

Tema 4 → Seguridad y Privacidad en App. Web

Red Team

Actividades Ofensivas (buscar vulnerabilidades)

Kali Linux

CERT/CSIRT

Analizan ciberincidentes de seguridad y solucionan

Criticó "APT" Stuxnet (central nuclear)

Muy Alto Malware, robo

Medio Ataque Dos

Bajo Spam

↓
Ataque grandes equipos

Aircrack-ng

Wireshark

John (the Ripper)

Nmap

Mitre

Repositorio con distintos ataques

NVD(Nist)

API para ataque a fu código

Blue Team

Actividades Defensivas (defender S.O.)

Seguridad E-Mail

Objetivo: Autenticación y Confidencialidad (capa Aplicación)

PGP (uso personal)

(uso comercial) S-MIME

Integrar varios algoritmos

Igual que "PGP" pero firma

RSA] Autenticidad

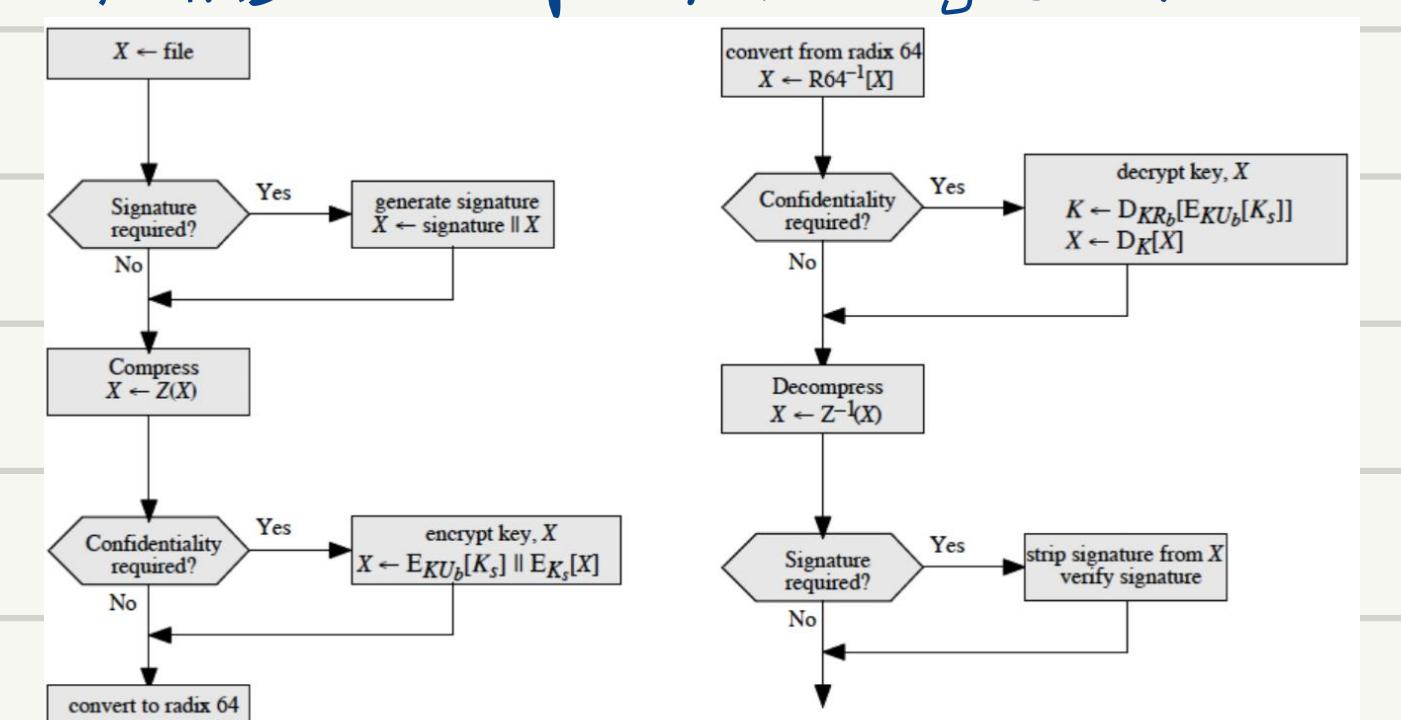
DH } Confidencialidad

DSS } Integridad

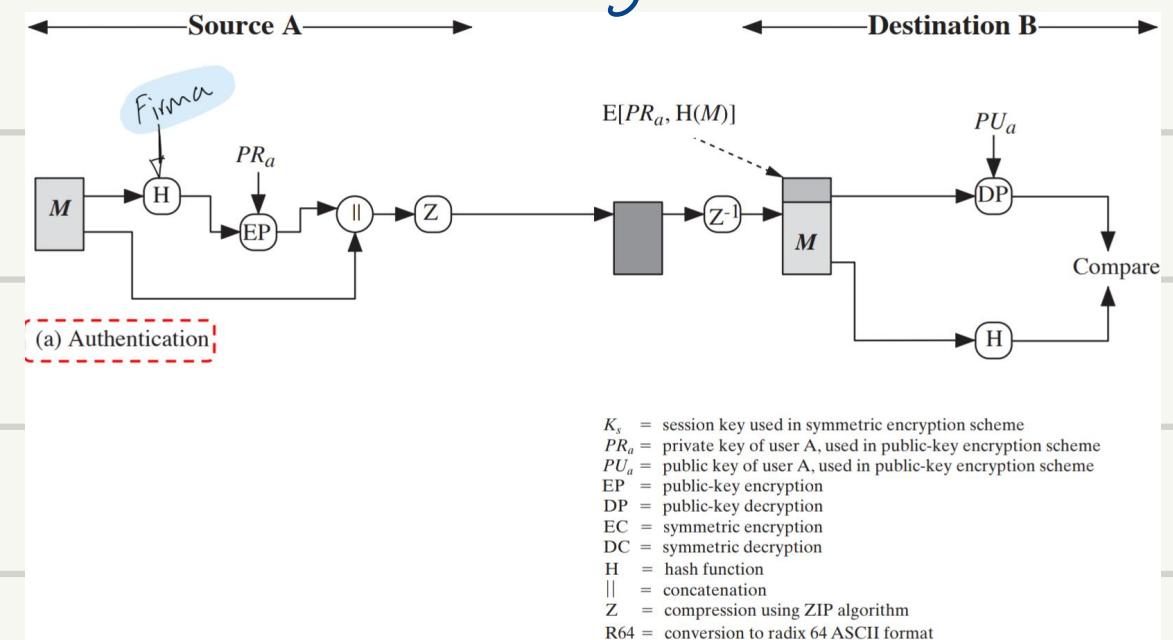
APP. E-Mail (incluye compresión)

Uso en Almacenamiento Ficheros

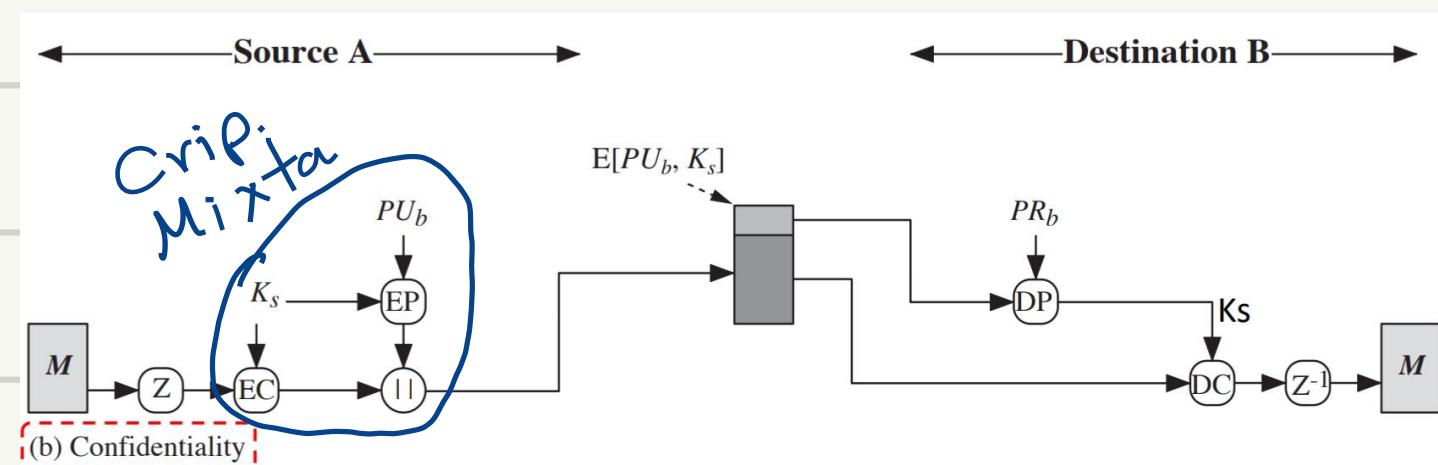
Trans. Recip. mensajes "PGP"



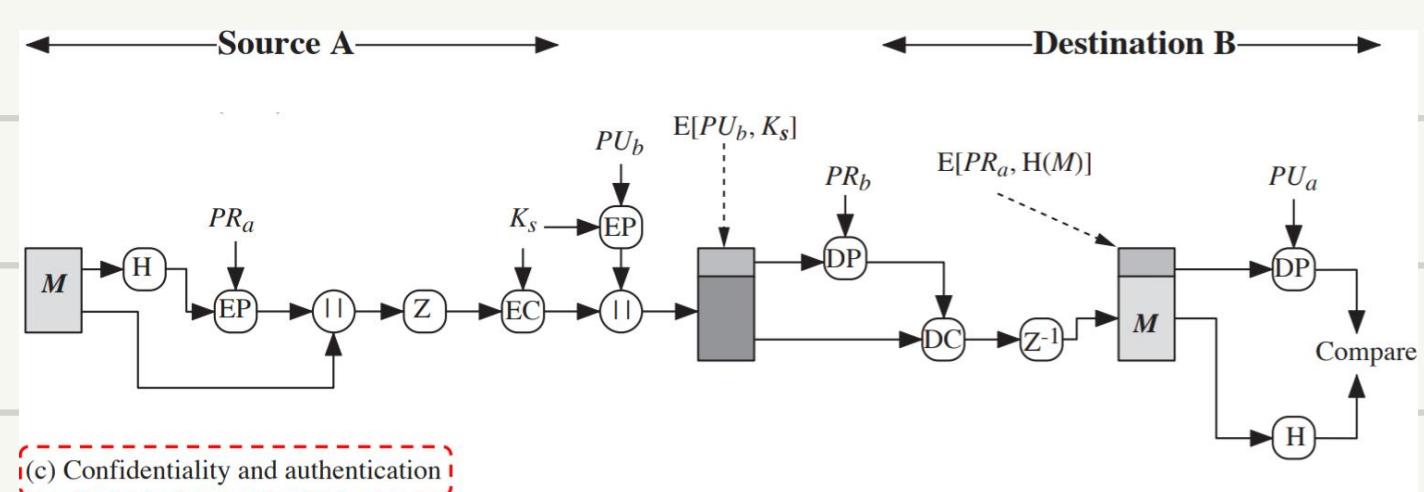
Autenticación



Confidencialidad



This is the Remix Brrr..



Conexión Remota Segura

Peligrosos: Siempre Cerrados

- Telnet 23: Control remoto por TCP
 - FTP 20-21: Transferencia de ficheros
- } Fácil de interceptar con "Wireshark"
 Información trans. en plano

Seguro: SSH - Secure Shell (v.2)

- SSH 22: Se cifra con criptografía pública.
- Algoritmos: Cifrado: AES

Integridad: MAC

Autenticación: RSA, DSA

Inter. Claves: DH

Putty: Windows

OpenSSH: Unix (Linux, Mac)

Aut Cliente
Int. claves / Negociación Modo cifrado
Conexión cliente-Servidor

Aplicación

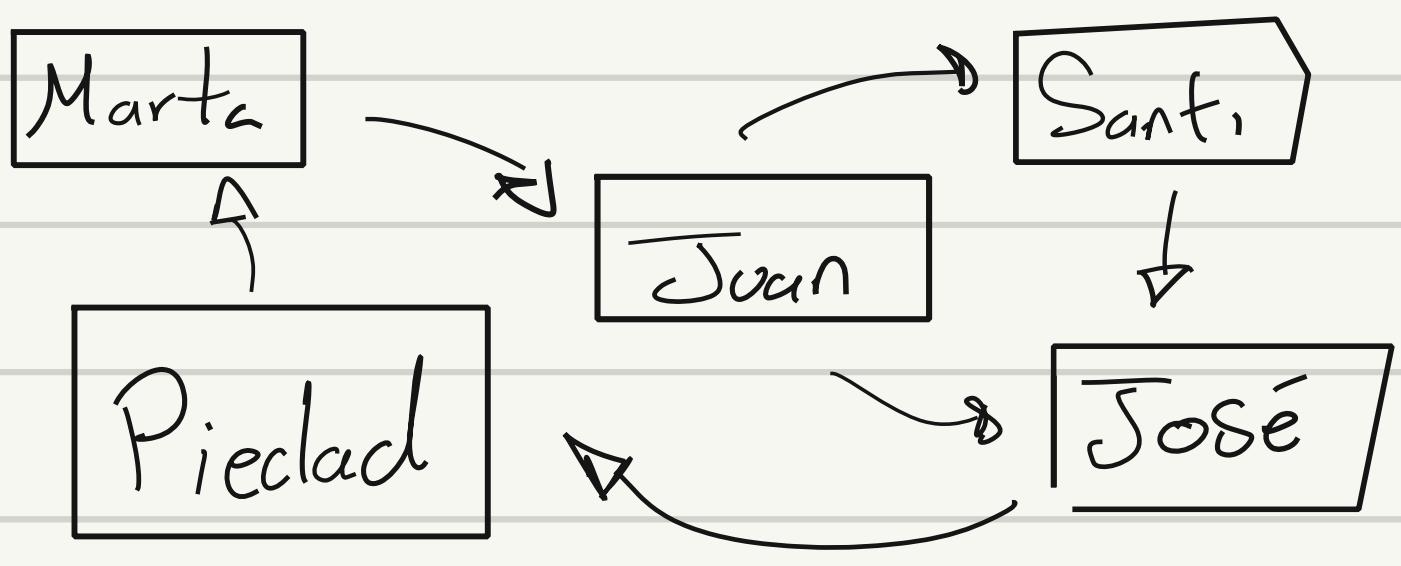
Transporte

Red → Modo túnel (P2P) cifrado

[Esteganografía: Ocultar info haciendo uso de archivos multimedia]

Modelo "PKI" en Malla

- Cada usuario emite certificados de claves públicas a otros usuarios



Flags (nivel de confianza): 1-10
 menor → mayor confianza

Se puede usar para transferir ficheros Seguros

◦ SFTP (FTP sobre SSH) → todo cifrado a nivel de red (tunel)

Modo Básico: Usuario/contraseña

Modo Avanzado: Claves Públicas SSH (transmitidas previamente)

▫ Cuando se quiere enviar datos, se vuelven a intercambiar las claves → Autenticación

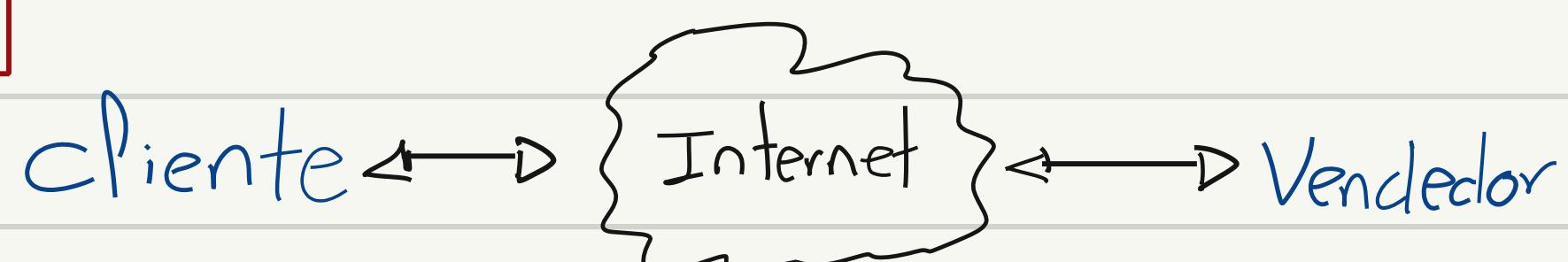
◦ FTPS (FTP sobre TLS) → Credenciales cifradas durante el proceso (tema 5)

EJ: BitLocker, Sealed

• Seguridad en Pagos Electrónicos

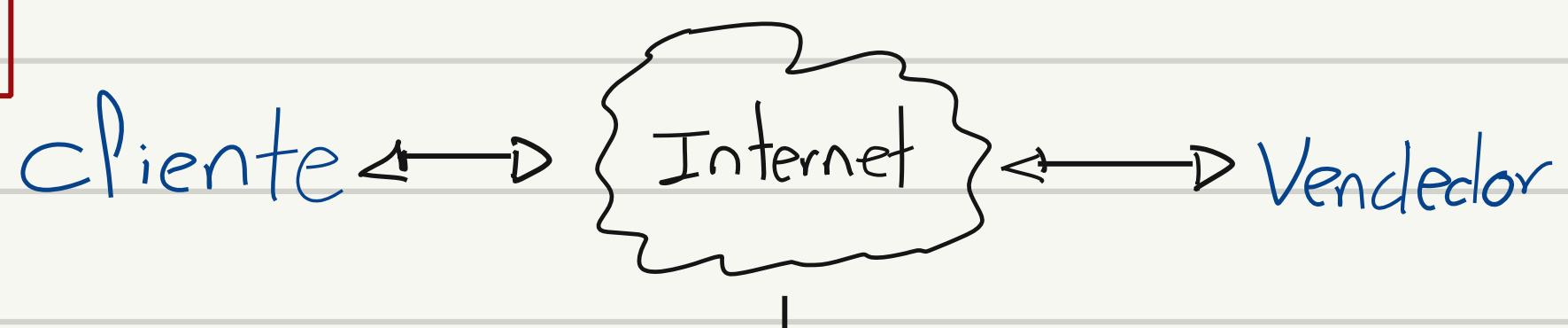
Objetivo: Misma funcionalidades que comprando en física

Antes



No intermediario → No Repudio info en plano

Ahora



Se controlan las transacciones

TTV → No-Repudio

Se cobra una comisión de uso

Protocolo "Set" Usado actual pero sobre TLS

◦ Formas de Pago

◦ Vendedor contacta con banco.

Online: Pago en el teléfono con tarjeta al contacto

Offline: Recargo tarjeta después de comprar (Amazon)

↳ Se deja un tiempo que la entidad financiera corrobore.

◦ Comprador procede con transacción

Sistema Pre-Pago: tarjetas prepago

Sistema Pago Instantáneo: tarjetas débito

Sistema Post-Pago: tarjetas crédito

↳ cierto tiempo después se decrementa la cantidad

◦ Cantidad de Dinero

Micropagos (- - 6 euros, España)

Coste elevado de implementación

Vendedor paga demasiada comisión por uso del gateway

Solución: Broker

Cupones "Script"

Comprar bonos (ej: 10 euros) y canjearlo en una cuenta para micropagos en distintas tiendas web.

Pagos: (6-100 euros)

Macropagos: (100- - euros)

◦ Primeros Protocolos → SSL

Uso de Cript. híbrida (RSA, DH, DES, RC4)

Autent. e integridad (MAC)

Problemas

- Vendedor recibe info del cliente en plano (tarjetas)
- No gateway → No bancos → No verificación de pagos



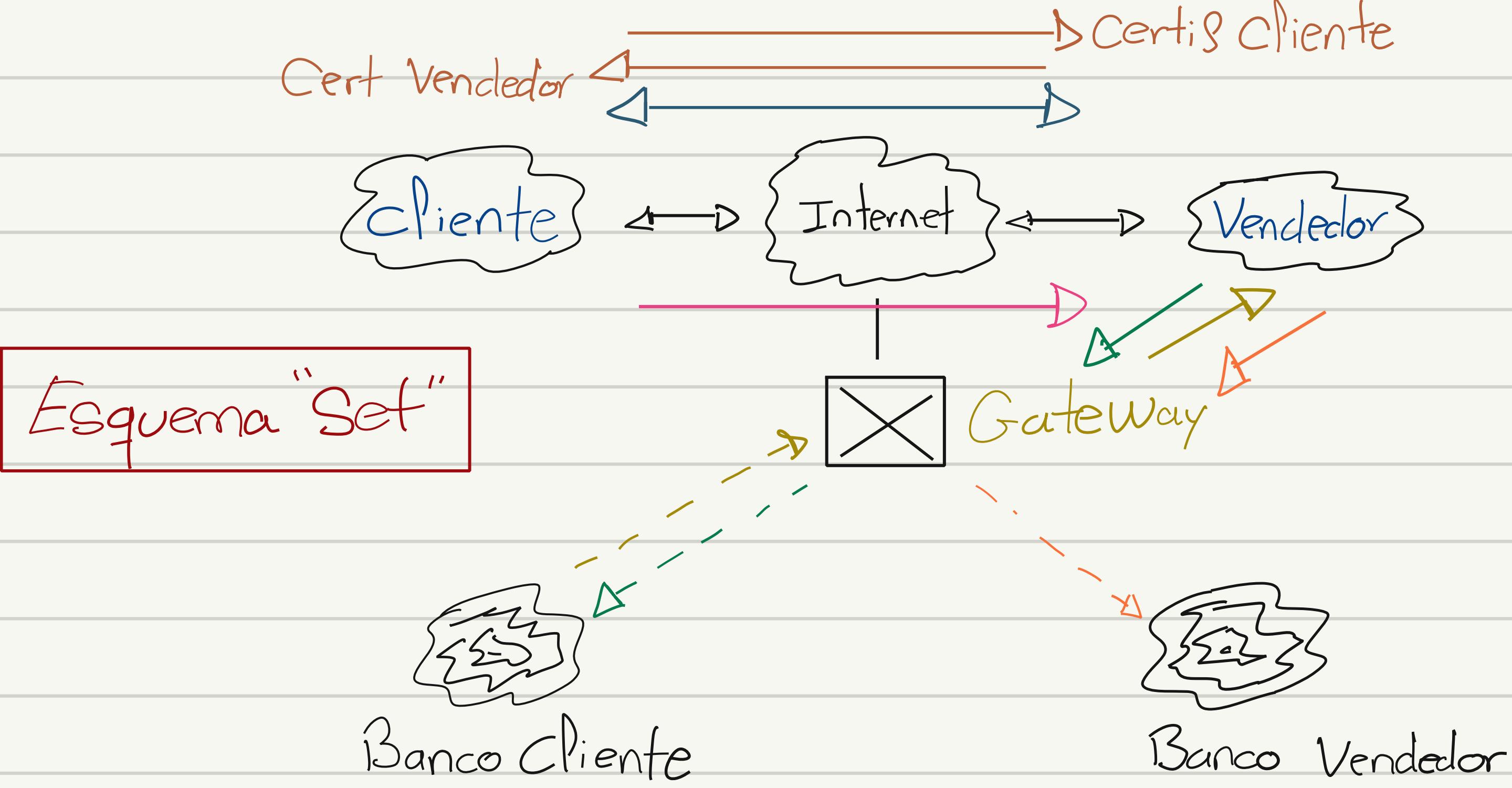
No recibos → No No-Reputio



Estafas

Surge Protocolo Set (conjunto protocolos seguridad)

Confidencialidad, Autenticación (cert. X.509), Privacidad (firma dual),
Integridad y no-reputio (firmas digitales), autorización.



1º Petición Producto

2º Envío de certificados (autentificación)

3º Pedido y Pago

4º Autorización del vendedor a sacarle dinero al banco cliente

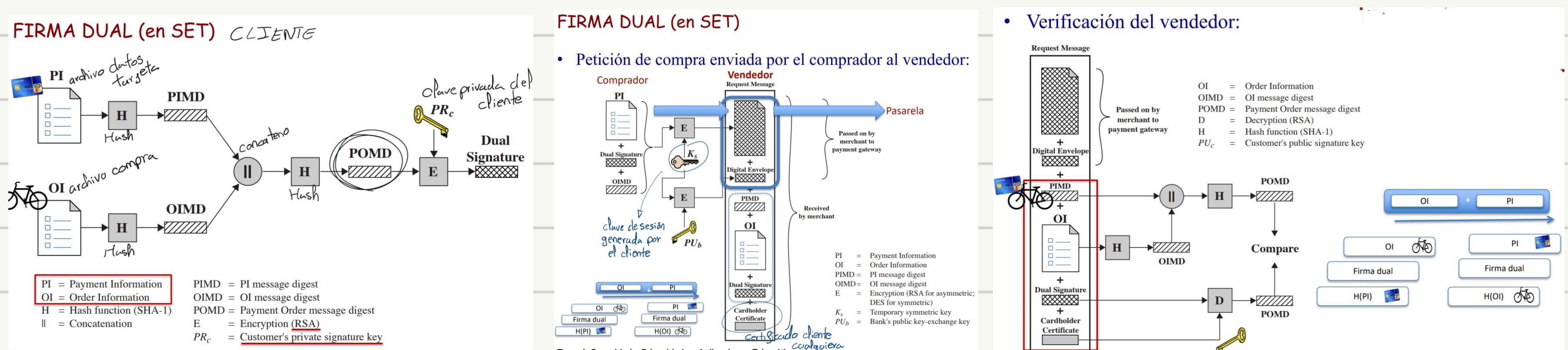
5º Aprobación

6º Vendedor reclama cantidad a Gateway

• Firma Dual → Privacidad

▪ Ni el comerciante ni vendedor necesitan conocer los datos de la tarjeta PI

▪ Ni el banco necesita saber lo que compró OI



Confidencialidad, autenticación, integridad, no-repudio, privacidad

Desventajas: Depende del protocolo → Si se descubre falla, inseguro

Dificultad con certificados

No para micropagos → Costosos

• Privacidad de los usuarios en aplicaciones

Privacidad: Mantener la integridad de la persona
 Confidencialidad: Mantener datos en secreto

Proteger la Privacidad =

- Leyes con Sanciones GDPR
- Algoritmos Criptográficos
- Atacante no puede relacionar 2 mensajes

Propiedades Privacidad =

- No puede rastrear el mensaje

Anonimato No Rastreable

Pseudónimos
"mote"

Privacidad
= Anonimato

Anonimato Rastreable
2 personas se conocen pero uno puede desvelar al otro

Anonimato No Rastreable y No Vinculante

Firma Digital

• Firma Digital

→ Firma verificada más tarde

▫ Firma a Ciega: Anonimato Rastreable

EJ: Voto Electrónico

"Mensaje M de Bob sea firmado por Alice sin saber que pone en M' "

e = clave pública
 d = "privada"
 r = $n \leq$ Aleatorio

1. Bob: r (un nonce)
2. Bob: $M' = M \cdot r^e \pmod n$
3. Bob → Alice: M'
4. Alice → Bob: $(M')^d \pmod n = [M^d \cdot (r^e)^d \pmod n] = [M^d \cdot r \pmod n]$
5. Bob: $M^d \cdot r^{-1} \pmod n = [M^d \pmod n]$

▫ Firma de Grupo: Anonimato Rastreable

EJ: Bancos

"Admin. del grupo emite clave de firma de grupo → todos firman con ella, en nombre de todos."

▫ Admin. puede verificar quién firmó (nadie más sabe quién firmó → Anonimato)

z = clave secreta admin.

Firma: $S = E_{x_i}(M)$

↳ Claves privadas de cada miembro

→ Si la firma es correcta o no

Verificación: $E_y(S) = M$

Clave pública grupo

▫ Firma Anillo: No vinculable ni Rastreable

EJ: Envío de un cheque firmado por cliente y trabajadores de la entidad bancaria.

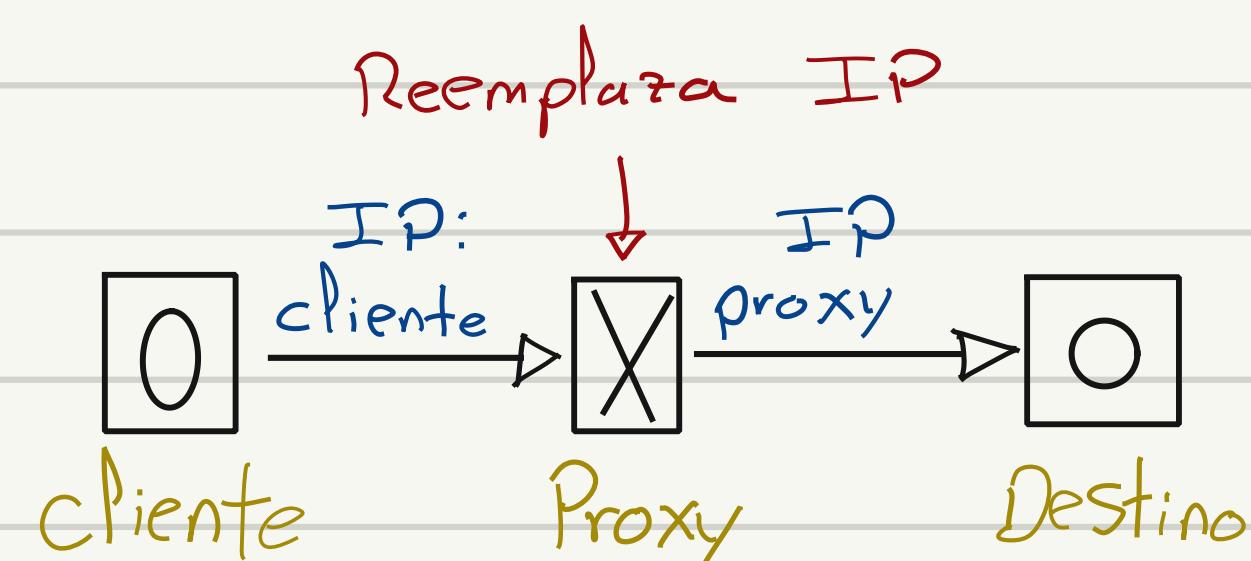
"Computa una firma con su clave privada y el resto de claves públicas".

◦ Protocolos Criptográficos y de Enrutado

▫ Basado en el uso de Proxy

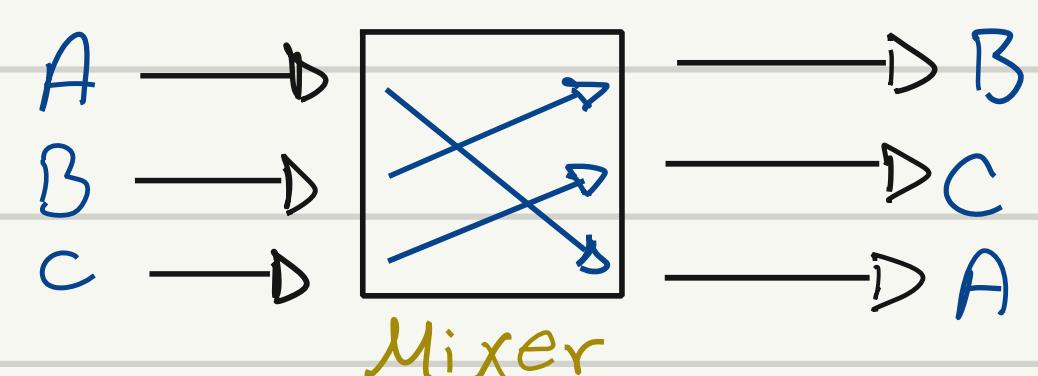
Servidor proxy hace de intermediario

- de llega un paquete con una IP y reemplaza por su propia IP antes de reenviar.



▫ Basado en uso de Mixers

Tipo de proxy → de llega un mensaje y lo almacena un tiempo. Al tiempo lo envía por un puerto diferente



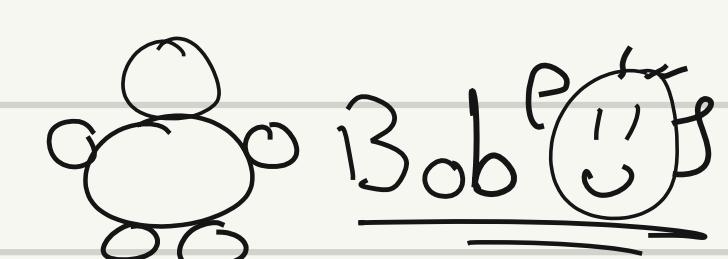
▫ Basado en el conocimiento parcial de la ruta

Paquete cifrado en varias capas → Se "pelean" rutas llegar al mensaje cada vez que pasa por un router.

Vulnerabilidad → El último router deja el mensaje en

Surge TOR (v2) plano.

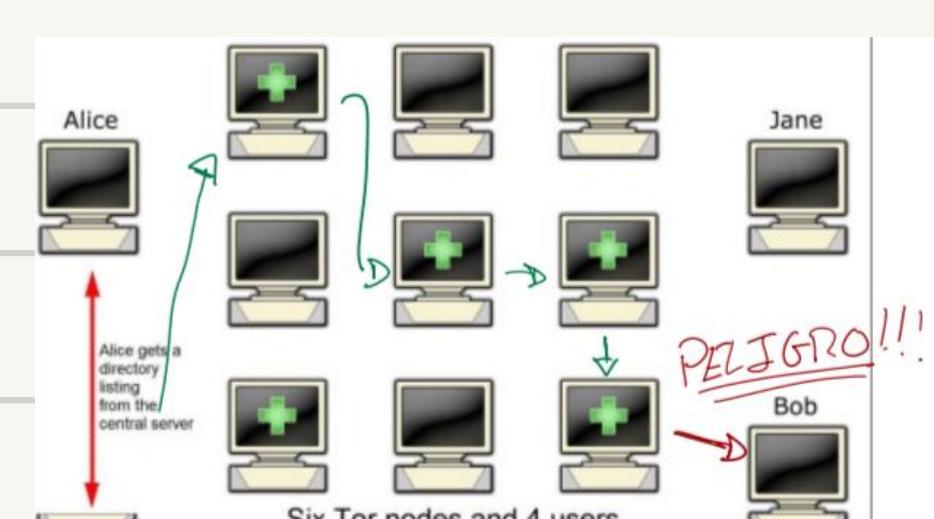
M Interceptable



Red Privada: Alice quiere enviar a Bob

- Alice habla con Proxy → establece ruta de encaminamiento
 - ↳ Cada 10 minutos cambia

↳ Paquete dentro de la red: Seguro (encriptado)
Fuera " " " : Peligro (texto plano)



▫ Basada en la creación de grupos

Crowds: Muchos emisores → Cada uno solo conoce el nodo destino y de donde viene el paquete.

Hordes: Igual al anterior, pero el reenvío hace broadcast a los vecinos para enviar.

Más rápido, muy costoso (Sobrecarga sistema)

Tema 5

Seguridad en la Capa de Transporte

Surge WTS (Web Transaction Security)

SHTTP → Capa Aplicación

SSL → Capa Transporte

↳ Trama HTTP pasa por

un filtro de SSL antes de salir y enviarlo por un puerto TCP



Proporciona: Autenticación, Confidencialidad, Integridad.

HTTPS: 443 "HTTP sobre SSL/TLS"

Existen 2 fases en SSL: Sesión SSL y Conexión SSL

Establecer una conexión segura

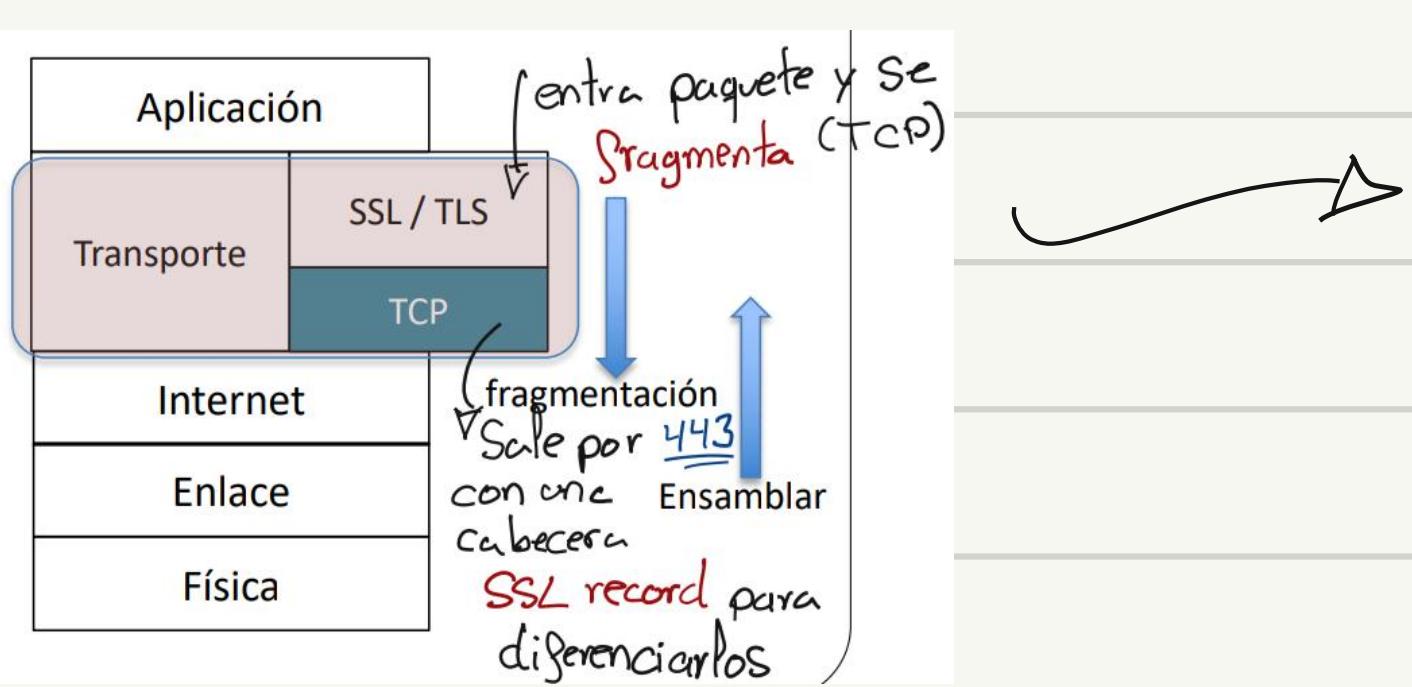
Transmitir por esa conexión

Todos esos paquetes se fragmentan

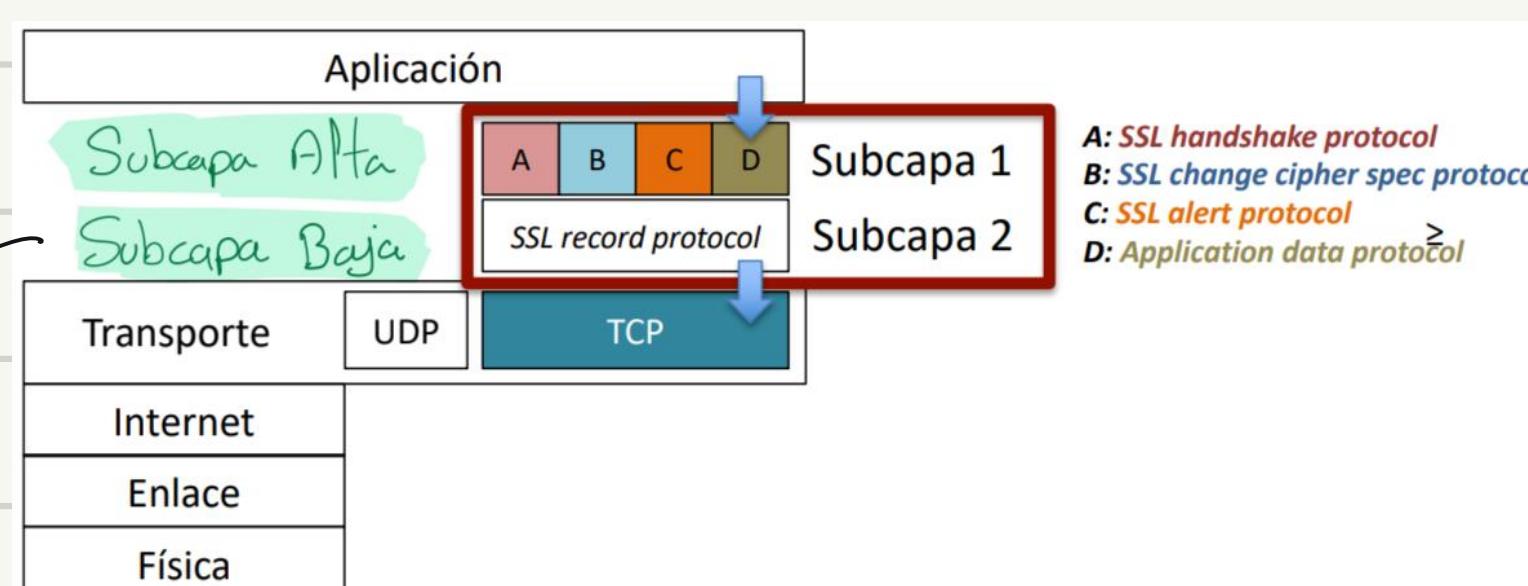
Dado que luego se envían por TCP

Fragmentación

Ensamblado



Para llevar a cabo las 2 fases, se divide en 2 Subcapas



Fragmenta los datos de la capa de aplicación y los procesa de forma individual

La subcapa alta contiene:

- (22) **SSL Handshake Protocol**: permite que los puntos de comunicación:

- Se autentican ambas partes, sin ser obligatorio.
- Se negocia la Suite de cifrado

- se autentiquen mutuamente, y que, además, se negocien un **cipher suite** (opcionalmente) y un método de compresión

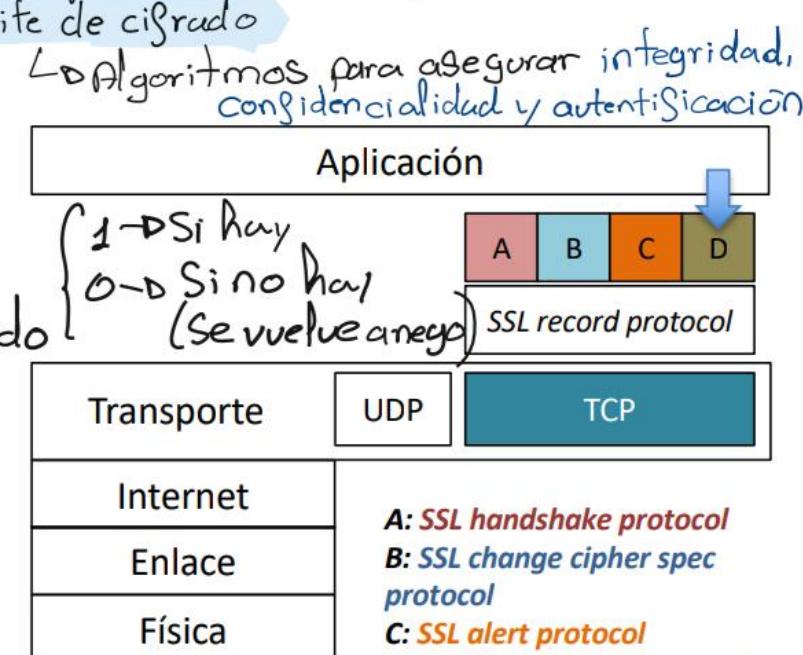
- (20) **SSL Change Cipher Spec Protocol**:

permite a los puntos de comunicación activar el cipher suite **Negociar algoritmos de cifrado**

- (21) **SSL Alert Protocol**: permite a los puntos de comunicación indicar posibles problemas potenciales e intercambiar los correspondientes mensajes de alerta

- (23) **SSL Application Data Protocol**: es el propio protocolo de la capa de aplicación (ej: HTTP) y alimenta al SSL Record Protocol

- Como un gateway, lo pasa todo de la capa de aplicación a la capa de red.



A: SSL handshake protocol
B: SSL change cipher spec protocol
C: SSL alert protocol
D: Application data protocol

Hebra (hilera) que analiza el estado de los paquetes (malformaciones).

Si ServerKeyExchange

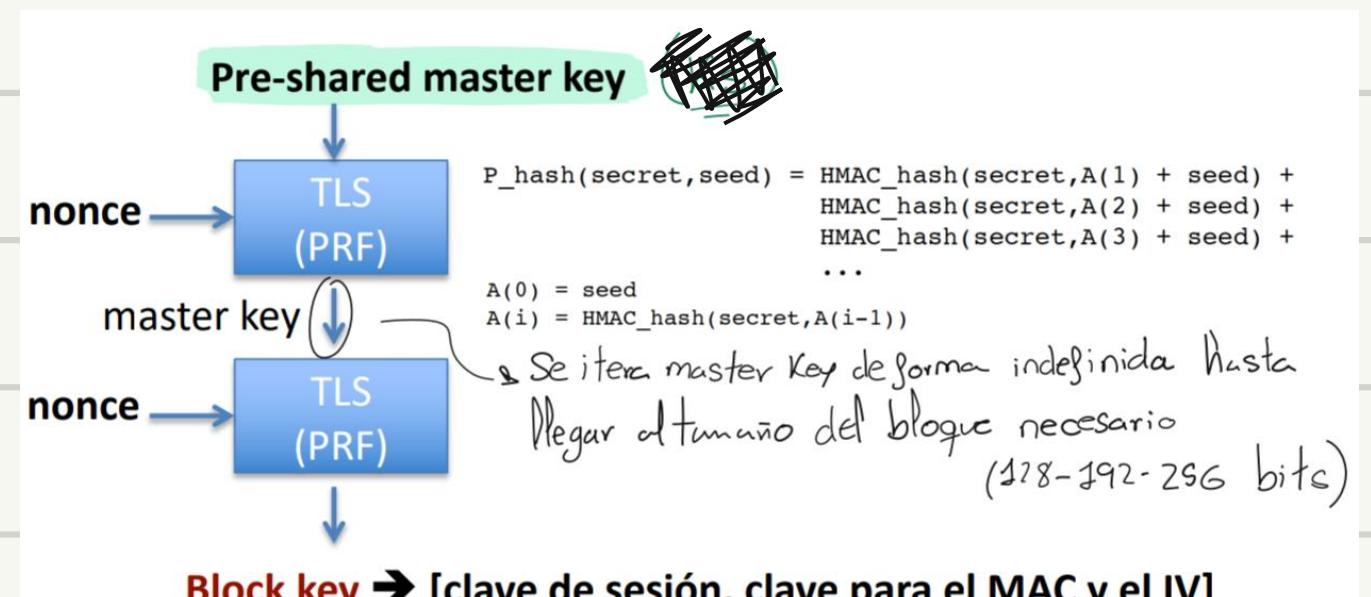
[$K_s = \text{Clave Sesión}$]

Hay → Se produce con los nonces y las claves públicas y se envía con las claves públicas cifradas.

No Hay → Se genera K_s con los nonces del cliente y de servidor.

• Intercambio de Claves

Necesario: Clave Sesión K_s / Clave MAC / Vector IV



Necesitamos: Se generan con una Salt

- Una Semilla "pre-shared master Key"
- Nonce (valor aleatorio)
- Función Pseudorandom "PRF"

▷ También se pueden generar las claves por "DHE" (Diffie Hellman)

esímero)

▫ "x" y "q" no son constantes

↳ Evitamos **Man in the Middle**

Garantizamos "PFS" (Perfect Forward Secrecy)
Claves no comprometidas incluso si ya las hay

• TLS 1.2 y TLS 1.3

TLS 1.2: ▷ Master Key con **SHA-256** (no "PRF")

▷ Uso de extensiones (opcional → obligatorio en TLS 1.3)

▷ Elimina DES y IDEA → Añade **AES**

SSL - Secure Sockets Layer

Servidor se autentica siempre? YO

" " está autenticando?

Buscar trama con un certificado

Siempre se genera

ClientHello nonceservidor

ServerHelloDone nonceservidor

[Certificate] [ServerKeyExchange] [Kpub], [Kpub]

[CertificateRequest]

ClientKeyExchange Exp(Yc)

Exp(Ys)

[CertificateVerify]

ChangeCipherSpec [0-1] valor

Finished

ChangeCipherSpec

Finished

changeCipherSpec a 1 en ambos

para empezar a cifrar

Si el cliente envía el certificado, entonces necesita enviar un certificado firmado para la verificación de la entidad origen

Fase 4: Se establece el suite de cifrado y se termina el proceso de handshake

Primera trama que se cifra → "Finished" → Ya están todas de acuerdo con el Suite de cifrado

https://www.

SSL

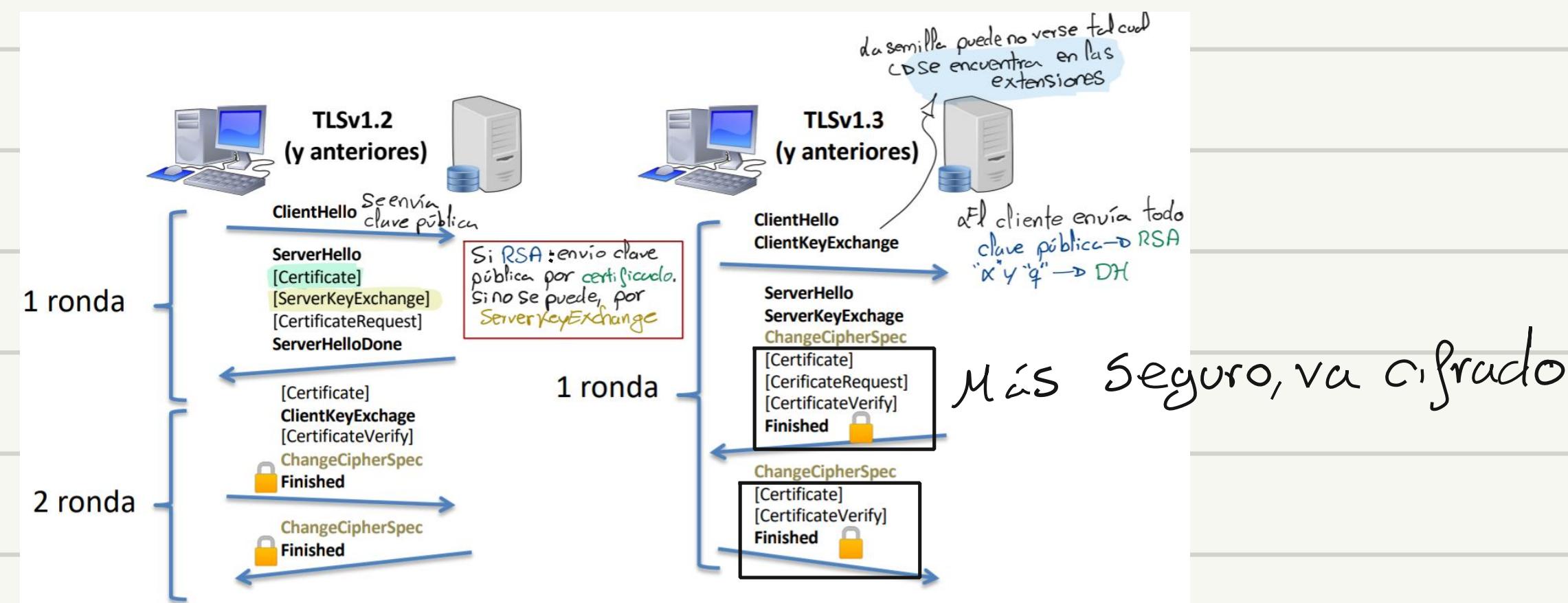
TLS

SSLv3

TLSv1.3

TLS 1.3

Más tardío (+ milisegundos) → Eficiencia ahorrando tramas



Nota

Existe una versión que funciona igual que TLS, pero para paquetes que usen el protocolo de transporte UDP

↳ Se le conoce como DTLS

◦ Seguridad en la Capa de Internet

IPSec: Especificaciones y Funciones Pidades de la capa de red para el modelo TCP/IP.

EJ: Acceso Remoto vía Internet / Apps. Comercio Electrónico

Se pretende: Autenticidad, Integridad, Confidencialidad

No proporciona Servicio de no-Reputación (ni de ataques DoS)

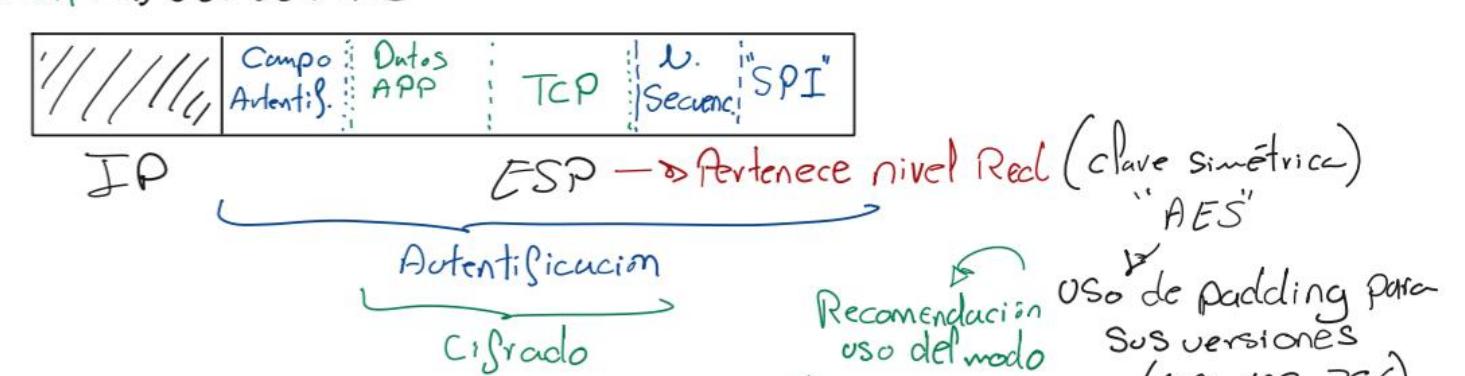
Si de repetición

◦ Protocols de IPSec

"IANA": los valores asignados a IPSec ayudan a mantener la coherencia y compatibilidad entre diferentes implementaciones de IPSec en distintos sistemas y dispositivos.

• Carga de Seguridad encapsulada "ESP"

- Cifra todo el campo de datos → Confidencialidad
↳ Uso de "Diffie-Hellman" (intercambio de claves públicas).
- Integridad y autenticación → Uso de MAC



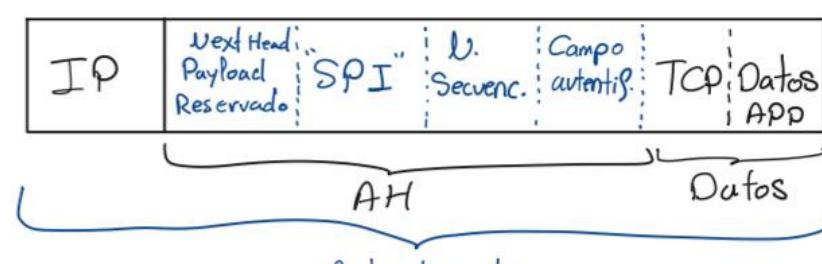
• Cabecera de autenticación "AH"

- Integridad y autenticación → Uso huella digital "HMAC"

↳ Se calcula una función Hash al contenido del paquete IP

- NO Confidencialidad → No cifra los datos del paquete IP. ↳ info vista por terceros

↳ Forma de evitarlo → Uso de HTTPS o FTPES con TLS



Pasos del "AH"

1º Emisor calcula la función Hash a partir del mensaje a transmitir.

2º Se transmiten los datos por Internet

3º Paquete llega al receptor, aplica la función Hash y compara con la clave secreta que ya tenía.

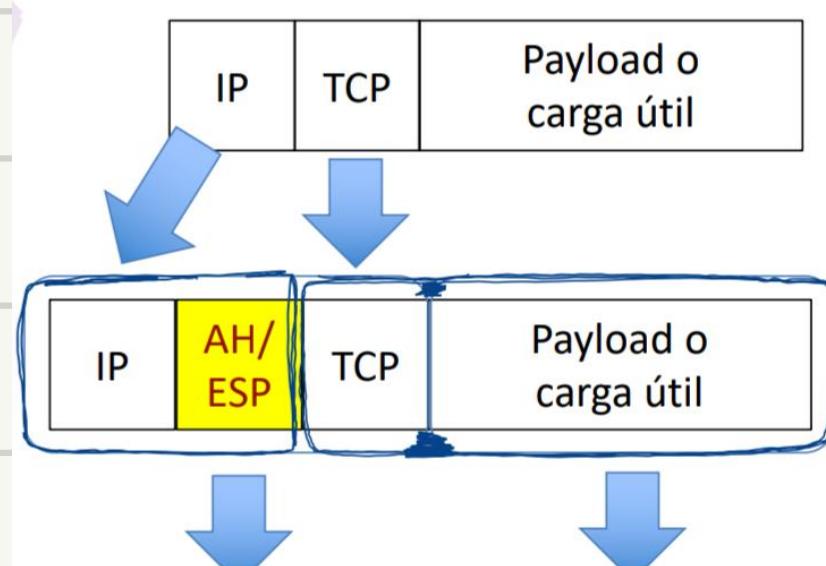
- IKE
 - ↳ Automatiza la generación y administración de claves
 - ↳ Negociación de claves
 - Se acuerda los tipos de cifrado y algoritmos de autenticación.

Modos de IPsec

Modo Transporte

entorno cerrado (intranet) · 2 dispositivos que no sean túneles

SOLO protege carga datagrama



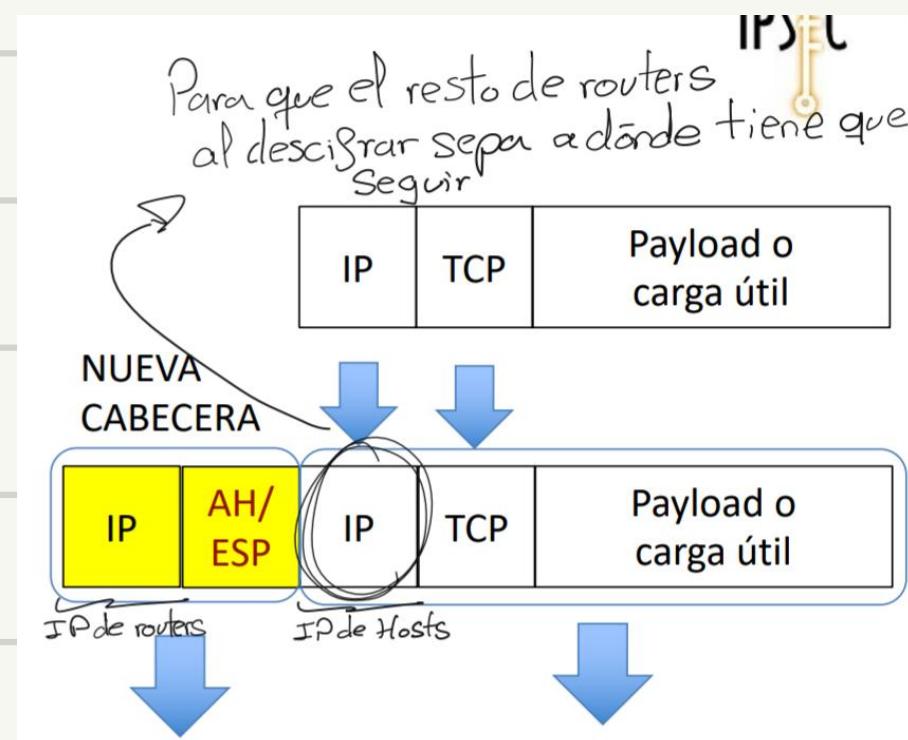
Con este modo, es visible la
• IP origen - host
• IP destino - host
Importante para saber a dónde voy

En el payload se puede aplicar:
• Cifrado
• Autenticación
• Integridad

Modo Túnel

entre routers: Salgo de la LAN

encapsula todo datagrama IP



Con este modo, es solo visible la
• IP origen - router
• IP destino - router

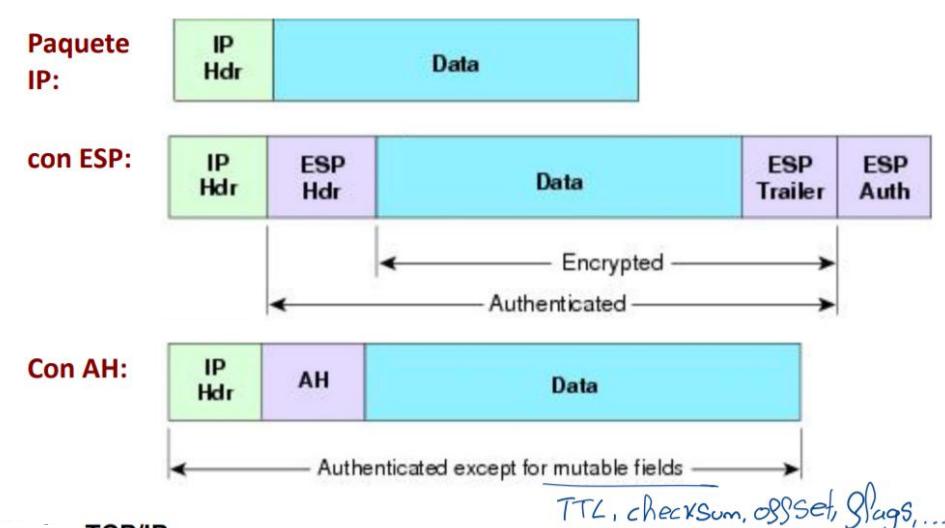
En el datagrama se puede aplicar:
• Cifrado
• Autenticación
• Integridad

- ESP:

- se cifra el payload y
- opcionalmente lo autentica, pero no la cabecera IP

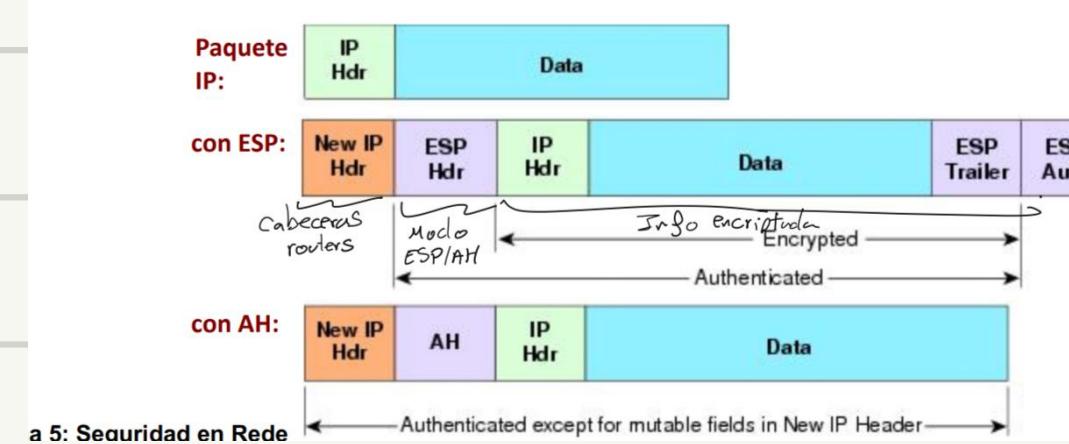
- AH:

- se autentica el payload y algunas porciones de la cabecera IP



Se puede aplicar:

- **ESP:**
 - se cifra y
 - opcionalmente autentica todo el paquete IP original (paquete interno), incluyendo la cabecera de ese paquete original
- **AH:**
 - se autentica todo el paquete original y algunas partes de la nueva cabecera externa



Protocolo "IKE"

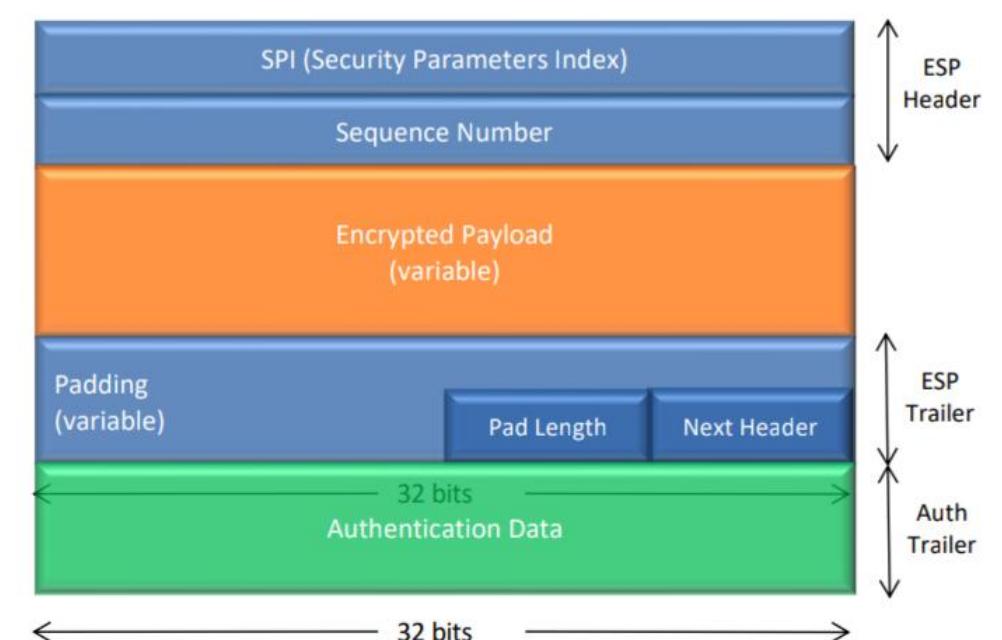
▷ Autenticación de las partes.

↳ Certificados X.509

▷ Establecimiento clave secreta
↳ "DH"

Encapsulating Security Payload – ESP

- cifrado + autenticación de dato+integridad:



Uso de ISAKMP

2 fases

Autentica Cada Parte.

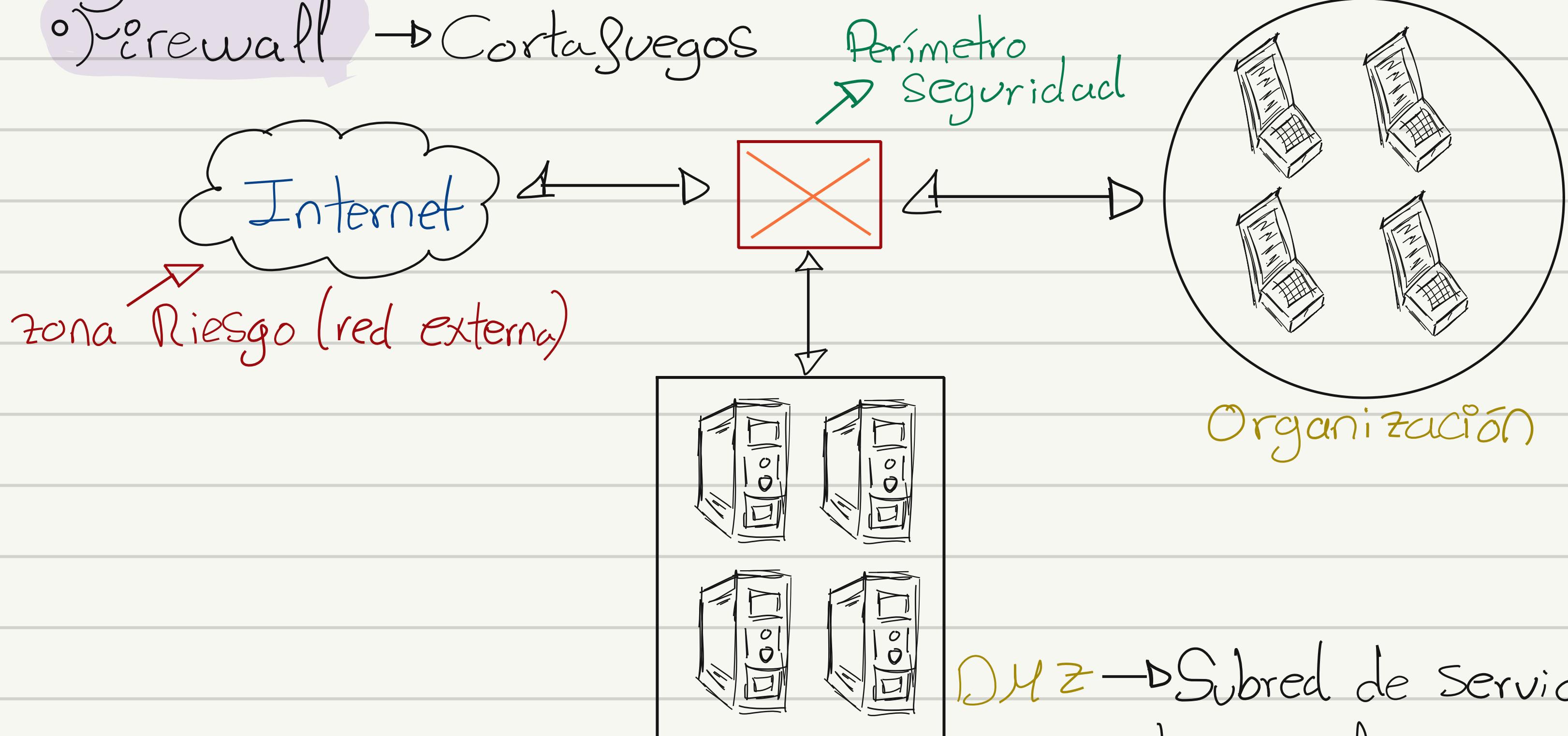
Modo Agresivo: Usa mitad de los mensajes.

No confidencialidad: ID cliente en claro

Modo Principal: Todos (Punto a punto y seguro)

Negocia y establece SAs de IPsec:

o) Firewall → Cortafuegos



DMZ → Subred de servidores
protege cualquier acceso a
los host del sistema.

1^{era} Generación

Revisa: IP emisor/receptor
Protocolo, puerto, DNS

2^{da} Generación

Revisa: IP emisor/receptor
Protocolo, puerto, DNS
Flags: SYN, ACK, ...

Generaciones

4^{ta} Generación (todo lo anterior)

+

- Control Móvil
- Validación APPs y eliminación
- Monitorización usuario y APPs

3^{ra} Generación

Revisa: IP emisor/receptor
Protocolo, puerto, DNS
Flags: SYN, ACK, ...
DPI: Uso firewall con
VPN, TLS, ...

o) Comandos

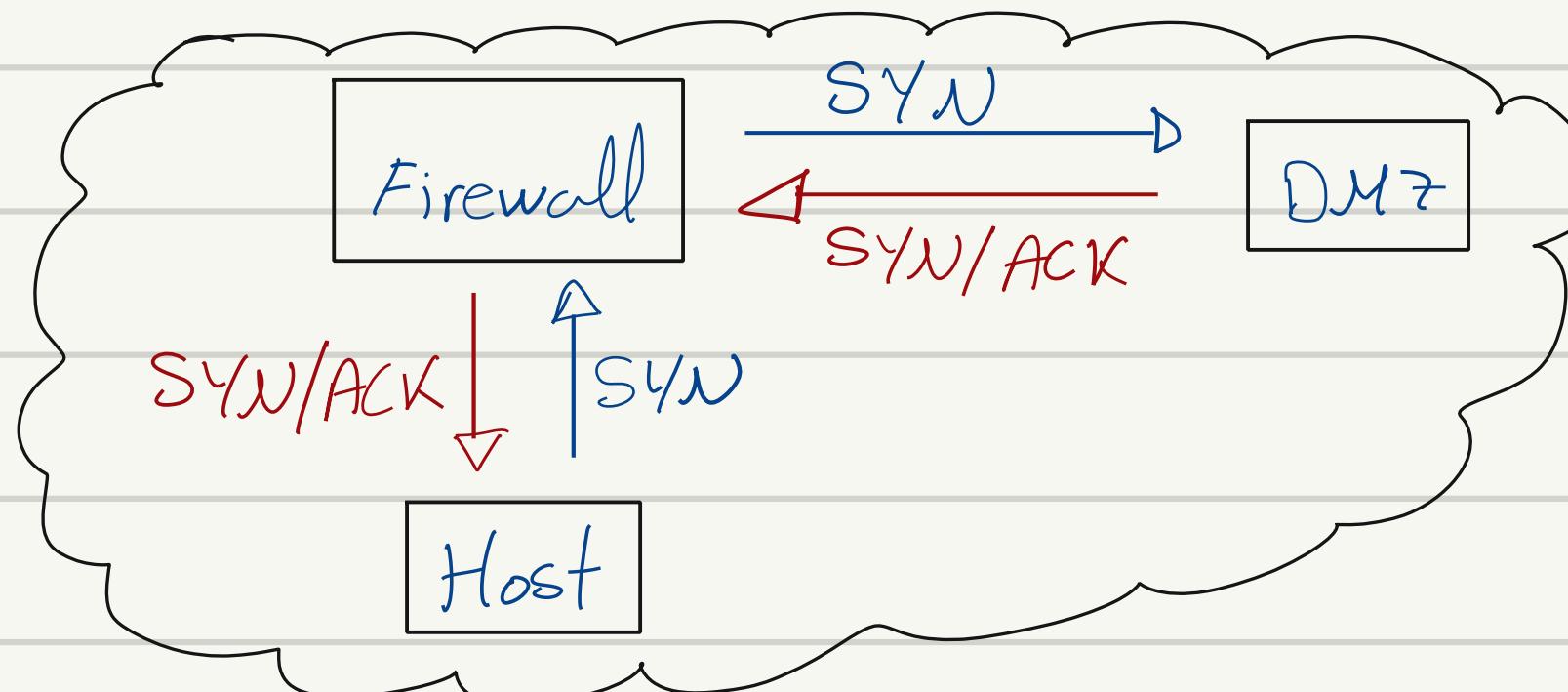
Input/Output: tráfico entrante / Saliente

Forward: Permitir (Accept) o denegar (drop) cierto tráfico por un puerto específico desde el exterior a una red interna o entre 2 LANs

Prerouting / Postrouting: Para tomar decisiones sobre un paquete antes de su enrutamiento EJ: redirigir un paquete de un puerto a otro antes de realizar enrutamiento.

El **postrouting** una vez tomada la decisión de enrutamiento, se usa para tomar acciones sobre los paquetes antes de salir del Sistema. EJ. **MASQUERADE** (traducir dirección NAT en conex. salientes).

Los **flags** (NEW, ESTABLISHED, RELATED) son importantes para comunicar la intención de establecer una conexión con otra máquina.



• **IDS** → Sistema detector de anomalías.

Pasivo "IDS": Detecta anomalía → Manda señal a una estación de un intruso. → **NO ACTUA**

Activo: "IPS": Detecta anomalía → Actua → **Elimina**

► Tipos

NIDS: Buscan evidencias en paquetes

DPI: "Patrones" = Repeticiones (DoS, cambios variables, uso excesivo CPU, ...)

Desventajas: tráfico cifrado = nada que hacer
Falsos positivos

HIDS: Supervisa Hosts (abuso de GPU, memoria, ...)

► Técnicas de detección

- **Firma o abusos**: Patrones predefinidos (baja tasa falso positivos)
- **Anomalías**: "Machine learning" (alta tasa falso positivos)
- **Híbridas**: Ambas

o Redes Inalámbricas

transm. Inalámbricas

Copia/modif mensajes

Inserción y **Dos (DoS)**

Ocultación / cifrado

Autenticación, nonce

Medidas Seguridad

Puntos Acceso(AP)

Acceso No Autorizado



Autenticación
(Usuario/contraseña)

Elem. Interconexión

· Cifrado

· desactivar broadcast

↳ Solo dejar ciertos equipos (por su MAC)

· Cambiar password

o Protocolos

Surge **WEP** (algoritmo RC4)

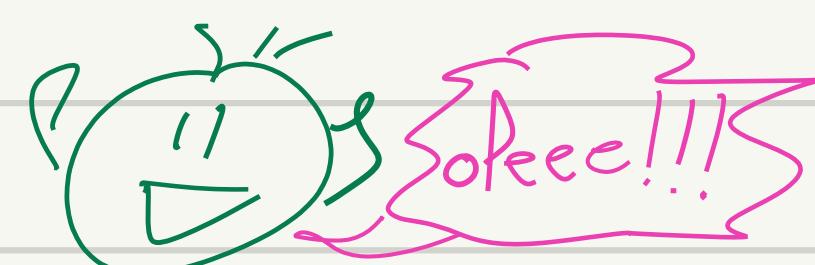
↳ Vulnerabilidad (24 bits vector inicialización IV)

↓ (con varios paquetes, se recupera clave Secreta)

Surge **WPA** (RC4)

· Igual que WEP, pero vector de 48 bits (**vulnerable**)

Con **WPA 2/3** → Usamos RSA



¡mínimo!!

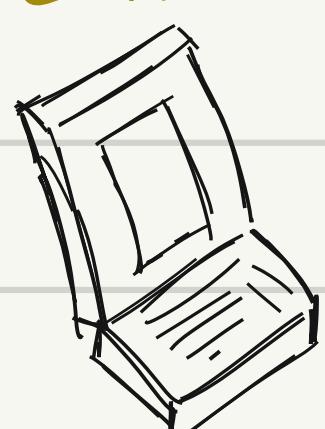
Autenticación: A través de un AP → Protocolo EAP

MAC/MIC usando HMAC (SHA-256)

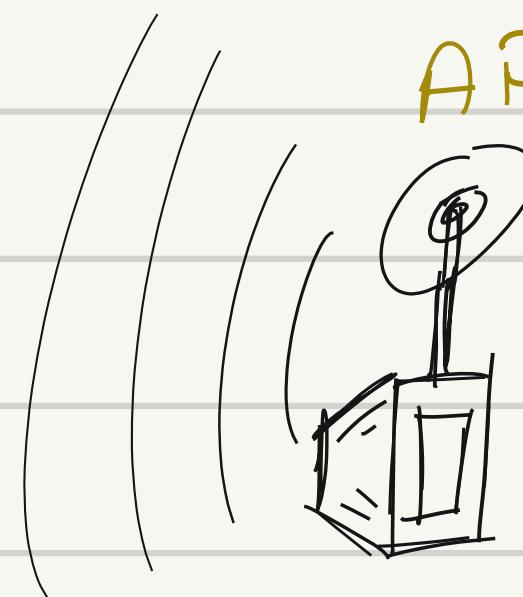
Ofrece Control Acceso: Basado en puertos (se bloquean hasta que se autentique).

Confidencialidad: Datos Se cifran

STA



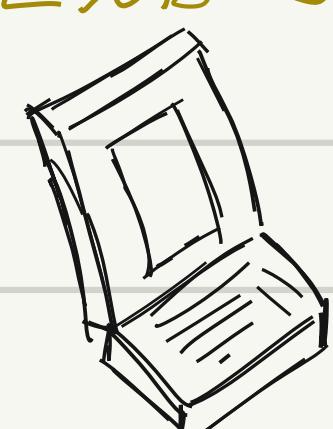
AP



AS



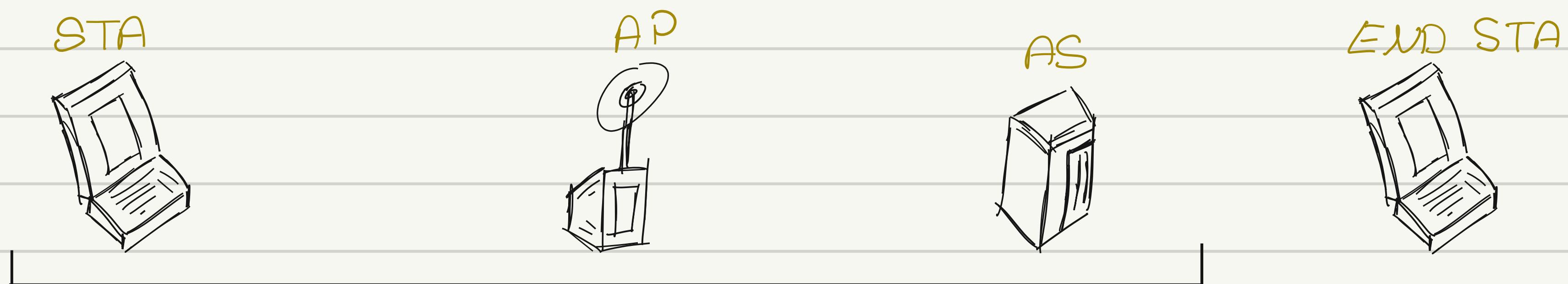
END STA



Fase 1: AP emite ondas beacon y probe responses para su política de seguridad.

↳ Se seleccione "Suite Cifrado" y mecanismo autenticación

Fase 2: STA y AS verifican la identidad AP bloquea tráfico hasta que la autenticación no acabe



Fase 3: AS y STA generan claves (Se guardan en AP y STA)

Comunicación Segura → Entre AP y STA

STA → AP → END STA

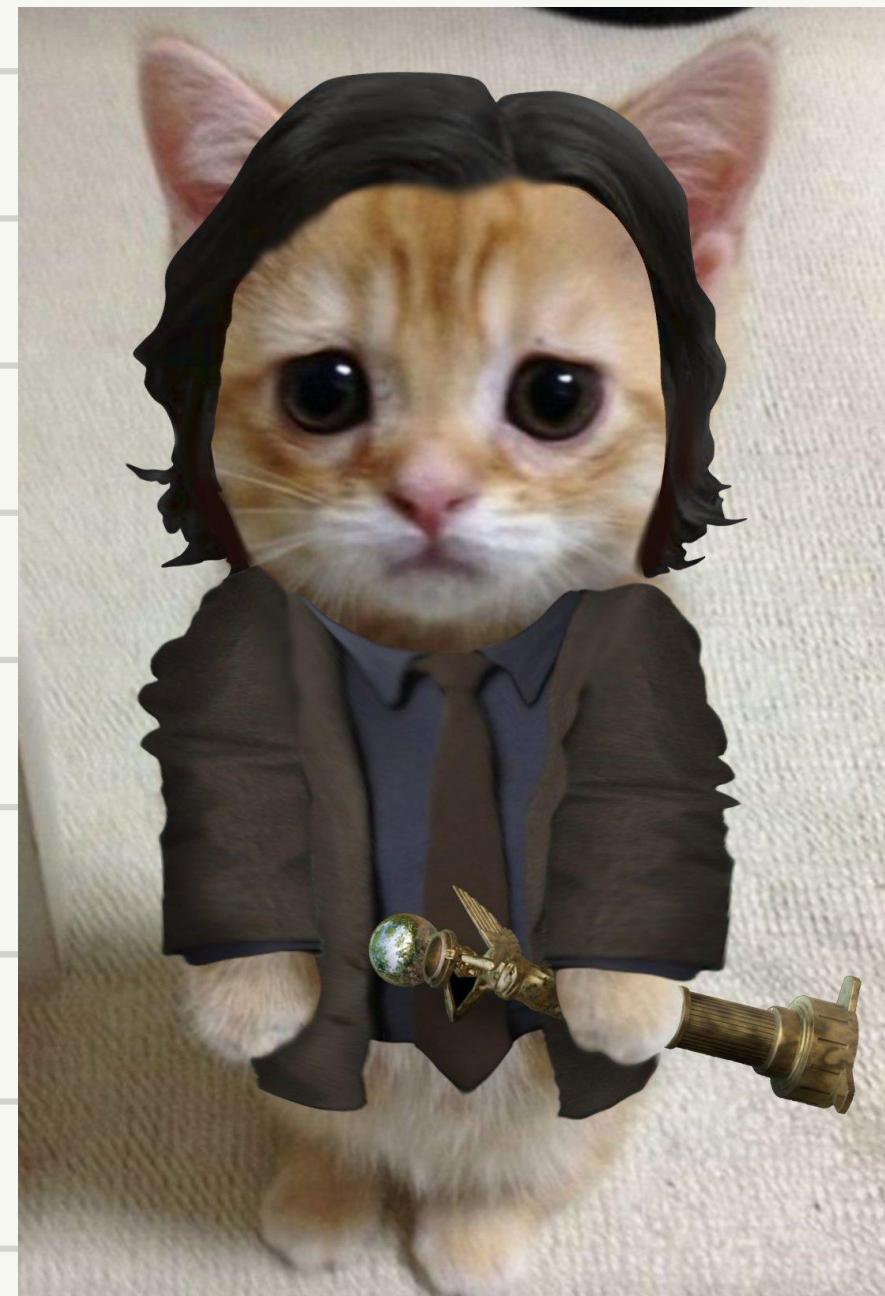
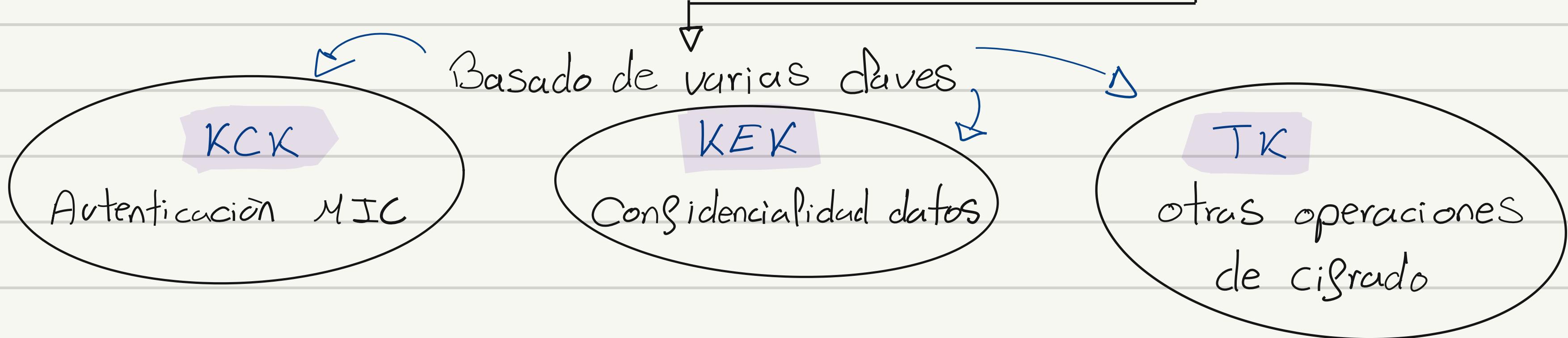
Fase 4: transferencia Segura SOLO pasando STA/END STA por AP

STA → AP → END STA

Fase 5: Finaliza Conexión

Generación de Claves

PMK → Se genera una clave temporal → PTK



Alejandro WAKE