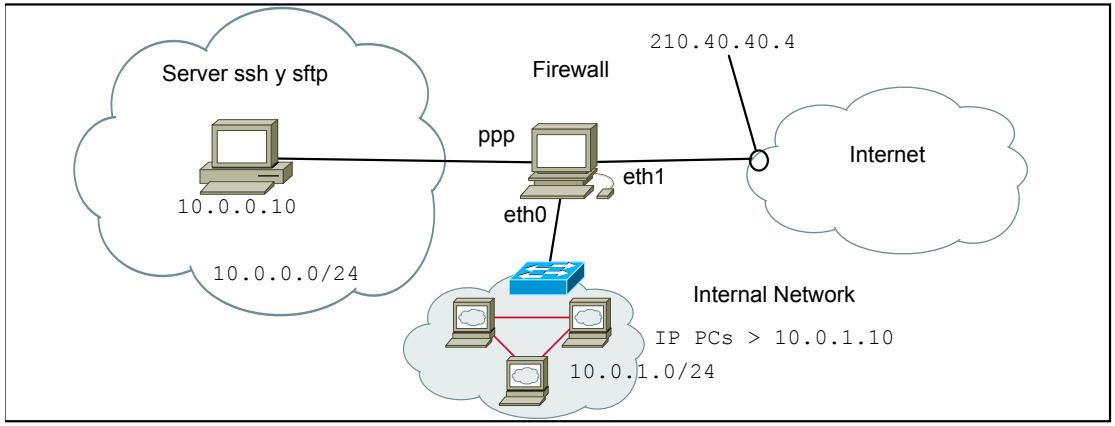
- Configurar el servicio NAT en el firewall para que la @IP del servidor 10.0.0.10 se vea en Internet como 210.40.40.4

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

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

2.1.4 – iptables

Ejemplos

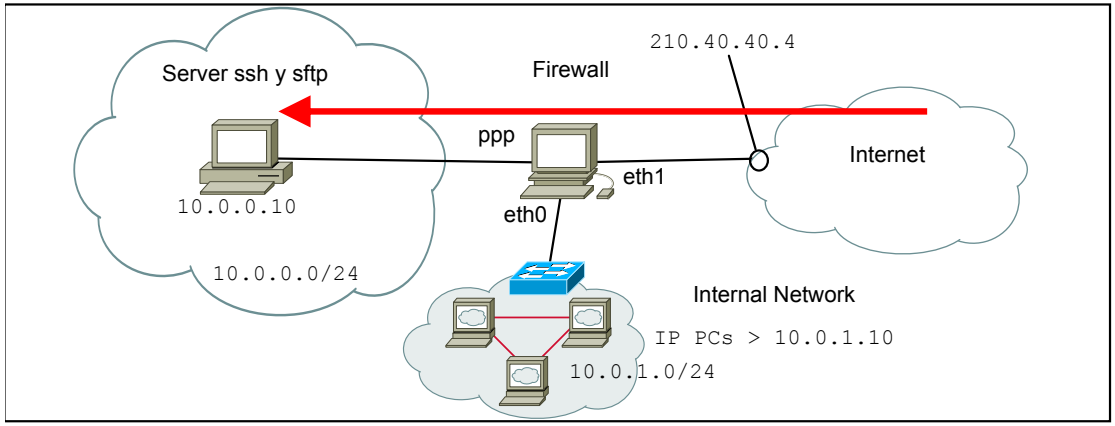


- Configurar el firewall para que los hosts de Internet solo puedan acceder a los servicios ssh y sftp (puerto 22) del servidor 10.0.0.10

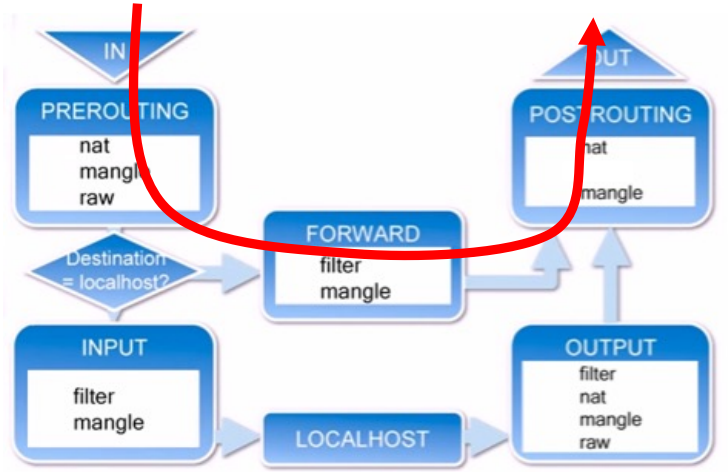
2.1.4 – iptables

FIB

Ejemplos

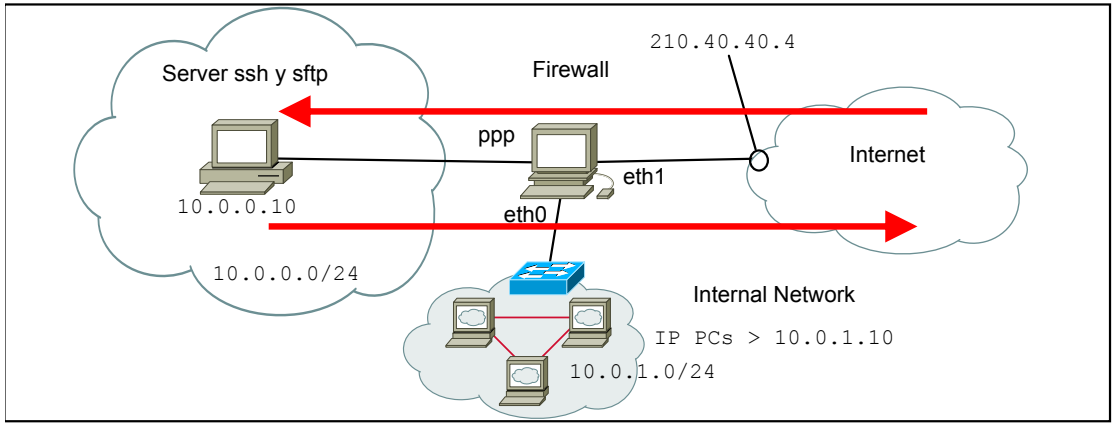


- Configurar el firewall para que los hosts de Internet solo puedan acceder a los servicios ssh y sftp (puerto 22) del servidor 10.0.0.10



2.1.4 – iptables

Ejemplos



- 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

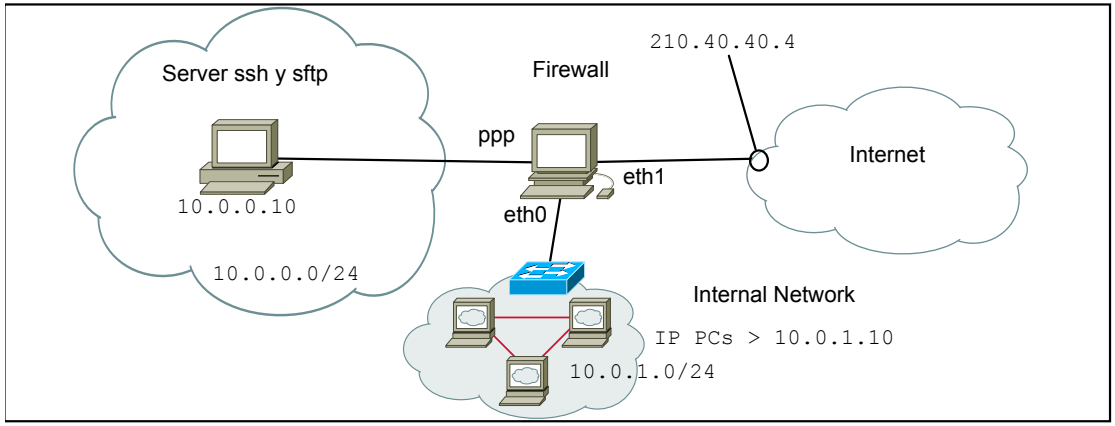
iptables –t filter –A FORWARD –i eth1 –o ppp –d 10.0.0.10 0.0.0.0 –dport 22 –j ACCEPT

iptables –t filter –A FORWARD –i ppp –o eth1 –s 10.0.0.10 0.0.0.0 –sport 22 –m conntrack --cstate ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP

2.1.4 – iptables

Ejemplos

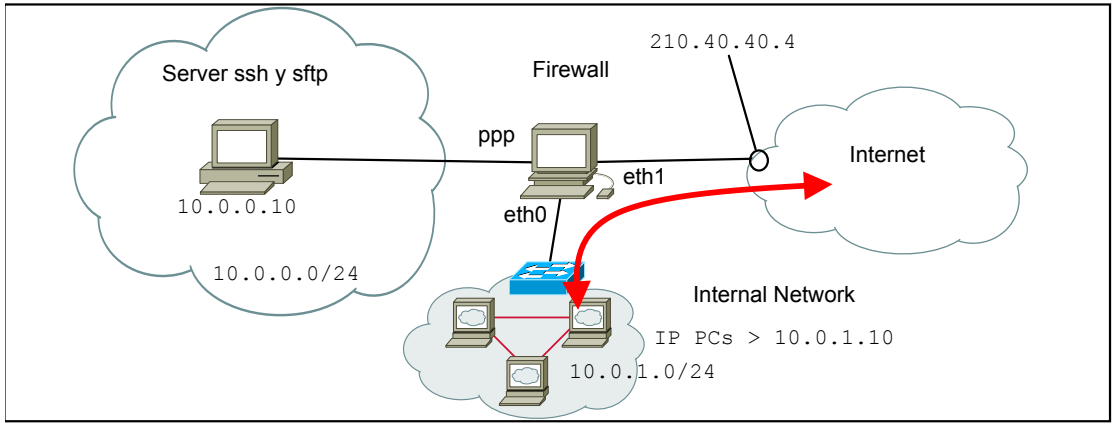


- Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar

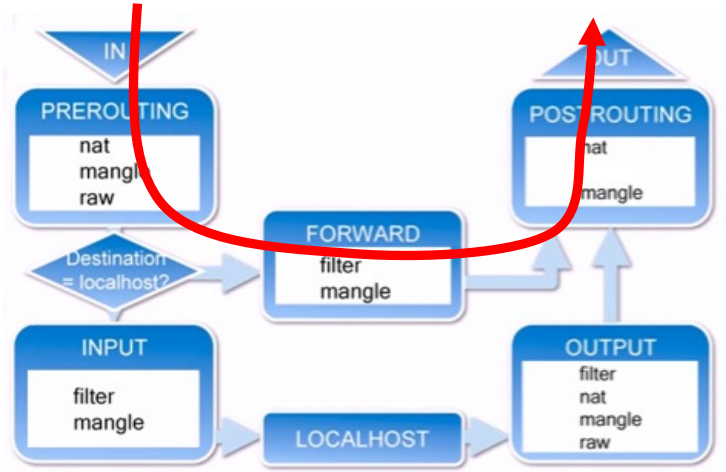
2.1.4 – iptables

FIB

Ejemplos

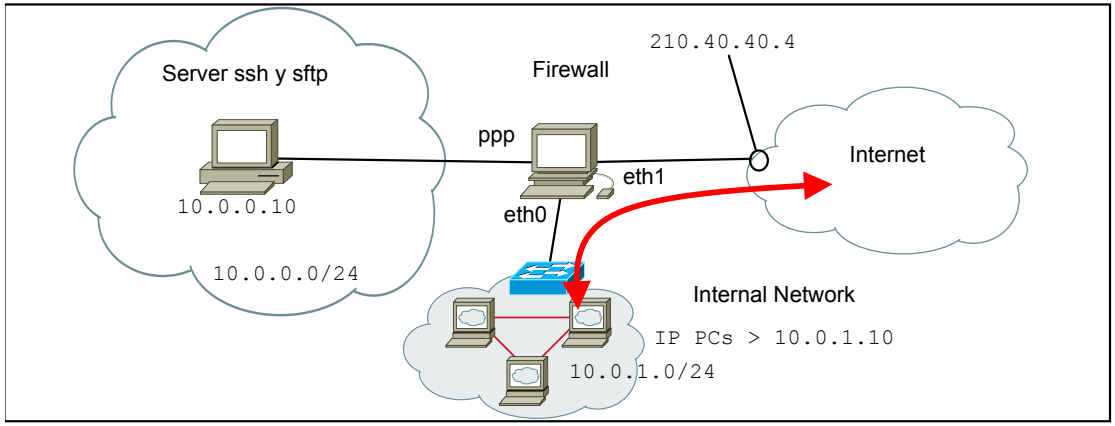


- Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar



2.1.4 – iptables

Ejemplos



- 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

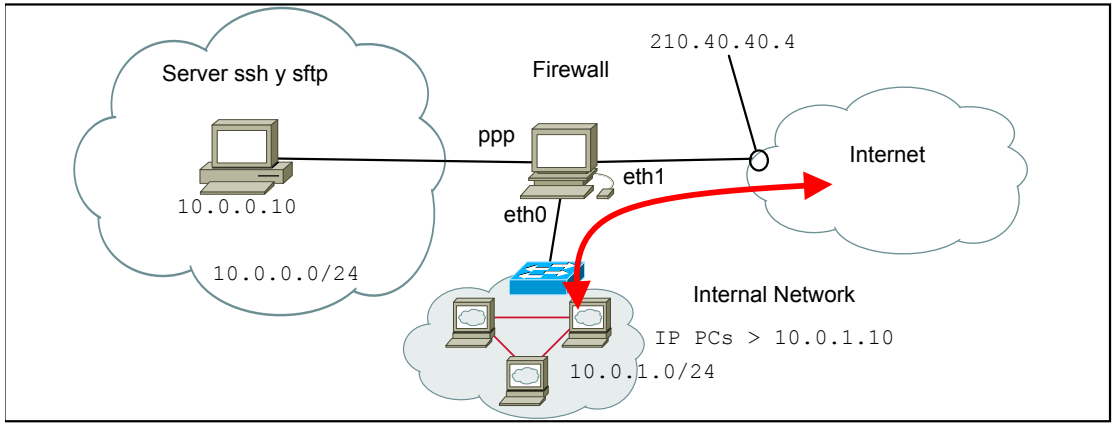
iptables –t filter –A FORWARD –i eth0 –o eth1 –s 10.0.1.0 0.0.0.255 –dport 80 –j ACCEPT

iptables –t filter –A FORWARD –i eth1 –o eth0 –d 10.0.1.0 0.0.0.255 –sport 80
–m conntrack --cstate ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP

2.1.4 – iptables

Ejemplos



- 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

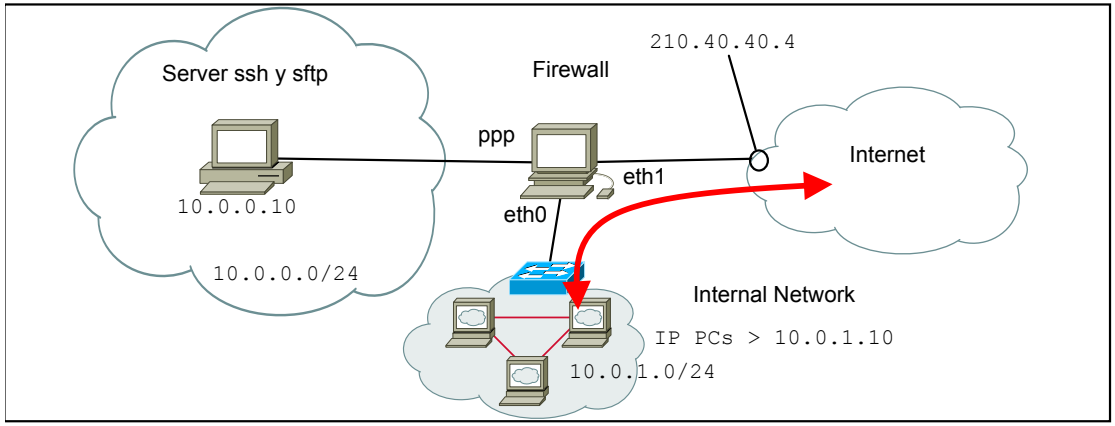
iptables –t filter –A FORWARD –i eth1 –o eth0 –d 10.0.1.0 0.0.0.255 –sport 80
–m conntrack --cstate ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP

- Funciona? Falta algo?

2.1.4 – iptables

Ejemplos



- 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

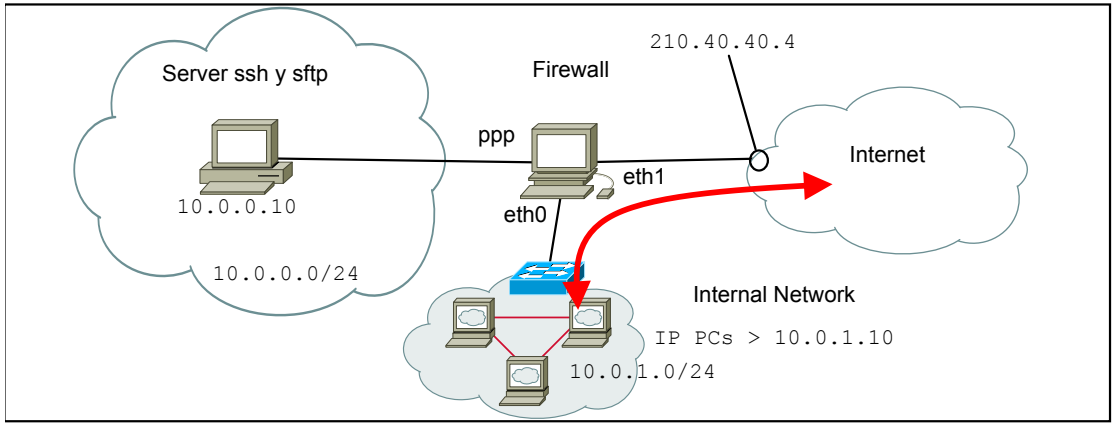
iptables –t filter –A FORWARD –i eth1 –o eth0 –d 10.0.1.0 0.0.0.255 –sport 80
–m conntrack --cstate ESTABLISHED –j ACCEPT

iptables –P FORWARD DROP

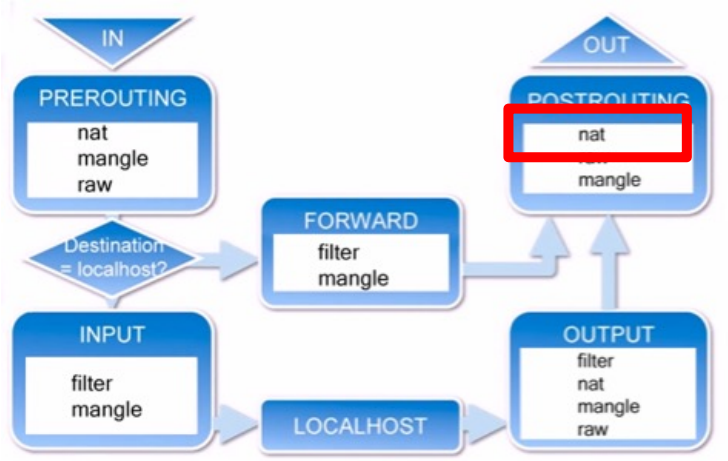
- Funciona? Falta algo? Los hosts tienen direccionamiento privado
- Falta el NAT

2.1.4 – iptables

Ejemplos

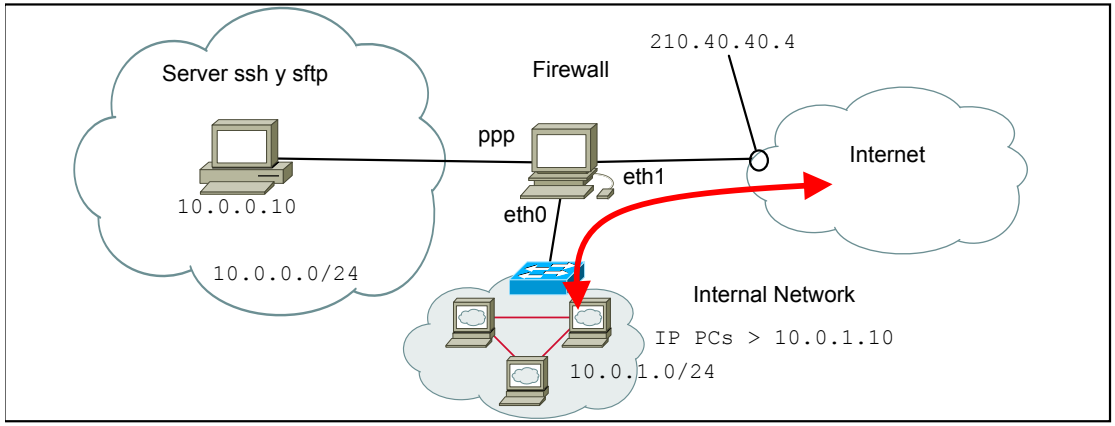


- Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar



2.1.4 – iptables

Ejemplos



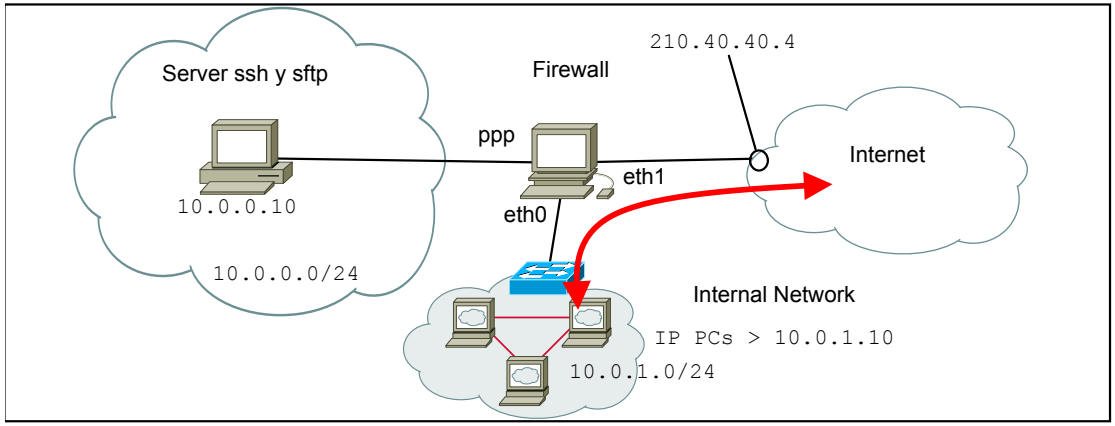
- Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar
- NAT dinámico: suponemos se reserva el rango 210.40.40.10-210.40.40.40

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

2.1.4 – iptables

FIB

Ejemplos



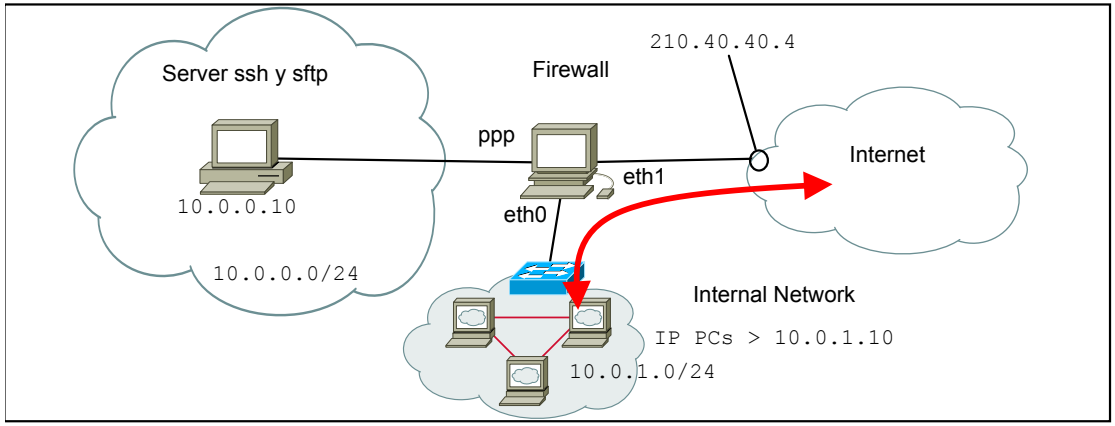
- Configurar el firewall para que los hosts de la red 10.0.1.0/24 tenga acceso al servicio HTTP de Internet y que este pueda solo contestar
- NAT dinámico: suponemos se reserva el rango 210.40.40.10-210.40.40.40

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 que iptables mantiene estados e Internet ya solo podrá contestar (no puede empezar una comunicación)

2.1.4 – iptables

Ejemplos



- 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
- PAT: todos los hosts usan la @IP publica del Firewall

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
- Como antes, no hace falta poner la vuelta ya que iptables mantiene estados e Internet ya solo podrá contestar (no puede empezar una comunicación)

Tema 2. Seguridad en las redes

FIB

Índice

- Introducción
- Firewalls
  - Arquitecturas
  - Tecnologías
  - Reglas de filtrado
- Seguridad en IP
  - Introducción y usos
  - Arquitecturas VPN
  - Familia IPsec
- Sistemas de detección de intrusos (IDS)
  - Funcionalidades y arquitecturas
  - Tecnologías

2.2.1 – Seguridad en IP

FIB

Introducción

- Capa de red
  - Todos los datagramas, independientemente de la aplicación y de la capa de transporte, deben encaminarse de forma segura en la red
  - Se pueden usar protocolos y dispositivos específicos (hardware y/o software) de protección y control de acceso (por ejemplo?)

2.2.1 – Seguridad en IP

FIB

Introducción

- Capa de red
  - Todos los datagramas, independientemente de la aplicación y de la capa de transporte, deben encaminarse de forma segura en la red
  - Se pueden usar protocolos y dispositivos específicos (hardware y/o software) de protección y control de acceso (por ejemplo?)
- Internet Protocol Security (IPsec)
  - RFC 4301 y 4309 (fundamentos)
  - Es el protocolo de seguridad de la capa de red más utilizado
  - Es un conjunto de protocolos (familia de protocolos) cuya función es asegurar las comunicaciones sobre IP autenticando y/o cifrando cada paquete en la capa de red

2.2.1 – Seguridad en IP

FIB

Introducción

- Capa de red
  - Todos los datagramas, independientemente de la aplicación y de la capa de transporte, deben encaminarse de forma segura en la red
  - Se pueden usar protocolos y dispositivos específicos (hardware y/o software) de protección y control de acceso (por ejemplo?)
- Internet Protocol Security (IPsec)
  - RFC 4301 y 4309 (fundamentos)
  - Es el protocolo de seguridad de la capa de red más utilizado
  - Es un conjunto de protocolos (familia de protocolos) cuya función es asegurar las comunicaciones sobre IP autenticando y/o cifrando cada paquete en la capa de red
- El uso más común de IPsec es proporcionar servicios VPN (red privada virtual)

2.2.1 – Seguridad en IP

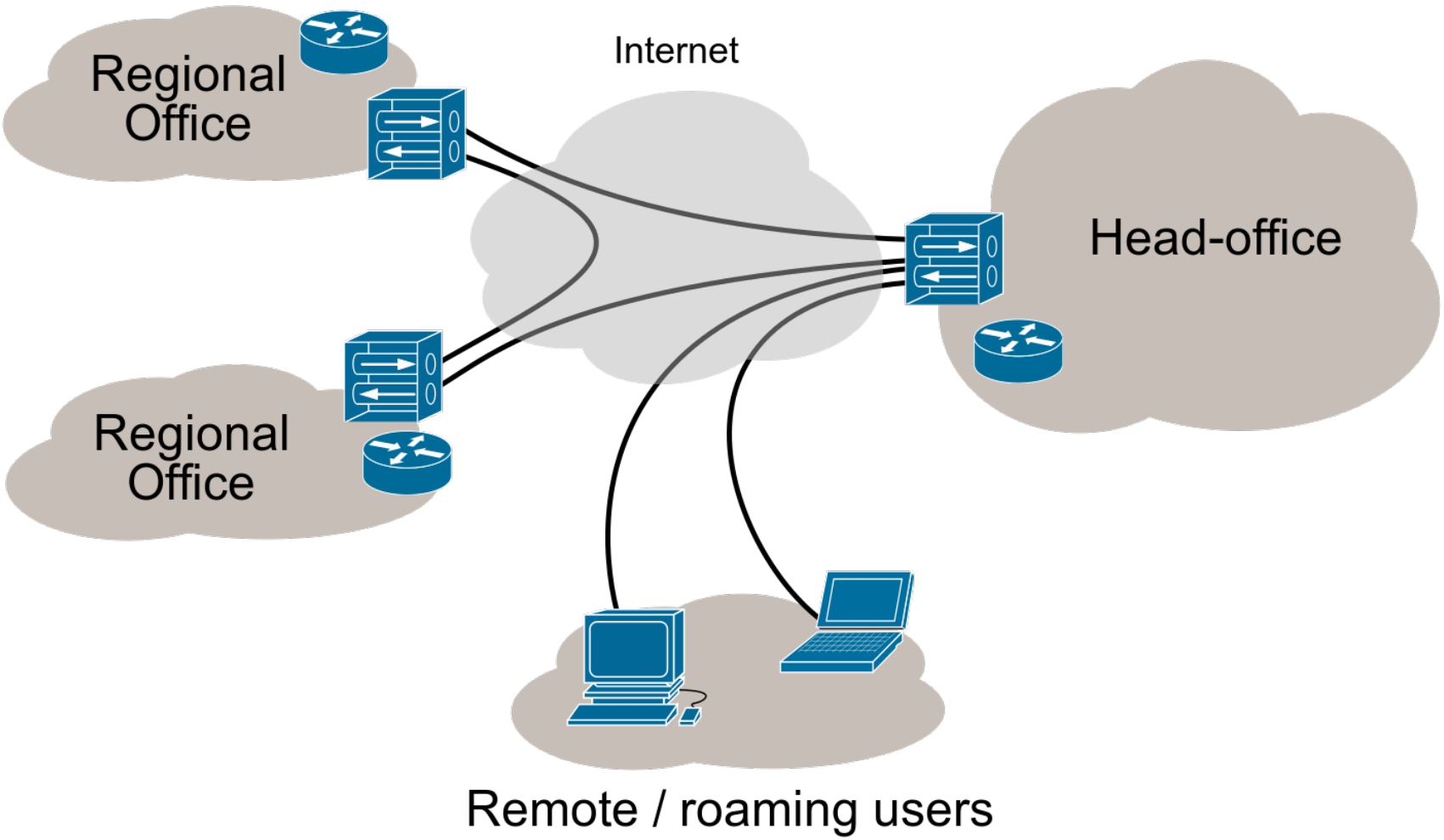
FIB

Definición de VPN

- Es un mecanismo para crear una conexión segura entre un dispositivo y una red, o entre dos redes, utilizando un medio de comunicación inseguro como es Internet
- Se suele usar para
  - Extender el acceso a una red privada a usuarios que no tienen acceso directo a ella. Como por ejemplo una red de oficina que permite el acceso seguro desde fuera a través de Internet
  - Evitar restricciones determinadas en Internet (censura, posición geográfica, etc.)
- Los beneficios de una VPN incluyen
  - Seguridad
  - Costes reducidos (no se necesita una infraestructura privada de interconexión)
  - Mayor flexibilidad para los trabajadores remotos

2.2.1 – Seguridad en IP

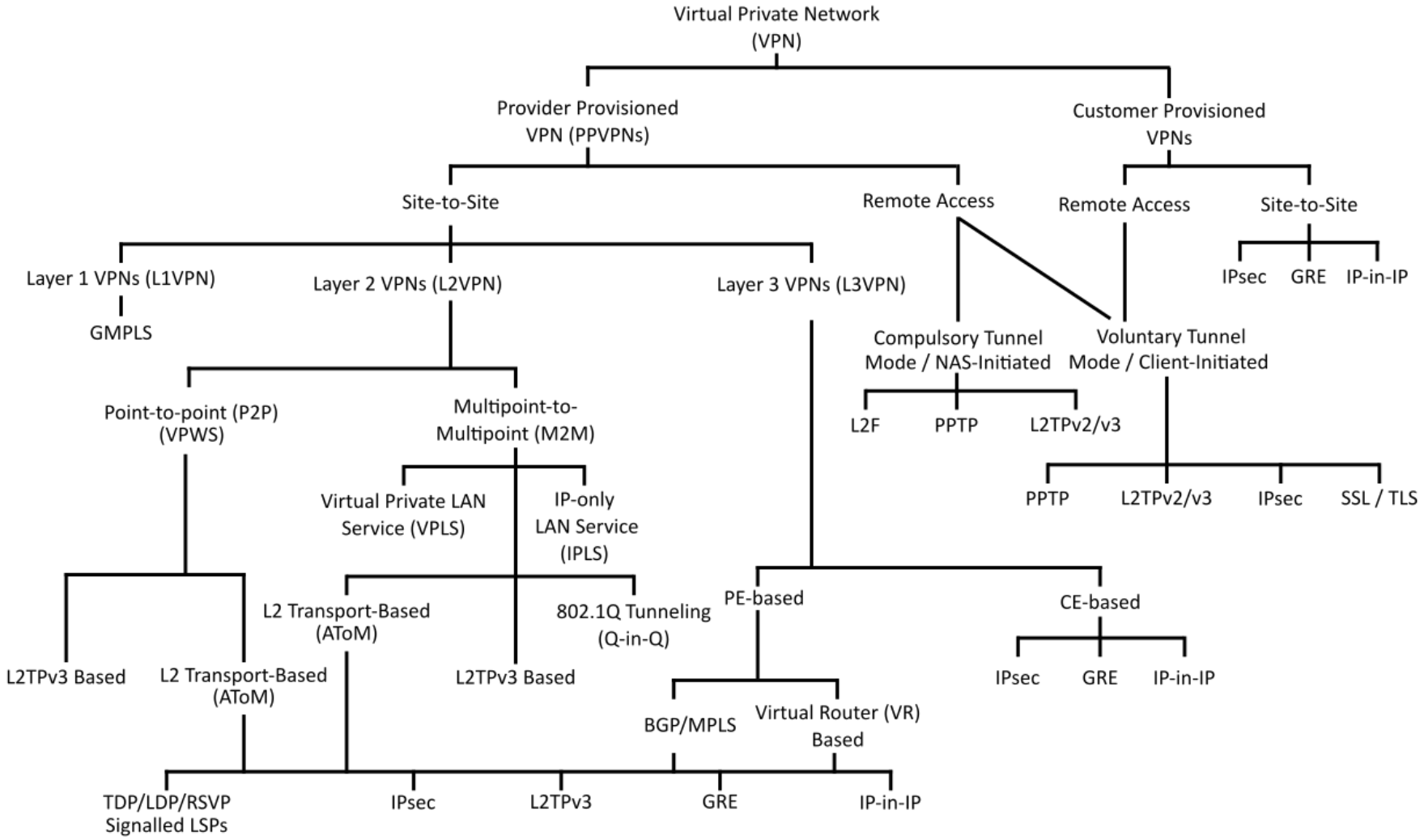
Definición de VPN



Fuente imagen: https://en.wikipedia.org/wiki/Virtual_private_network

2.2.1 – Seguridad en IP

Protocolos para VPN



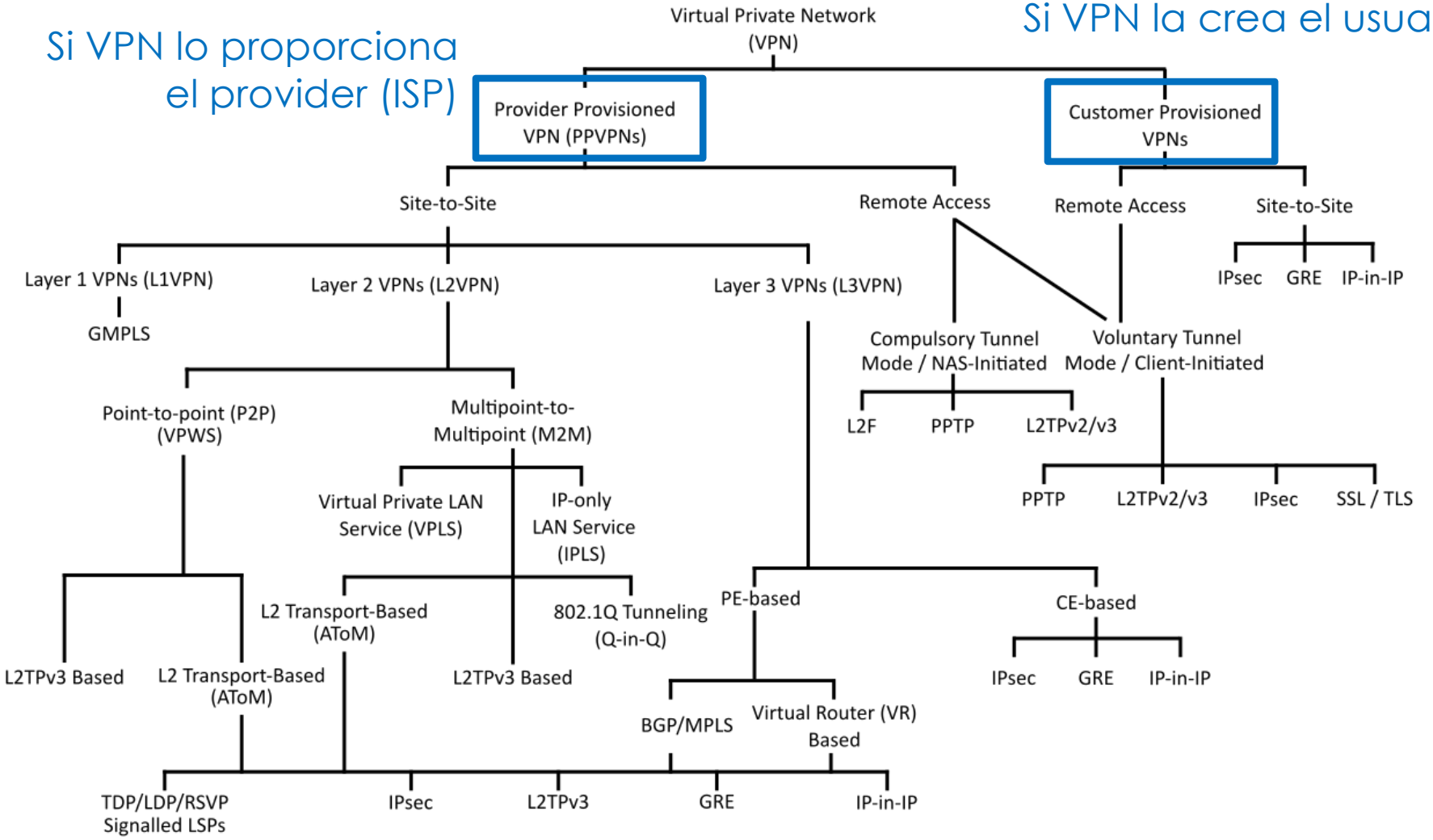
Fuente imagen: https://en.wikipedia.org/wiki/Virtual_private_network

2.2.1 – Seguridad en IP

Protocolos para VPN

Si VPN lo proporciona el provider (ISP)

Si VPN la crea el usuario

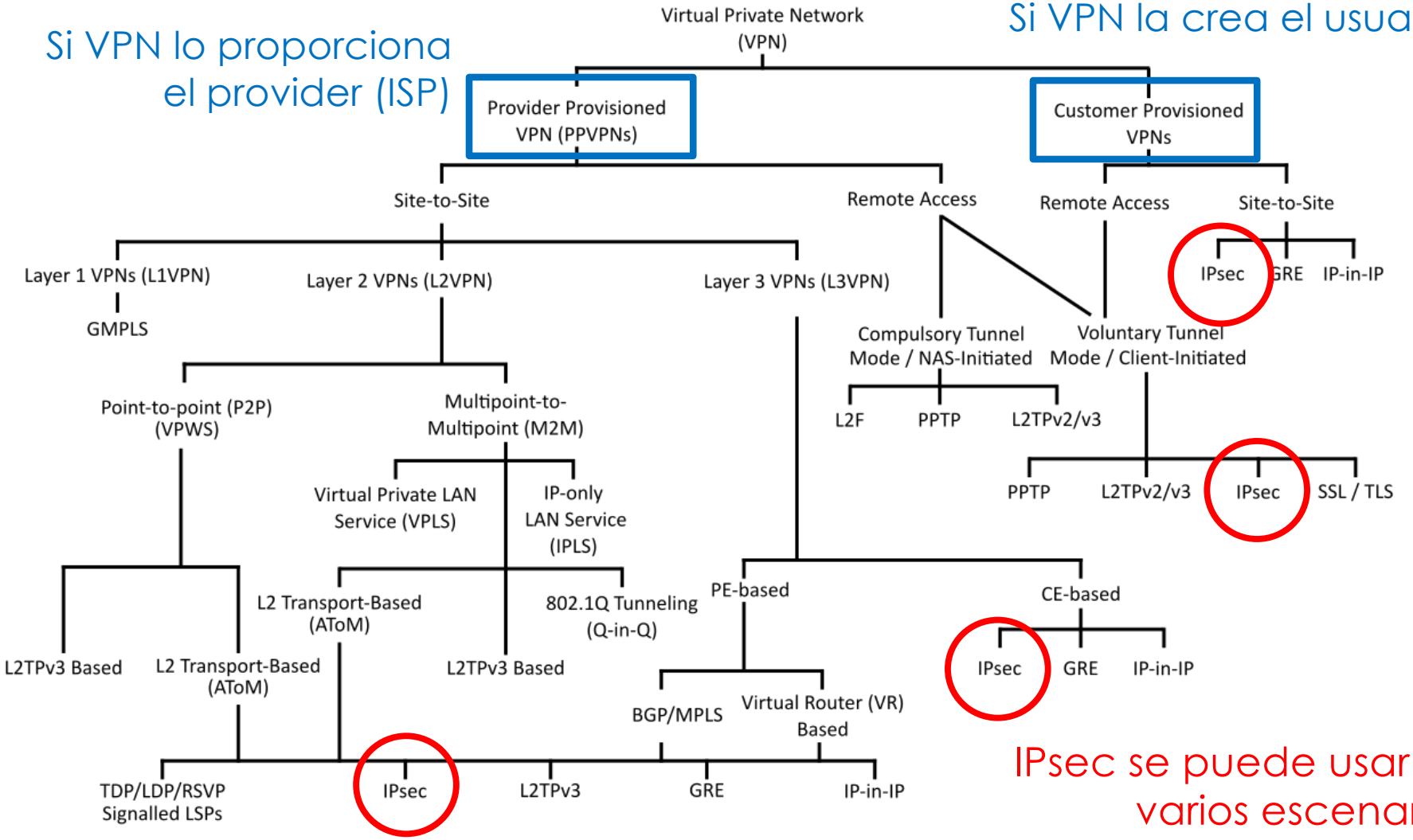


2.2.1 – Seguridad en IP

Protocolos para VPN

Si VPN lo proporciona el provider (ISP)

Si VPN la crea el usuario



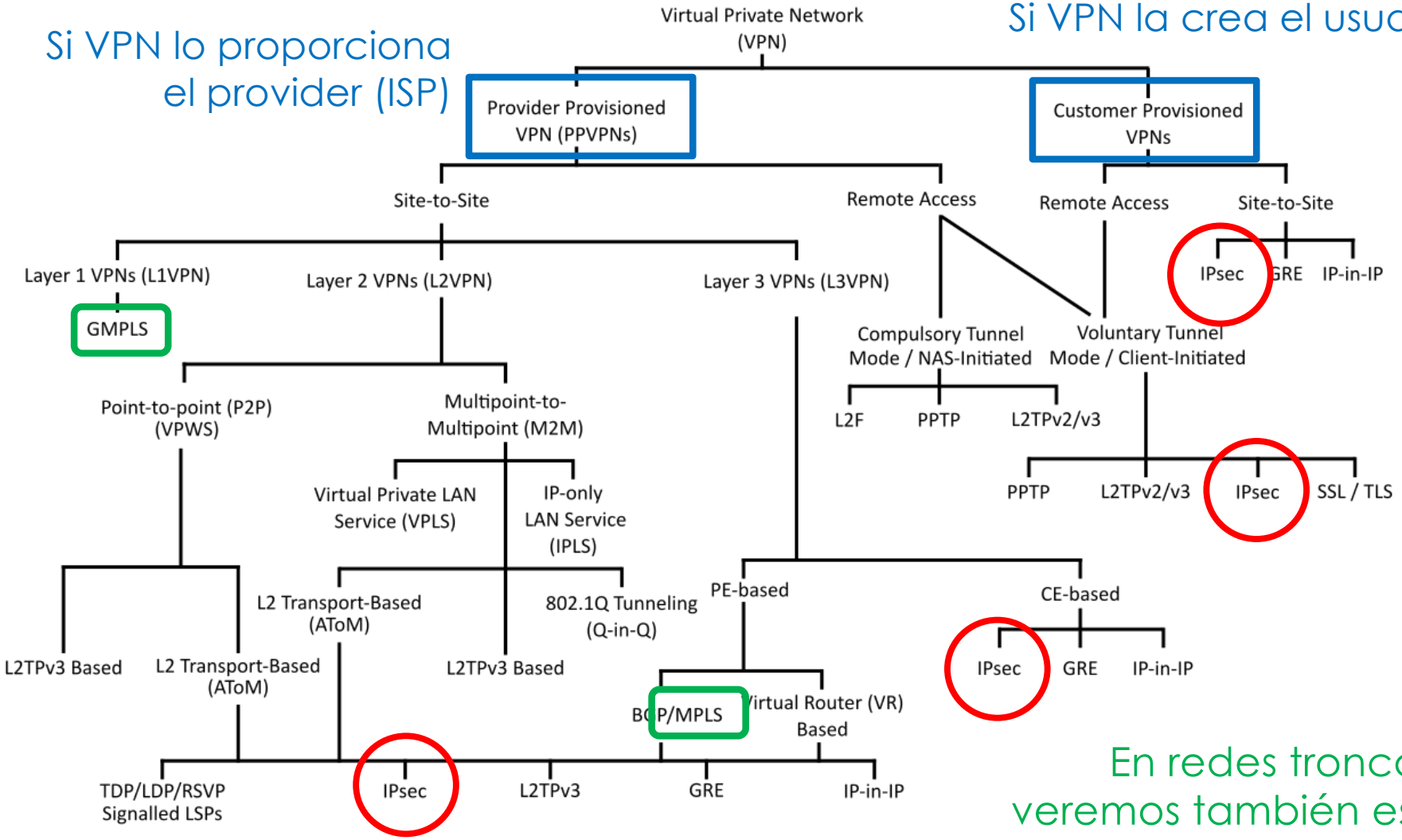
IPsec se puede usar en varios escenarios diferentes

2.2.1 – Seguridad en IP

Protocolos para VPN

Si VPN lo proporciona el provider (ISP)

Si VPN la crea el usuario



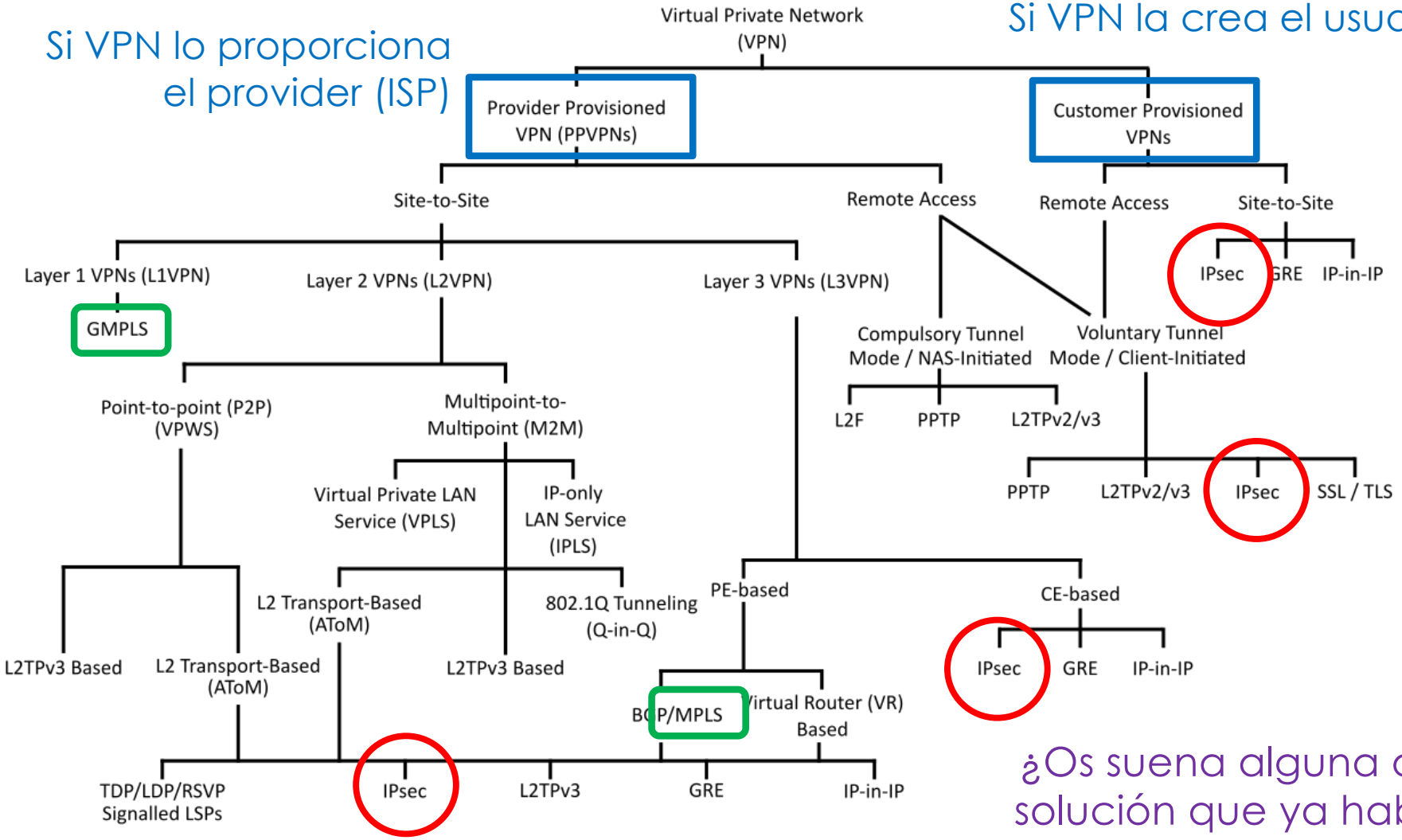
En redes troncales veremos también estas dos tecnologías

2.2.1 – Seguridad en IP

Protocolos para VPN

Si VPN lo proporciona el provider (ISP)

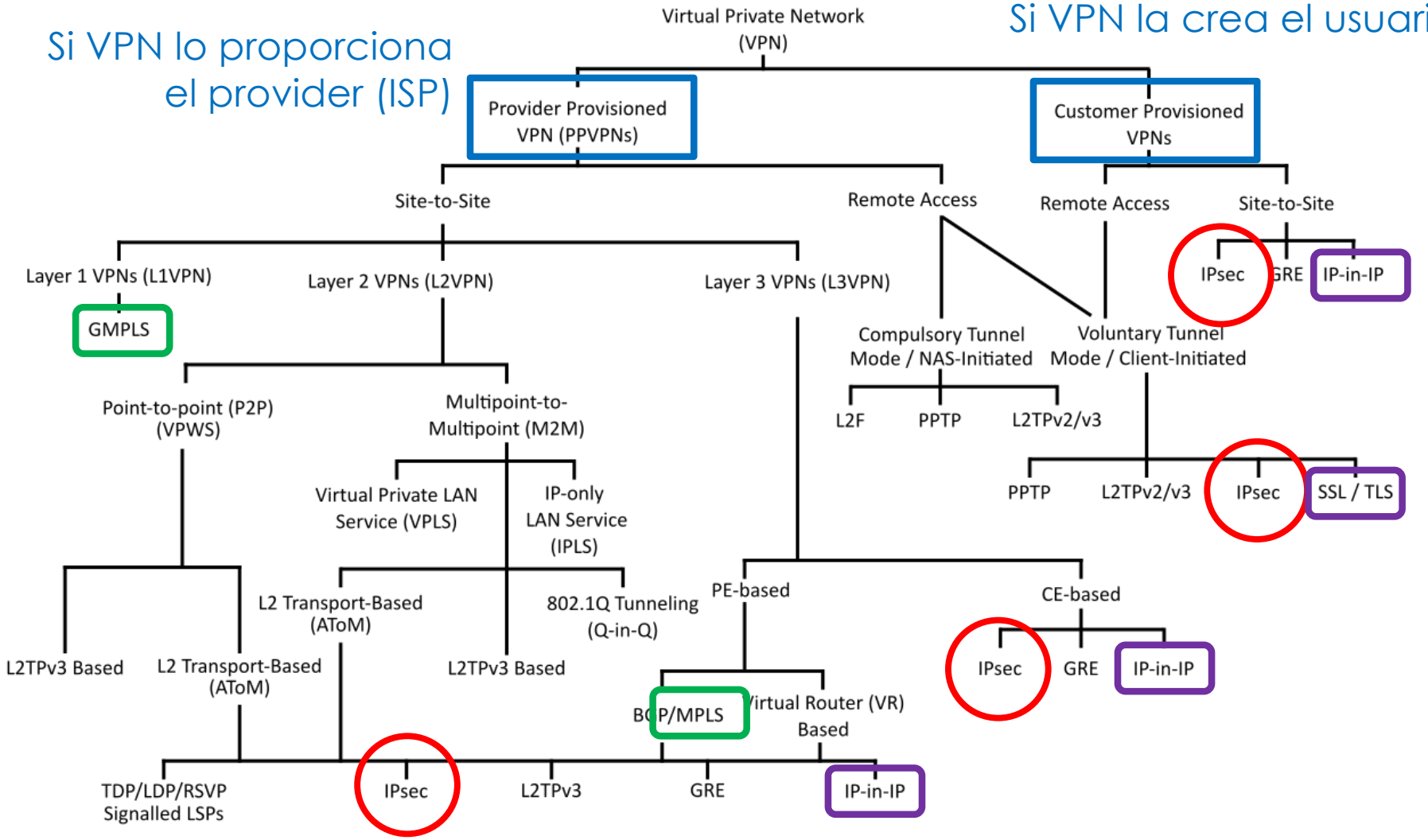
Si VPN la crea el usuario



¿Os suena alguna otra solución que ya habéis visto en otra asignatura?

2.2.1 – Seguridad en IP

Protocolos para VPN



2.2.2 – Arquitecturas de VPN

FIB

Escenarios

- Gateway-to-Gateway
  - También se suele llamar site-to-site
- Host-to-Gateway
  - También se suele llamar remote access
- Host-to-host

2.2.2 – Arquitecturas de VPN

FIB

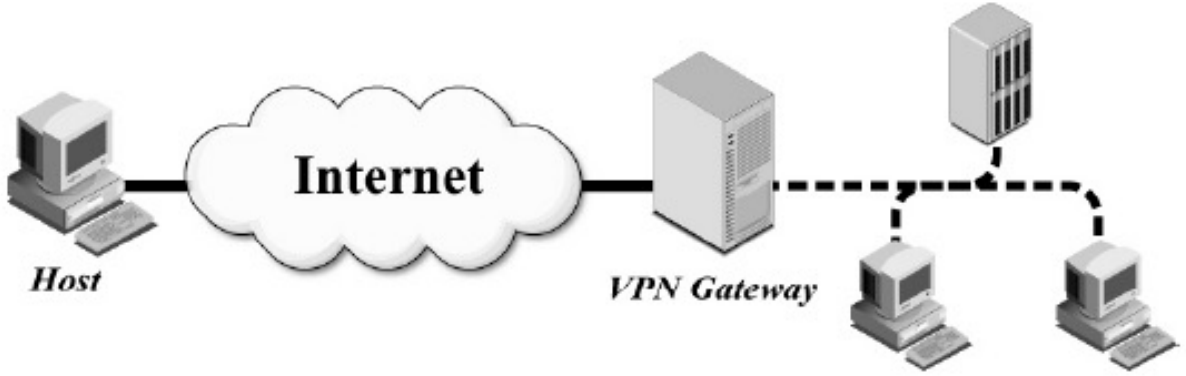
Gateway-to-Gateway

- 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
- Suele ser una **conexión permanente**
- El encaminamiento en cada sistema está configurado de modo que los paquetes que van de un sistema al otro, se encaminan a través de 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

2.2.2 – Arquitecturas de VPN

Host-to-Gateway

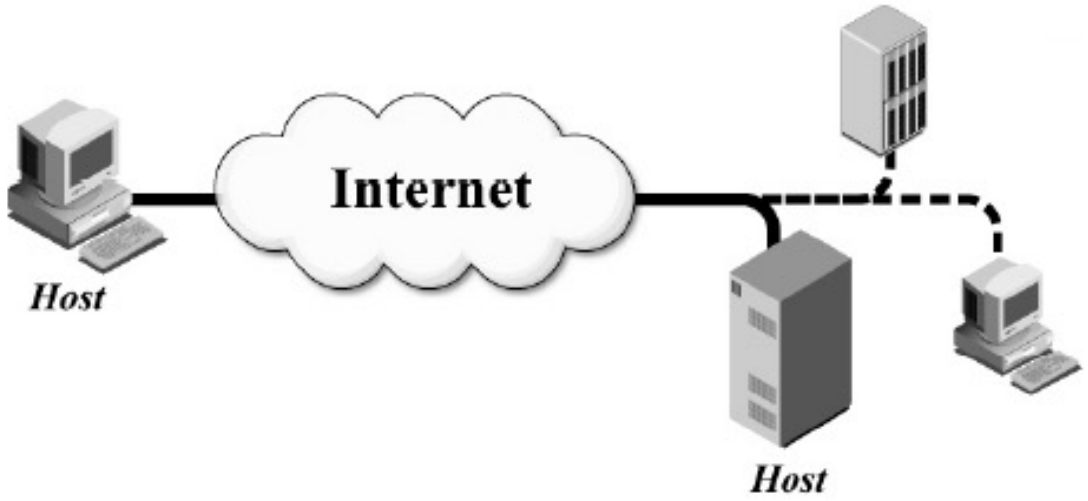
- Este modelo se utiliza para proporcionar acceso remoto seguro desde una red externa a los servicios internos de un sistema/empresa
- La empresa implementa un gateway VPN de entrada en su sistema; cada usuario establece una conexión VPN entre su host y este gateway
- La **conexión VPN la establece el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



2.2.2 – Arquitecturas de VPN

Host-to-Host

- Usado generalmente en casos especiales y puntuales; p.e., un administrador necesita acceder remotamente a un ordenador
- La **conexión la establece uno de los dos extremos**
- Este modelo es el único que proporciona seguridad extremo a extremo: 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



2.2.3 – Protocolos IPsec

FIB

Fundamentos

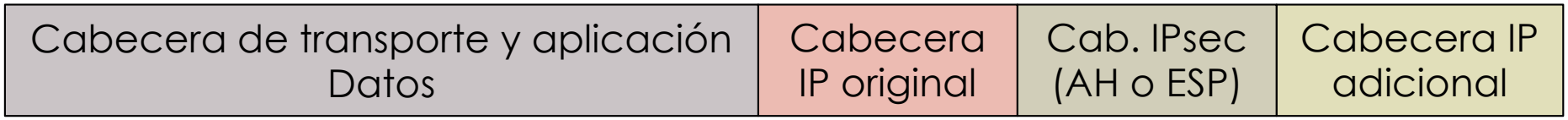
- IPsec es una **colección de protocolos** que ayudan a proteger las comunicaciones a través de redes IP
- Dependiendo de su implementación, puede proporcionar cualquier combinación de los siguientes tipos de protección:
  - **Confidencialidad**: aseguro que el datagrama no puede ser leído por alguien 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
  - **Protección de acceso**: los extremos pueden filtrar para asegurar que solo los usuarios autorizados IPsec pueden acceder a recursos particulares de la red
  - **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

2.2.3 – Protocolos IPsec

FIB

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-to-Gw y H-to-Gw

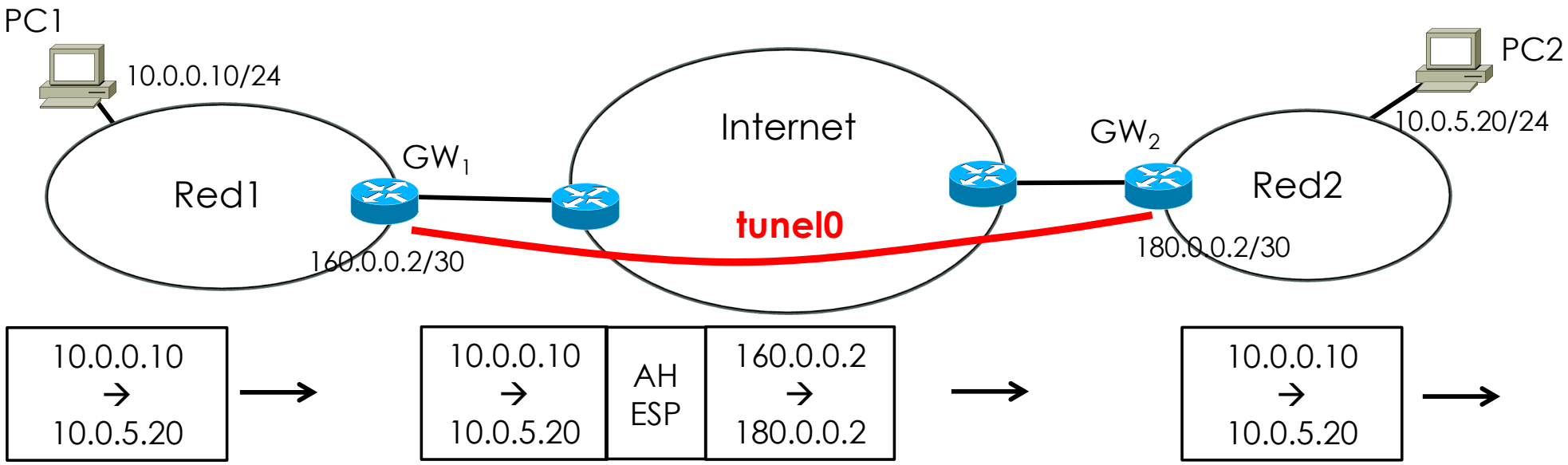


2.2.3 – Protocolos IPsec

FIB

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-to-Gw y H-to-Gw



2.2.3 – Protocolos IPsec

FIB

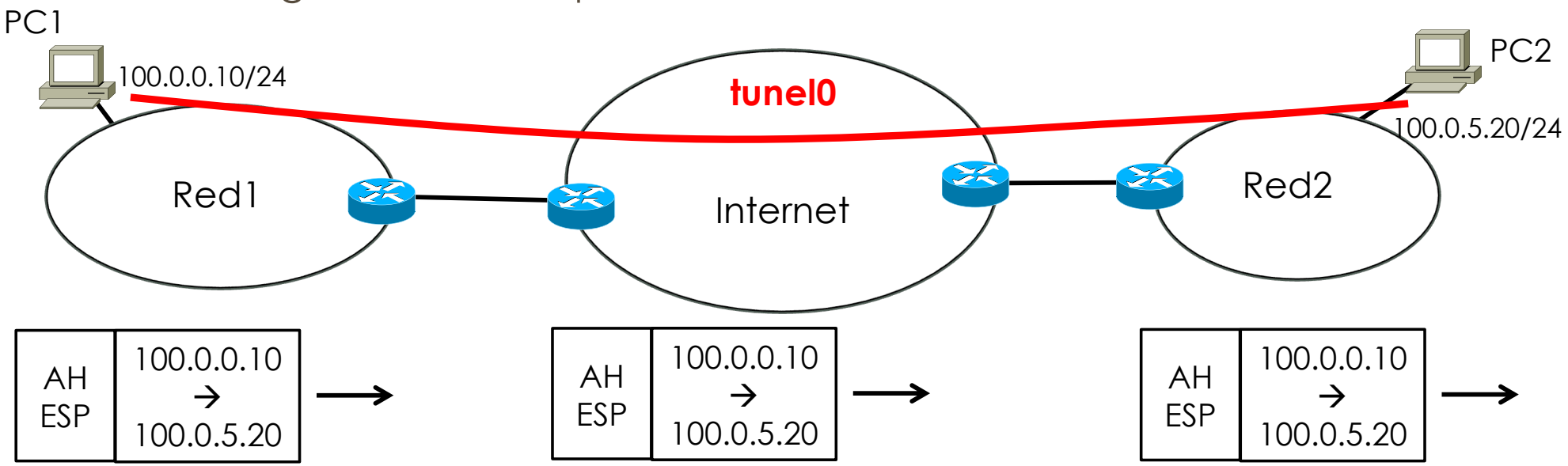
Modos de funcionamiento

- Transport mode
  - No se añade ninguna cabecera IP adicional
    - Sería inútil ya que una eventual cabecera IP adicional contendría exactamente lo mismo que la cabecera IP original
  - Se añade entre las dos cabecera, una cabecera IPsec (el contenido de esta cabecera depende del protocolo usado)
  - Se usa generalmente para VPN H-to-H

2.2.3 – Protocolos IPsec

Modos de funcionamiento

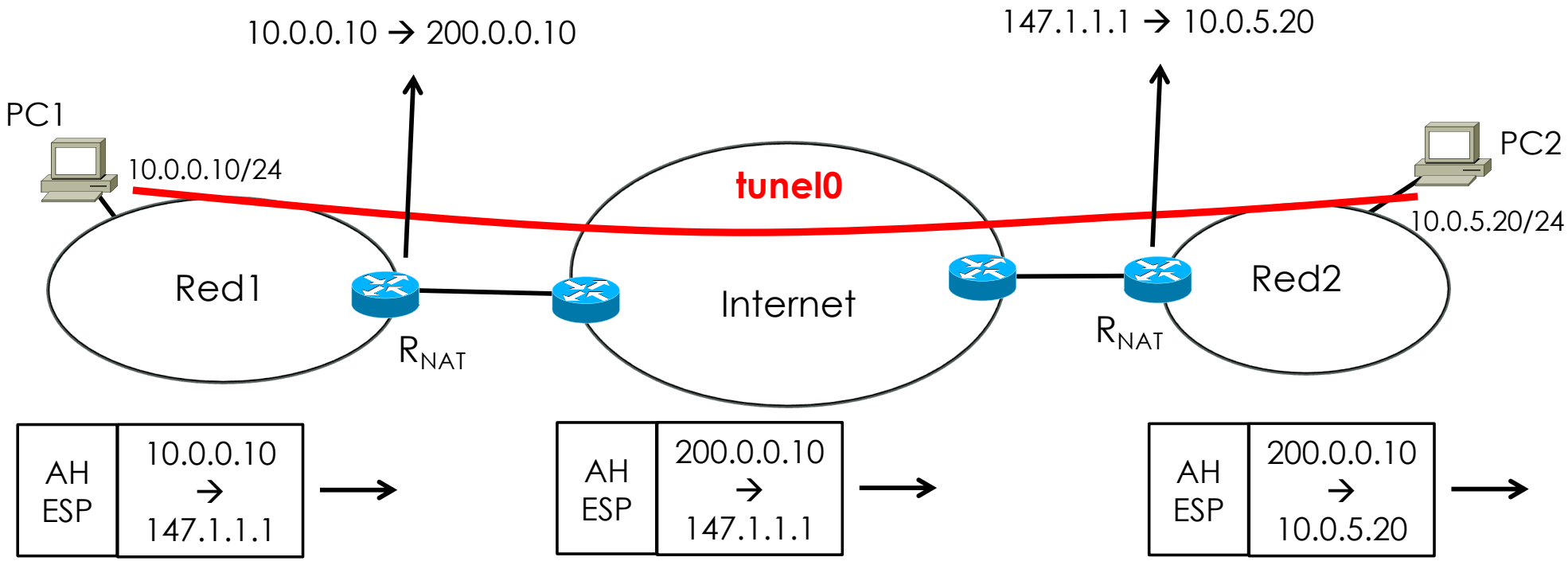
- Transport mode
  - No se añade ninguna cabecera IP adicional
    - Sería inútil ya que una eventual cabecera IP adicional contendría exactamente lo mismo que la cabecera IP original
  - Se añade entre las dos cabecera, una cabecera IPsec (el contenido de esta cabecera depende del protocolo usado)
  - Se usa generalmente para VPN H-to-H



2.2.3 – Protocolos IPsec

Modos de funcionamiento

- Transport mode
  - Si los hosts usan IP privadas, los routers NAT traducen las IPs a públicas



2.2.3 – Protocolos IPsec

FIB

Protocolos principales

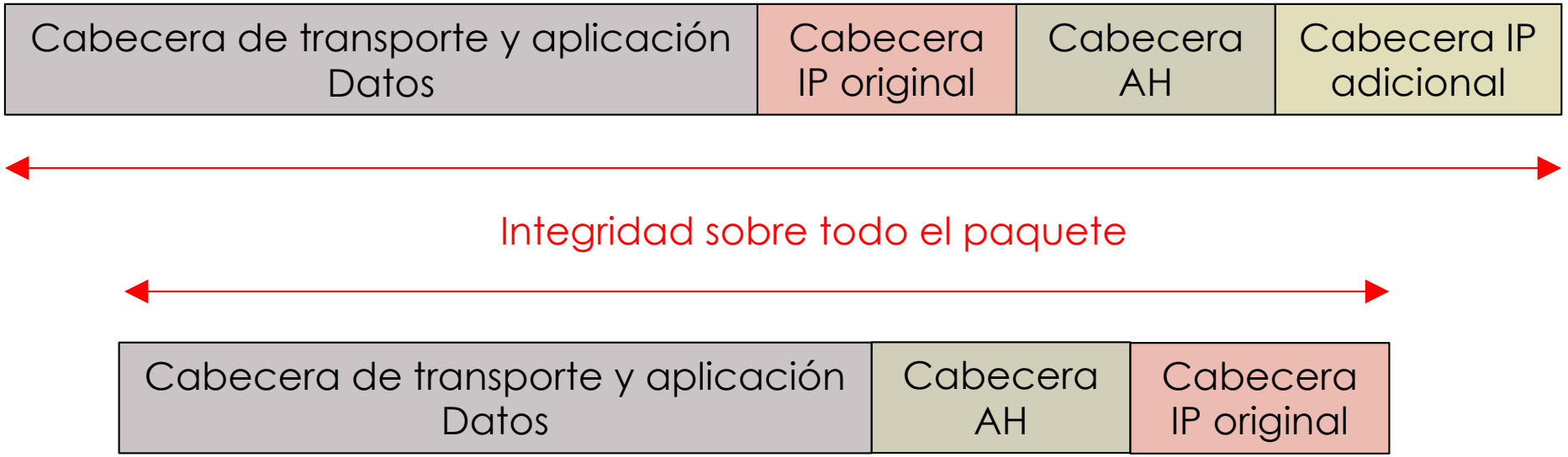
- Authentication Header (AH)
  - Proporciona integridad, autenticación y no repudio si se eligen los algoritmos criptográficos apropiados
- Encapsulating Security Payload (ESP)
  - Proporciona confidencialidad y la opción de autenticación y protección de integridad
- Security Association (SA)
  - Por 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

2.2.3 – Authentication Header (AH)

FIB

Servicios proporcionados

- AH proporciona protección de integridad para cabeceras IP y datos, así como autenticación de usuario
  - Opcionalmente, puede proporcionar protección de acceso
  - En cambio, no puede cifrar ninguna parte de los paquetes
- Se puede usar en Transport o Tunnel mode



2.2.3 – Authentication Header (AH)

FIB

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 falso positivo en el control de integridad
  - Para evitar este problema, la integridad se verifica excluyendo estos campos variables
- 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

2.2.3 – Encapsulating Security Payload

FIB

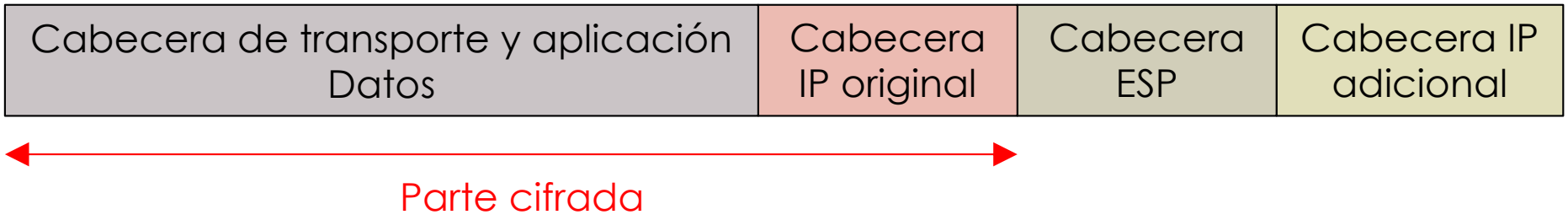
Servicios proporcionados

- 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

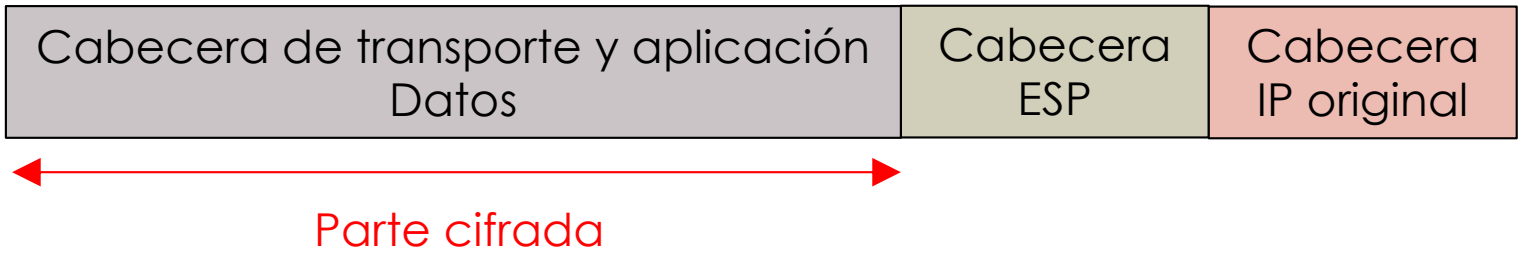
2.2.3 – Encapsulating Security Payload

Cifrado

- Tunnel mode



- Transport mode



2.2.3 – Security Association

FIB

Servicios proporcionados

- 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)

2.2.3 – Security Association

FIB

Servicios proporcionados

- Algoritmo usado para cifrar el paquete IP, por ejemplo AES
- Función Hash usada para garantizar la integridad, como SHA256
- La vida útil de esta asociación
- La clave secreta, por ejemplo usando Diffie-Hellman
- Algoritmo de autenticación usado
  - Una clave previamente conocida por ambos extremos
  - 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

2.2.4 – Ejemplos

FIB

VPN Gw-to-Gw

- Se necesita mantener dos sistemas A y B conectados entre ellos a través de Internet que proporcione confidencialidad

2.2.4 – Ejemplos

FIB

VPN Gw-to-Gw

- Se necesita mantener dos sistemas A y B conectados entre ellos a través de Internet que proporcione confidencialidad
- Se establece una conexión IPsec permanente entre el gateway G_A y el G_B con ESP en modo Tunnel
  - G_A inicia una negociación con G_B para definir los parámetros de seguridad (entre otros, autenticación, integridad del payload y cifrado) y crear la SA con identificador SPI_{AB}
  - G_A usa los parámetros configurados en la SA para negociar la conexión IPsec. En este caso se usa ESP en tunnel mode

2.2.4 – Ejemplos

FIB

VPN Gw-to-Gw

- Se necesita mantener dos sistemas A y B conectados entre ellos a través de Internet que proporcione confidencialidad
- Se establece una conexión IPsec permanente entre el gateway G_A y el G_B con ESP en modo Tunnel
- Un usuario H_A del sistema A quiere enviar datos al usuario H_B del sistema B
  - H_A envía datagramas IP al H_B
  - La red del sistema A se ocupa de re-enviar los datagrama hacia G_A
  - G_A recibe los datagramas de H_A y ve que el destino está en el sistema B
  - G_A cifra los datagramas según los parámetros indicados en la SPI_{AB}
  - G_A añade una nueva cabecera IP (@IP origen G_A e @IP destino G_B) a todos los datagramas
  - G_A envía los datagramas por Internet con destino G_B

2.2.4 – Ejemplos

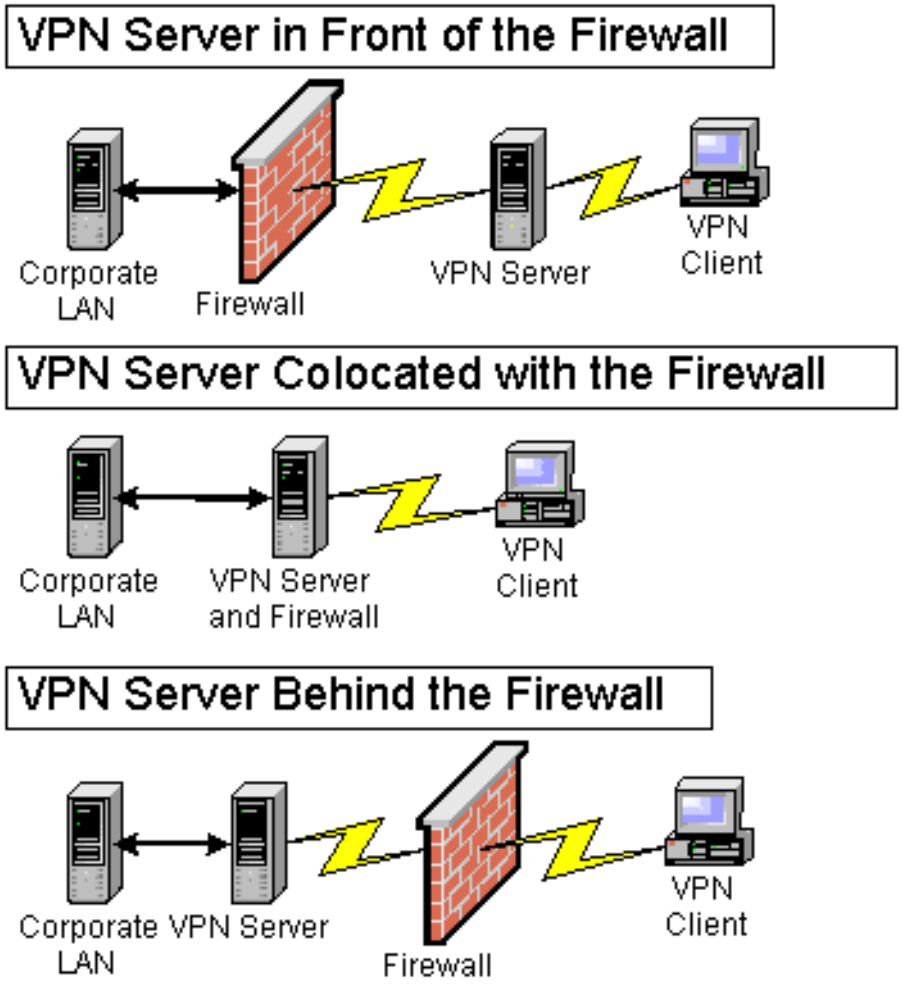
FIB

VPN Gw-to-Gw

- Se necesita mantener dos sistemas A y B conectados entre ellos a través de Internet que proporcione confidencialidad
- Se establece una conexión IPsec permanente entre el gateway G_A y el G_B con ESP en modo Tunnel
- Un usuario H_A del sistema A quiere enviar datos al usuario H_B del sistema B
  - G_B recibe el paquete y usa el SPI_{AB} de la cabecera ESP para reconocer la conexión segura
  - G_B quita la cabecera IP adicional, comprueba la integridad del paquete y descifra el paquete original
  - G_B envía el paquete a H_B

2.2.5 – Firewall + VPN

Soluciones

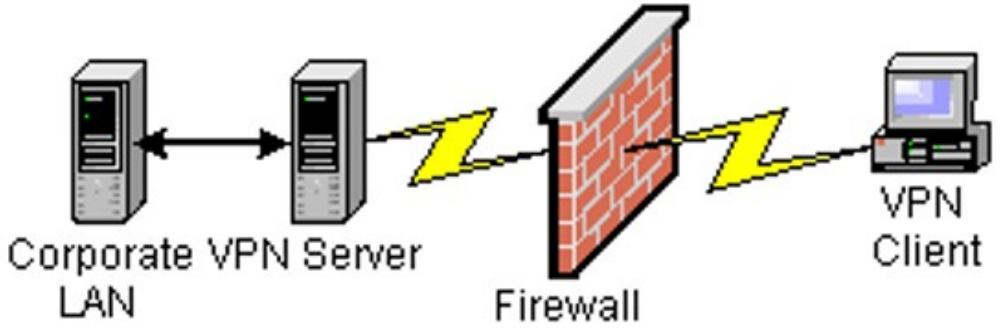


Fuente imagen: https://www.techrepublic.com/article/configuring-vpn-connections-with-firewalls/

2.2.5 – Firewall + VPN

VPN Gateway detrás del Firewall

- El VPN Gateway (VPN Server en la figura) se suele poner en la DMZ o en la red interna



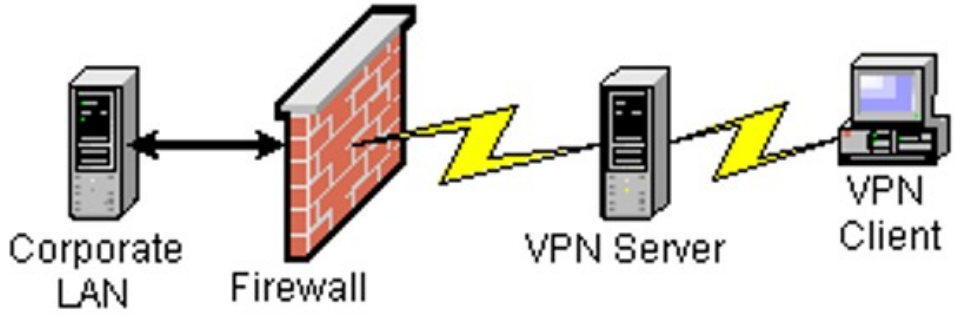
- Ventajas
  - El VPN Gateway está protegido por el Firewall
  - La configuración es como un servidor más en la DMZ/red interna
- Desventajas
  - El Firewall no puede controlar el contenido de los paquetes
  - El Firewall necesita reglas de filtrado para dejar pasar el trafico VPN

2.2.5 – Firewall + VPN

FIB

VPN Gateway antes del Firewall

- El VPN Gateway (VPN Server en la figura) está entre el Firewall e Internet



- Ventajas
  - El Firewall puede controlar que recursos internos son accesibles para clientes autenticados por el VPN Gateway
  - Si un atacante compromete el VPN Gateway, el Firewall puede aún actuar para bloquear este atacante
- Desventajas
  - El trafico entre el VPN Gateway y el Firewall no está cifrado

2.2.5 – Firewall + VPN

VPN Gateway y Firewall en un mismo dispositivo

- El VPN Gateway suele ser el primero que actúa sobre trafico desde Internet
- Luego se aplican las reglas del Firewall

- Ventajas
  - El Firewall puede controlar que recursos internos son accesibles para clientes autenticados por el VPN Gateway
  - Todo el trafico va cifrado
- Desventajas
  - Un único dispositivo para dos cosas
  - Único punto de defensa

Tema 2. Seguridad en las redes

FIB

Índice

- Introducción
- Firewalls
  - Arquitecturas
  - Tecnologías
  - Reglas de filtrado
- Seguridad en IP
  - Introducción y usos
  - Arquitecturas VPN
  - Familia IPsec
- Sistemas de detección de intrusos (IDS)
  - Funcionalidades y arquitecturas
  - Tecnologías

2.3 – Sistemas de detección de intrusos

FIB

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
  - Herramienta especializada que sabe leer e interpretar el contenido de los archivos de registro (logs) de routers, firewalls, servidores y otros dispositivos de red

2.3 – Sistemas de detección de intrusos

FIB

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

2.3 – Sistemas de detección de intrusos

FIB

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 (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

2.3 – Sistemas de detección de intrusos

FIB

Clasificación por funcionalidades

- Signature-based IDS
- Anomaly-based IDS

2.3.1 – Clasificación por funcionalidades

FIB

Signature-based IDS

- De manera similar al antivirus, los IDS basados en firmas mantiene una base de datos 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
  - Por ejemplo, GET /etc/passwd
- Si una regla coincide, se activa una alerta
- Sistema simple y efectivo
- Problema: nuevos ataques no se pueden reconocer ya que no hay entradas en la base de datos

2.3.1 – Clasificación por funcionalidades

FIB

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, …

2.3 – Sistemas de detección de intrusos

FIB

Clasificación por arquitecturas

- Network IDS
- Host IDS
- Distributed IDS

2.3.1 – Clasificación por arquitecturas

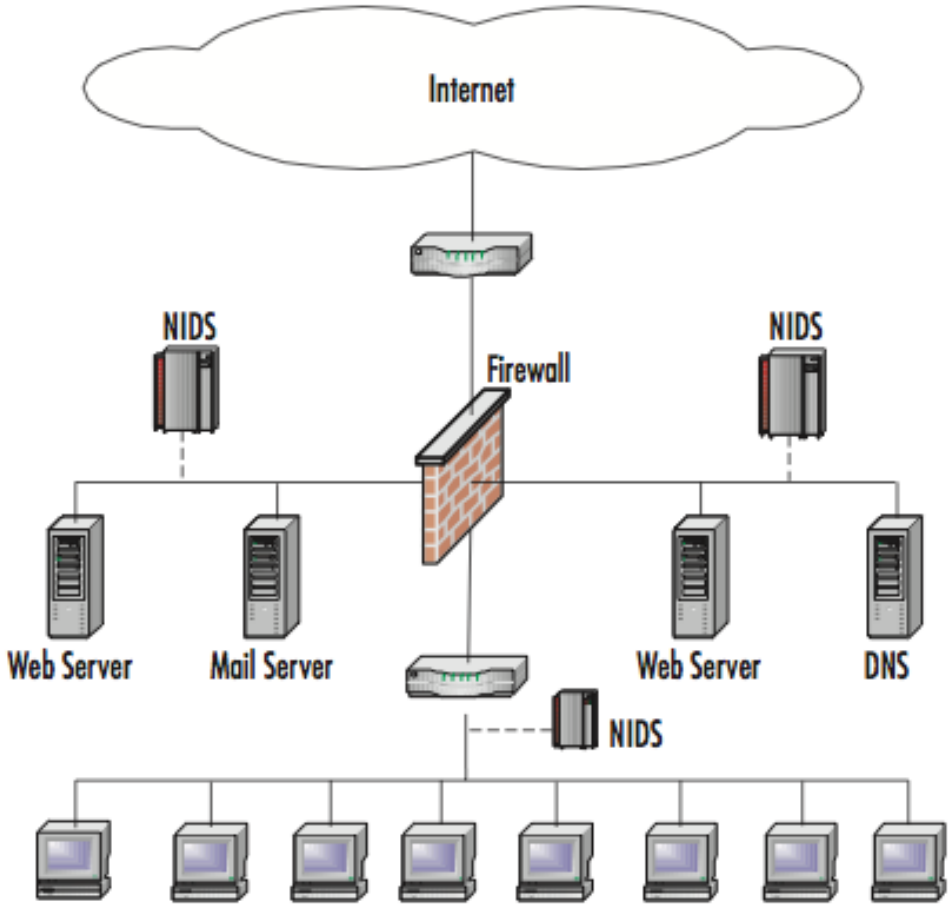
FIB

Network IDS (NIDS)

- Se suelen usar dispositivos llamados sensores que se instalan en varios puntos de la infraestructura
- Monitorean el tráfico de red en segmentos de red o dispositivos de red particulares
- Analizan los protocolos de red, transporte y aplicación para identificar actividades sospechosas y buscan de patrones indicativos de un ataque

2.3.1 – Clasificación por arquitecturas

Network IDS (NIDS)



- 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

2.3.1 – Clasificación por arquitecturas

FIB

Network IDS (NIDS)

- Recopilación de información
  - Los NIDS pueden recopilar información sobre hosts y actividad de red para identificar usuarios, SO, 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 pueden usar firmas y/o detección basada en anomalías
- Prevención
  - Una vez que se activa una alerta, se pueden abortar otras conexiones similares

2.3.1 – Clasificación por arquitecturas

FIB

Host IDS (HIDS)

- 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, …
- Funciones principales: verificadores de integridad del sistema (SIV), monitores de archivos de registro (LFM) y sistemas de trampa (tipo honeypot*)

2.3.1 – Clasificación por arquitecturas

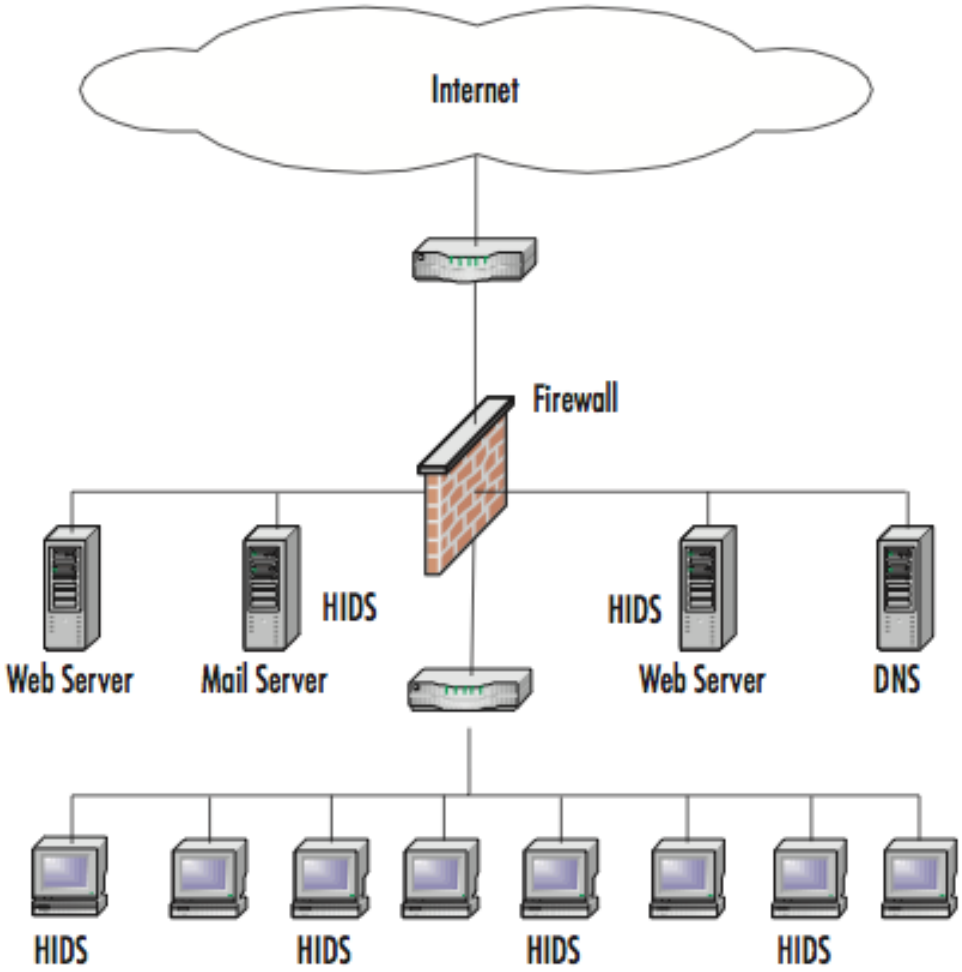
FIB

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
  - …

2.3.1 – Clasificación por arquitecturas

Host IDS (HIDS)



- HIDS instalados en host concretos
- Pueden ser servidores públicos o privados
- Pueden ser hosts de la red interna particularmente vulnerables o importantes

2.3.1 – Clasificación por arquitecturas

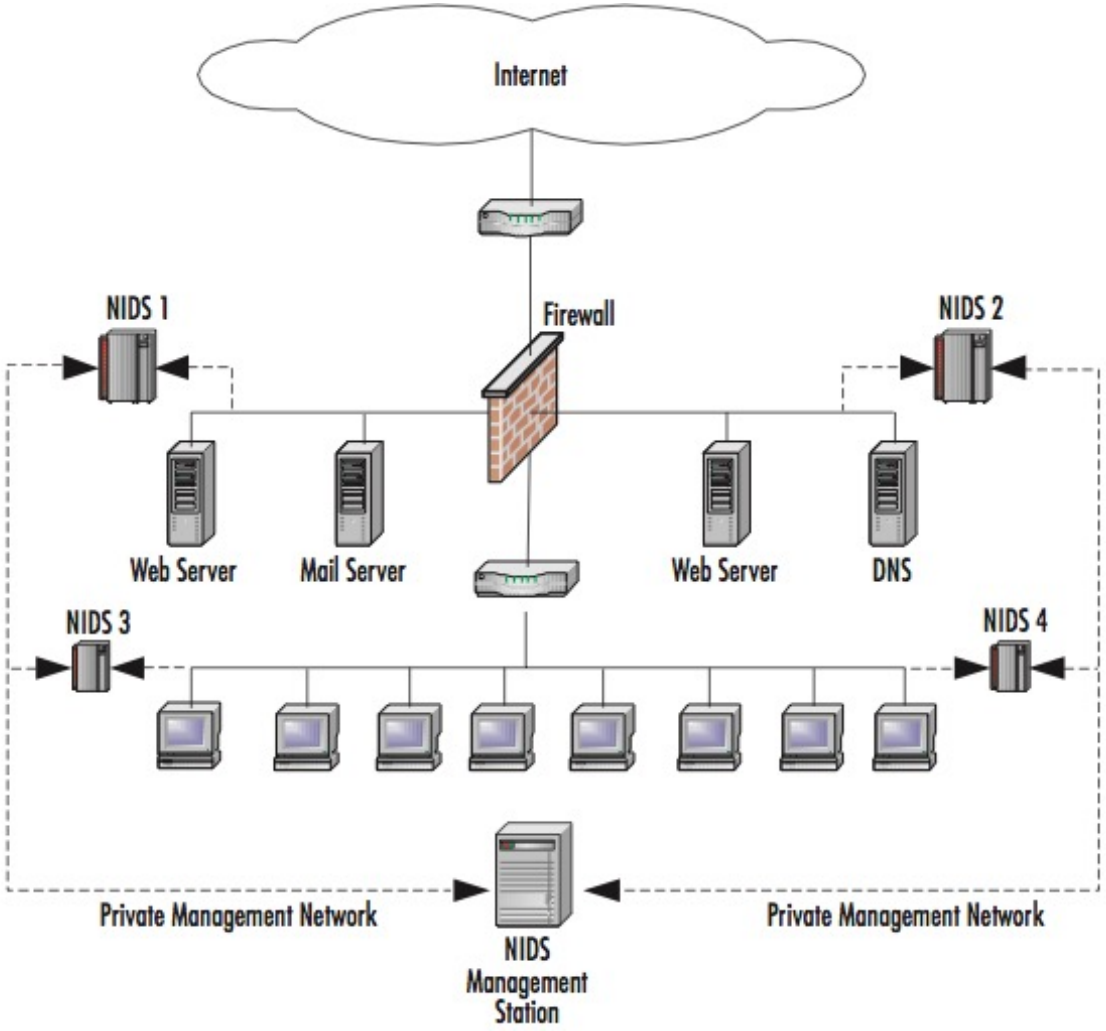
FIB

Distributed IDS (DIDS)

- 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

2.3.1 – Clasificación por arquitecturas

Distributed IDS (DIDS)



- Un gestor recibe toda la información de los sensores HIDS y NIDS
- Este la analiza y decide en tiempo real
- Puede actuar de diferentes maneras: alarma, desconexión, apagar, aislamiento, etc.

2.3.2 – Tecnologías

FIB

Host IDS (HIDS)

- Sagan
  - Linux/FreeBSD/OpenBSD/etc
  - Disponible versión gratuita
  - Es compatible con Snort, sino también con todas las herramientas integrables a Snort como Anaval o Squil
- OSSEC
  - Linux, OpenBSD, FreeBSD, OS X, Solaris y Windows
  - Open source
  - Análisis de registros, verificación de integridad, monitoreo del registro de Windows, detección de rootkits, alertas basadas en el tiempo y respuesta activa
- SolarWinds
  - Ideal para grandes corporaciones (coste a partir de $4.000)
  - Puede procesar registros generados por Windows, Mac-OS, Linux y Unix
  - Técnicamente es un Intrusion Prevention System (IPS) debido a la capacidad de detectar y actuar en lugar de solo analizar y notificar

2.3.2 – Tecnologías

FIB

Distributed IDS (HIDS)

- AlienVault OSSIM
  - Sistema gratuito
  - Descubrimiento y creación de inventario de red para la gestión de activos y la detección de intrusiones
- Suricata
- IBM QRadar
- Security Onion
- SolarWinds Security Event Manager
- Splunk

2.3.2 – Tecnologías

Network IDS (NIDS)

- Snort
- Zeek
- Sguil
- OpenWIPS-ng

2.3.3 – Snort

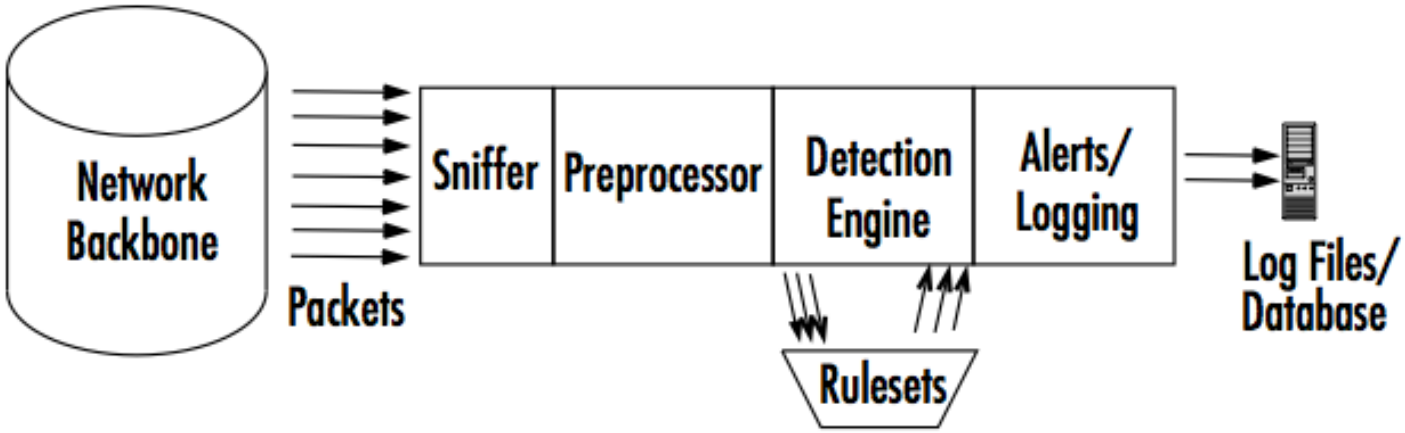
FIB

Introducción

- 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

2.3.3 – Snort

Arquitectura



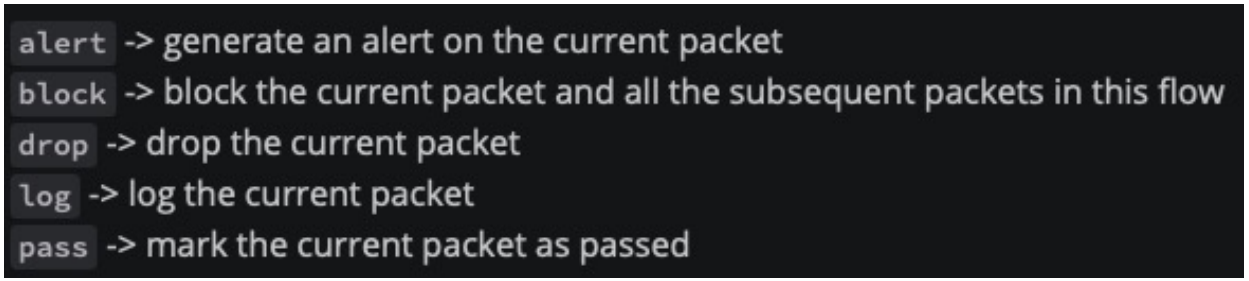
- Sniffer: captura los paquetes
- Preprocessor: determina el tipo de paquetes o comportamiento hay que tratar. Se usan plug-ins para detectar tipos (HTTP) o comportamientos específicos
- Detection engine: compara un paquete con las firmas (reglas), se hay una coincidencia, se envía al siguiente bloque
- 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

2.3.3 – Snort

Formato reglas

```
acción 1 2 3 -> 4 5 (msg: "6"; 7; sid:8; rev:9)
```

- Acción: alert, block, log, drop, pass
- 1: protocolo
- 2: @IP origen
- 3: Puerto origen
- 4: @IP destino
- 5: Puerto destino
- 6: Mensaje de alerta que se quiere enviar
- 7: Parámetros de detección
- 8: Número identificando una determinada regla del snort
- 9: Número de revisión de una determinada regla del snort



Fuente imagen: https://docs.snort.org/rules/headers/actions, acceso 3/2024

2.3.3 – Snort

FIB

Ejemplos

- Una regla para descartar un ping externo a la red interna 10.0.0.0/8
- Para esta regla se usa drop y se descarta si se usa el protocolo ICMP hacia la red interna

```
drop icmp any any -> 10.0.0.0/8 any (msg: “ICMP”;
sid:1000001; rev:1;)
```

2.3.3 – Snort

FIB

Ejemplos

- Una regla para detectar un posible ataque DoS de tipo SYN flooding al servidor HTTP 147.83.2.135
  - Consiste que un origen envía TCP SYN masivos al servidor sin pero completar el 3WH
  - De esta forma el servidor va abriendo sesiones incompletas hasta quedarse sin recursos
- Para esta regla se usa alert, se detecta por envío desde una misma origen de un número elevado de SYN (20 en 60s)

```
alert TCP any any -> 147.83.2.135 80 (msg: "Possible
DoS attack"; flags: S; detection_filter: track by_dst,
count 20, seconds 60; sid:1000001; rev:1;)
```

2.3.3 – Snort

FIB

Ejemplos

- Una regla para detectar una solicitud HTTP GET a un dominio determinado
- Esta regla crea una alerta si ve una conexión TCP en el puerto 80 (HTTP) con una solicitud GET al dominio “txc.com”

```
alert tcp any any -> any 80 (msg: “Possible HTTP GET
request”; content: “GET”; http_method; content:
“txc.com”; http_host; sid:1000001; rev:1;)
```

2.3.3 – Snort

FIB

Ejemplos

- Una regla para detectar una firma de malware conocida
- Esta regla crea una alerta si ve una datos que contienen una secuencia específica de bytes (firma conocida) del botnet Zeus en una conexión TCP ya establecida con un servidor

```
alert tcp any any -> any any (msg:"Possible Zeus Botnet
C&C Traffic"; flow:established,to_server; content:"|5a
4f 4f 4d 00 00|"; depth:6; sid:1000005; rev:1;)
```

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

FIB

Tecnologies de Xarxes de Computadors

Tema 2. Seguridad en las redes

Davide Careglio