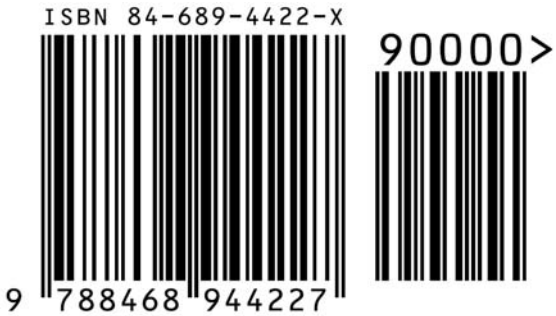
# Seguridad en redes y protocolos asociados

Ingeniería de Protocolos
Curso 2003/2004




MariCarmen Romero Ternero
mcromero@dte.us.es

# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# Sumario

- **Introducción**
- ACLs (Access Control Lists)
- **VLAN (Virtual LAN)**
- Criptografía en redes
- **Protocolos seguros**
- VPN (*Virtual Private Network*)
- **Cortafuegos**

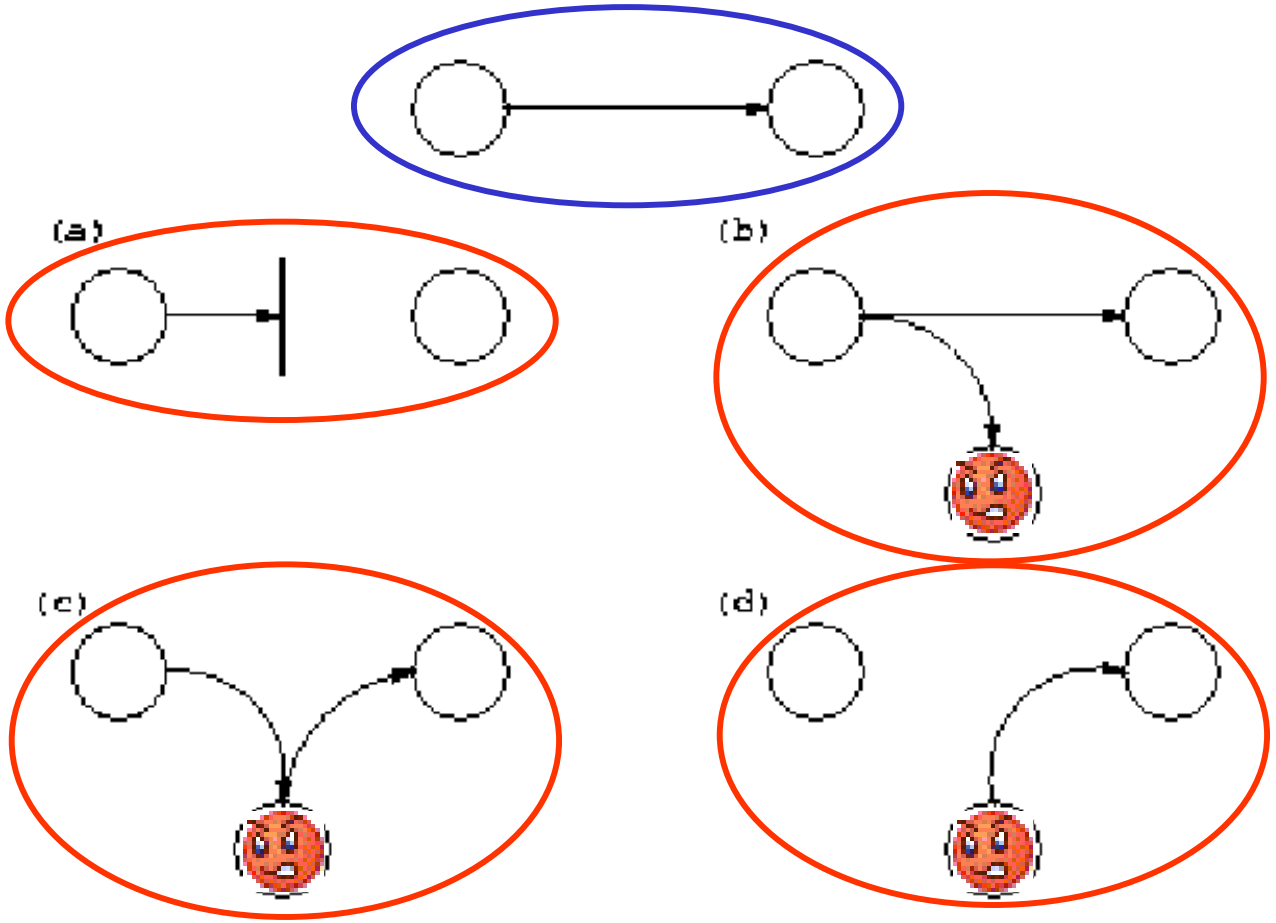
# Ventajas del uso de redes

- Permite la compartición de gran cantidad de recursos (hardware y software) e información
- Aumenta la fiabilidad de los recursos porque permite su replicación (servidores)
- Permite que las aplicaciones que necesiten muchos recursos se ejecuten de forma distribuida
- La red de Internet puede crecer de forma transparente al usuario, lo cual facilita la expansión de las redes empresariales

# Seguridad informática, ¿para qué?

- Compartición ⇨ muchos usuarios involucrados, más atacantes potenciales
- Complejidad del sistema ⇨ complejidad de los controles de seguridad
- Límite desconocido ⇨ identidad incierta de usuarios
- Múltiples puntos de ataque ⇨ mecanismos de protección en todo el camino de la información
- En cuanto a la información, se compromete:
  - su privacidad
  - su integridad
  - su autenticidad
  - su disponibilidad

# Ataques más comunes




Flujo normal de información entre emisor y receptor y posibles amenazas: (a) interrupción, (b) interceptación, (c) modificación y (d) fabricación.

# Ataques más comunes (II)

- Rastreadores o *sniffers*
- Suplantaciones de IP o *spoofing*
- Ataques de contraseñas
- Control de salida ilegal de información sensible desde una fuente interna
- Ataques de hombre en el medio (o *man-in-the-middle attacks*)
- Ataques de denegación de servicio, *Denial of Service* o ataques DoS.
- Ataques a nivel de aplicación para explotar vulnerabilidades conocidas
- Caballos de Troya (*Trojan Horses*), virus y otros códigos maliciosos

# Mecanismos de seguridad

- De prevención:
  - mecanismos de autenticación e identificación
  - mecanismos de control de acceso
  - mecanismos de separación (física, temporal, lógica, criptográfica y fragmentación)
  - mecanismos de seguridad en las comunicaciones (cifrado de la información)
- De detección:
  - IDS (*Intruder Detected System*)
- De recuperación:
  - copias de seguridad (*backup*)
  - mecanismos de análisis forense: averiguar alcance, las actividades del intruso en el sistema y cómo entró

# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# ACL. Sumario

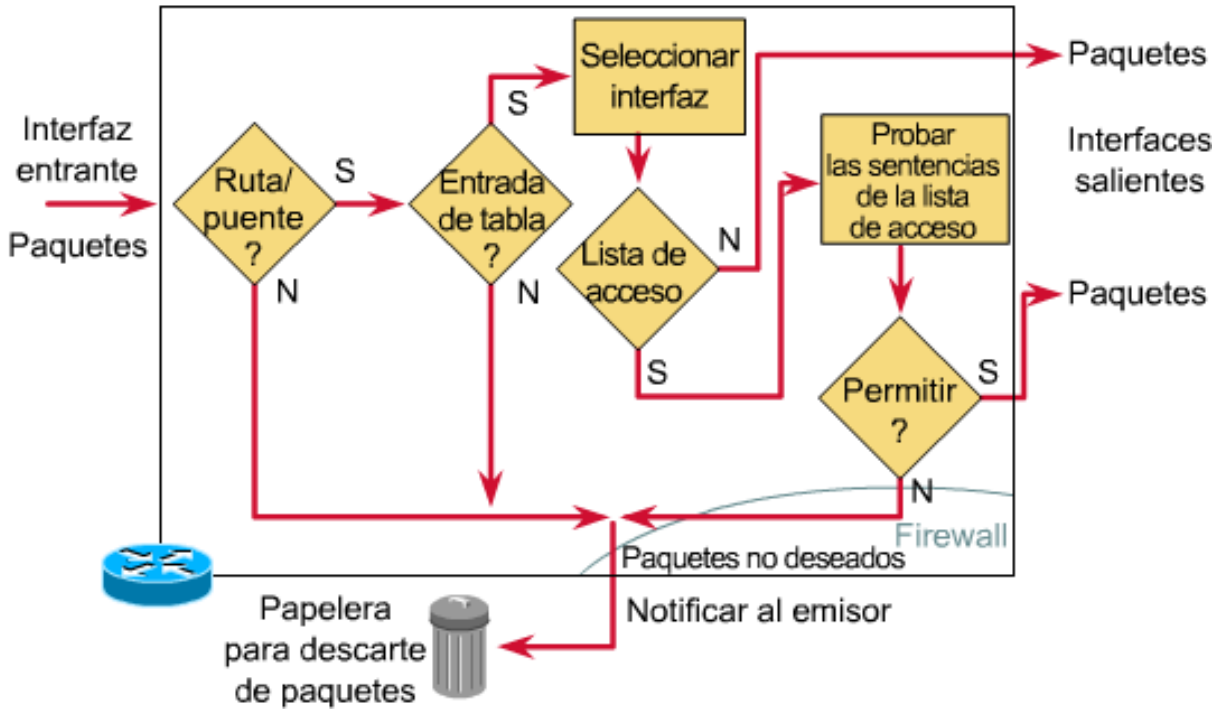
- Definición
- ACLs estándares
  - Ejemplos
- ACLs extendidas
  - Ejemplos
- ACLs nombradas
  - Ejemplos
- ACLs y protocolos
- Ubicación de las ACLs
- ¿Por qué usar las ACLs?

# ACL (Access Control List)

- Listas de sentencias que se aplican a una interfaz del router
- Indican al router qué tipos de paquetes se deben aceptar y qué tipos de paquetes se deben denegar
- La aceptación y rechazo se pueden basar en dirección origen, dirección destino, protocolo de capa superior y número de puerto
- Se pueden crear para todos los protocolos enrutados de red (IP, IPX) (1 ACL por cada protocolo enrutado)
- Se pueden configurar en el router para controlar el acceso a una red o subred
- Filtrado de tráfico entrante y saliente de interfaces

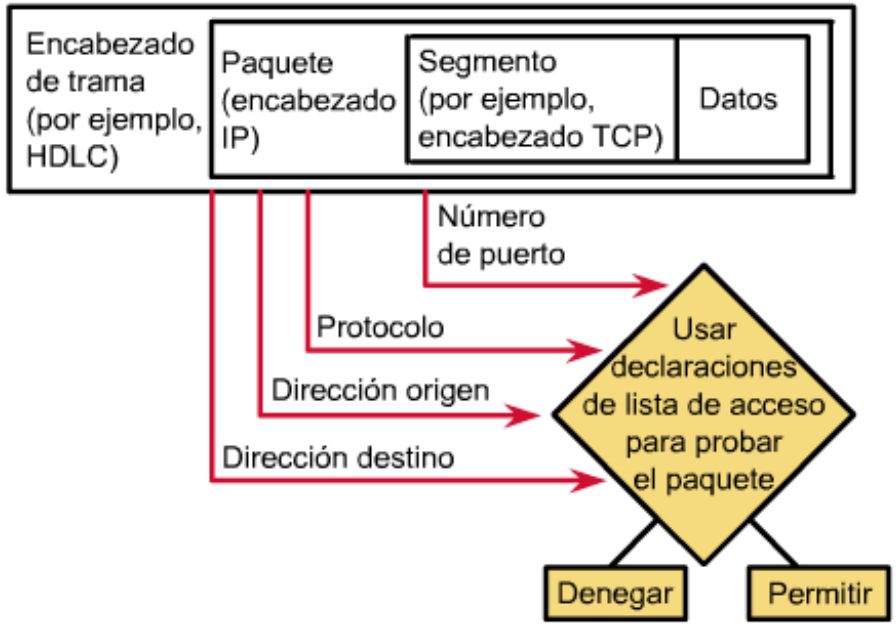
# ACLs

- Una ACL es un grupo de sentencias que define cómo se procesan los paquetes:
  - Entran a las interfaces de entrada
  - Se reenvían a través del router
  - Salen de las interfaces de salida del router



# Ejecución de las ACLs

- El orden de las sentencias de la ACL es importante
- Cuando el router está decidiendo si desea enviar o bloquear un paquete, el IOS prueba el paquete, verificando si cumple o no cada sentencia de condición, en el orden en que se crearon las sentencias
- Una vez que se verifica que existe una coincidencia, no se siguen verificando otras sentencias de condición
- Para añadir sentencias en una ACL hay que eliminar la ACL completa y volver a crearla con las nuevas sentencias de condiciones



# Creación de las ACLs

- Desde el modo de configuración global: **(config)#**
- 2 tipos de ACL:
  - ACL estándar ⇨ ACL del 1 al 99
  - ACL extendida ⇨ ACL del 100 al 199
- Es importante seleccionar y ordenar lógicamente las ACL de forma cuidadosa
- Se deben seleccionar los protocolos IP que se deben verificar; todos los demás protocolos no se verifican
- Aplicar ACL a interfaces oportunos (tráfico entrante y saliente) ⇨ se prefiere ACL para saliente (+ eficiente)
- Hay que asignar un número exclusivo para cada ACL:

| Protocolo | Intervalo |
|---|---|
| IP | 1-99 |
| IP extendido | 100-199 |
| AppleTalk | 600-699 |
| IPX | 800-899 |
| IPX extendido | 900-999 |
| Protocolo de publicación de servicio IPX | 1000-1099 |

# Creación de las ACLs (II)

Paso 1: Definir ACL

Router(config)# **access-list** *num_ACL* {**permit**|**deny**}
{*condición*}

Paso 2: Asociar ACL a un interfaz específico

Router(config-if)# {protocolo} **access-group** *num_ACL* {**in**|**out**}

- Generalmente, usaremos el protocolo IP:

  Router(config-if)# **ip access-group** *num_ACL* {**in** | **out**}
  ⇨ asocia ACL existente a una interfaz (sólo se permite una ACL por puerto por protocolo por dirección). Por defecto out

# ACLs estándar

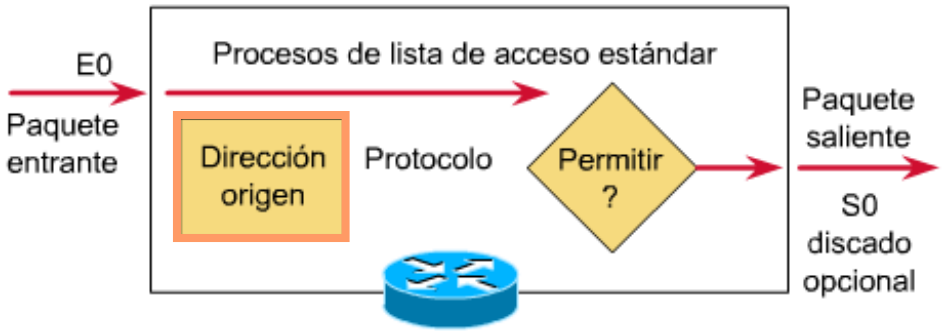
• Una ACL estándar puede servir para bloquear todo el tráfico de una red o de un host específico, permitir todo el tráfico de una red específica o denegar paquetes por protocolos

• Definir sentencias para una ACL:

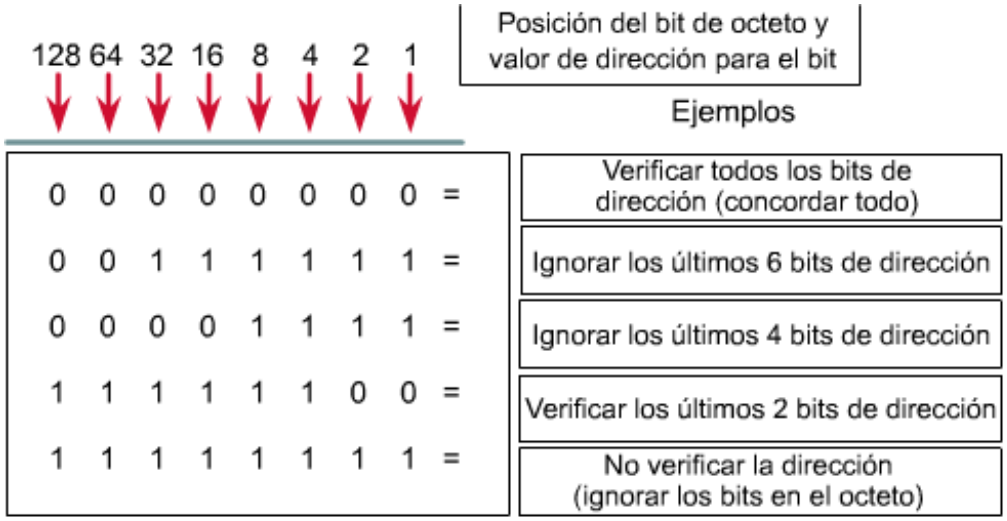
Router(config)# **access-list** *num_ACL* {**deny** | **permit**}
*fuente* [*wildcard_fuente*] [*log*]

• Eliminar una ACL:

1º. Router(config-if)# **no ip access-group** *num_ACL*
2º. Router(config)# **no access-list** *num_ACL*

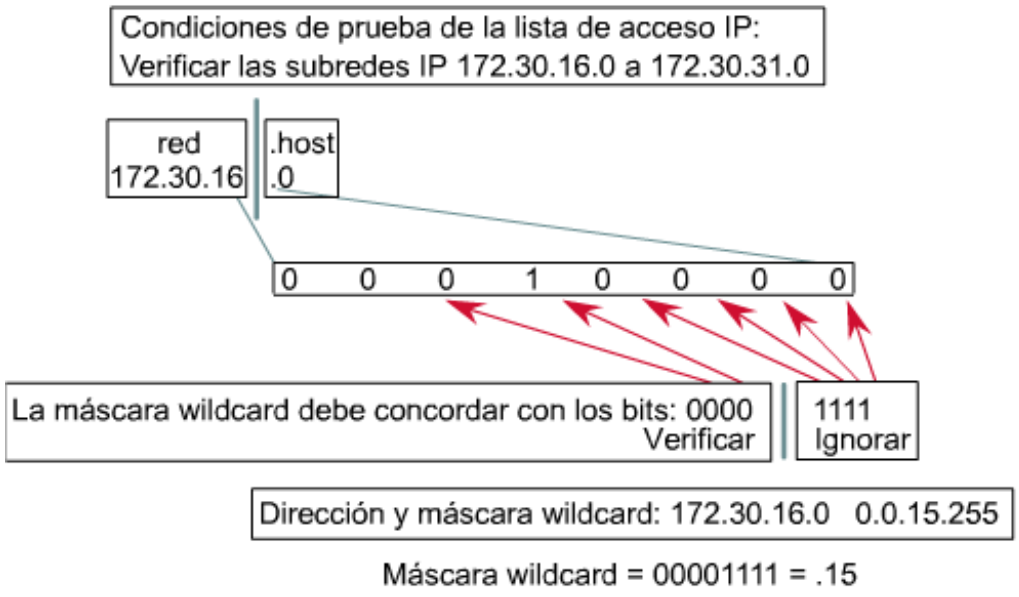


# Bits de la máscara de wildcard




0 ≡ verificar valor del bit
1 ≡ ignorar valor del bit

ignorar ≡ permitir sin comprobar



# ACLs estándar. Ejemplos

- Ejemplo (permite acceso a todos los hosts de las 3 redes especificadas):

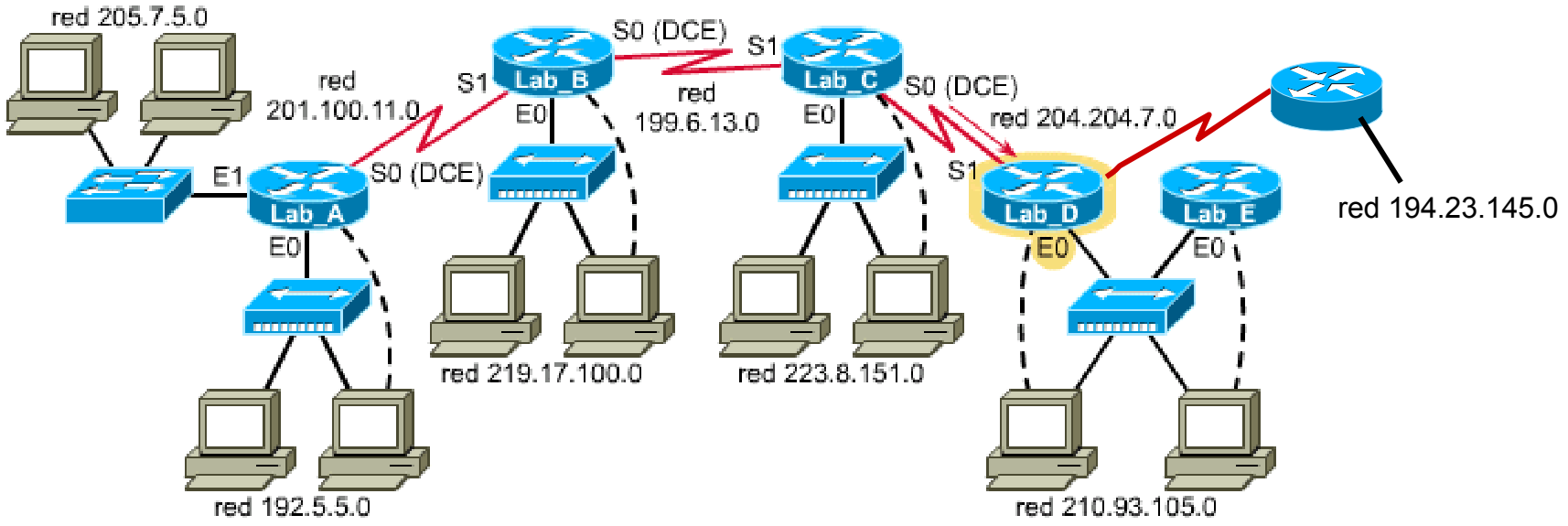
```
Router(config)# access-list 1 permit 192.5.34.0 0.0.0.255
Router(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Router(config)# access-list 1 permit 36.0.0.0 0.255.255.255
!(Nota: cualquier otro acceso está implícitamente denegado)
(access-list 1 deny any)
```

- Son equivalentes:

```
Router(config)# access-list 2 permit 36.48.0.3 0.0.0.0
Router(config)# access-list 2 permit host 36.48.0.3
```

```
Router(config)# access-list 2 permit 0.0.0.0 255.255.255.255
Router(config)# access-list 2 permit any
```

# ACLs estándar. Ejemplos (II)




| Nombre de router - Lab_A | Nombre de router - Lab_B | Nombre de router - Lab_C | Nombre de router - Lab_D | Nombre de router - Lab_E |
|---|---|---|---|---|
| Tipo de router - 2514 | Tipo de router - 2501 | Tipo de router - 2501 | Tipo de router - 2501 | Tipo de router - 2501 |
| E0 = 192.5.5.1 | E0 = 219.17.100.1 | E0 = 223.8.151.1 | E0 = 210.93.105.1 | E0 = 210.93.105.2 |
| E1 = 205.7.5.1 | S0 = 199.6.13.1 | S0 = 204.204.7.1 | S1 = 204.204.7.2 | SM = 255.255.255.0 |
| S0 = 201.100.11.1 | S1 = 201.100.11.2 | S1 = 199.6.13.2 | SM = 255.255.255.0 | |
| SM = 255.255.255.0 | SM = 255.255.255.0 | SM = 255.255.255.0 | | |

**Crear una lista de acceso IP estándar que deniegue paquetes provenientes del host 192.5.5.2 hacia cualquier host en la red 210.93.105.0 y que permita el tráfico desde todas las demás redes.**

**⇨ ¿Dónde colocamos la ACL y por qué?**

# ACLs extendidas

• Ofrecen una mayor cantidad de opciones de control que las ACLs estándares, son más versátiles

• Verifican direcciones origen y destino de los paquetes, protocolos, números de puerto y otros parámetros específicos

• Las ACLs extendidas usan un número dentro del intervalo del 100 al 199

• Al final de la sentencia de la ACL extendida, se puede especificar opcionalmente el número de puerto de protocolo TCP o UDP para el que se aplica la sentencia:

- 20 y 21: datos y programa FTP
- 23: Telnet
- 25: SMTP
- 53: DNS
- 69: TFTP

...

# ACLs extendidas

- Definir ACL extendida:

Router(config)# **access-list** *num_ACL* {**permit** | **deny**} *protocolo fuente* [*mascara-fuente destino mascara-destino operador operando*] [**established**]

**num_ACL** ⇨ Identifica número de lista de acceso utilizando un número dentro del intervalo 100-199
**protocolo** ⇨ IP, TCP, UDP, ICMP, GRE, IGRP
**fuente** | **destino** ⇨ Identificadores de direcciones origen y destino
**mascara-fuente** | **mascara-destino** ⇨ Máscaras de wildcard
**operador** ⇨ lt, gt, eq, neq
**operando** ⇨ un número de puerto
**established** ⇨ permite que pase el tráfico TCP si el paquete utiliza una conexión establecida (p.e. tiene bits de ACK establecidos)

- Asociar ACL a interfaz:

Router(config-if)# **ip access-group** *num_ACL* {**in** | **out**}

# ACLs extendidas. Ejemplos

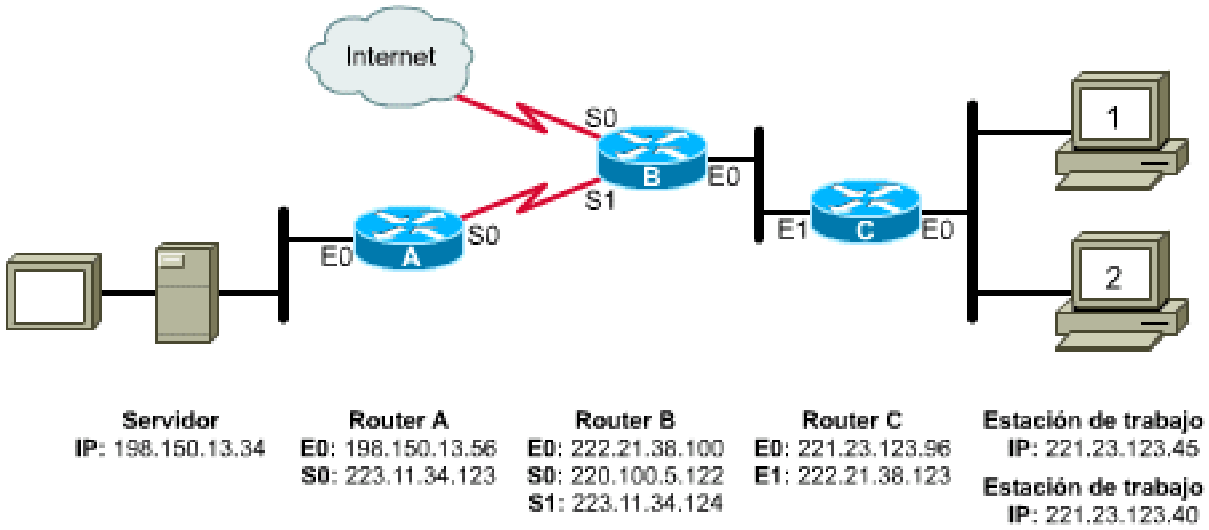
- Ejemplo 1 (denegar FTP entre dos redes y permitir todo lo demás):

```
Router(config)# access-list 101 deny tcp 172.16.4.0 0.0.0.255
                        172.16.3.0 0.0.0.255 eq 21
Router(config)# access-list 101 permit ip 172.16.4.0 0.0.0.255
                        0.0.0.0 255.255.255.255
Router(config)# access-list 101 deny ip 0.0.0.0 255.255.255.255
                        0.0.0.0 255.255.255.255 (implícito)
```

- Ejemplo 2 (denegar Telnet a hosts de una red y permitir todo lo demás):

```
Router(config)# access-list 102 deny tcp 172.16.4.0 0.0.0.255
                        any eq 23
Router(config)# access-list 102 permit ip any any
Router(config)# access-list 102 deny ip 0.0.0.0 255.255.255.255
                        0.0.0.0 255.255.255.255 (implícito)
```

# ACLs extendidas. Ejemplos (II)




**Crear una ACL que bloquee todo el tráfico desde la red 221.23.123.0 al servidor 198.150.13.34 provocando la menor cantidad de tráfico en la red.**

**⇨ ¿Dónde colocamos la ACL y por qué?**

# ACLs nombradas

- Permiten que las ACL IP estándar y extendidas se identifiquen con una cadena alfanumérica (nombre) en lugar de la representación numérica actual (1 a 199)

- Se usan si:
  - Se desea identificar intuitivamente las ACL utilizando un nombre alfanumérico, o
  - Existen más de 99 ACL simples y 100 extendidas que se deben configurar en un router para un protocolo determinado

- Tener en cuenta que:
  - No son compatibles con versiones < 11.2 del IOS
  - No se puede usar el mismo nombre en varias ACLs

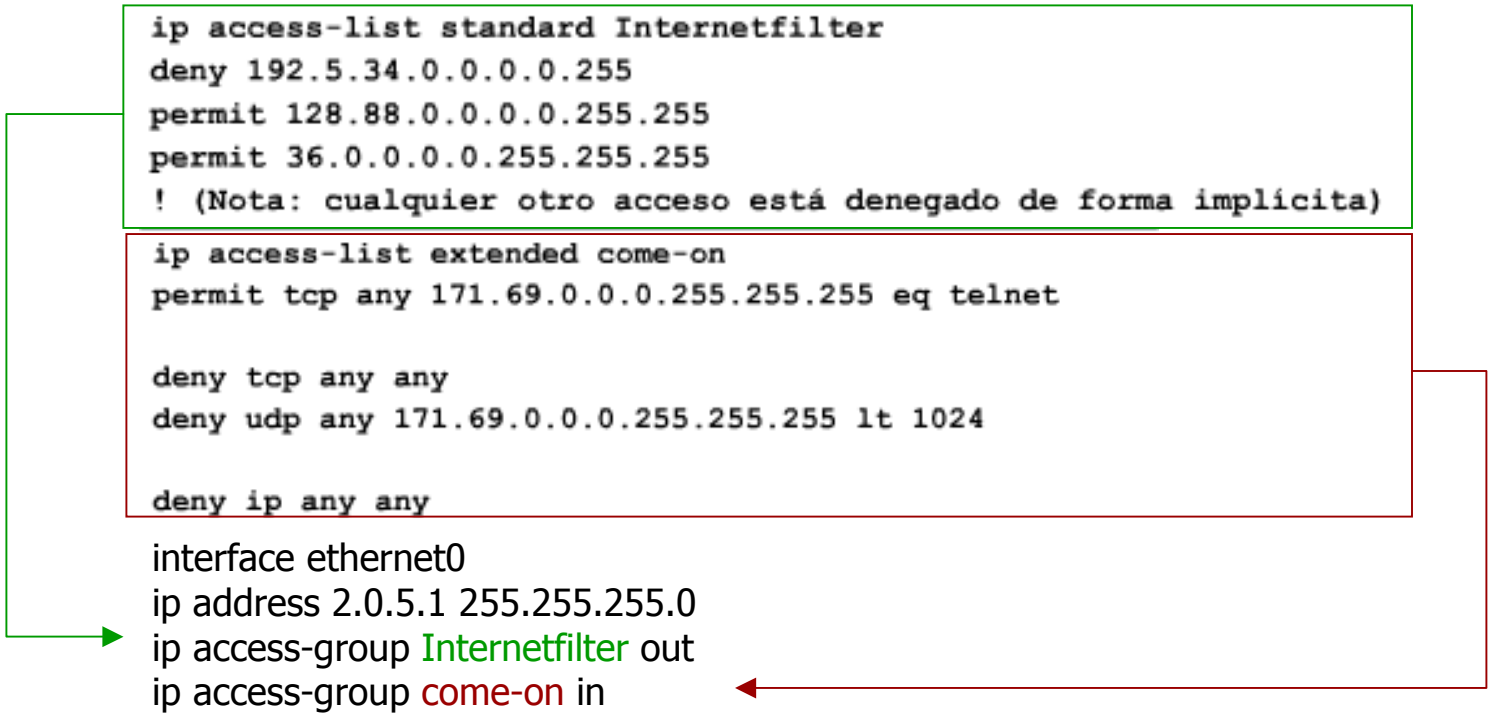
```
ip interface ethernet0/5
ip address 2.0.5.1.255.255.255.0
ip access-group Internetfilter out
ip access-group marketinggroup in
...
ip access-list standard Internetfilter
permit 1.2.3.4

deny any
ip access-list extended marketing_group
permit tcp any 171.69.0.0.0.255.255.255 eq telnet

deny tcp any any
deny udp any 171.69.0.0.0.255.255.255 lt 1024

deny ip any log
```

# ACLs nombradas (II)




**Router(config)# ip access-list {standard | extended} nombre**

**Router(config-{std- | ext-}nacl)# {permit | deny} {ip ACL test conditions}**
**no {permit | deny} {ip ACL text conditions}**

**Router(config-if)# ip access-group {nombre | 1-199} {in | out}**

# ACLs y protocolos

- ACLs pueden controlar la mayoría de los protocolos en un router Cisco
- El protocolo al que tiene que aplicarse la ACL se indica como un número en el intervalo de números de protocolo

- Sólo se puede especificar una ACL por protocolo y por interfaz
- Para algunos protocolos, se pueden agrupar hasta 2 ACL a una interfaz (entrante y saliente). Con otros protocolos, se agrupa sólo 1 ACL

- Si ACL es entrante, se comprueba al recibir el paquete
- Si ACL es saliente, se comprueba después de recibir y enrutar un paquete a la interfaz saliente

- *Nombrar o numerar un protocolo IP:*
  - usando las palabras reservadas: eigrp, gre, icmp, igmp, igrp, ip, ipinip, nos, ospf, tcp, o udp, o bien
  - con un nº entero (0 a 255) , que representa un nº de protocolo IP
  - la palabra reservada *ip* indica cualquier protocolo Internet
  - los protocolos y sus números correspondientes se enumeran en RFC 1700, junto con los números de puerto

# Ubicación de las ACLs

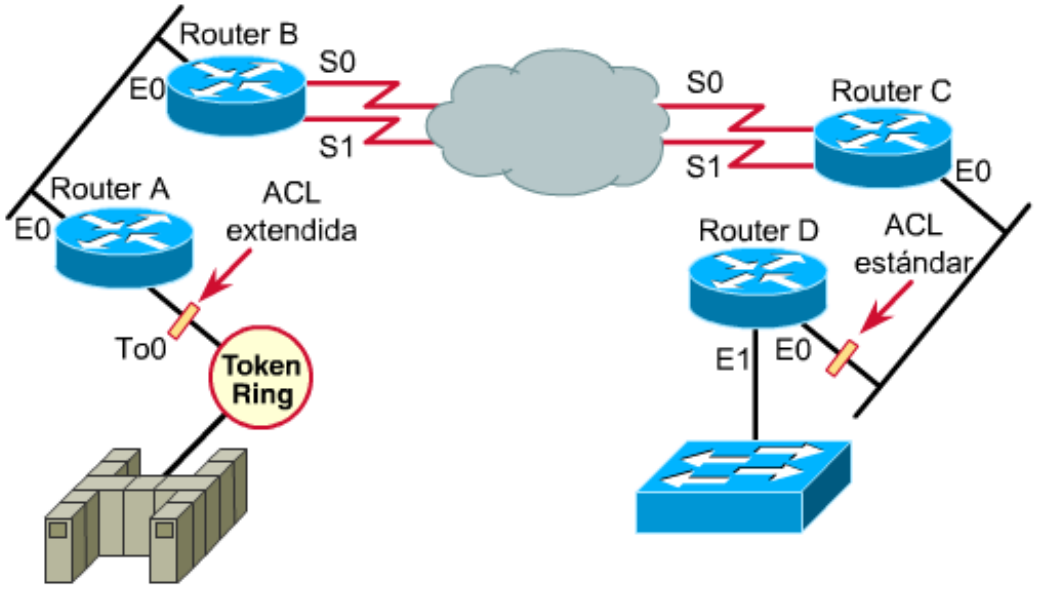
• El lugar donde se ubique una sentencia de ACL influye en la reducción del tráfico innecesario

• El tráfico que será denegado en un destino remoto no debe usar los recursos de la red en el camino hacia ese destino

• La regla es colocar las:

⇨ ACL extendidas lo más cerca posible del origen del tráfico denegado

⇨ ACL estándar lo más cerca posible del destino (no especifican direcciones destino)



# Verificación de las ACLs. Comandos

- show ip interface ⇨ muestra información de interfaz IP e indica si hay alguna ACL asociada a dicho interfaz

- show access-lists ⇨ muestra el contenido de todas las ACLs definidas en el router

- show access-lists *num_o_nombre_ACL* ⇨ muestra contenido de ACL específica indicada como parámetro

# ipchain, iptable

- Herramientas gratuitas en entornos UNIX (habría que compilar el kernel con la opciones de *firewalling*), que sirven para lo mismo que las ACLs de los routers Cisco
- Ambas herramientas permiten aplicar sentencias de filtrado a los diferentes interfaces que tengamos configurados y, además, permite la redirección de puertos y NAT

- **Ejemplos**:

```
pepito:~# /sbin/ipchains -A input -p tcp -j ACCEPT -d 158.42.22.41 80
pepito:~# /sbin/iptables -A INPUT -p TCP -j ACCEPT -d 158.42.22.41 --dport 80
```

⇨ Indica que se añada (`-A') en la chain `input' (tramas de entrada) una regla que permita (`ACCEPT') el tráfico tcp (`-p') cuyo destino (`-d') sea el puerto 80 de la dirección 158.42.22.41 (ip del servidor web).

# ¿Por qué usar ACLs?

- Limita el tráfico de red, mejorando su rendimiento
- Proporciona control de flujo del tráfico que debe pasar por el router
- Proporciona un nivel básico de **seguridad** de acceso a la red en función de distintos parámetros
- El administrador puede decidir qué tipo de tráfico se envía o bloquea en los interfaces del router
- Pertenecen a la categoría de cortafuegos de filtrado de paquetes (capa 3 y 4)

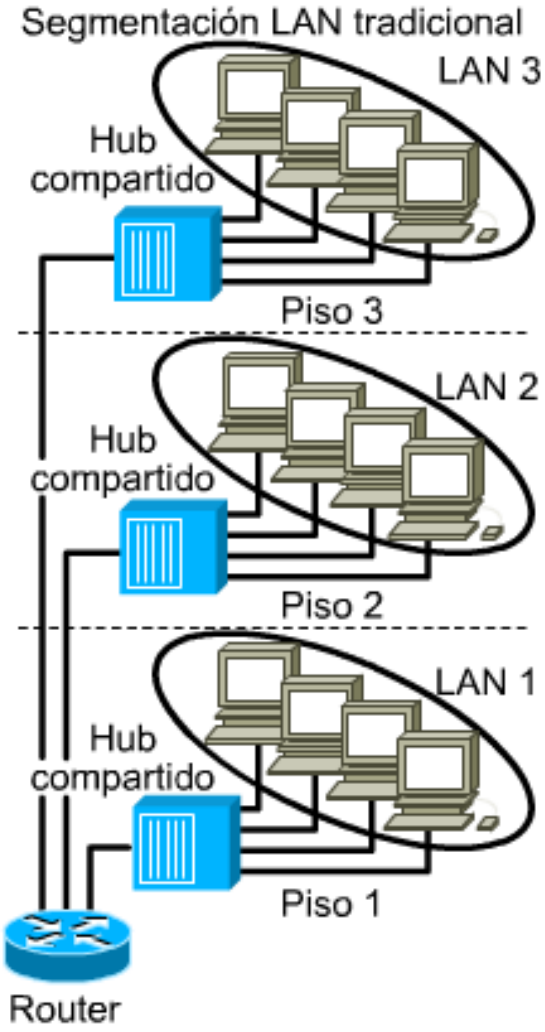
# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# VLAN. Sumario

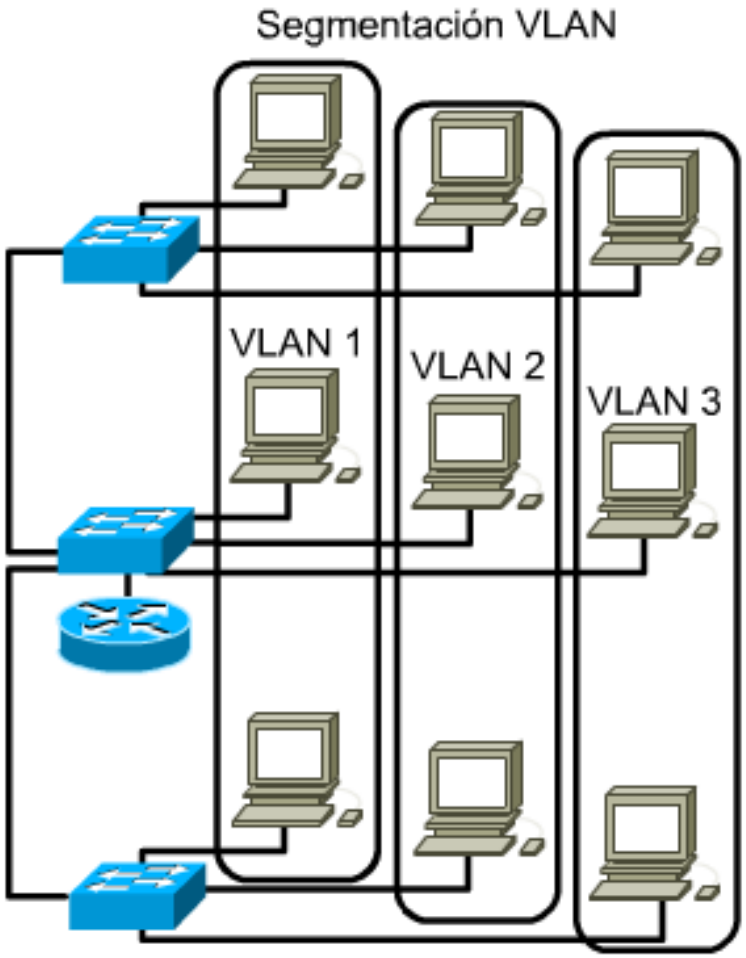
- Segmentación tradicional vs VLAN
- Características de las VLANs
- VLANs y switches
- VLANs y routers
- VLANs y hubs
- VLANs y VTP (*Virtual Trunking Protocol*)
- Asignación a VLANs
- Ventajas de las VLANs

# Segmentación tradicional




- Cada usuario se conecta al hub/switch más próximo físicamente
- La pertenencia de un usuario a una red u otra está limitada por el cableado físico
- Si un segmento emplea hubs para la interconexión, todos los usuarios pertenecen al mismo dominio de colisión (no así si se usan switches)
- Los dominios de broadcast están delimitados por el router

# Segmentación con VLANs




- Cada usuario se conecta al switch VLAN más próximo físicamente
- Se definen varias VLANs en los switches
- Los usuarios se agrupan en las VLANs, según criterio del administrador
- La pertenencia de un usuario a una VLAN no depende del cableado físico
- Cada VLAN es un dominio de broadcast
- El router permite la comunicación entre VLANs

# Características de las VLANs

- Crean una topología virtual independiente de la física
- Permiten agrupar a los usuarios en grupos de trabajo flexibles
- Funcionan en los niveles 2 y 3 de OSI
- La comunicación entre VLANs requiere enrutamiento de capa 3 (routers)
- Permiten controlar el tamaño de los dominios de *broadcast*
- Necesitan administración
- Pueden ayudar a aumentar la seguridad de la red

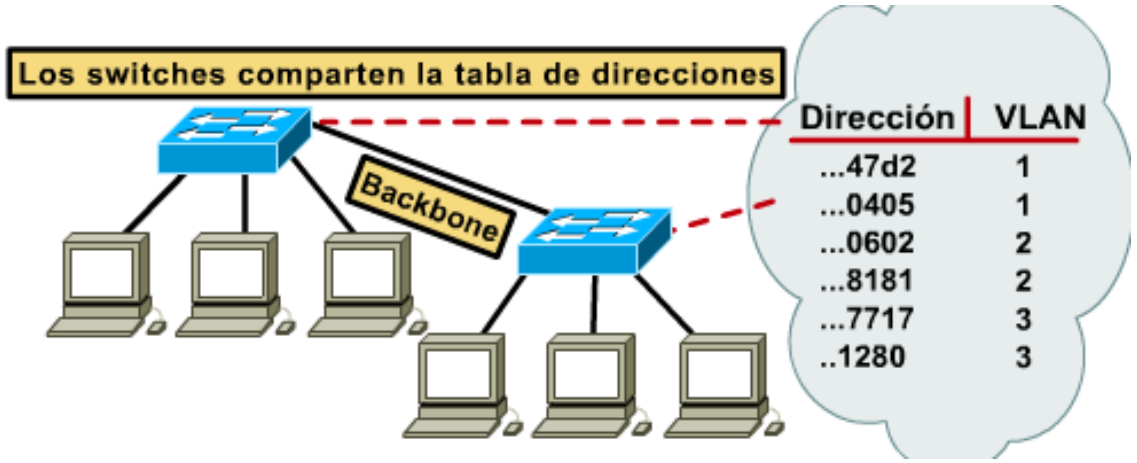
# Transporte VLANs entre switches

- Normalmente, la infraestructura de VLANs estará distribuida entre varios switches
- Los switches VLAN se interconectan mediante puertos de mayor capacidad (***trunks***)
- Esta interconexión es parte del cableado backbone


- La información sobre la pertenencia de los usuarios a las distintas VLANs se transmite a través del backbone. Existen dos métodos:
  - filtrado de tramas
  - etiquetado (identificación, rotulado) de tramas
- La información sobre las VLANs definidas también se distribuye a través del backbone entre los distintos switches, mediante el protocolo VTP (Vlan Trunk Protocol)

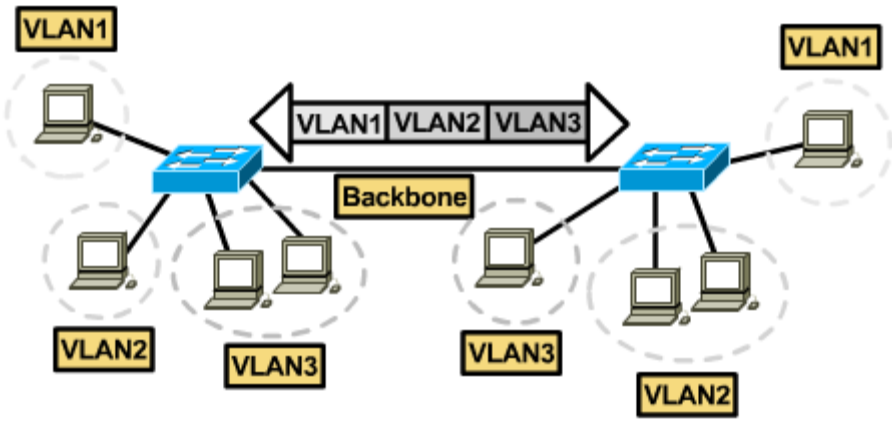
# VLAN entre switches: filtrado

- Cada switch desarrolla una tabla de filtrado, que asocia cada dirección física con la VLAN a la que pertenece
- Los switches comparten las tablas a través del backbone
- Cuando una trama llega a un switch, éste puede determinar a qué VLAN pertenece empleando la tabla
- Esta técnica permite filtrar en función de cualquier parámetro de la trama (dirección física, lógica, ...)
- No es escalable; no se emplea actualmente



# VLAN entre switches: etiquetado

- Cada VLAN tiene asociado un identificador
- Las tramas procedentes de los usuarios se etiquetan con el identificador correspondiente a la VLAN a la que pertenecen
- El etiquetado se lleva a cabo en el switch (capa 2 OSI)
- Las tramas etiquetadas atraviesan el backbone
- Cuando una trama etiquetada va a abandonar el backbone, el switch elimina el identificador
- Estándar IEEE 802.1Q

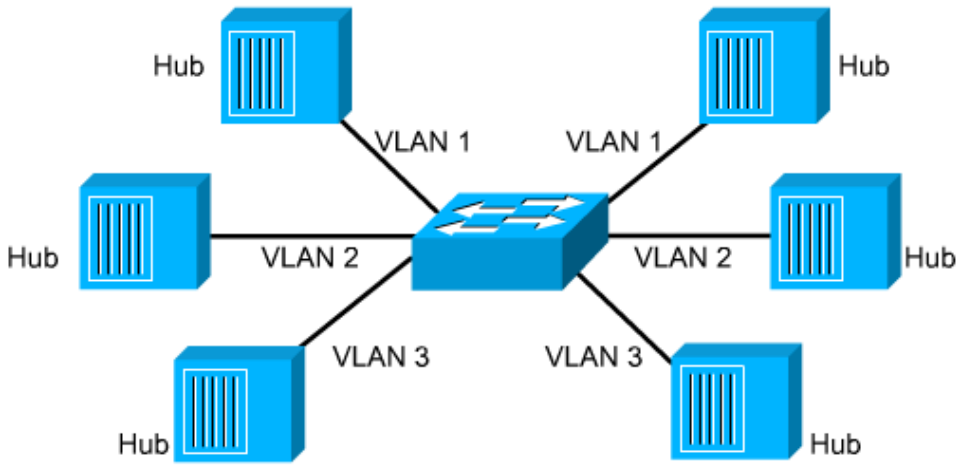


# Routers y VLANs

- Las VLANs son dominios de broadcast separados, por lo tanto, no se pueden comunicar directamente
- Normalmente, cada VLAN se corresponde con una subred
- La comunicación entre VLANs se hace a través de un router
- La conexión entre los switches VLAN y el router se hace normalmente mediante enlaces de alta capacidad
- El router admite el etiquetado 802.1Q e ISL (Cisco Inter-Switch Link), de modo que puede conectarse directamente al enlace backbone conmutado

# Hubs y VLANs

- Es posible usar hubs en VLANs para reducir costes
- El hub se conecta
  - a un puerto de un switch VLAN
  - a un cierto número de usuarios
- Todos los usuarios conectados al mismo hub:
  - pertenecen a la misma VLAN
  - pertenecen al mismo dominio de colisión (comparten el ancho de banda correspondiente al puerto del switch)



# Definición de VLANs y VTP

- La definición de VLANs se lleva a cabo en un único switch (servidor)
- La información sobre las VLANs se transmite mediante el backbone hacia los demás switches (clientes) usando el protocolo VTP (*VLAN Trunk Protocol*)
- Un switch se puede configurar para ignorar los mensajes VTP (modo transparente)
- La configuración del switch sólo puede hacerse dentro de la VLAN de gestión (*Management VLAN*), por defecto, VLAN 1

# Asignación a VLANs

- La asignación de usuarios a las VLANs definidas puede ser:
  - **Estática**: cada puerto del switch es asignado a una VLAN. Por tanto, el usuario conectado a ese puerto pertenecerá a la VLAN.
    - El administrador debe realizar la configuración VLAN manualmente
    - Fácil de administrar
    - Implementación más eficiente
  - **Dinámica**: la pertenencia se determina en función de la dirección física (capa 2), dirección lógica (capa 3), tipo de protocolo, etc.
    - Necesita de un servidor de configuración VLAN (que hay que mantener)
    - Al conectar un usuario a un puerto, el switch consulta el servidor de configuración para determinar a qué VLAN pertenece
    - No necesita administración al realizar desplazamientos de usuarios
    - Seguridad: notificación cuando usuarios no autorizados acceden a la red
- La configuración del switch sólo puede hacerse dentro de la VLAN de gestión (*Management VLAN*), por defecto, VLAN 1

# Ventajas de las VLANs

- Facilitan el alta/baja de nuevos usuarios y los desplazamientos. Por ejemplo, para mover a un usuario de ubicación física sin cambiar su dirección IP ni su VLAN:
  - VLAN estática: conectar el usuario a un puerto libre en un switch VLAN y configurar el puerto para que pertenezca a la VLAN del usuario
  - VLAN dinámica: no requiere cambios
- Contención de broadcasts: sin VLANs, el único medio de controlar el tamaño de los dominios de broadcast es mediante routers. Las VLANs son una alternativa menos costosa
- Seguridad:
  - un usuario sólo puede ver el tráfico broadcast de su VLAN
  - un usuario no puede conectarse a la red sin la aprobación del administrador
  - configuración de los switches: sólo desde la VLAN de gestión
  - los routers pueden incorporar listas de control de acceso para filtrar el tráfico entre VLANs

# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# Criptografía en redes

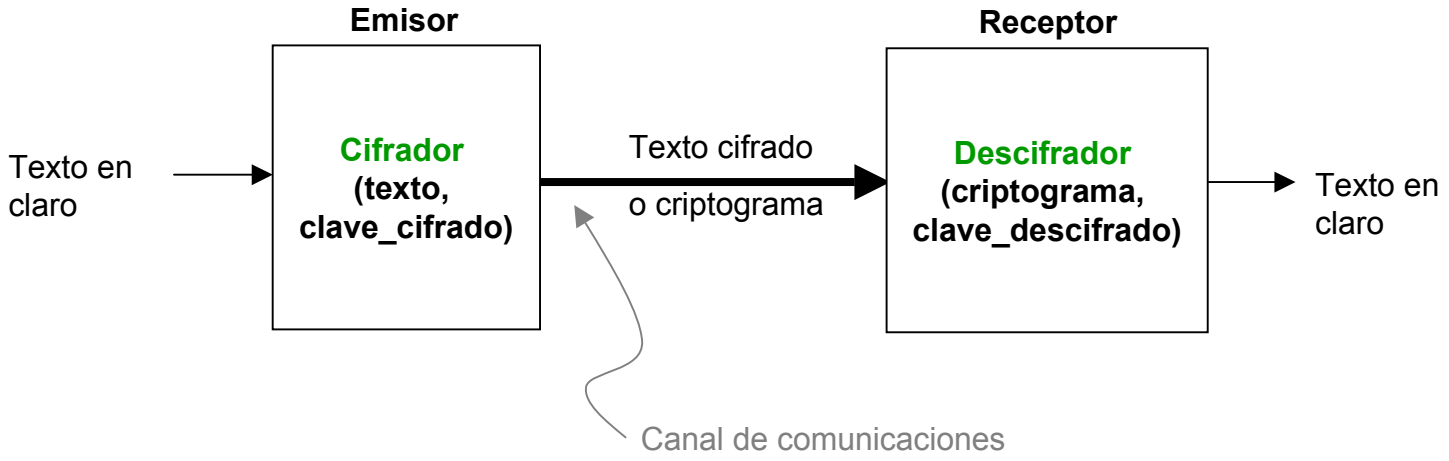
- Conceptos
- Relación entre criptografía y redes
- Métodos básicos de cifrado en redes
  - Cifrado de enlace
  - Cifrado extremo a extremo

# Conceptos

- Criptología (del griego krypto y logos, estudio de lo oculto, lo escondido) es la ciencia que trata los problemas teóricos relacionados con la seguridad en el intercambio de mensajes en clave entre un emisor y un receptor a través de un canal de comunicaciones (en términos informáticos, ese canal suele ser una red de computadoras). Esta ciencia está dividida en dos grandes ramas:
  - la criptografía, ocupada del cifrado de mensajes en clave y del diseño de criptosistemas, y
  - el criptoanálisis, que trata de descifrar los mensajes en clave, rompiendo así el criptosistema

# Conceptos (II)

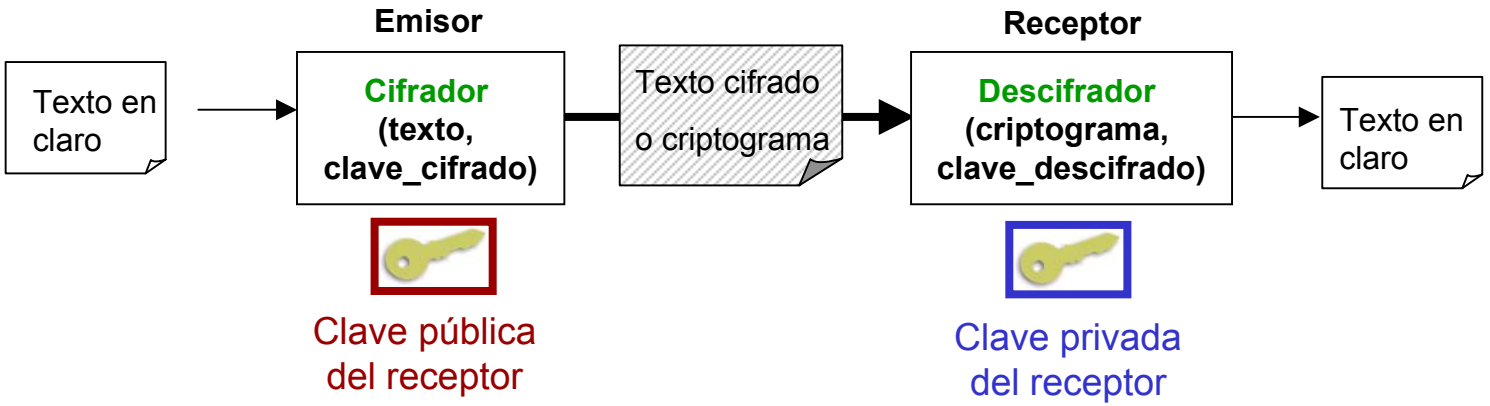
- Criptosistema, formado por:
  - un alfabeto
  - un espacio de claves
  - un conjunto de transformaciones de cifrado
  - un conjunto de transformaciones de descifrado



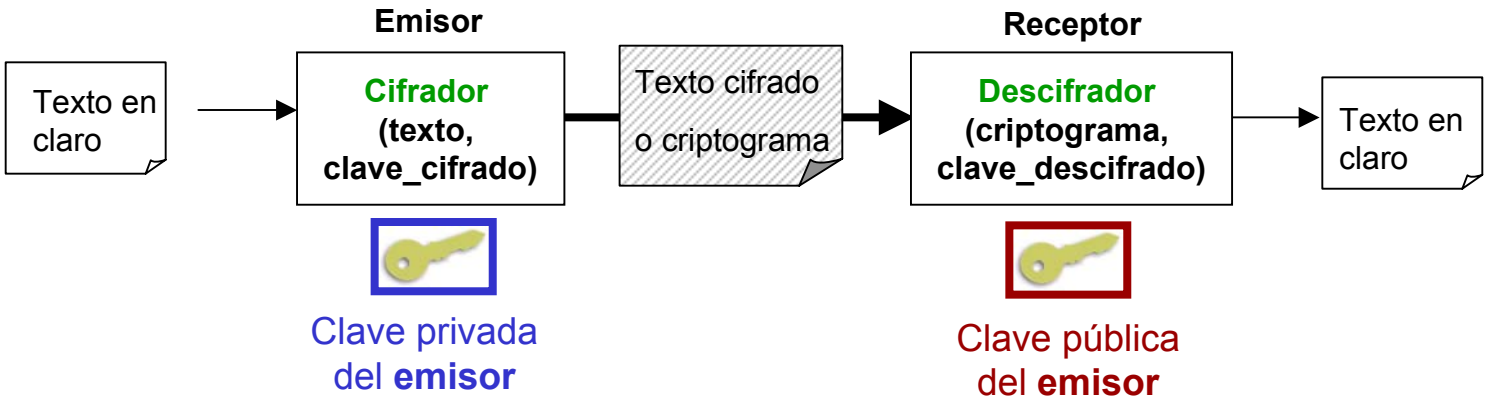

- Tipos de criptosistema:
  - de clave privada o secreta (simétricos): DES (*Data Encryption Standard*)
  - de clave pública (asimétricos): RSA (*Rivest-Shamir-Adleman*)

# Conceptos (III)

- Cifrado/Descifrado asimétrico con clave pública:




- Firma digital y autenticación:



# Conceptos (IV)

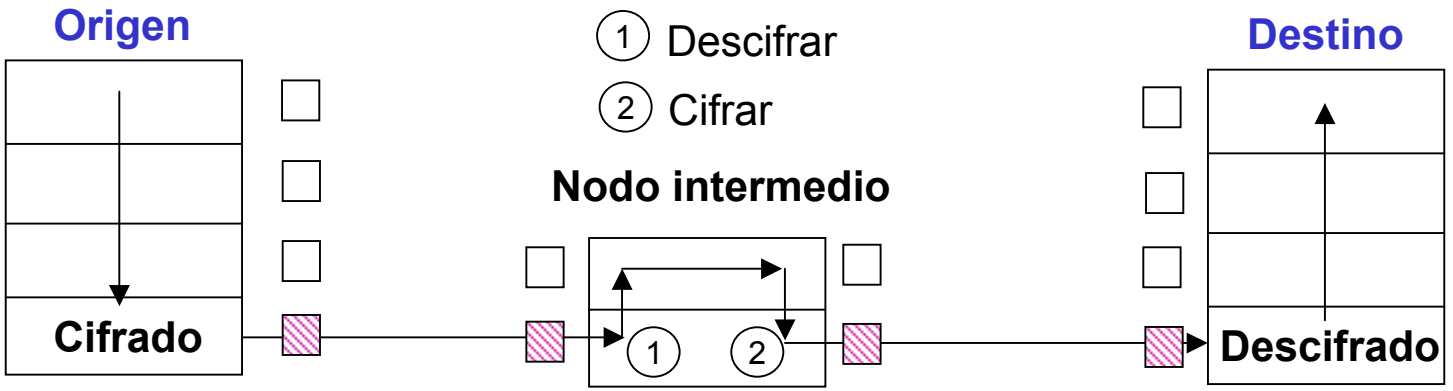
- Esteganografía (del griego stegos (cubierta)): ciencia que estudia los procedimientos encaminados a ocultar la existencia de un mensaje en lugar de ocultar su contenido
  - El objetivo de la **criptografía** es que un atacante que consigue un mensaje no sea capaz de averiguar su contenido y el objetivo de la **esteganografía** es ocultar ese mensaje dentro de otro sin información importante, de forma que el atacante ni siquiera se entere de la existencia de dicha información oculta
  - No sustituye al cifrado convencional sino que lo complementa, pues ocultar un mensaje reduce las posibilidades de que sea descubierto. Sin embargo, en caso de ser descubierto, el que ese mensaje haya sido cifrado introduce un nivel adicional de seguridad

# Relación entre criptografía y redes

- La criptografía es el mecanismo más utilizado para proporcionar seguridad en redes
- Permite crear conexiones seguras sobre canales inseguros
- La Criptografía podrá entonces ser empleada en diferentes niveles de abstracción (protocolos de distintos niveles)
- Según el tipo de red, puede ser más o menos necesaria:
  - Redes internas (LAN): la red es propietaria de la empresa ⇨ control total sobre su seguridad
  - Redes externas: no se controlan las infraestructuras públicas ⇨ no controlamos la seguridad ⇨ Criptografía
  - Intranet (redes externas que se comportan de cara a los usuarios como redes privadas internas) ⇨ no controlamos la seguridad ⇨ Criptografía

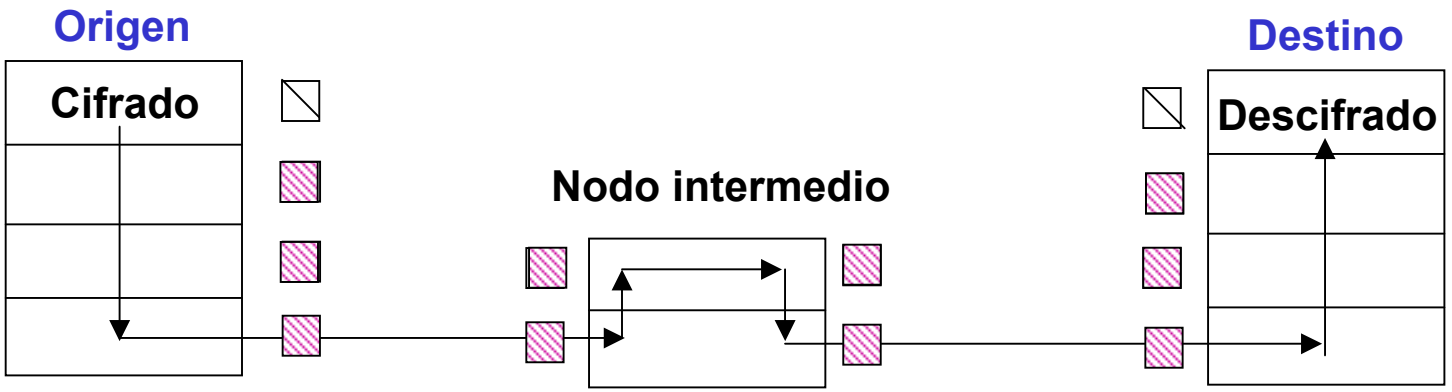
# Métodos básicos de cifrado de redes

- **Cifrado de enlace**
  - De capa 2 de OSI
  - Cifra todo el mensaje, incluidas las cabeceras de niveles superiores
  - Requiere nodos intermedios con capacidades de cifrado/descifrado
  - La información está protegida entre cada par de nodos consecutivos (distintas claves para cada par)
  - Es necesario descifrarla, aunque sea parcialmente, para procesos de encaminamiento, control de errores...



# Métodos básicos de cifrado (II)

- **Cifrado extremo a extremo**
  - De capa 7 de OSI
  - Sólo se cifran los datos, las cabeceras se añaden y se transmiten sin cifrar
  - El cifrado de datos se mantiene desde origen hasta destino



# Métodos básicos de cifrado (III)

| Cifrado de enlace | Cifrado extremo a extremo |
|---|---|
| **Seguridad en los nodos** | |
| El mensaje queda expuesto en el nodo emisor y receptor, y en todos los nodos intermedios | El mensaje sólo queda expuesto en el nodo emisor y en el receptor |
| **Papel del usuario** | |
| El cifrado se aplica en el nodo emisor y en todos los intermedios | El cifrado es aplicado sólo por el proceso emisor |
| El cifrado es transparente al usuario | El usuario aplica el cifrado (puede decidir qué partes de la información quiere cifrar) |
| El nodo se encarga del cifrado | El usuario debe encontrar la aplicación de cifrado y preocuparse de usarla |
| Un solo servicio para todos los usuarios | Cada usuario selecciona su criptosistema |
| Suele realizarse por hardware | Suele realizarse por software |
| Se cifran todos los mensajes o ninguno | El usuario elige qué mensajes quiere cifrar y qué claves usar en cada caso |
| **Aspectos de implementación** | |
| Se necesita una clave para par de nodos. Si se compromete uno de los nodos, sólo se comprometen las claves relacionadas con sus adyacentes, y no toda la red | Si se usa cifrado simétrico se necesita una clave para cada par de usuarios, si se usa cifrado asimétrico se necesitan dos claves para cada par de usuarios |
| Porporciona autentificación de nodos (protege información de cabeceras potencialmente útiles para ataques) | Proporciona autentificación de usuarios (las cabeceras se transmiten sin cifrar, luego se pueden interceptar fácilmente) |
| **En resumen** | |
| Más rápido y más fácil para el usuario | Más flexible y no requiere que los nodos intermedios tengan capacidad de cifrado/descifrado |

# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# Protocolos seguros

- Servicios de seguridad:
  - Manejo de claves
    (negociación y almacenamiento de claves)
  - Confidencialidad / Privacidad
  - No repudio
  - Integridad
  - Autenticación
  - Autorización

# Protocolos seguros . Niveles

| Seguridad de Nivel de | Ventajas | Desventajas | Ejemplos de protocolos |
|---|---|---|---|
| **Aplicación** | - Se puede extender la aplicación para brindar servicios de seguridad sin tener que depender del SO<br>- Facilita el servicio de no repudio | - Los mecanismos de seguridad deben ser diseñados de forma independiente para cada aplicación<br>- Mayores probabilidades de cometer errores | Kerberos<br>PGP<br>SSH<br>S/MIME<br>SET<br>IPSec (ISAKMP)<br>RADIUS<br>TACACS |
| **Transporte** | - En teoría, no se requieren modificaciones por aplicación | - Mantener el contexto del usuario es complicado<br>- TLS requiere que las aplicaciones sean modificadas | SSL (Netscape Corp.)<br>TLS (IETF) |
| **Red** | - Disminuye el flujo excesivo de negociación de claves<br>- Las aplicaciones no requieren modificación alguna<br>- Permite crear VPNs e Intranets | - Difícil manejar el no repudio | IPSec (AH, ESP)(IETF)<br>NLSP (ISO)<br>Protocolos de tunneling:<br>PPTP, L2TP |
| **Enlace de datos** | - Más rápido | - No son soluciones estables y funcionan bien sólo para enlaces dedicados<br>- Los dispositivos deben estar físicamente conectados | ATMs (IEEE)<br>SILS (IEEE)<br>CHAP<br>PAP<br>MS-CHAP,<br>EAP, LEAP, PEAP |

PGP = Pretty Good Privacy
SSH = Secure Shell
S/MIME = Secure Multi-Purpose Internet Mail Extensions
SET = Secure Electronic Transaction
ISAKMP = Internet Security Association and Key Management Protocol
RADIUS = Remote Access Dial-In User Service
TACACS = Terminal Access Controller Access Control System
SSL = Secure Socket Layer
TLS = Transport Layer Security
IPSec = IP Security

AH = Authentication Header
EDS = Encapsulation Security Payload
NLSP = Network Layer Security Protocol
PPTP = Point-to-Point Tunneling Protocol
L2TP = Layer 2 Tunneling Protocol
SILS = Standards for Interoperable LAN/MAN Security
CHAP = Challenge-Handshake Authentication Protocol
PAP = Password authentication Protocol
MS-CHAP = Microsoft CHAP
EAP = Extensible Authentication Protocol
LEAP = Lightweight EAP
PEAP = Protected EAP

# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
  - **Nivel de Aplicación**
  - **Nivel de Transporte**
  - **Nivel de Red**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# SSH (Secure SHell)

- SSH es un protocolo de nivel de aplicación para crear conexiones seguras entre dos sistemas sobre redes no seguras (SSH2)
- Alternativa a programas de acceso remoto no seguros, como telnet, ftp, rlogin, rsh y rcp (slogin, ssh y scp)
- Proporciona terminal de sesión cifrada con autenticación fuerte del servidor y el cliente, usando criptografía de clave pública
- Incluye características como:
  - una variedad de mecanismos de autenticación de usuarios
  - conexiones TCP arbitrarias de tunneling a través de la sesión SSH, protegiendo protocolos inseguros como IMAP y permitiendo el paso seguro a través de cortafuegos
  - reenvío automático de conexiones X windows
  - soporte para métodos de autenticación externa, inluyendo Kerberos
  - transferencias seguras de ficheros
- SSH está basado en protocolos documentados por el IETF

# SSH (Secure SHell) (II)

- Otros tipos de protección que proporciona SSH:
  - Después de la conexión inicial, el cliente puede verificar que se está conectando al mismo servidor durante sesiones posteriores
  - El cliente puede transmitir su información de autentificación al servidor, como el nombre de usuario y la contraseña, en formato cifrado
  - El cliente tiene la posibilidad de usar X11 en aplicaciones lanzadas desde el indicador de comandos de la shell. Esta técnica proporciona una interfaz gráfica segura (llamada *reenvío por X11*)
  - Si el servidor usa la técnica del reenvío de puerto, los protocolos considerados como inseguros (POP, IMAP...), se pueden cifrar para garantizar una comunicación segura
- Versiones gratuitas y comerciales (*):
  - proyecto openSSH ⇨ http://www.openssh.org (OpenSSH v3.8 Febrero 2004)
  - (*) http://www.ssh.com

# SSH (Secure SHell) (III)

- Secuencia de eventos de una conexión SSH:

1. Se crea una capa de transporte segura para que el cliente sepa que está efectivamente comunicando con el servidor correcto. Luego se cifra la comunicación entre el cliente y el servidor por medio de un código simétrico

2. Con la conexión segura al servidor en su lugar, el cliente se autentifica ante el servidor sin preocuparse de que la información de autentificación pudiese exponerse a peligro. OpenSSH usa claves DSA o RSA y la versión 2.0 del protocolo SSH para autenticaciones predeterminadas

3. Con el cliente autentificado ante el servidor, se pueden usar varios servicios diferentes con seguridad a través de la conexión, como una sesión shell interactiva, aplicaciones X11 y túneles TCP/IP

# SSH. Reenvío por puertos

- El reenvío por TCP/IP trabaja con el cliente SSH y pide que un determinado puerto en el lado del cliente o del servidor sea asignado a la conexión SSH existente.

- Para asignar un puerto local del cliente a un puerto remoto del servidor, primero hay que saber los números de puerto de ambas máquinas. Es posible asignar dos puertos no estándar, diferentes el uno del otro.

  **ssh -L *<puerto-local>*:*<maquina-remota>*:*<puerto-remoto>* *<nombre-usuario>*@*<maquina>***

- P.e., para controlar su correo electrónico en un servidor llamado correo.dominio.com usando POP y SMTP, y SSH está a disposición en ese servidor, el reenvío por TCP/IP se configura con:

  **ssh -L 1100:correo.dominio.com:110 1025:correo.dominio.com:25 usuario@correo.dominio.com**

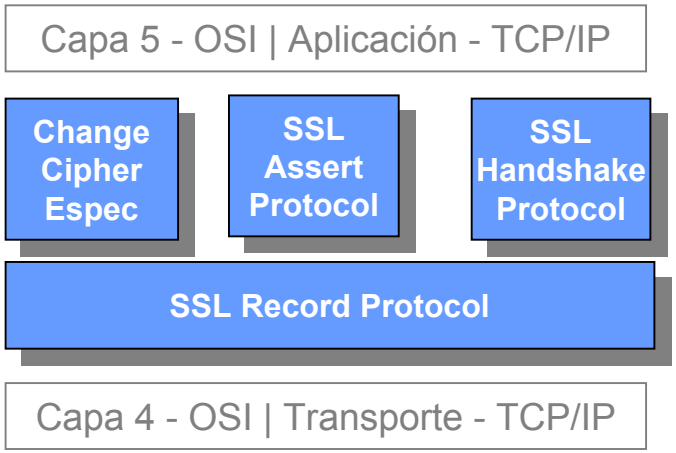
- El reenvío por puertos es especialmente útil si se tiene un cortafuegos. Si el cortafuegos está configurado para permitir el tráfico SSH a través de su puerto estándar (22) pero bloquea el acceso a través de otros puertos, sigue siendo posible una conexión entre dos hosts que usen los puertos bloqueados desviando su comunicación a través de una conexión SSH establecida entre ellos

# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
  - **Nivel de Aplicación**
  - **Nivel de Transporte**
  - **Nivel de Red**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

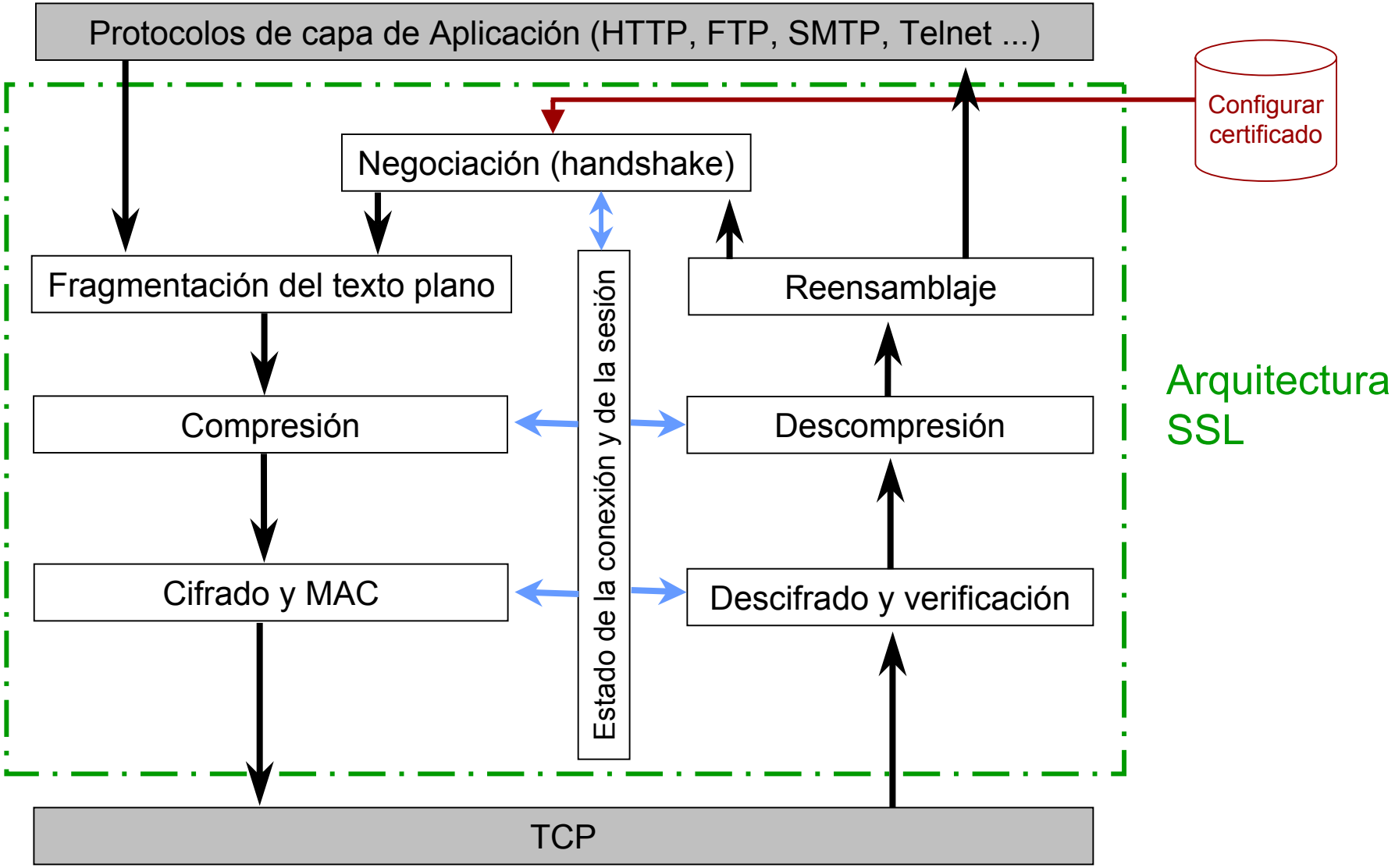
# SSL (Secure Socket Layer)

- El protocolo SSL fue desarrollado por Netscape en 1994 y puesto en dominio público para la definición de canales seguros sobre TCP. Su objetivo es la realización de conexiones seguras a los servidores independientemente del SO de los extremos del canal.
- Está compuesto por dos capas:
  - La primera capa (**SSL Record Protocol**), encapsula los protocolos de nivel más alto y construye el canal de comunicaciones seguro
  - La segunda capa está formada por tres protocolos:




- **SSL Handshake protocol** se encarga de gestionar la negociación de los algoritmos de cifrado, y la autenticación entre el cliente y el servidor
- **SSL Assert Protocol** señaliza errores y problemas en la sesión establecida
- **Change Cipher Espec Protocol** consiste en un solo mensaje de 1 byte que sirve para notificar cambios en la estrategia de cifrado

# SSL. Arquitectura



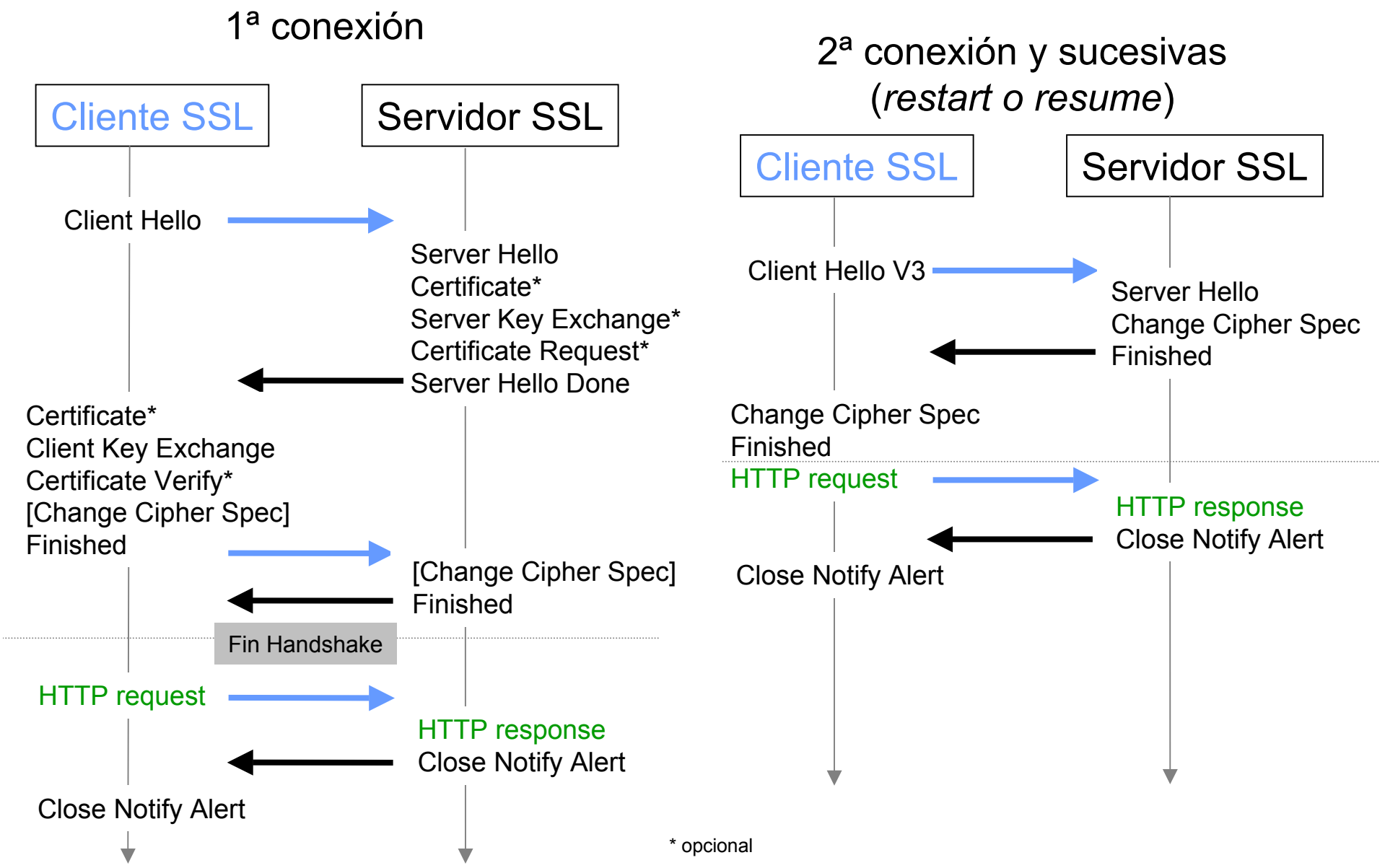
# SSL. Funcionamiento

- Su funcionamiento es el siguiente:
  - El cliente al hacer la conexión informa sobre los sistemas criptográficos que tiene disponibles, y el servidor responde con un identificador de la conexión, su clave certificada e información sobre los sistemas criptográficos que soporta
  - El cliente deberá elegir un sistema criptográfico, verificará la clave pública del servidor. Entonces se genera una clave cifrada con la clave del servidor
  - Este es uno de los puntos importantes del protocolo SSL, porque si alguien pudiese descifrar la información, sólo conseguiría romper esa conexión, y una conexión posterior requeriría una clave criptográfica **diferente**
  - Una vez finalizado este proceso, los protocolos toman el control de nivel de aplicación, de modo que SSL nos asegura que:
    - los mensajes que enviamos o recibimos no han sido modificados
    - ninguna persona sin autorización puede leer la información transmitida
    - efectivamente recibe la información quien debe recibirla

# SSL Handshake Protocol

- Genera los parámetros criptográficos del estado de la sesión
- Opera sobre el SSL Record Layer Protocol
- Tiene dos mecanismos de negociación de sesión:
  - Full Handshake (1ª conexión)
  - Abbreviated Handshake (conexiones posteriores)

# SSL Handshake Protocol (II)



# SSL. Protocolos de capa superior

- Versión actual SSL 3.0
- SSL es capaz de trabajar de forma transparente con todos los protocolos que trabajan sobre TCP
- Para ello el IANA tiene asignado un número de puerto por defecto a cada uno de ellos:

| Identificador de protocolo | Puerto TCP | Descripción |
|---|---|---|
| https | 443 | HTTP sobre SSL |
| smtps | 465 | SMTP sobre SSL |
| nttps | 563 | NTTP sobre SSL |
| ladps | 646 | LDAP sobre SSL |
| telnets | 992 | TELNET sobre SSL |
| imaps | 993 | IMAP sobre SSL |
| ircs | 994 | IRC sobre SSL |
| pop3s | 995 | POP3 sobre SSL |
| ftps-data | 989 | FTP-Datos sobre SSL |
| ftps-control | 990 | FTP-Control sobre SSL |

Nota: estos puertos también son válidos para las implementaciones de dichos protocolos sobre TLS

# SSL. Servicios de seguridad

- Confidencialidad ⇨
  - Cifrado y descifrado
- Autenticación ⇨
  - Autenticación basada en certificado
  - Criptosistema de clave pública
- Integridad ⇨
  - Message Authentication Code (MAC)
- No repudio ⇨
  - Certificado
  - Firma digital

# SSL. Problemas

- Sólo trabaja sobre TCP (no UDP ni IPX)
  - crear sesión SSL sobre TCP y cifrar los paquetes UDP con el fruto de esa negociación. Requiere que cada paquete UDP pueda descifrarse por separado y se cifre con claves distintas
- No repudio de transacciones
  - SSL lo implementa si ambos extremos tienen certificados
  - Usar S/MIME sobre SSL
- Ineficiencia debido al handshake inicial
  - Cachear las sesiones (válido para HTTP, pero no para otros)
  - Uso de hardware especializado que acelere el tráfico SSL
    - Tarjeta aceleradora integrada en cada servidor SSL
    - Dispositivo externo y autónomo dedicado exclusivamente al cifrado y descifrado SSL (compartido por todos los servidores SSL)
    - Dispositivo que integra el balanceo de carga con el cifrado SSL

# Sumario

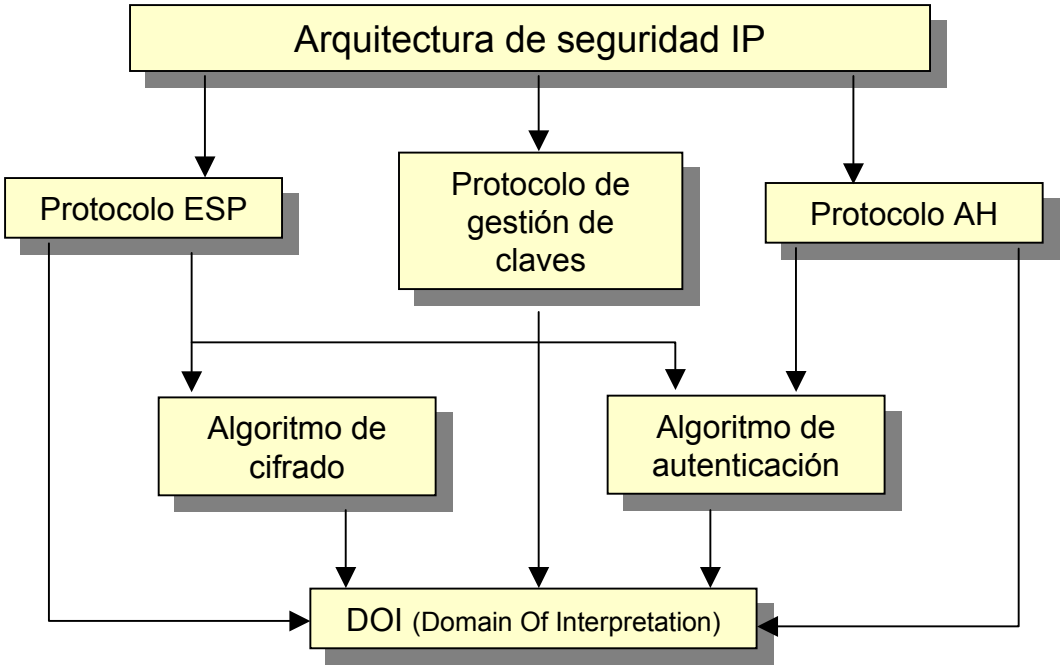
- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
  - **Nivel de Aplicación**
  - **Nivel de Transporte**
  - **Nivel de Red**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# IPSec (IP Security)

- RFC 2401, estándar IETF desde 1999
- Suministra seguridad a nivel de red, proporcionando seguridad para IP y los protocolos de capas superiores
- Provee:
  - Control de accesos
  - Integridad no orientada a la conexión
  - Autenticación del origen de datos
  - Rechazo o reenvío de paquetes
  - Confidencialidad
  - Negociación de compresión IP
- Independiente de los algoritmos criptográficos actuales
- Contempla su implementación con IPv4 e IPv6
- Es un componente obligado en IPv6

# IPSec. Arquitectura

- Componentes fundamentales de esta arquitectura:
  - Protocolos de seguridad:
    - AH (Authentication Header), RFC 2402
    - ESP (Encapsulation Security Payload), RFC 2406
  - Asociaciones de seguridad: SA (Security Association)
  - IKE (Internet Key Exchange) RFC 2409
  - Algoritmos de autenticación y cifrado



# IPSec. Authentication Header

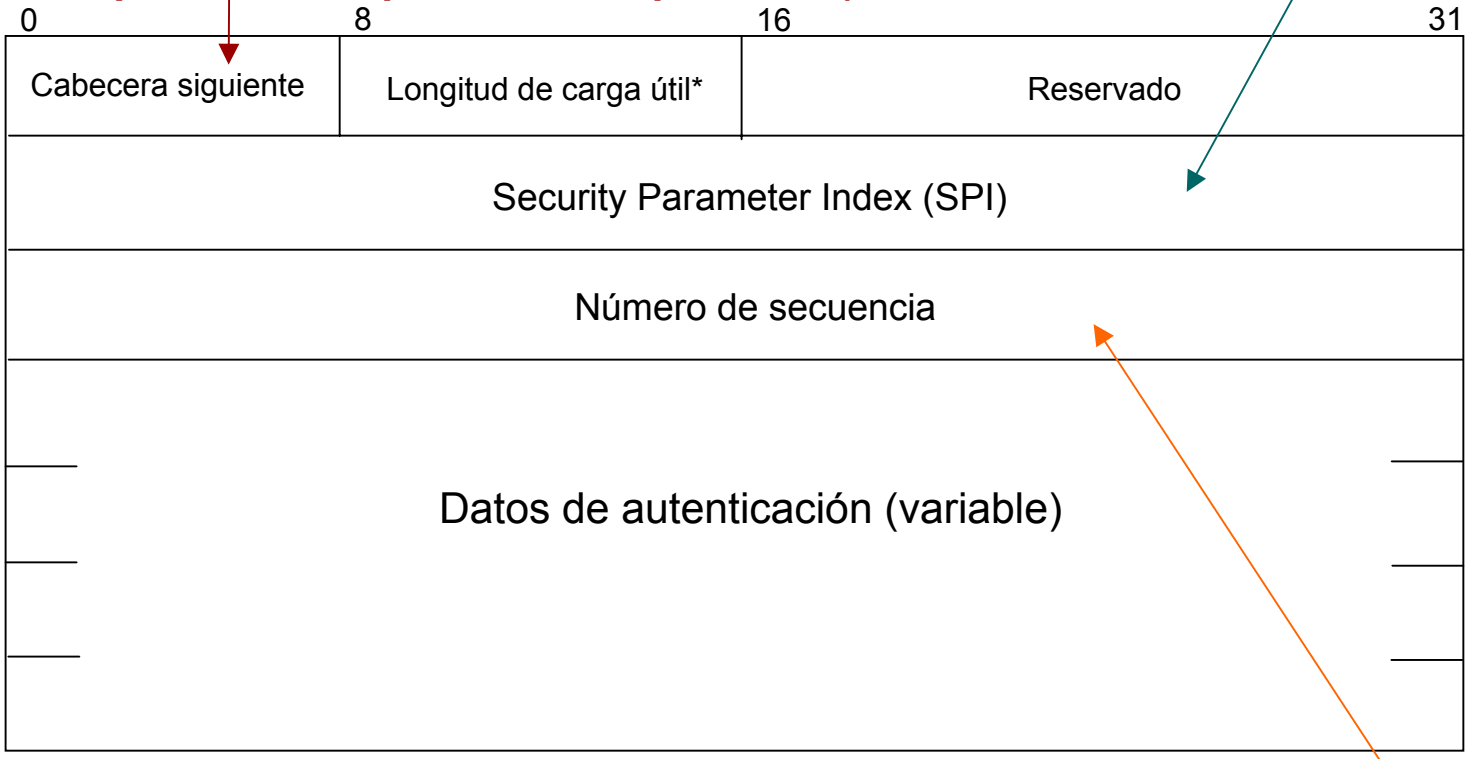
- Encabezado IPSec para proveer servicios de integridad de datos, autenticación del origen de los datos, *antireplay* para IP
- Para proteger la cabecera IP y los datos contra las modificaciones se calcula un MAC en clave (Message Authentication Code) sobre *la mayoría* de los octetos del datagrama IP
- Estándar definido en la RFC 2402
- AH puede ser implementado solo o en combinación con ESP o anidado en modo túnel de IPSec
- Los servicios de seguridad que ofrece pueden ser entre:
  - Dos hosts
  - Un host y un gateway de seguridad
  - Dos gateways de seguridad
- Valor 51d  en el campo Protocol (IPv4), o Next Header (IPv6)

- **Garantiza que el datagrama fue enviado por el remitente y que no ha sido alterado durante su viaje**

# Formato cabecera AH

**Número de protocolo correspondiente al protocolo original (que no va indicado en el campo Protocol de IP) (por ejemplo, 1 para ICMP, 6 para TCP, 17 para UDP)**

Identificador unívoco de la SA




**Número de secuencia para prevenir ataques de *replay***

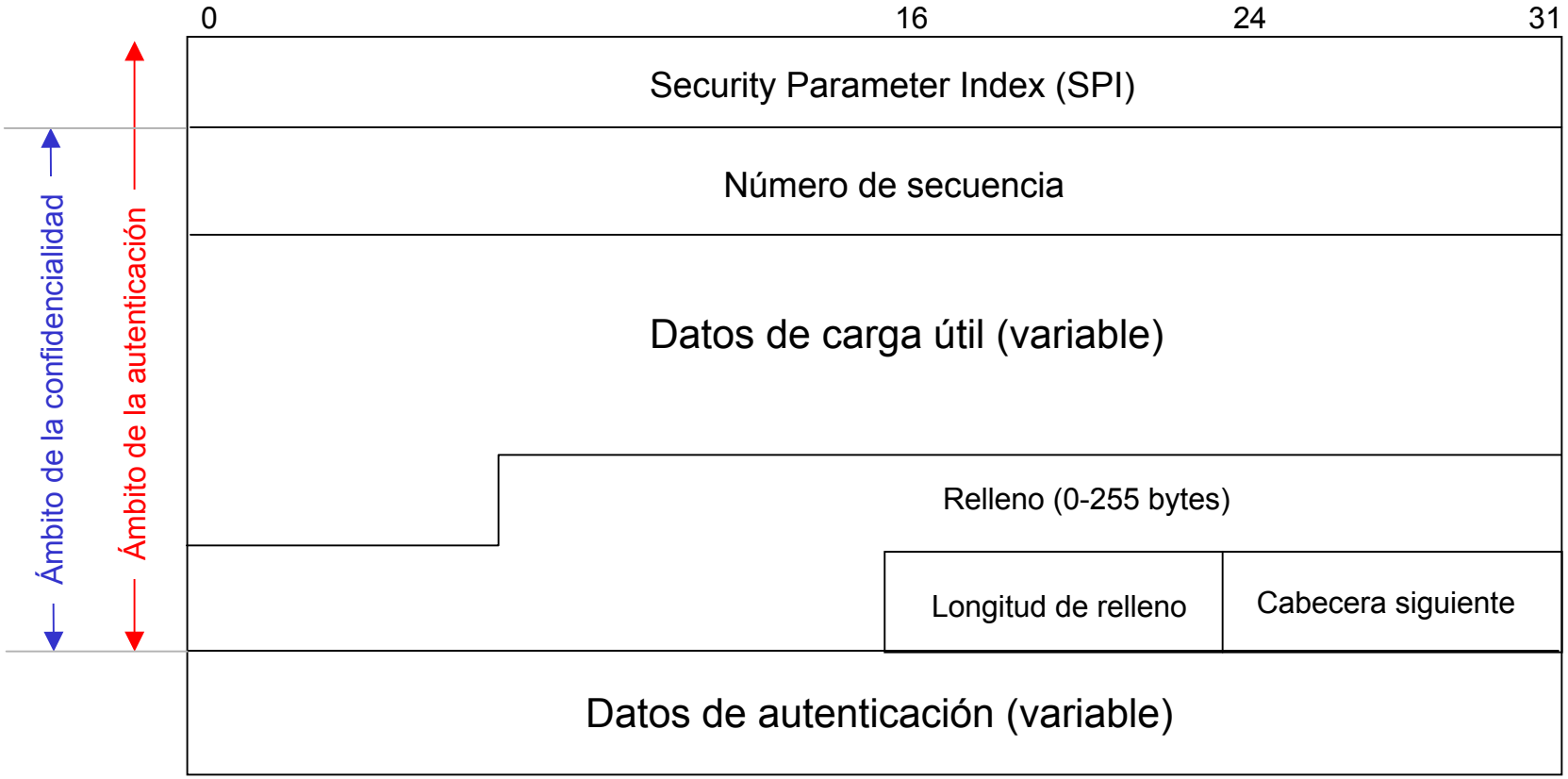
* payload

# IPSec. Encapsulation Security Payload

- Encabezado insertado en el datagrama IP para proveer servicios de confidencialidad, autenticación del origen de los datos, *antireplay* e integridad de datos a IP
- Estándar definido en la RFC 2406
- Valor 50 en el campo Protocol (IPv4), o Next Header (IPv6)

- **Garantiza que el contenido no pueda ser examinado por terceros o, que si lo es, no pueda ser interpretado. Opcionalmente puede incluir la función de AH**

# Formato cabecera ESP



# IPSec. AH vs EPS

- ESP provee todo lo que ofrece AH más confidencialidad de datos
- La principal diferencia entre la autenticación provista entre ESP y AH tiene que ver con la cobertura, ESP **no** protege los campos del encabezado IP, a menos que sean encapsulados por ESP (modo túnel)

| | AH | ESP (sólo cifrado) | ESP (cifrado + autenticación) |
|---|---|---|---|
| **Control en el acceso** | √ | √ | √ |
| **Integridad sin conexión** | √ | | √ |
| **Autenticación en el origen de datos** | √ | | √ |
| **Rechazo de paquetes retocados (antireplay)** | √ | √ | √ |
| **Confidencialidad** | | √ | √ |

# IPSec. Security Association

- Una SA es una clase de conexión que permite establecer los servicios de seguridad del tráfico
- En cada SA los servicios de seguridad pueden hacer uso de AH o ESP, pero **no** de ambos simultáneamente
- Para utilizar los dos, es necesario establecer dos SA
- Una SA se identifica unívocamente por tres valores:
  - SPI (Security Parameter Index)
  - Dirección IP destino
  - Identificador del protocolo de seguridad de IPSec (AH o ESP)
- Se pueden definir dos tipos de SA:
  - modo transporte: se trata de una SA entre dos hosts
  - modo túnel: se trata de una SA aplicada a un túnel IP (en este modo existen dos encabezados IP, uno que es el externo que lleva los datos del *destino del túnel* y otro interno a este que indica el destino final)
- Un host debe soportar ambos modos, un gateway de seguridad sólo debe soportar el modo túnel

# IPSec. Modos de funcionamiento

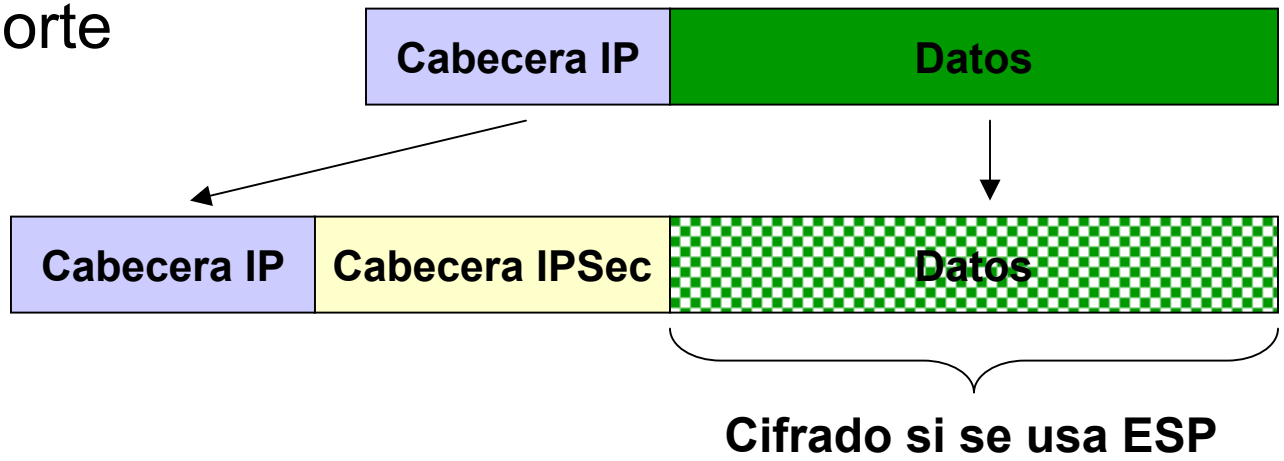
- Modo **transporte** (IP seguro):
  - se protege la carga útil IP (payload) (capa de transporte)
  - comunicación segura extremo a extremo
  - requiere implementación de IPSec en ambos hosts

- Modo **túnel** (IP seguro dentro de IP estándar):
  - se protegen paquetes IP (capa de red)
  - para la comunicación segura entre routers/gateways de seguridad sólo se puede usar este modo
  - permite incorporar IPSec sin afectar a los hosts
  - se integra cómodamente con VPNs

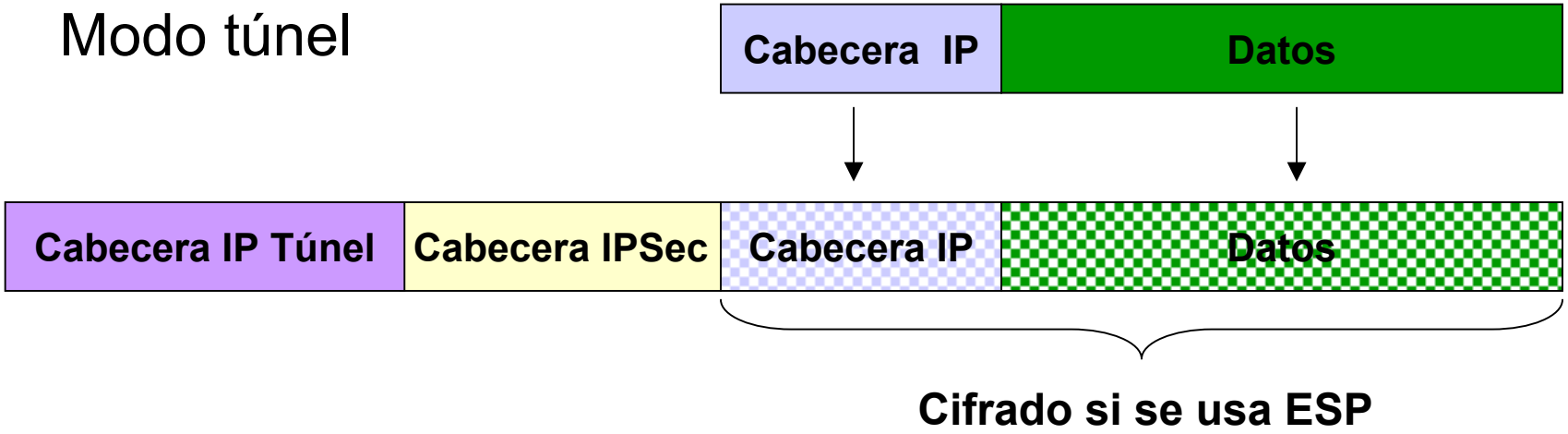
- Combinaciones:
  - AH en modo transporte
  - AH en modo túnel
  - ESP en modo transporte
  - ESP en modo túnel

# IPSec. Encapsulado

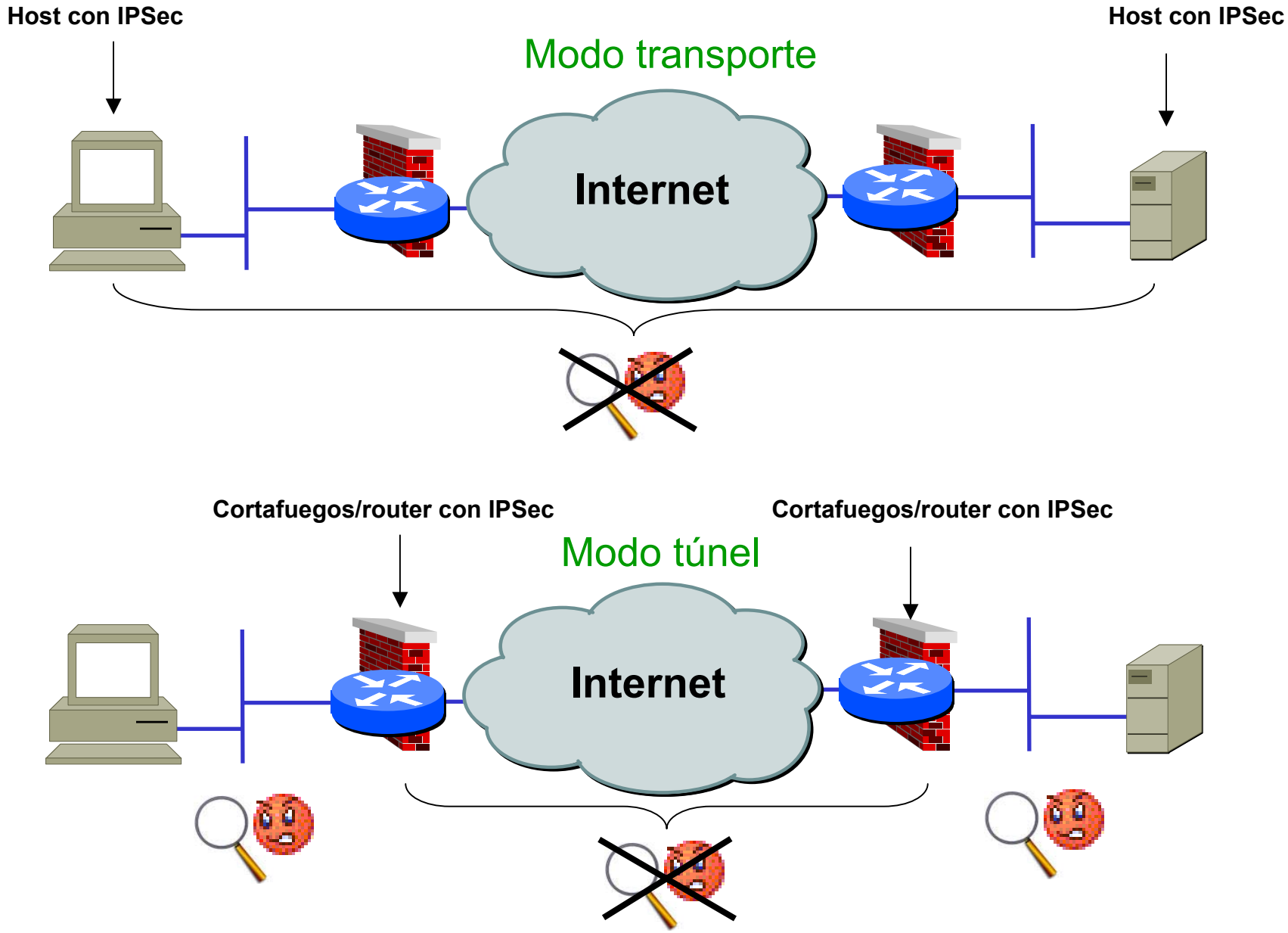
Modo transporte




Modo túnel



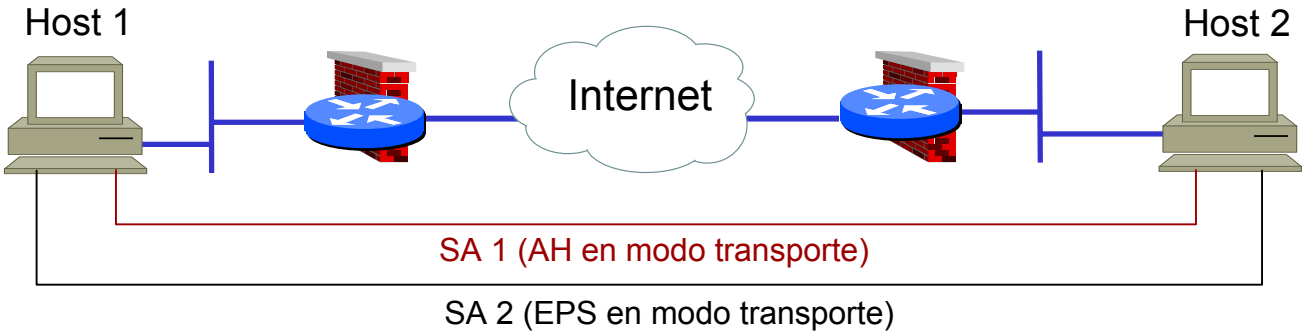
# IPSec. Modos de funcionamiento (II)



# IPSec. Modos de funcionamiento (III)

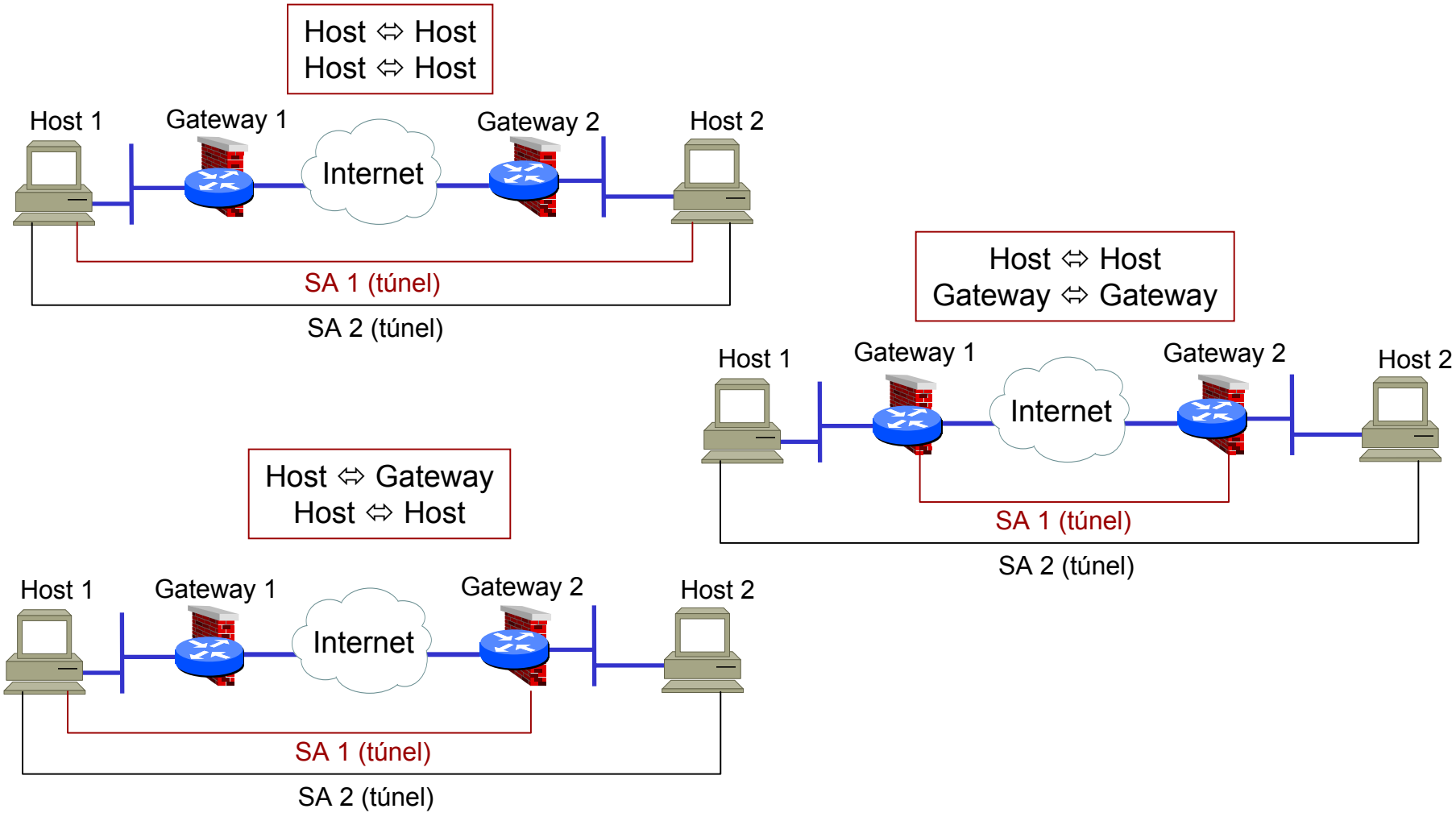
*SA bundle*

- Las diferentes SA pueden iniciarse y finalizar en los mismos puntos o no y se pueden combinar de dos formas:
  - **Transporte adyacente**: se trata de aplicar más de un protocolo de seguridad a un mismo datagrama sin invocar un modo túnel, aprovechando la combinación de AH y ESP



# IPSec. Modos de funcionamiento (IV)

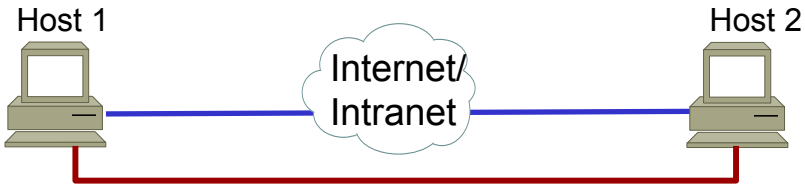
– **Túnel iterado**: son también varias SA, pero implementadas a través de modo túnel, y se puede llevar a cabo de tres formas:



# IPSec. Modos de funcionamiento (V)

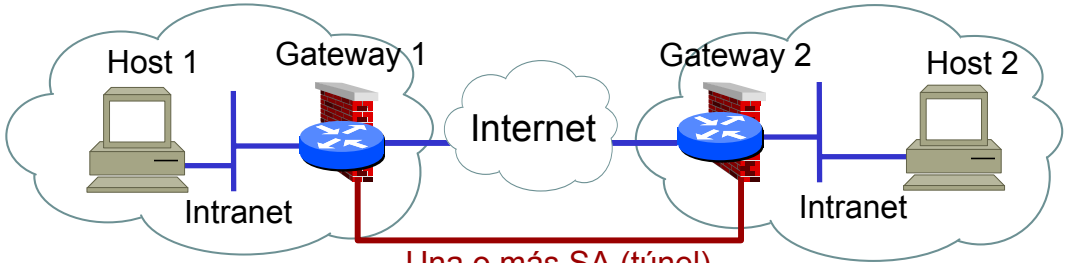
- **Combinaciones**: cualquiera de las propuestas anteriores puede ser combinada con otras, generando empaquetados de SA mixtos. Hay cuatro casos básicos de estas combinaciones que deben ser soportados por todo host o gateway de seguridad que implemente IPSec:

**Seguridad extremo a extremo entre dos hosts a través de Internet o Intranet**




Una o varias SA (AH, ESP, transporte o túnel)

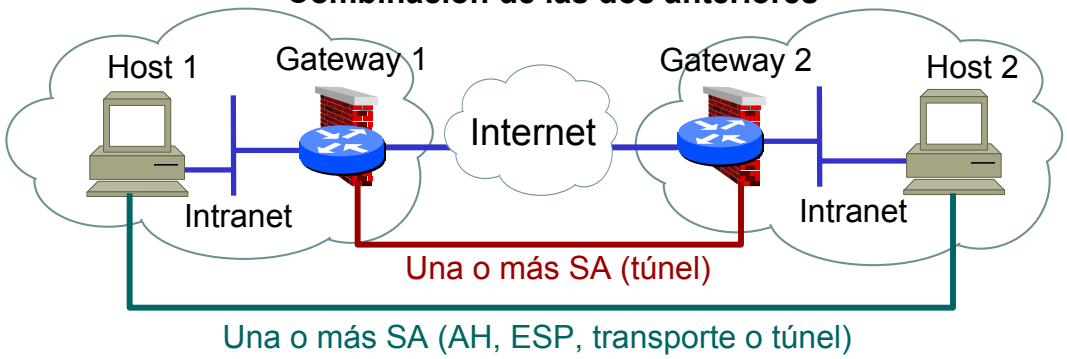
**Soporte con simple VPN**



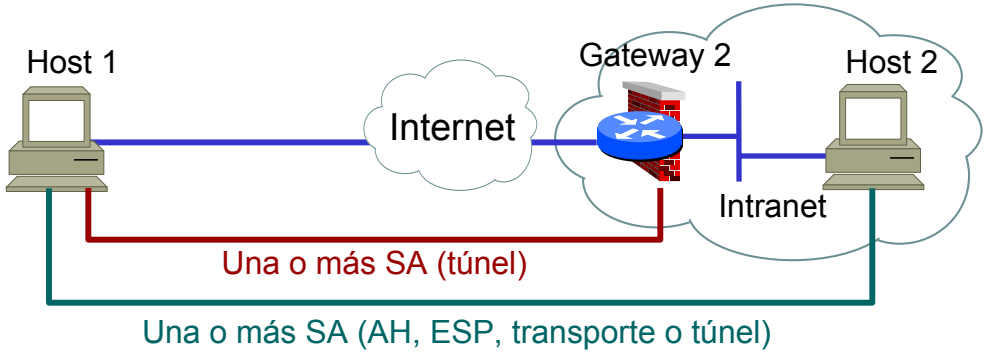

Una o más SA (túnel)

# IPSec. Modos de funcionamiento (VI)

**Combinación de las dos anteriores**



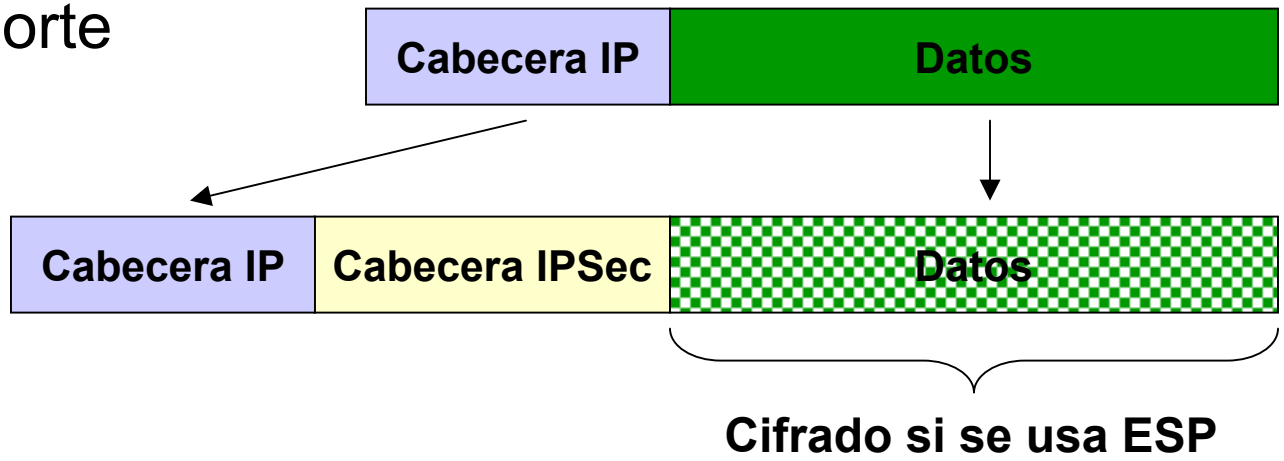

**Host remoto a través de Internet**



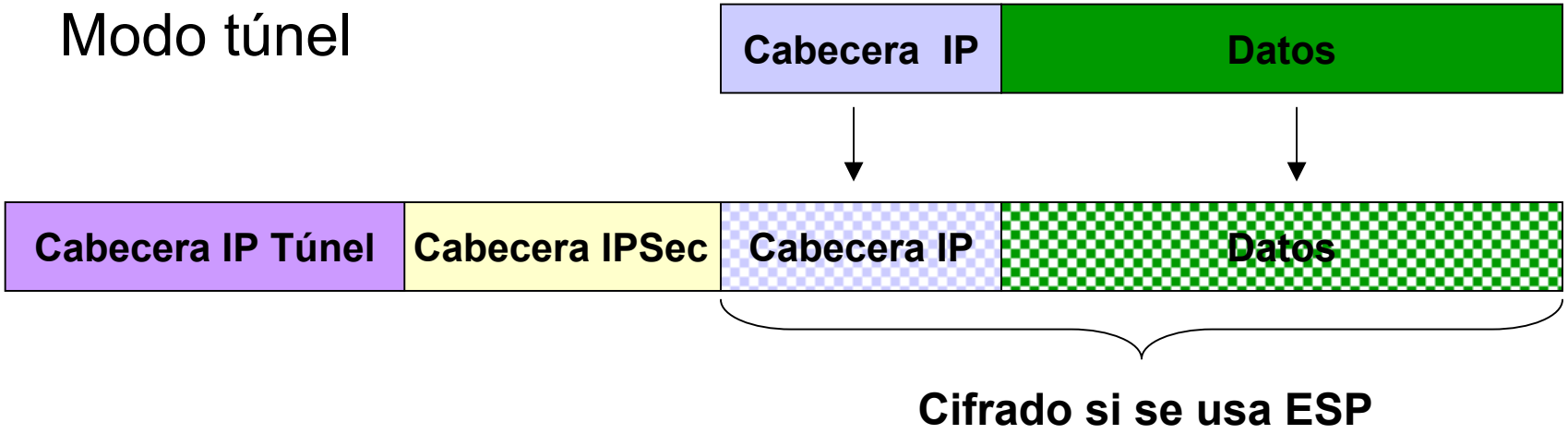

En este caso, entre el host 1 y el gateway de seguridad sólo se puede emplear el modo túnel, y entre los hosts, cualquiera de los modos

# IPSec. Encapsulado

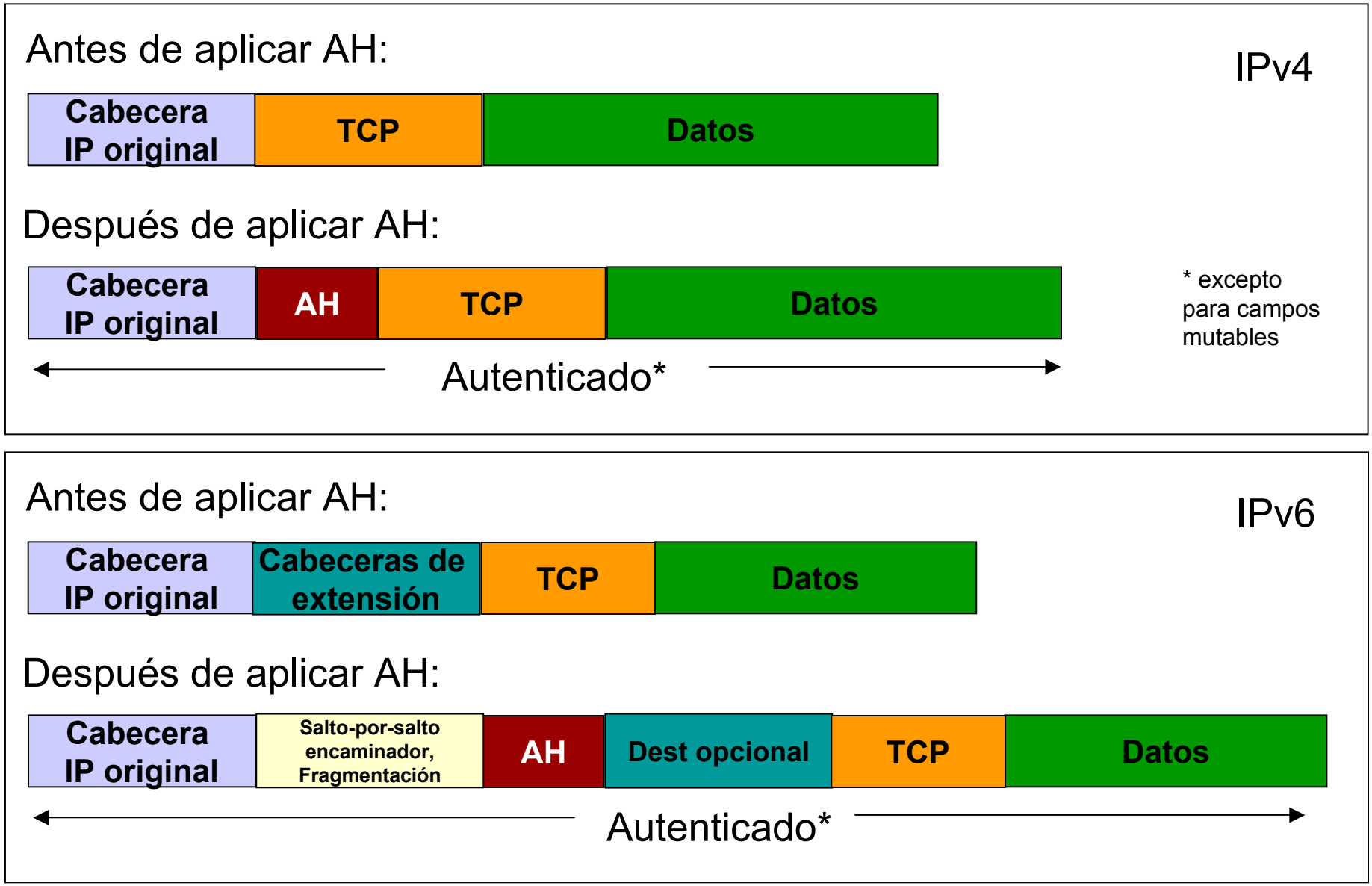
Modo transporte




Modo túnel



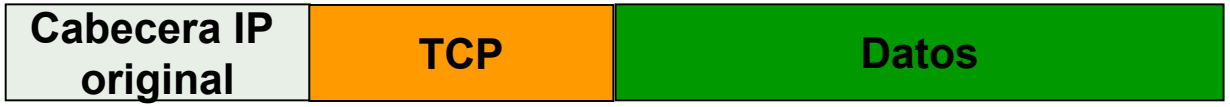
# IPSec. Encapsulado AH modo transporte



# IPSec. Encapsulado AH modo túnel

Antes de aplicar AH:

IPv4



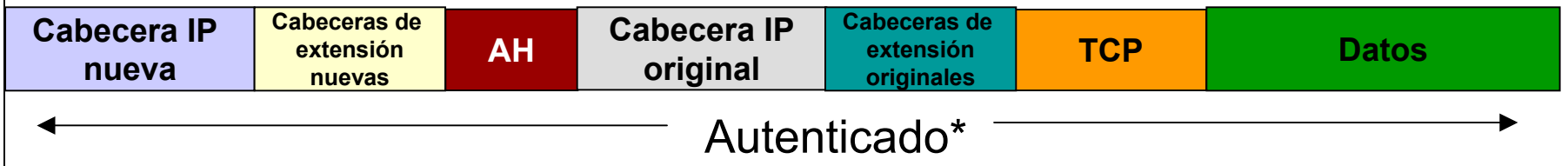

* Excepto para campos mutables en nueva cabecera IP

Después de aplicar AH:


Antes de aplicar AH:

IPv6


Después de aplicar AH:

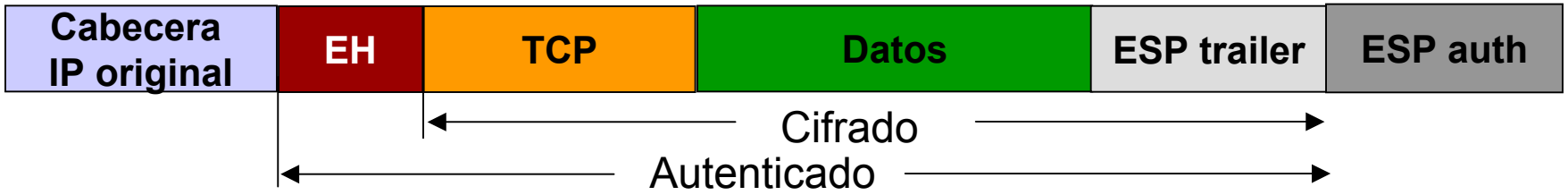


# IPSec. Encapsulado EPS modo transporte

Antes de aplicar EPS:

IPv4

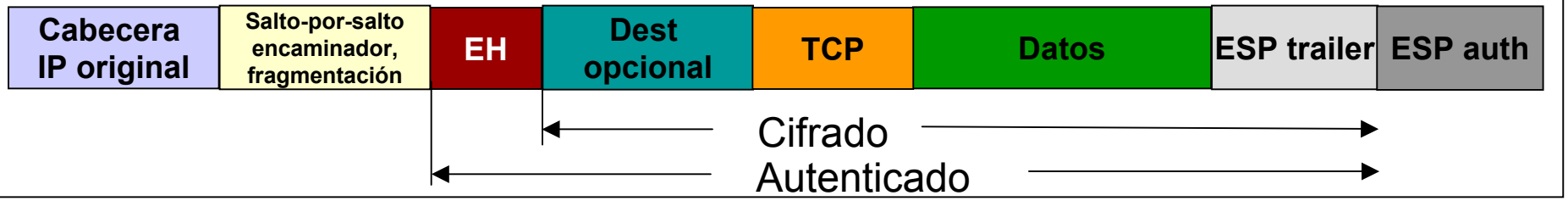

Después de aplicar EPS:




Antes de aplicar EPS:

IPv6

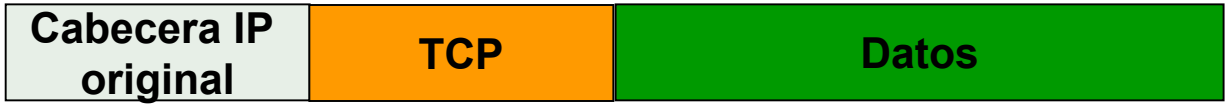

Después de aplicar EPS:



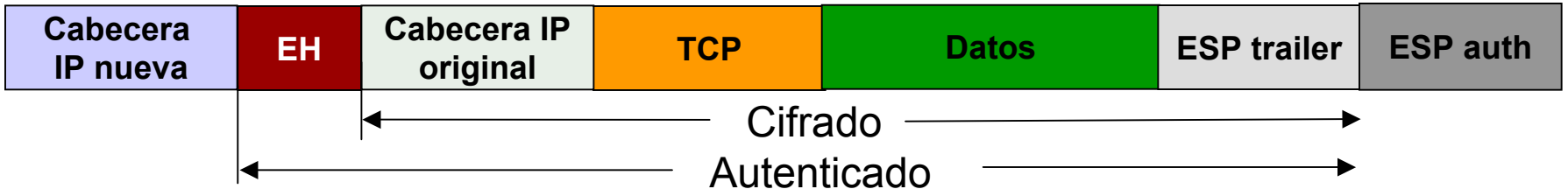
# IPSec. Encapsulado EPS modo túnel

Antes de aplicar ESP:

IPv4



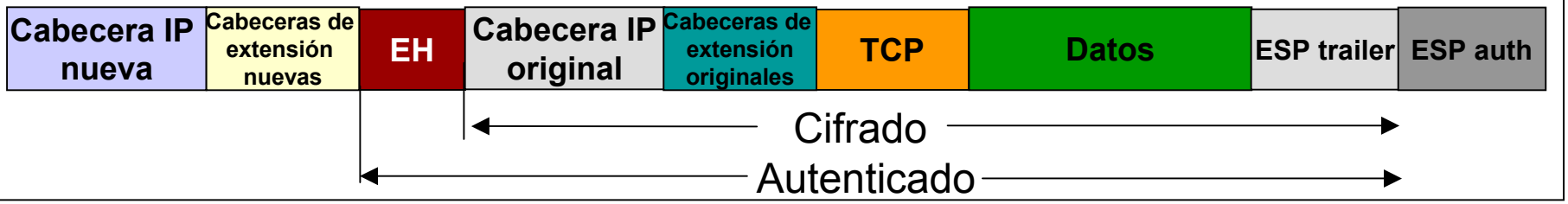

Después de aplicar ESP:




Antes de aplicar ESP:

IPv6


Después de aplicar ESP:



# IPSec. Cifrado (DOI)

- DES (Data Encryption Standard):
  - Claves de 56 bits
  - Rápido
  - No 100% seguro
- Triple DES:
  - Más costoso de procesar (más lento)
  - 100% seguro
- AES (Advanced Encryption Standard):
  - Aún no implementado en productos comerciales
- Otros: Blowfish, CAST-128, IDEA, RC5, Triple IDEA
- Normalmente en routers y servidores de túneles el cifrado se hace por hardware cuando el tráfico es elevado (a partir de 100 sesiones, 2-4 Mbps)

# IPSec. IKE (Internet Key Exchange)

- El protocolo IKE no es parte de IPSec
- Crea Asociaciones de Seguridad (SA) de forma *dinámica* y está definido en la RFC 2409, 1998
- Es un protocolo híbrido basado en el marco definido por ISAKMP y otros dos protocolos de manejo de claves: Oakley y SKEME
- El manejo manual de claves es obligatorio y sólo algunas implementaciones consideran IKE, que ha resultado demasiado complejo e inapropiado
- El uso de IKE fue congelado en el 2001 por la IETF
- El manejo dinámico en el 2001 es llamado “son of IKE” y se encuentra en discusión en el área de seguridad de la IETF

# IPSec. IKE (II)

- Oakley (RFC 2412) define una serie de modos de intercambio de claves detallando los servicios que provee cada uno

- SKEME (*Secure Key Exchange Mechanism for Internet*) describe una técnica de intercambio de claves muy versátil que provee anonimato, no repudio y refresco rápido de claves

- ISAKMP (*Internet Security Association and Key Management Protocol*), definido en la RFC 2408, consiste en un mecanismo seguro (manual y automático) de intercambio de claves utilizadas en las tareas de cifrado y autenticación de AH y ESP. Dos fases:
  1ª. Establecer un canal seguro y autenticado (SA)
  2ª. Negociar parámetros de seguridad (KMP)

- Se usa una combinación ISAKMP/Oakley

# IPSec vs SSL/TLS

| | SSL/TLS | IPSec |
|---|---|---|
| **Control de accesos** | | |
| Conexiones permanentes | | ✓ |
| Conexiones efímeras o puestos móviles | ✓ | |
| Ambos tipos de acceso | ✓ | ✓ |
| **Usuarios** | | |
| Todos los usuarios son empleados de la compañía | | ✓ |
| No todos los usuarios son empleados de la compañía | ✓ | |
| No todos los usuarios son empleados de la compañía y, además, algunos trabajan con sus propios sistemas | ✓ | ✓ |
| **Software cliente** | | |
| Todos los usuarios han de tener acceso a todos los recursos de la red | | ✓ |
| Deseamos controlar el acceso a determinadas aplicaciones | ✓ | |
| Necesitamos niveles variables de control de acceso en las diferentes aplicaciones | ✓ | ✓ |

| **Confidencialidad y Autenticidad** | | |
|---|---|---|
| Precisamos de un alto nivel de seguridad en el cifrado y autenticación | | ✓ |
| La confidencialidad y autenticidad no son especialmente críticas en nuestros sistemas | ✓ | |
| Precisamos de niveles moderados de confidencialidad e integridad | ✓ | |
| **Criticidad de los recursos accedidos** | | |
| Alta | | ✓ |
| Moderada | ✓ | |
| Variable | ✓ | ✓ |
| **Criticidad de las funciones realizadas** | | |
| Alta | | ✓ |
| Moderada | ✓ | |
| Variable | ✓ | ✓ |
| **Nivel técnico de los usuarios** | | |
| Entre moderado y alto | | ✓ |
| Entre moderado y bajo | ✓ | |
| **Implantación, flexibilidad y escalabilidad** | | |
| Deseamos una implantación rápida y facilidad de mantenimiento | ✓ | |
| Deseamos flexibilidad en las modificaciones futuras | | ✓ |
| Ambas consideraciones son importantes | ✓ | ✓ |

# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# VPN. Sumario

- ¿Qué es *tunneling*?
- Protocolos de tunneling
- Comparativa
- ¿Qué es una VPN?
- Clasificación general de las VPNs
- Elementos de una VPN
- Implementación de una VPN
- Funcionamiento básico
- ¿Para qué sirven las VPNs?
- Ventajas e inconvenientes de usar VPNs
- Evolución de los estándares de VPNs
- Alternativas a las VPNs

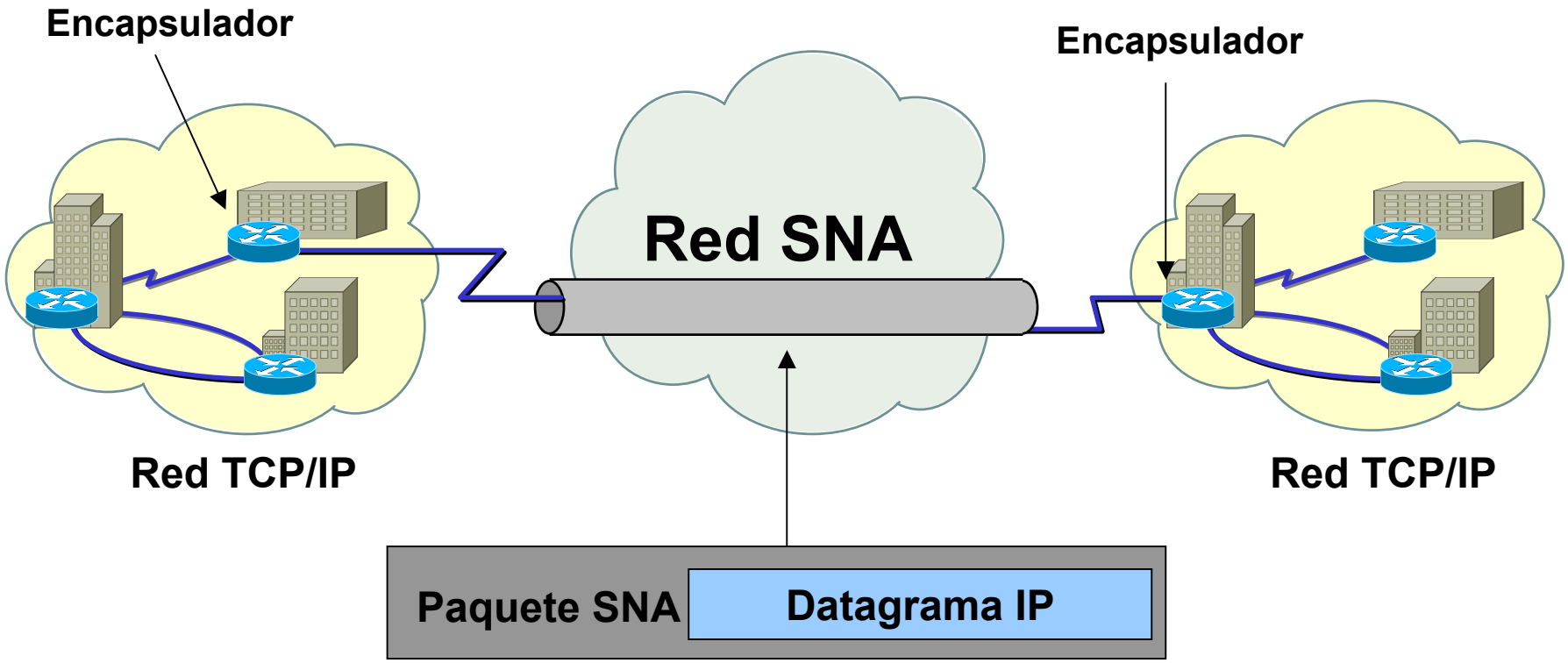
# ¿Qué es tunneling?

- La transmisión de paquetes de datos de un determinado protocolo encapsulados en otro, de manera que el contenido del paquete original puede llegar inalterado a su destino, creando algo así como una conexión punto a punto virtual a través de una red

- Ejemplos:
  - Túnel SNA para enviar paquetes IP
  - MBone: túneles multicast sobre redes unicast
  - 6Bone: túneles IPv6 sobre redes IPv4
  - Túneles IPv4 para hacer enrutamiento desde el origen

- También permiten crear redes privadas virtuales o VPNs (*Virtual Private Networks*)

# ¿Qué es tunneling? Ejemplo




- Túnel SNA transportando datagramas IP
- Los datagramas IP viajan encapsulados en paquetes SNA

SNA = Systems Network Architecture (IBM)

# Protocolos de tunneling

- Hay múltiples protocolos de tunneling:
  - **IPSec**: IP Secure (RFC 2401) [capa 3]
  - **L2F**: Layer 2 Forwarding Protocol (Cisco) (RFC 2341) [capa 2]
  - **PPTP**: Point-to-Point Tunneling Protocol (RFC 2637) [capa 2]
  - **L2TP**: Layer 2 Tunneling Protocol (RFC 2661) [capa 2 y 3]
  - **GRE**: Generic Routing Encapsulation (RFC 1701) [capa 3]
  - **IP/IP**: IP over IP (RFC 2003) [capa 3]
  - **IPSec**: IP Secure (RFC 2475) [capa 3]
  - **MPLS**: Multi-Protocol Label Switching (RFC 2917) [capas 2 y 3]
  - **MPOA**: Multi-Protocol Over ATM [capa 3]

- PPTP: orientado al usuario ⇨ permite establecer un túnel de forma transparente al proveedor de Internet
- L2TP: orientado al proveedor ⇨ permite establecer un túnel de forma transparente al usuario (generalmente se utiliza junto con IPSec)

# Protocolos de tunneling (II)

- PPTP (Point-to-Point Tunneling Protocol):
  - Protocolo desarrollado por Microsoft y normalizado por la IETF (RFC 2637)
  - Permite el tráfico seguro de datos desde un cliente remoto a un servidor corporativo privado
  - PPTP soporta múltiples protocolos de red (IP, IPX, NetBEUI… )
  - Tiene una mala reputación en seguridad
  - Muy usado en entornos Microsoft

- L2F (Layer 2 Forwarding):
  - Protocolo desarrollado por Cisco Systems
  - Precursor del L2TP
  - Ofrece métodos de autenticación de usuarios remotos
  - Carece de cifrado de datos

# Protocolos de tunneling (III)

- L2TP (Layer 2 Tunneling Protocol):
  - Estándar aprobado por la IETF (RFC 2661)
  - Mejora combinada de PPTP y L2F
  - No posee cifrado o autentificación por paquete, por lo que ha de combinarse con otro protocolo, como IPSec
  - Combinado con IPSec ofrece la integridad de datos y confidencialidad exigidos para una solución VPN
  - Permite el encapsulado de distintos protocolos (IP, IPX, NetBEUI…)

# PPTP vs L2TP

- Tanto PPTP como L2TP utilizan PPP para proporcionar una envoltura inicial para los datos y luego guardan las cabeceras adicionales para transporte a través de la red. Los dos protocolos son muy similares, sin embargo existen ciertas diferencias significativas:
  - PPTP requiere que la red sea IP, mientras que L2TP requiere que sólo el túnel provea conectividad punto a punto orientada a paquetes. L2TP puede ser utilizado sobre IP (utilizando UDP), Circuitos virtuales permanentes (PVC's) en Frame Relay, Circuitos virtuales en X.25 o ATM
  - PPTP solamente soporta un túnel simple entre dos puntos, mientras que L2TP permite la utilización de múltiples túneles entre dos puntos. Además se pueden crear diferentes túneles para diferentes QoS
  - L2TP permite compresión de cabeceras. Cuando esta compresión está habilitada, éste opera con 4 bytes de overhead en comparación con los 6 bytes de PPTP
  - L2TP permite autenticación de túnel, mientras que PPTP no lo permite. Sin embargo, cuando cualquiera de estos protocolos es utilizado sobre IPSec, la autenticación de túnel es realizada por IPSec de manera que la autenticación en la capa 2 no es necesaria.

# ¿Qué es una VPN?

- Proporciona el medio para usar una infraestructura de red pública como un canal apropiado para comunicaciones privadas de datos
- Con las tecnologías de cifrado y encapsulación adecuadas, una VPN constituye un túnel (generalmente túnel IP) cifrado y/o encapsulado a través de Internet
- Utiliza protocolos de tunneling
- Proporciona los servicios de las redes privadas (confianza)
- Utiliza conexiones temporales (virtuales)
- Es una combinación de hardware y/o software que:
  - Extiende una intranet o red corporativa a través de la insegura y pública Internet
  - Permite comunicación segura con las oficinas sucursales, usuarios móviles o remotos y clientes

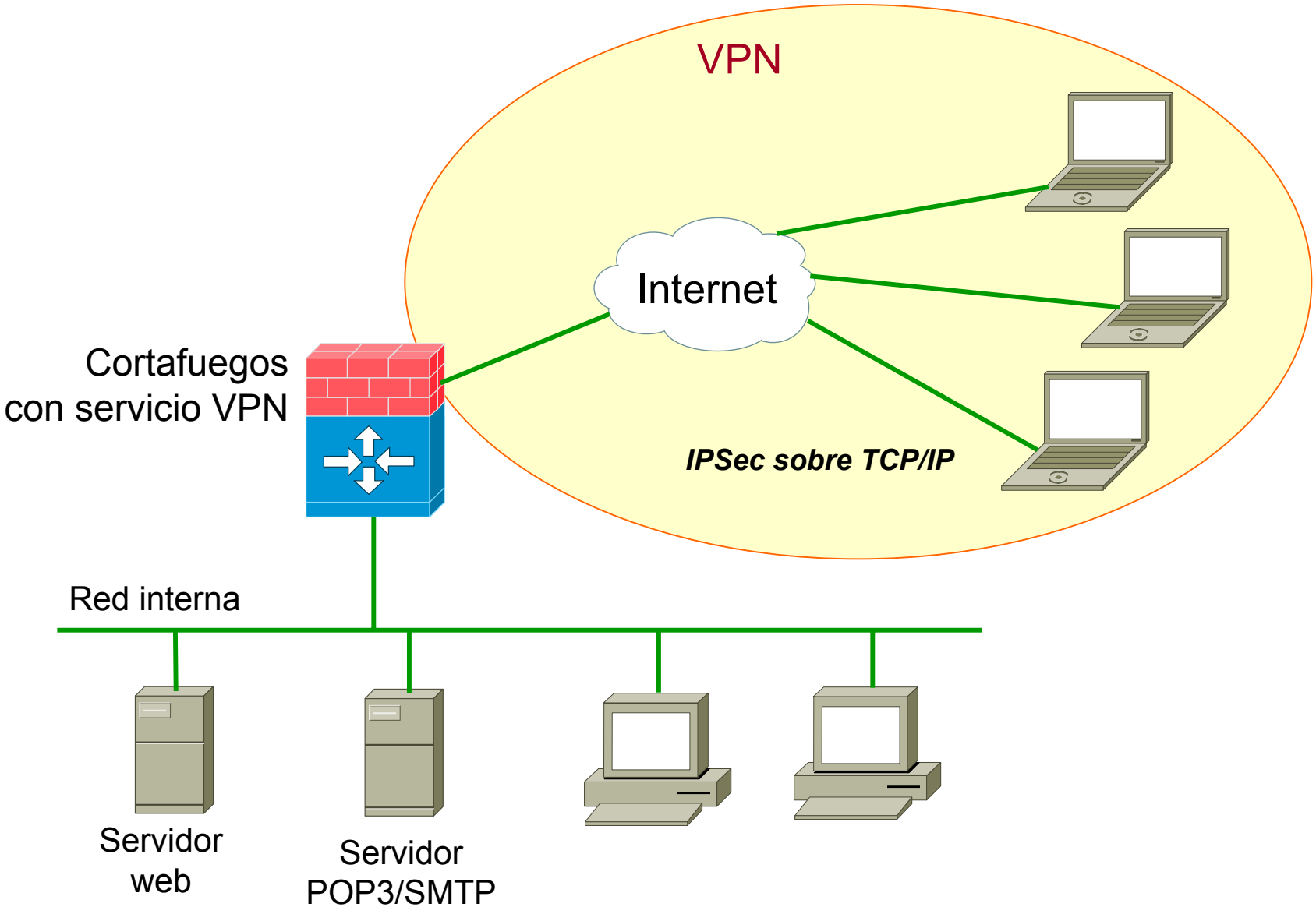
# Clasificación general de VPNs

- VPNs de Intranets
  - proporcionan conectividad interna entre distintos emplazamientos de una misma empresa
- VPNs de Acceso Remoto
  - amplían la red interna a los teletrabajadores, trabajadores itinerantes y a las oficinas remotas
- VPNs de Extranets
  - amplían la red de las empresas e incluyen proveedores, empresas asociadas y/o clientes

# Elementos de una VPN

- Las VPNs se basan en las siguientes tecnologías
  - Firewalls
    - Como mecanismo de protección adicional
  - Autenticación
    - Para dar acceso sólo a sistemas permitidos
  - Cifrado
    - Para asegurar confidencialidad e integridad
  - Tunneling
    - Como mecanismo de intercambio de información

# Elementos de una VPN (II)



# Implementación de una VPN

- Hay que realizar las siguientes operaciones
  - Diseñar una topología de red y firewalls
    - Teniendo en cuenta los costes y la protección
  - Escoger un protocolo para los túneles
    - Teniendo en cuenta los equipos finales
    - Teniendo en cuenta las aplicaciones finales
  - Diseñar una PKI (Public Key Infraestructure)
    - Teniendo en cuenta las necesidades del protocolo
- En el mercado hay ofertas de productos que tienen integradas varias de las opciones anteriores:
  - Altavista Tunnel, *Digital (*para redes IP y protocolo propietario)
  - Private Internet Exchange (PIX), *Cisco Systems (*para redes IP y protocolo propietario)
  - S/WAN, *RSA Data Security (*para redes IP y protocolo estándar (IPSec))

# Funcionamiento básico de una VPN

1. El usuario remoto marca a su ISP local y se conecta a la red del ISP de forma normal

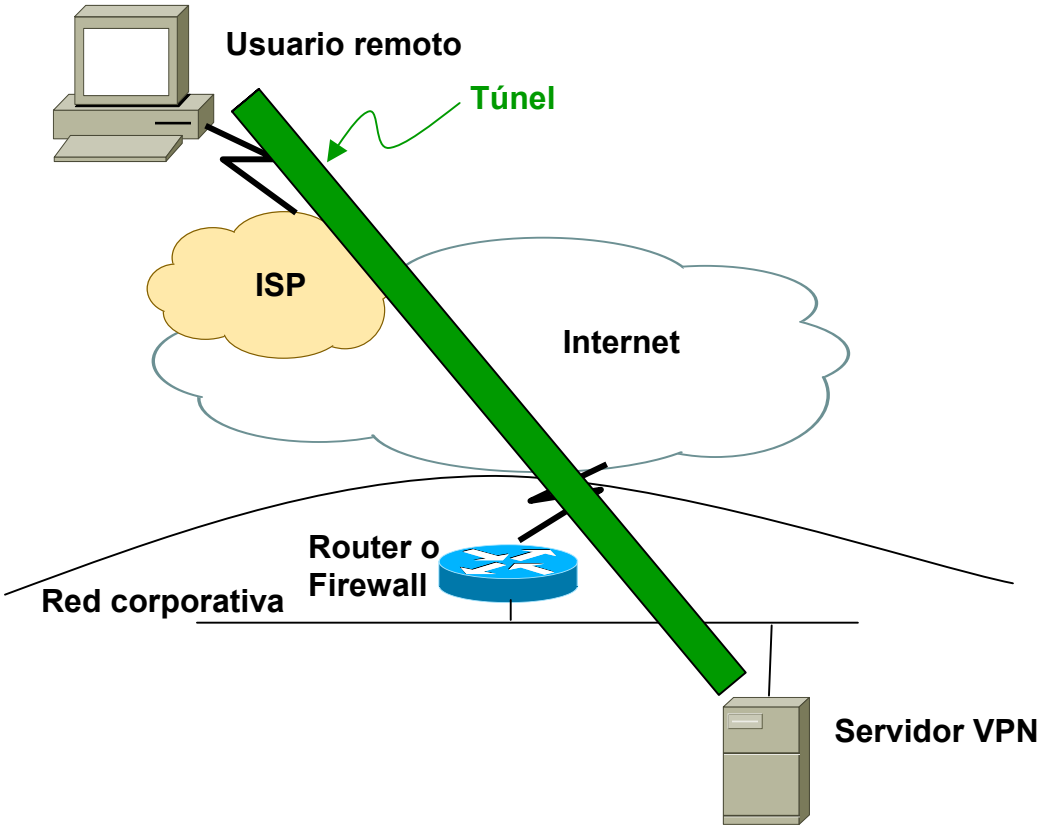
2. Cuando desea conectarse a la red corporativa, el usuario inicia el túnel mandando una petición a un servidor VPN de la red corporativa

3. El servidor VPN autentica al usuario y crea el otro extremo del túnel

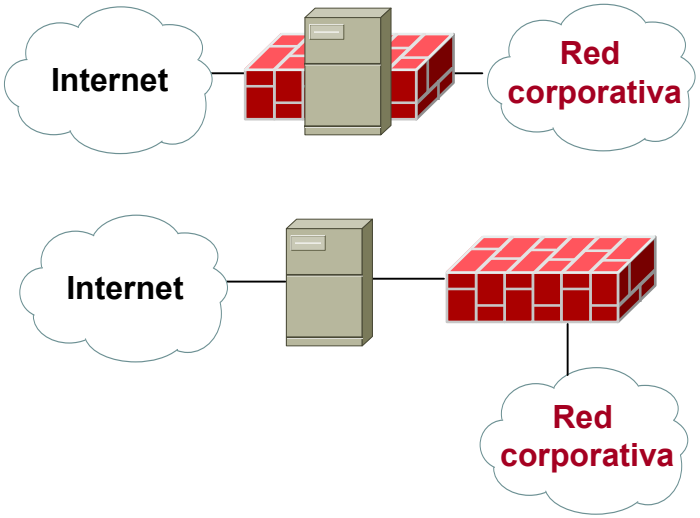
4. El usuario comienza a enviar datos a través del túnel, que son cifrados por el software VPN (del cliente) antes de ser enviados sobre la conexión del ISP

5. En el destino, el servidor VPN recibe los datos y los descifra, propagando los datos hacia la red corporativa. Cualquier información enviada de vuelta al usuario remoto también es cifrada antes de enviarse por Internet

# Funcionamiento básico de una VPN (II)

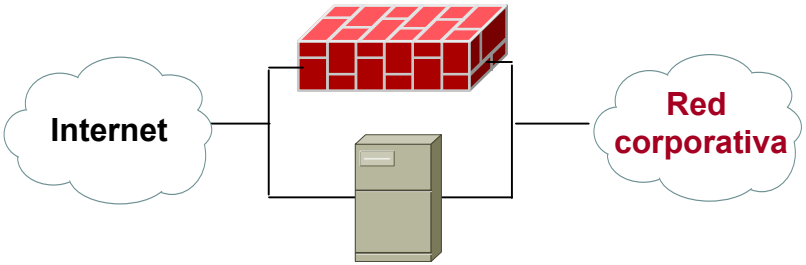



Puede haber distintas configuraciones Firewall - servidor VPN:









# ¿Para qué sirven las VPNs?

- Se pueden usar como una Extranet
- Son más seguras que una Extranet
- Permiten conectar diferentes delegaciones de una empresa, simulando una red local de una manera transparente y económica
- Proporcionan acceso a los diferentes recursos de la red de forma remota a todos los usuarios de la red corporativa (clientes, socios, consultores...)
- Las VPNs seguras proporcionan:
  - Confidencialidad (cifrado de los datos)
  - Integridad (IPSec, asegura que los datos no son modificados en tránsito)
  - Autenticación de usuarios (certificados X.509; identificación de usuarios y protección contra ataques de suplantación)
  - Control de acceso a la red (políticas VPN)
  - No repudio

# Ventajas de las VPNs

- Ahorro en costes
- No se compromete la seguridad de la red empresarial
- El cliente remoto adquiere la condición de miembro de la LAN (permisos, directivas de seguridad)
- El cliente tiene acceso a todos los recursos ofrecidos en la LAN (impresoras, correo electrónico, base de datos, …)
- Acceso desde cualquier punto del mundo (siempre y cuando se tenga acceso a Internet)

# Inconvenientes de las VPNs

- No se garantiza disponibilidad ( NO Internet ⇨ NO VPN)
- No se garantiza el caudal (red pública)
- Gestión de claves de acceso y autenticación delicada y laboriosa
- La fiabilidad es menor que en una línea dedicada
- Mayor carga en el cliente VPN (encapsulación y cifrado)
- Mayor complejidad en la configuración del cliente (proxy, servidor de correo, … )
- Una VPN se considera segura, pero no hay que olvidar que la información sigue viajando por Internet (no seguro y expuestos a ataques)

# Evolución de los estándares de VPN

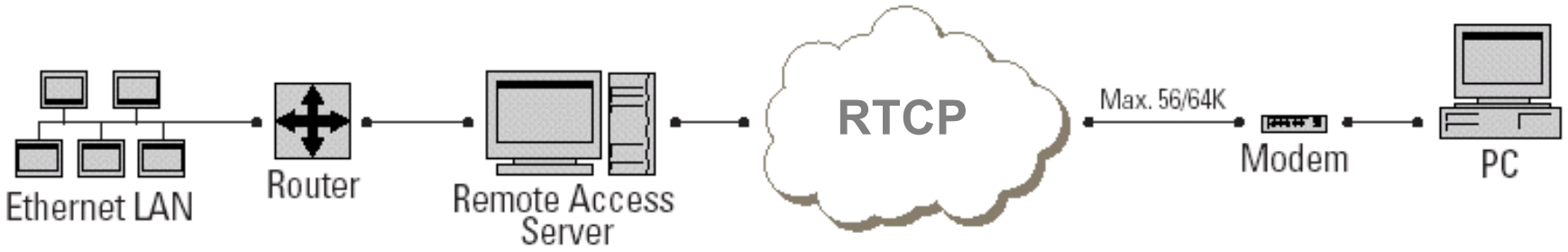
- 1996: Tunneling protocols
  - PPTP (Microsoft), L2F (Cisco)
  - Enfoque al tunneling forzoso
- 1997: VPN standardization
  - Estandarización del tunneling: L2TP
  - Autenticación y cifrado: EAP, IPSEC
  - Incremento del desarrollo del tunneling voluntario
- 1998: VPN solutions
  - Gestión de usuario centralizada: RADIUS,LDAP
  - Auditoría, contabilidad y aviso de alarmas: RADIUS
  - Gestión de red: SNMP

---

- IP Tunneling IP
  - IPSec
- PPP Tunneling
  - PPTP/L2TP
- Circuit-Level Proxying
  - SOCKS v5

# Alternativas a las VPNs

- **RAS** (Remote Access Service)
  - Sistemas de acceso remoto basado en llamadas conmutadas (RTC, RDSI)
  - Se produce una llamada del cliente al servidor de RAS
  - El coste de esta llamada es el de una llamada conmutada entre los dos extremos de la comunicación
  - Se pueden tener tantas conexiones simultáneas como dialers (módems) tengamos disponibles



# Alternativas a las VPNs (II)

- **Alquiler de líneas dedicadas**:
  - Son seguras ya que sólo circulamos nosotros
  - Alto coste económico
  - El ancho de banda del que queramos disponer va en proporción a lo que se esté dispuesto a pagar
  - Ej: líneas T1/E1, frame-relay, RDSI

- **WAN** :
  - Coste muy elevado, no asumible por la mayoría de empresas
  - Ej: FDDI, ATM, ...

# Sumario

- **Introducción**
- **ACLs (Access Control Lists)**
- **VLAN (Virtual LAN)**
- **Criptografía en redes**
- **Protocolos seguros**
- **VPN (*Virtual Private Network*)**
- **Cortafuegos**

# Cortafuegos. Sumario

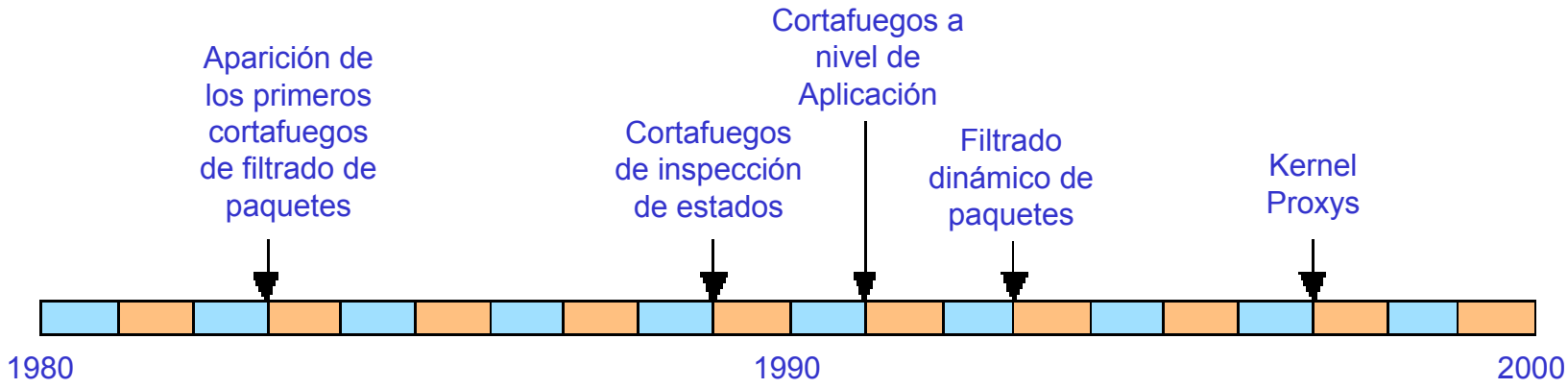
- ¿Qué es un cortafuegos?
- ¿Hasta qué nivel nos protegen?
- Funciones de los cortafuegos
- Componentes de los cortafuegos
- Técnicas aplicadas en los cortafuegos
- Arquitecturas de los cortafuegos
- Distintas clasificaciones de cortafuegos
- Servicios adicionales de un cortafuegos

# ¿Qué es un cortafuegos?

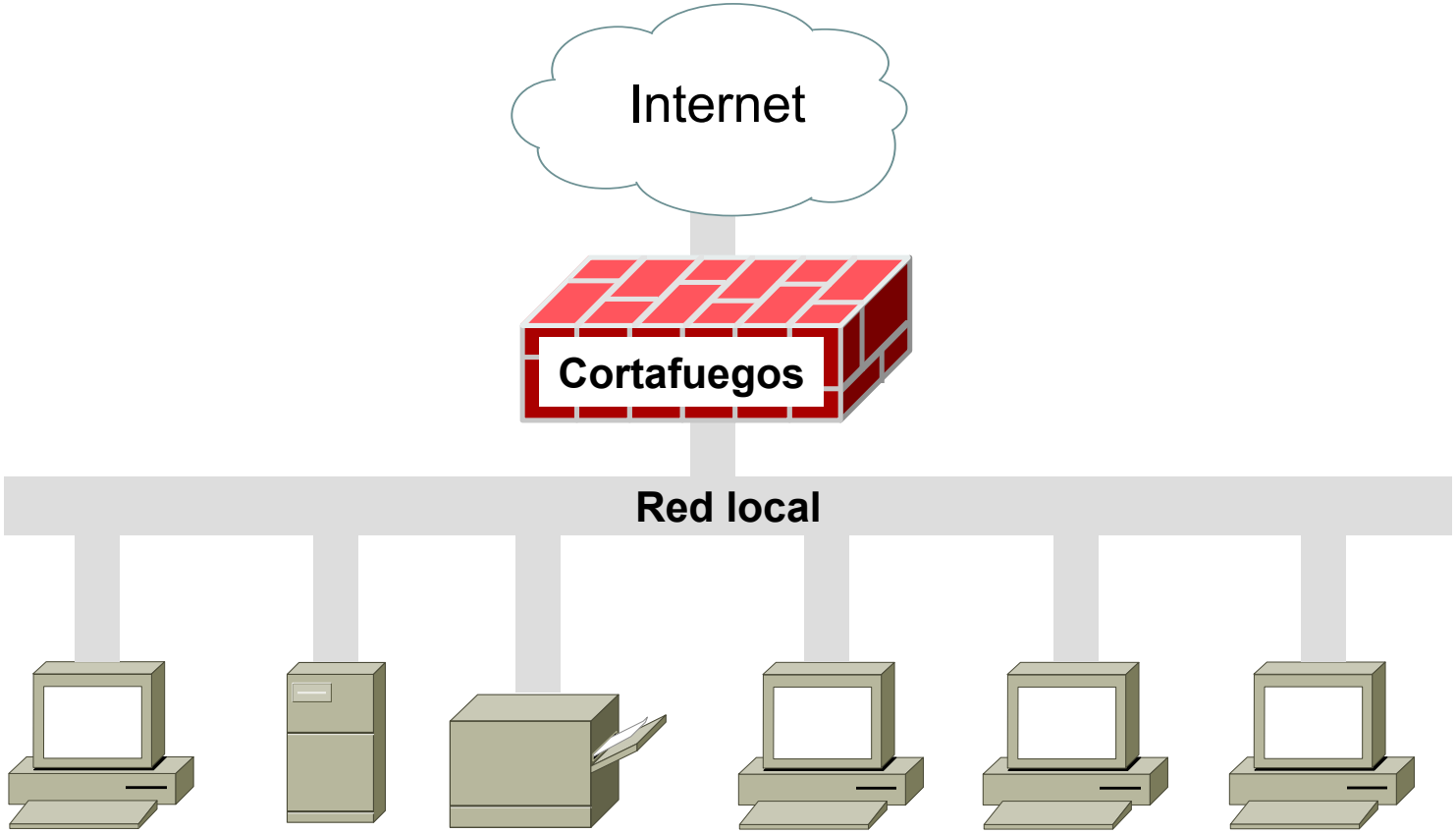
- Combinación de técnicas, políticas de seguridad y tecnologías (hardware y software) encaminadas a proporcionar seguridad en la red, controlando el tráfico que circula entre dos o más redes (y más concretamente, entre una red privada e Internet)

- Cortafuegos = *firewall* o gateway de seguridad

- Evolución:



# ¿Qué es un cortafuegos? (II)



# ¿Hasta qué nivel nos protege un cortafuegos?

- El nivel de protección que ofrece un cortafuegos depende de las necesidades concretas
- Un cortafuegos proporciona un único punto de acceso donde centralizar las medidas de seguridad y auditoría de la red
- No puede protegernos de:
  - amenazas que no pasan a través del cortafuegos
  - amenazas que provienen de nuestra propia red
  - clientes o servicios que admitimos como válidos pero que resultan vulnerables (tunneling sobre HTTP, SMTP...)

- **Los cortafuegos deben combinarse con otras medidas de seguridad en redes ⇨ protocolos seguros**

# Funciones de un cortafuegos

- Controlar, permitiendo o denegando, los accesos desde la red local hacia el exterior y los accesos desde el exterior hacia la red local (redes, subredes o nodos específicos y servicios)
- Filtrar los paquetes que circulan, de modo que sólo los servicios permitidos puedan pasar
- Monitorizar el tráfico, supervisando destino, origen y cantidad de información recibida y/o enviada
- Almacenar total o parcialmente los paquetes que circulan a través de él para analizarlos en caso de problemas
- Establecer un punto de cifrado de la información si se pretende comunicar dos redes locales a través de Internet

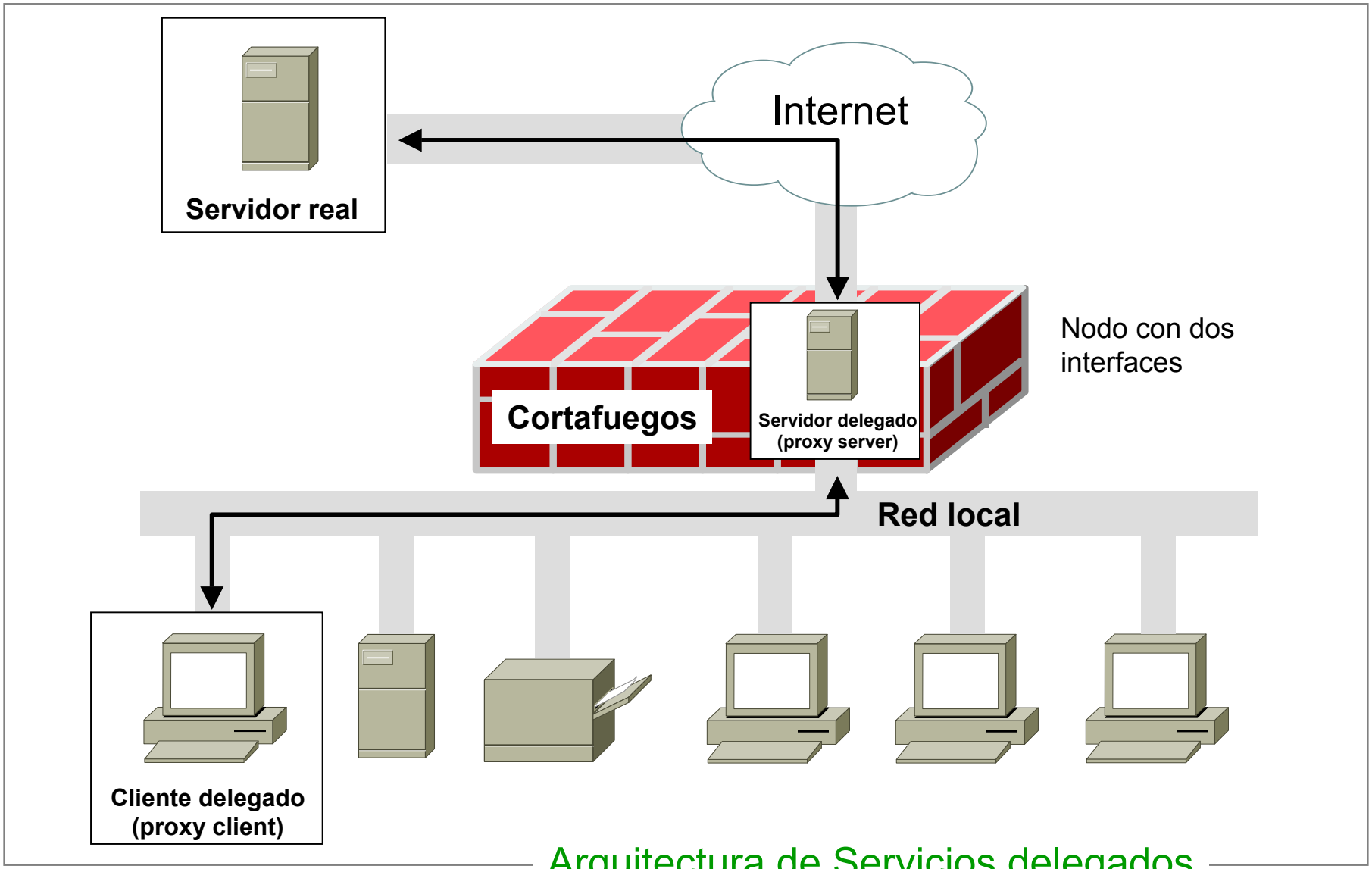
# Componentes de un cortafuegos

- Filtros
  - dispositivos que permiten bloquear selectivamente determinados paquetes
  - normalmente se trata de routers con capacidad de filtrado o computadoras con utilidades de filtrado
- Nodos bastión (*bastion host* o *gate*)
  - son computadoras altamente seguras que sirven como punto de contacto entre la red local e Internet
  - se trata de máquinas vulnerables por estar expuestas directamente a Internet
  - generalmente máquinas UNIX en las que se han extremado las medidas de seguridad (sólo se instalan los servicios absolutamente imprescindibles)

# Técnicas aplicadas a cortafuegos

- Filtrado de paquetes
  - se controla selectivamente el tráfico de la red definiendo una serie de reglas que especifican qué tipos de paquetes pueden circular en cada sentido y cuáles deben bloquearse
  - las reglas para definir qué paquetes se permiten o no se basan en las cabeceras de los paquetes
- Servicios delegados (*proxy service*)
  - son aplicaciones especializadas que funcionan en un cortafuegos (normalmente en el nodo bastión) y que hacen de intermediarios entre los servidores y los clientes reales
  - estas aplicaciones reciben las peticiones de servicios de los usuarios, las analiza y en su caso modifica, y las transmiten a los servidores reales
  - es transparente al usuario y al servidor real

# Técnicas aplicadas a cortafuegos (II)

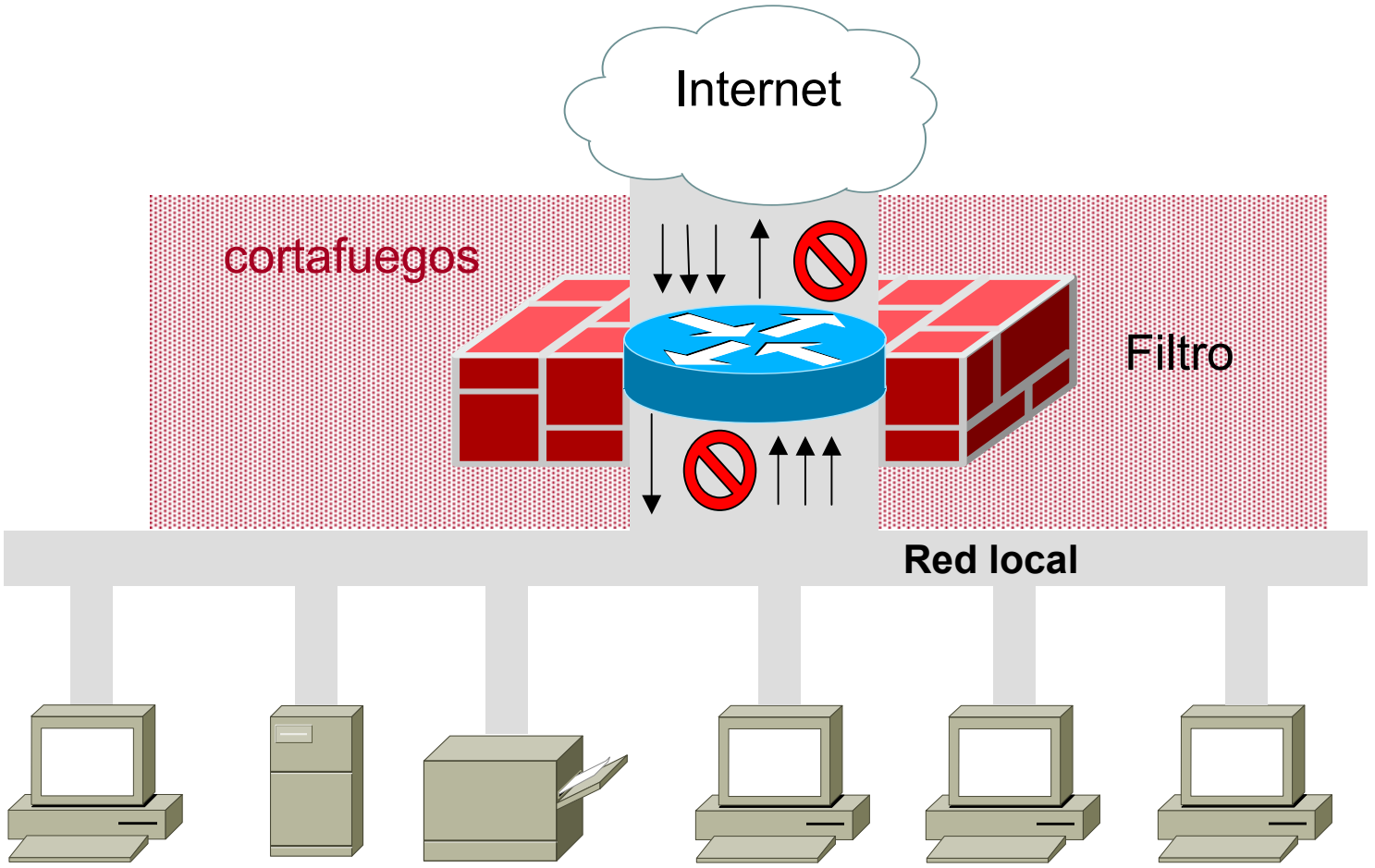



Arquitectura de Servicios delegados

# Arquitecturas de cortafuegos

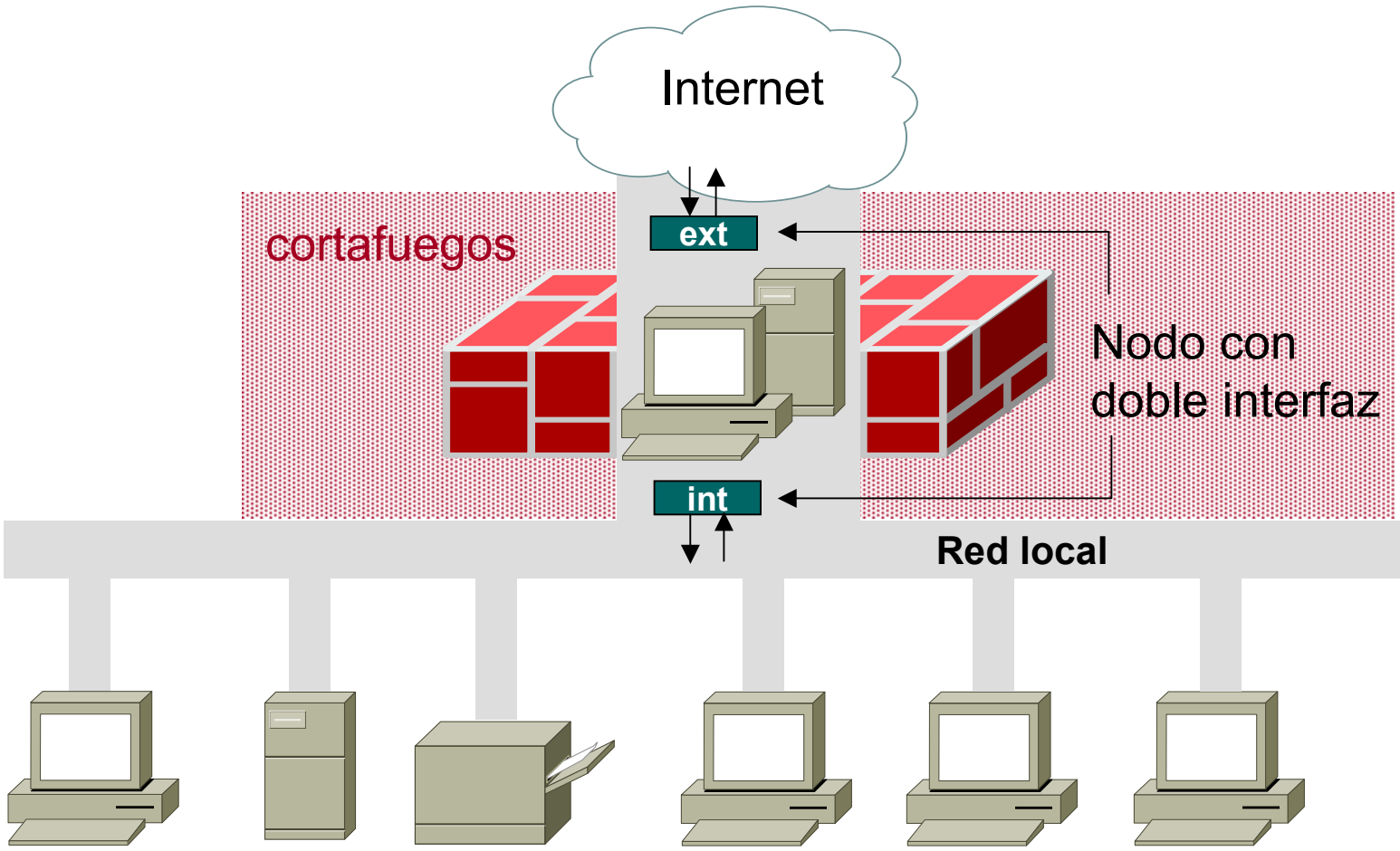
- La combinación de los dos componentes básicos, filtro y nodo bastión, y las técnicas de filtrado y delegación, permite definir múltiples arquitecturas para los cortafuegos:
  - Cortafuegos de filtrado de paquetes (*Screening Router*)
  - Cortafuegos con nodo de doble interfaz (*Dual-Homed host arquitecture*)
  - Cortafuegos con nodo pantalla
  - Cortafuegos con red pantalla (*DMZ, DeMilitarized Zone*)

# Arquitecturas de cortafuegos (II)



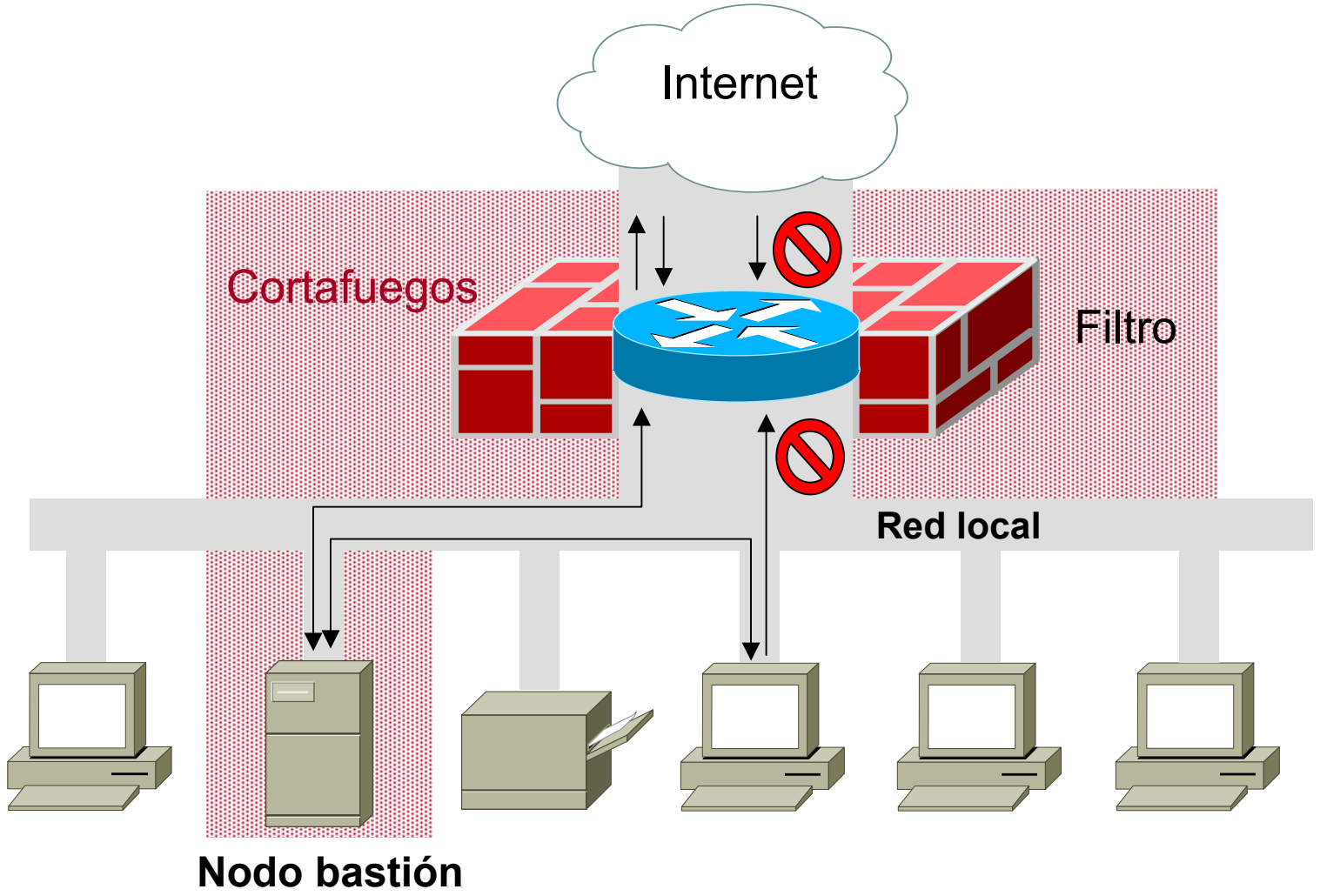

Cortafuegos con filtrado de paquetes
(*Screening Router*)

# Arquitecturas de cortafuegos (III)



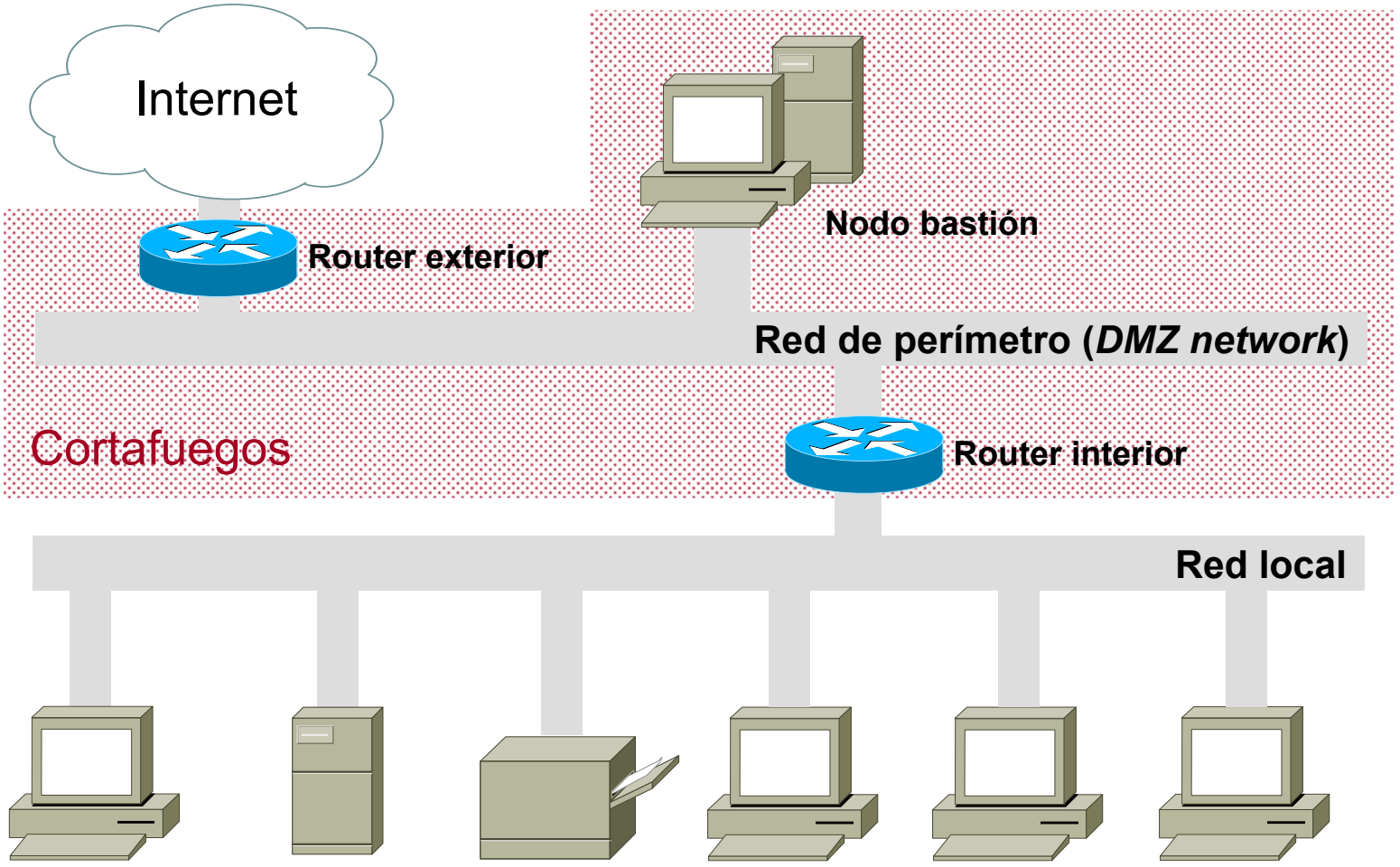

Cortafuegos con nodo con doble interfaz
(*Dual-homed host arquitecture*)

# Arquitecturas de cortafuegos (IV)




Cortafuegos con nodo pantalla

# Arquitecturas de cortafuegos (V)




Cortafuegos con red pantalla
(*DMZ, DeMilitarized Zone*)

# Clasificaciones de cortafuegos

- Clasificación según las capas OSI en que opera:
  - **Cortafuegos a nivel de red (capas 2, 3 y/o 4)**
    - Evaluación más rápida y transparente
  - **Cortafuegos a nivel de aplicación (capas 5, 6 y/o 7)**
    - Mayor capacidad de acción frente ataques complejos
- Clasificación desde el punto de vista de la industria:
  - **1ª generación** ⇨ filtrado de paquetes
  - **2ª generación** ⇨ filtrado con inspección de estado
  - **3ª generación** ⇨ filtrados a nivel de Aplicación
  - **4ª generación** ⇨ filtrado dinámico de paquetes
  - **Última generación** ⇨ híbridos

# Clasificaciones de cortafuegos

- Clasificación desde el punto de vista de la arquitectura del sistema:
  - **Servicios que se ejecutan sobre SO robustos**
    - IPTables en UNIX
    - Cisco Centri Firewall en Windows NT/2000
  - **Complejas herramientas modulares que se instalan en varias máquinas**
    - Firewall-1 de Central Point (2 módulos: inspección y gestión)
  - **Sistemas dedicados que incluyen hardware y software propietario**
    - Cisco PIX Firewall

# Cortafuegos de filtrado de paquetes

- **Analizan el tráfico a nivel de Red (capa 3)**
- **Criterios de filtrado:**
  - Dirección IP origen y destino (capa 3)
  - Protocolos usados en capa 2 y 3
  - Tipo de tráfico: ICMP, TCP, UDP... (capas 3 y 4)
  - Números de puertos origen y destino para una sesión (capa 4)
  - Interfaz físico por donde entra y sale el paquete (capa 1)
- **Varias políticas de acceso:**
  - 2 listas de reglas: una de permisos y otra de denegaciones
  - 1 lista de reglas: en ella aparece lo que se permite o lo que no
  - la acción por defecto puede ser aceptar o denegar
  - la última acción puede ser la acción por defecto

# Cortafuegos de filtrado de paquetes (II)

- **Ejemplos**
  - IPTables (UNIX)
  - **ACLs** (Cisco)

| | Dirección de origen | Puerto de origen | Dirección de destino | Puerto de destino | Acción | Descripción |
|---|---|---|---|---|---|---|
| 1 | Cualquiera | Cualquiera | 192.168.1.0 | >1023 | Aceptar | Permite que pasen los paquetes de retorno de una conexión originada en la red interna. |
| 2 | 192.168.1.1 | Cualquiera | Cualquiera | Cualquiera | Rechazar | Previene conexiones directas del cortafuegos con otro host. |
| 3 | Cualquiera | Cualquiera | 192.168.1.1 | Cualquiera | Rechazar | Previene de accesos directos desde hosts externos al cortafuegos. |
| 4 | 192.168.1.0 | Cualquiera | Cualquiera | Cualquiera | Aceptar | Permite acceso al exterior sin restricciones a los usuarios internos. |
| 5 | Cualquiera | Cualquiera | 192.168.1.2 | SMTP | Aceptar | Permite a los usuarios externos enviar e-mail. |
| 6 | Cualquiera | Cualquiera | 192.168.1.3 | HTTP | Aceptar | Permite a los usuarios externos acceder al servidor web interno. |
| 7 | Cualquiera | Cualquiera | Cualquiera | Cualquiera | Rechazar | Cualquier regla que no haya sido previamente definida es explícitamente denegada. |

Aceptar = Accept
Denegar / Rechazar = Deny o Drop
Descartar = Discard

# Cortafuegos de filtrado de paquetes (III)

- Ventajas:
  - Rapidez, transparencia, flexibilidad
  - Alto rendimiento y escalabilidad a bajo coste
  - Útiles para bloquear ataques DoS
- Inconvenientes:
  - funcionalidad limitada
  - complejidad de su configuración (por expertos) ⇨ susceptibles a error en la implementación de las reglas
  - fácilmente vulnerables mediante técnicas de spoofing
  - no previenen contra ataques que exploten vulnerabilidades de aplicaciones
  - históricos de accesos imprecisos

- Muy efectivos, como primera barrera, si se combinan con más medidas de seguridad

# Cortafuegos con inspección de estado

- También conocidos como ***Stateful Inspection Firewalls*** o ***Circuit Level Firewalls***
- Son cortafuegos de filtrado por paquetes en los que, además, a la hora de aceptar o rechazar un paquete se comprueba si es una petición de nueva conexión o pertenece a una sesión (o circuito virtual) previamente establecido entre un host externo y otro interno
- Se examina el establecimiento de la conexión (capa 4) para asegurar su legitimidad y si está permitida
- El cortafuegos mantiene una tabla con las conexiones establecidas e información sobre ellas:
  - identificador de sesión único asignado por el cortafuegos para cada conexión establecida
  - estado de la conexión: negociándose, establecida o cerrándose (capa 4)
  - número de secuencia del último paquete (capa 4)
  - dirección IP del origen y el destino (capa 3)
  - interfaces físicas de entrada y salida de los paquetes (capa 1)

# Cortafuegos con inspección de estado (II)

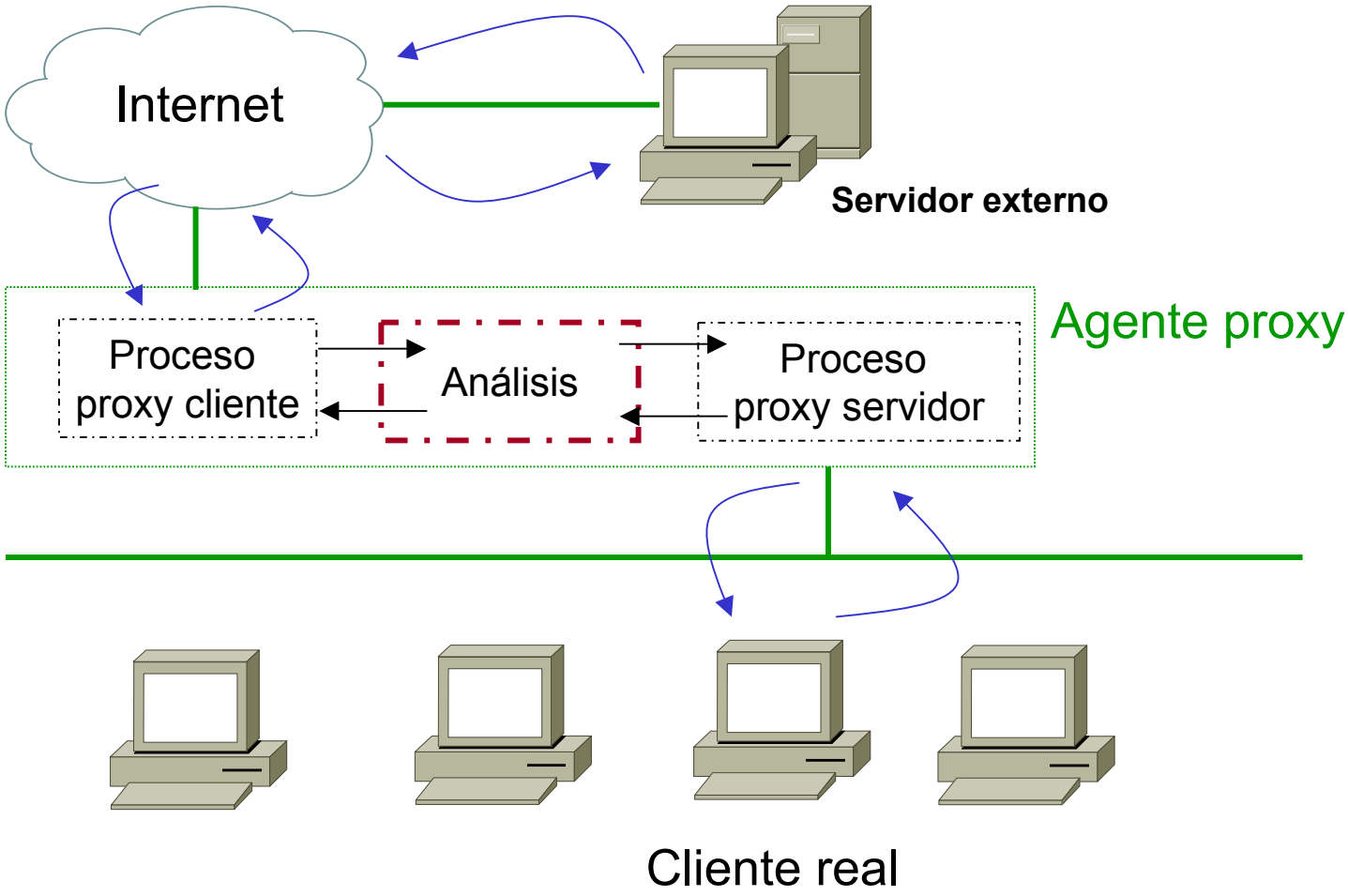
- Ventajas:
  - velocidad de filtrado
  - menor riesgo de ataques por puertos no controlados (>1023)
  - solidez de su funcionamiento para establecer políticas de seguridad
  - servicio NAT, que proporciona fuerte protección a las direcciones IP internas

- Inconvenientes:
  - limitación a la comprobación del protocolo TCP
  - imposibilidad de examinar protocolos superiores
  - limitación inherente a su funcionamiento para llevar histórico de sucesos
  - imposibilidad de implementar servicios adicionales (por ejemplo, filtrado de URL's)

# Cortafuegos de nivel de Aplicación

- Comprobación a nivel de Aplicación
- Al igual que cortafuegos de inspección de estado, llevan un control del estado de las conexiones y números de secuencia de los paquetes
- Suelen prestar servicio de autenticación de usuarios
- Prestan servicios de Proxy (DNS, Finger, FTP, HTTP, HTTPS, LDAP, NMTP, SMTP y Telnet) ⇨ impiden la comunicación directa entre red interna y externa

# Cortafuegos de nivel de Aplicación (II)



# Cortafuegos de nivel de Aplicación (III)

- Ventajas:
  - detallados registros de tráfico (ya que pueden examinar la totalidad del paquete de datos)
  - valor añadido del servicio de autenticación de cara a asegurar nuestra red
  - casi nula vulnerabilidad ante ataques de suplantación (spoofing)
  - aislamiento de la red interna
  - mayor flexibilidad de configuración (seguridad a alto nivel de los protocolos que inspeccionan y los servicios añadidos, como caché y filtro de URL's, que prácticamente todos implementan)
- Inconvenientes:
  - menores prestaciones (- velocidad de inspección)
  - necesidad de contar con servicios específicos para cada tipo distinto de tráfico e imposibilidad de ejecutar muchos otros servicios en él (puesto que escucha en los mismos puertos)
  - imposibilidad de inspeccionar protocolos como UDP, RPC y otros servicios comunes
  - vulnerables ante ataques directos al SO sobre el que se suelen ejecutar

# Cortafuegos de filtrado dinámico de paquetes

- Igual a filtrado de paquetes, pero proporcionan mecanismos de seguridad sobre UDP
- Este tipo de cortafuegos asocia el tráfico UDP con conexiones virtuales, de modo que si se genera un paquete de respuesta y envía de vuelta al solicitante original, se establece una conexión virtual y se permite al futuro paquete de respuesta atravesar el cortafuegos
- La información asociada a una conexión virtual se guarda durante un periodo de tiempo muy corto y si no se recibe dicho paquete de respuesta durante el mismo, se invalida la conexión
- Algunos modelos de este tipo de cortafuegos pueden realizan controles similares a este sobre el protocolo ICMP

# Cortafuegos híbridos

- Combinan las mejores características de dos o más de los anteriores tipos
- Ejemplo comercial muy utilizado actualmente: *CheckPoint Firewall-1*
  - Es de inspección de estado, pero intercepta los paquetes entre las capas 2 y 3, extrae información relativa al estado de la conexión y mantiene dinámicamente unas tablas con información sobre el estado de las conexiones abiertas o en trámites de ser establecidas
  - El módulo de inspección del Firewall-1 se carga dinámicamente en el núcleo (kernel) del SO de la máquina que lo aloja, inspeccionando todos los paquetes entrantes y salientes que pasan por las interfaces de red
  - Ningún paquete es procesado por las capas superiores hasta que el motor de inspección verifica que cumple con las políticas de seguridad establecidas

# Servicios adicionales de los cortafuegos

- NAT
- DHCP
- VPN
- Modeladores o reguladores de AB
- Inspección de contenidos
- Autenticación de usuarios
- Alta disponibilidad y balanceo de carga
- Integración con IDS (*Intruder Detection System*)

# Referencias

Referencias.doc