

# Capítulo 8

## Seguridad en Redes:

### Integridad de Mensajes e e-mail seguro

*Basado en:*  
*Computer Networking: A Top Down Approach,*  
Jim Kurose, Keith Ross.

# Capítulo 8 contenidos

8.1 ¿Qué es la seguridad en la red?

8.2 Principios de criptografía

8.3 Integridad de mensajes

8.4 Dando seguridad a e-mail

8.5 Conexiones TCP seguras: SSL

8.6 Seguridad en capa de Red: IPsec

8.7 Seguridad en redes locales inalámbricas

8.8 Cortafuegos y Sistemas de detección de intrusión (IDS)

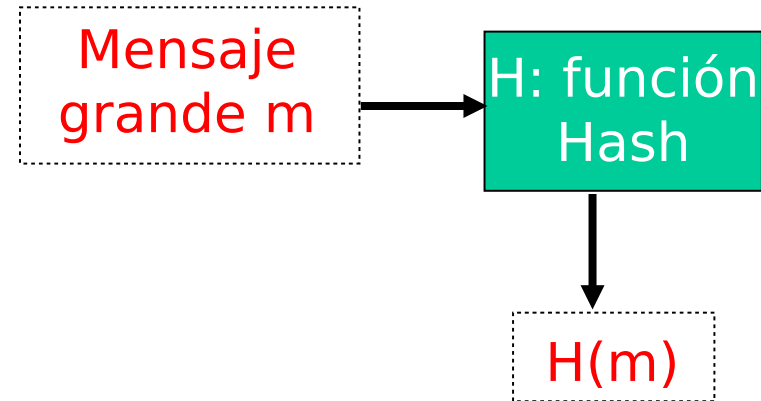
# Integridad de Mensajes

- ❑ Permite al Tx y Rx verificar que los mensajes son auténticos.
  - El contenido no ha sido alterado
  - La fuente del mensaje es quién o qué el Rx piensa que es.
  - El mensaje no ha sido reproducido (no es un duplicado de uno previo).
  - La secuencia de mensajes es mantenida
- ❑ Consideremos primero el concepto de resumen de un mensaje (message digest).

*Los requisitos de integridad listados arriba demanda autenticación y más.*

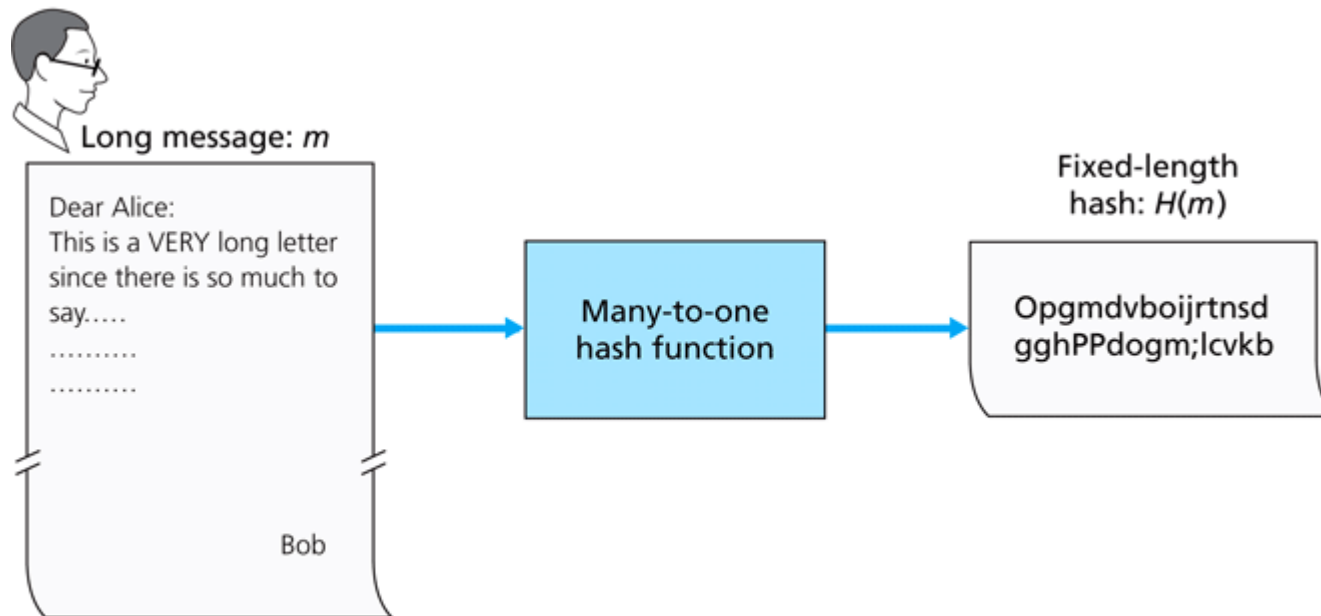
# Resumen de un mensaje (Message Digests)

- Una función  $H()$  toma como entrada un mensaje de largo arbitrario y genera un string de largo fijo: “**firma del mensaje**”
- Notar que  $H()$  es una función muchos a 1; en otras palabras, hay más secuencias de entrada que de salida.
- $H()$  es conocida como “función hash”
- Ejemplo: Códigos de redundancia cíclica (CRC)



- Propiedades deseables:
  - Fácil de calcular
  - Irreversible: No se pueda determinar  $m$  a partir de  $H(m)$
  - Resistencia a colisiones: que sea difícil generar  $m$  y  $m'$  tal que  $H(m) = H(m')$
  - Salida de apariencia aleatoria

# Resumen de un mensaje (Message Digests)



**Figure 8.7** ♦ Hash functions

# La suma de chequeo en Internet es un resumen pobre del mensaje

- La suma de chequeo de Internet tiene algunas propiedades de función hash:
  - Produce resumen de largo fijo (suma de 16-bit)
  - Es del tipo muchos es a uno
- **Pero** dado un mensaje con un valor hash, es fácil encontrar otro con el mismo valor.
- Ejemplo: suma de chequeo simplificada: suma grupos de 4-byte a la vez:

<u>mensaje</u>	<u>Formato ASCII</u>	<u>mensaje</u>	<u>Formato ASCII</u>
I O U 1	49 4F 55 31	I O U <u>9</u>	49 4F 55 <u>39</u>
0 0 . 9	30 30 2E 39	0 0 . <u>1</u>	30 30 2E <u>31</u>
9 B O B	39 42 4F 42	9 B O B	39 42 4F 42

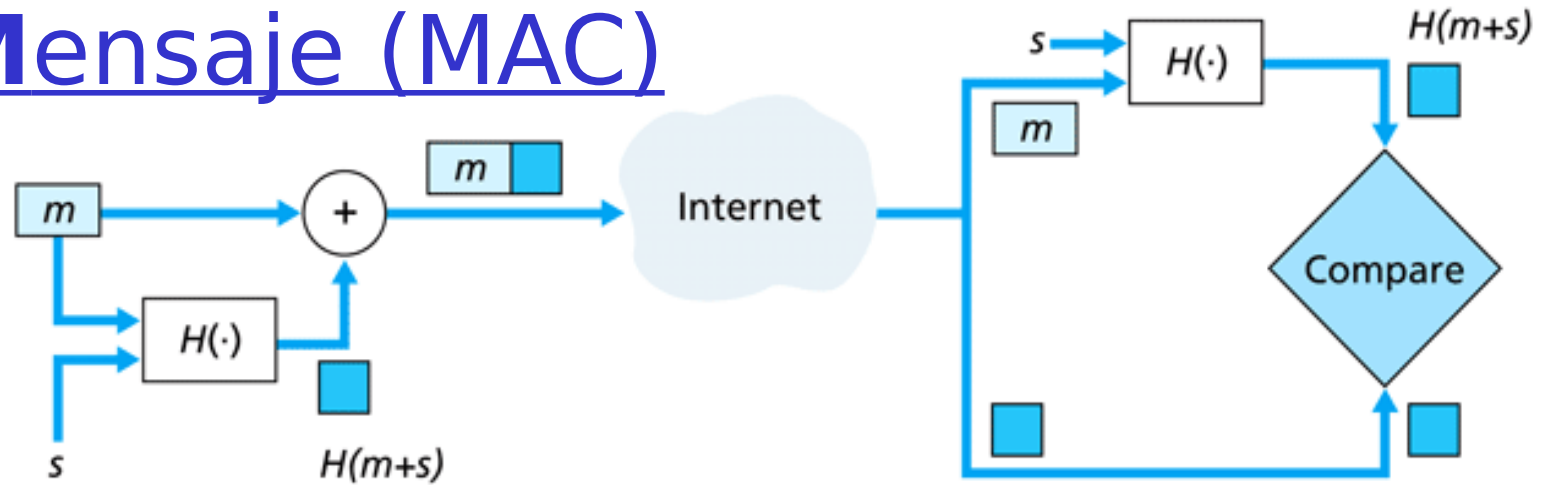
B2 C1 D2 AC — Mensajes diferentes — B2 C1 D2 AC

Pero suma de chequeo idéntica!

# Algoritmos de Función Hash

- MD5 es una función hash usada ampliamente (RFC 1321)
  - Genera un resumen del mensaje de 128-bit en un proceso de 4 pasos.
- SHA-1 también es usado.
  - Es un estándar en US
  - Genera un resumen del mensaje de 160-bit

# Código de Autenticación de Mensaje (MAC)



Key:

$m$  = Message

$s$  = Shared secret

**Figure 8.9** ♦ Message authentication code (MAC)

- ❑ **Autentica al Tx. (porque el Tx tiene la misma  $s$ )**
- ❑ **Verifica la integridad del mensaje**
- ❑ No encripta !
- ❑ Notación:  $MD_m = H(s+m)$  ; enviamos  $m+MD_m$  Aquí + es concatenación.



# Ejemplo: OSPF

- ❑ Recordar que OSPF es un protocolo intra-SA (tipo Dijkstra)
- ❑ Cada router crea un mapa del Sistema Autónomo entero y corre el algoritmo de ruta más corta sobre el mapa.
- ❑ Los routers reciben los avisos de estado de enlace desde otros routers en el SA.

## Ataques:

- ❑ Inserción de mensajes
- ❑ Borrado de mensajes
- ❑ Modificación de mensajes
- ❑ ¿Cómo sabemos si un aviso OSPF es auténtico?

# Ejemplo OSPF:Autenticación

- ❑ Dentro del sistema autónomo, router envían mensajes OSPF a otros.
- ❑ OSPF provee alternativas de autenticación:
  - No autenticar
  - Uso de hash
- ❑ Uso de hash con MD5
  - Campo de 64-bit de autenticación incluye 32 bits de número de secuencia.
  - MD5 es corrido sobre la concatenación del paquete OSPF y la clave compartida.
  - Luego el hash MD5 es agregado al paquete OSPF; el cual es encapsulado en un datagrama IP.

# Firmas Digitales

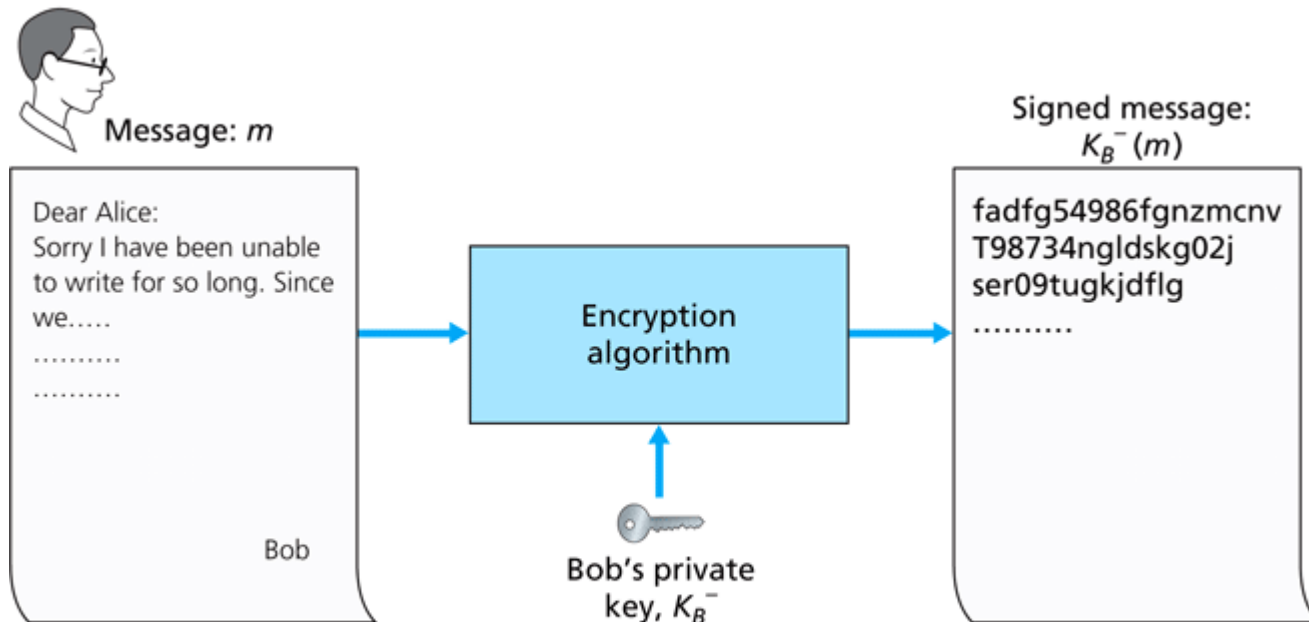
## Técnica criptográfica análoga a las firmas a mano

- ❑ Rx (Bob) digitalmente firma un documento, establece así que él es su dueño/creador.
- ❑ Objetivo es similar a Message Authentication Code, **excepto que ahora se usa el método de encriptación de clave pública**
- ❑ **verificable, no-repudiable**: receptor (Alice) puede probar que Bob, y nadie más, debió haber firmado el documento.

# Firma Digital: Motivación

## Firma digital simple para mensaje $m$ :

- Bob firma  $m$  encriptándolo con su clave privada  $K_B^-$ , se crea un mensaje firmado,  $K_B^-(m)$

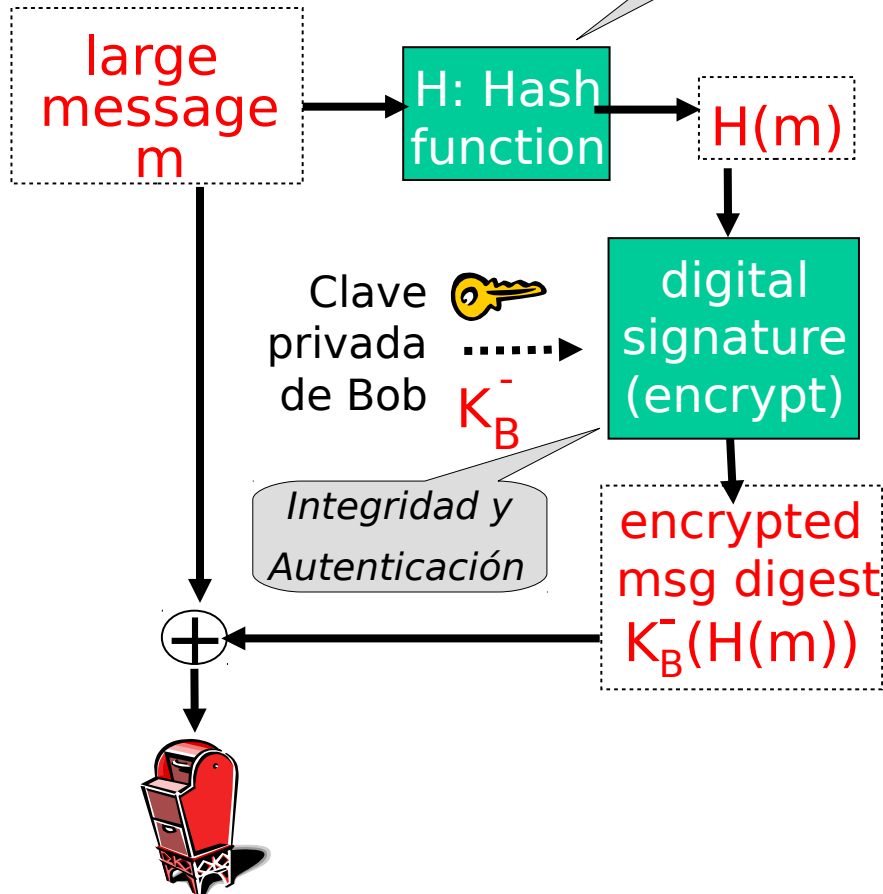


**Problema:** Cifrar y descifrar el mensaje completo es computacionalmente caro.

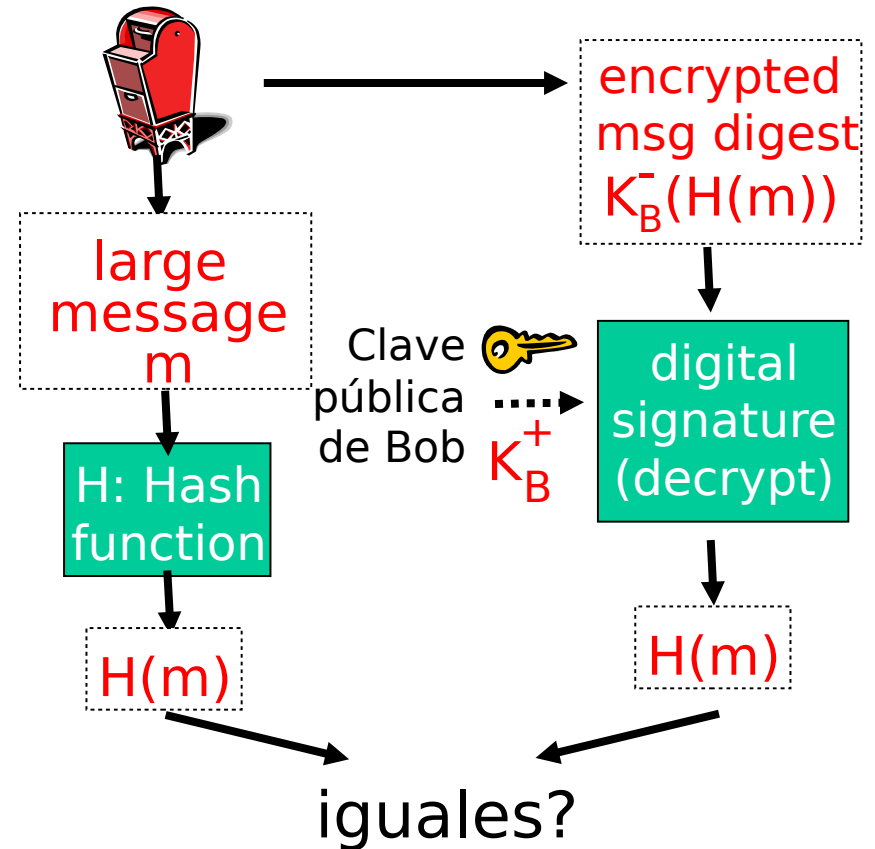
# Firma digital = firmar sólo el resumen del mensaje

Envío de un mensaje firmado:

Para hacerlo rápido



Alicia verifica firma e integridad del mensaje:



# Firma Digital (más)

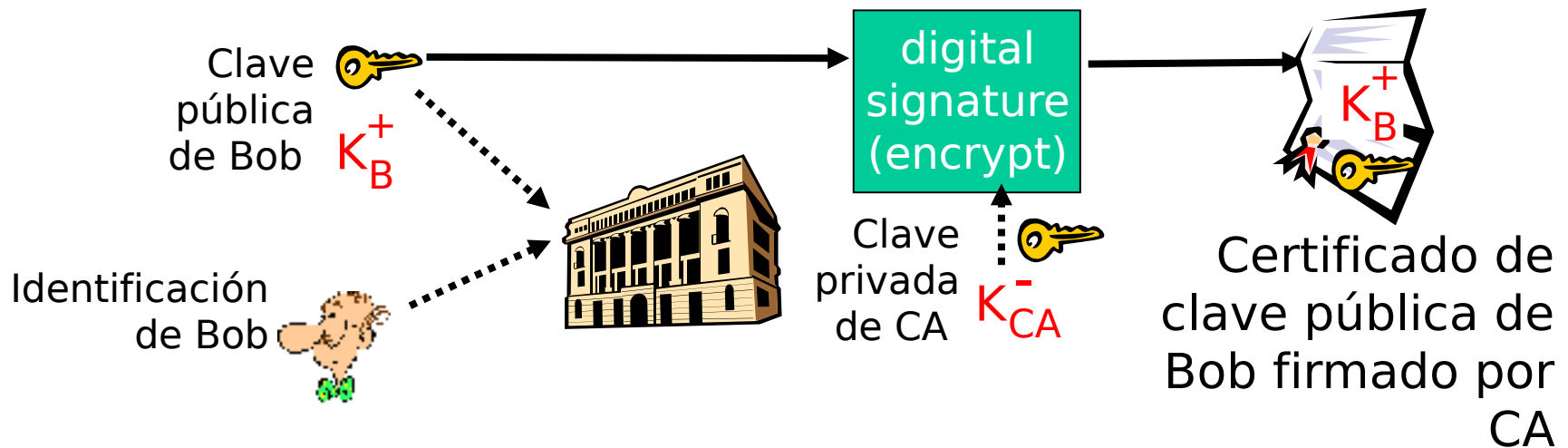
- Supongamos que Alicia recibe mensaje  $m$  y su firma digital  $K_B^-(H(m))$
- Alicia verifica  $m$  aplicando la clave pública de Bob  $K_B^+$  a  $K_B^-(H(m))$  así chequea  $K_B^+(K_B^-(H(m))) = H(m)$ .
- Si  $K_B^+(K_B^-(H(m))) = H(m)$ , quien sea que firmó  $m$  debe haber usado la clave privada de Bob.
  - **Así Alicia verifica que:**
    - Bob firmó  $m$ , nadie más lo hizo.
    - Bob firmó  $m$  y no  $m'$ .
  - **No-repudiación:**
    - Alicia puede llevar  $m$  y la firma de  $m$ ,  $K_B^-(H(m))$ , a un juez y probar que Bob lo firmó.

# Certificación de Clave Pública

- Motivación: Un intruso hace una broma a Bob
  - Intruso hace una orden de pizza por mail:  
*Estimado Negocio: Por favor envíeme 4 pizzas de peperoni. Gracias, Bob.*
  - Intruso firma la orden con su clave privada.
  - Intruso envía al negocio su clave pública, pero dice que es la clave de Bob.
  - El negocio verifica la firma y envía las 4 pizzas a Bob.
  - ¿Qué haría usted?

# Autoridad Certificadora

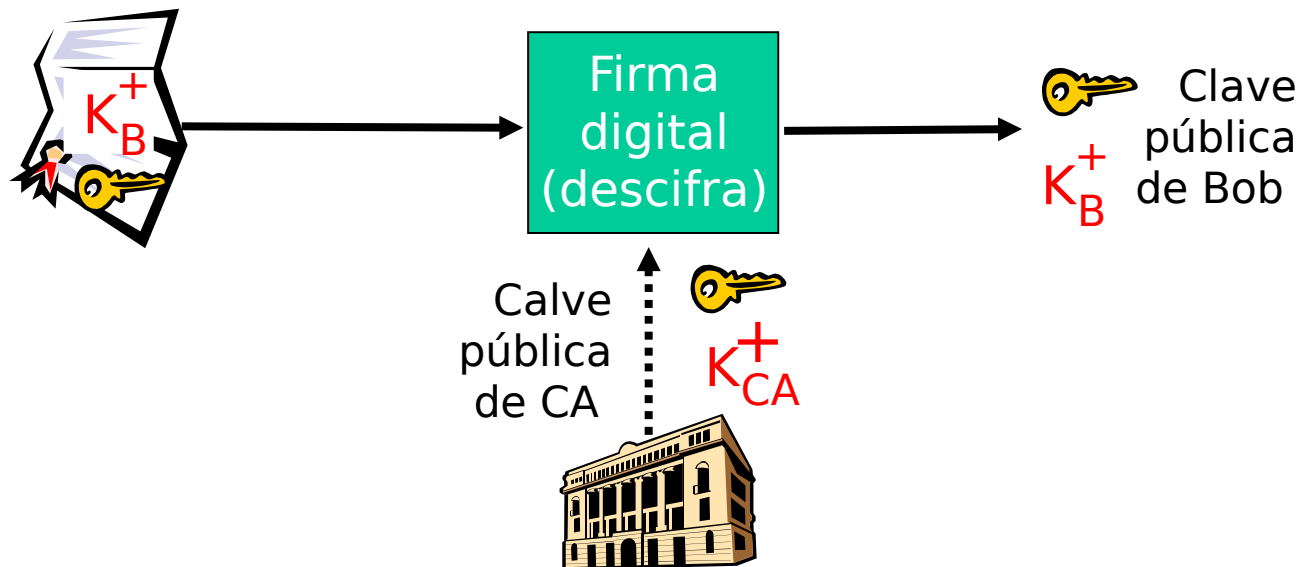
- **Autoridad Certificadora (CA):** Asocia la clave pública con un ente particular, E.
- E (persona, router) registra su clave pública con CA.
  - E provee “prueba de identidad” a CA.
  - CA crea un certificado asociando E a su clave pública.
  - Certificado contiene la clave pública de E firmada digitalmente por CA. CA afirma “esta es la clave pública de E”





# Autoridad Certificadora

- Cuando Alicia necesita la clave pública de Bob:
  - Obtiene el certificado de Bob (desde Bob u otro lugar).
  - Aplica la clave pública de CA al certificado de Bob, así verifica la clave pública de Bob.



# Certificados: resumen

- ❑ Estándar primario X.509 (RFC 2459)
- ❑ Cada certificado contiene:
  - Nombre de quien lo emite
  - Nombre de la entidad, dirección, nombre de dominio, etc.
  - Clave pública de la Entidad
  - Firma digital (firmado con la clave privada del emisor)
- ❑ Public-Key Infrastructure (PKI)
  - Todo lo necesario: software, procedimientos, personas etc. para gestionar certificados digitales
  - A menudo considerada “pesada o excesiva”

# Autenticación del extremo

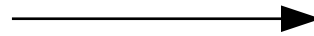
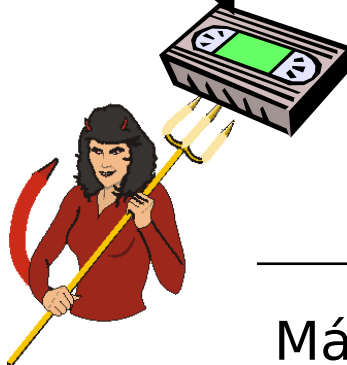
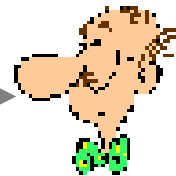
- ❑ Se trata de asegurar quién originó el mensaje.
- ❑ Si suponemos que Alicia y Bob tienen una clave secreta compartida, ¿proveerá MAC (Código de autenticación de mensaje) autenticación en extremo?
  - Sabemos que Alicia creó el mensaje.
  - ¿Pero lo habrá enviado ella?

# Ataque de reproducción

MAC =  
 $f(\text{msg}, s)$



Transfiera \$1M de Bill a intruso	MAC
--------------------------------------	-----



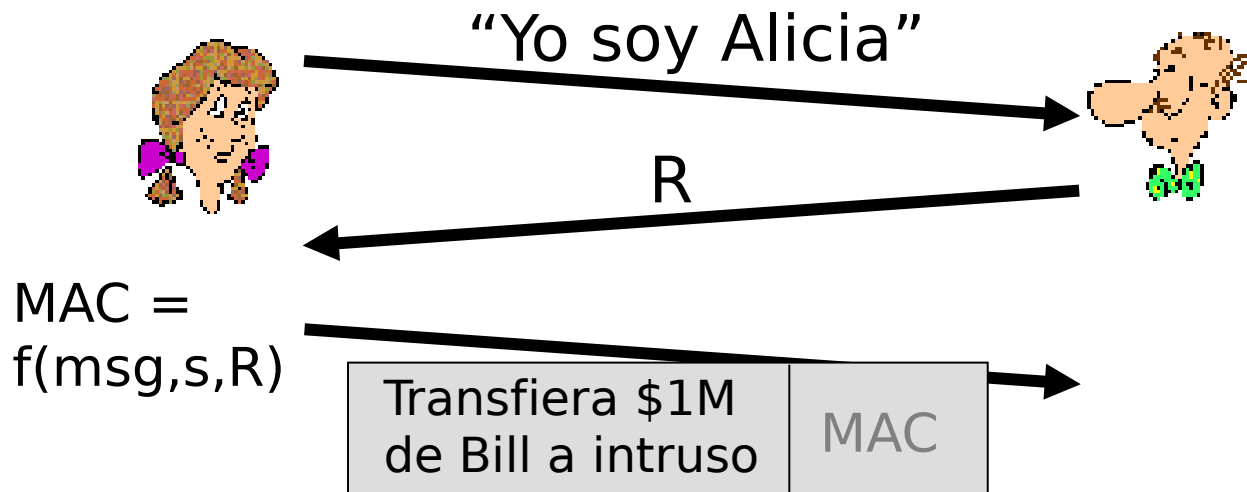
Más tarde ...



Transfiera \$1M De Bill a intruso	MAC
--------------------------------------	-----

Conclusión: MAC no basta para proveer integridad.

# Derrota a ataques de reproducción: Número único



- Al generar un número único e incluir éste en la respuesta, el mensaje enviado por Alicia no puede ser usado nuevamente a futuro.
- El receptor debe asegurar que su número sea único cada vez.

# Capítulo 8 contenidos

8.1 ¿Qué es la seguridad en la red?

8.2 Principios de criptografía

8.3 Integridad de mensajes

8.4 Dando seguridad a e-mail

8.5 Conexiones TCP seguras: SSL

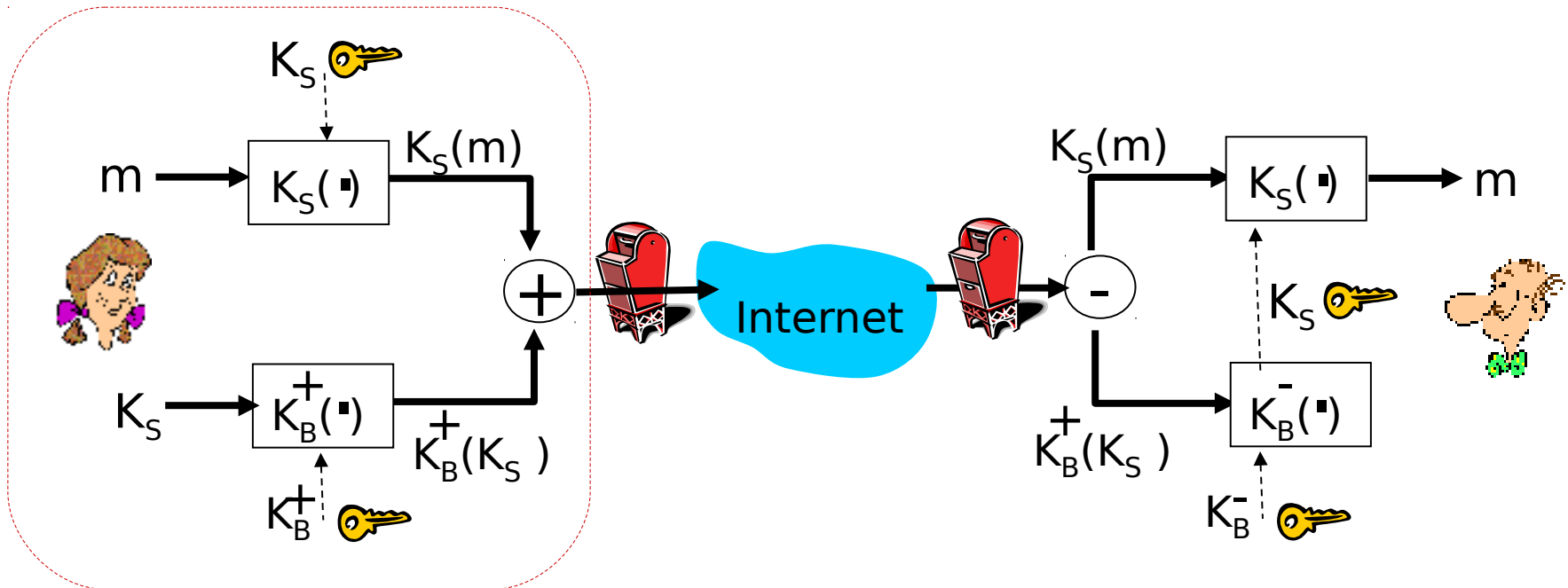
8.6 Seguridad en capa de Red: IPsec

8.7 Seguridad en redes locales inalámbricas

8.8 Cortafuegos y Sistemas de detección de intrusión (IDS)

# E-mail seguros

- Alicia desea enviar un email **confidencial**,  $m$ , a Bob.

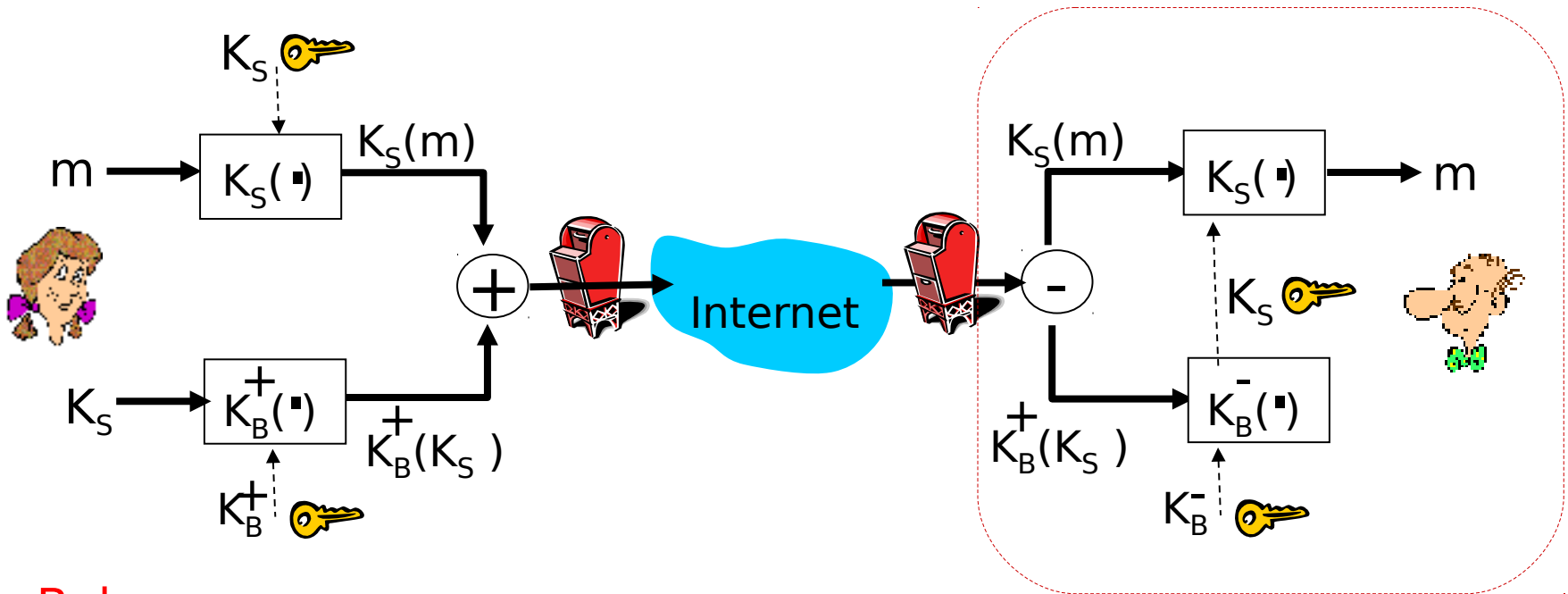


**Alicia:**

- Genera una clave *simétrica* privada,  $K_S$ .
- Cifra mensaje  $m$  con  $K_S$  (por eficiencia)
- También cifra  $K_S$  con clave pública de Bob.
- Envía ambos  $K_S(m)$  y  $K_B^+(K_S)$  a Bob.

# E-mail seguros (cont.)

- Alicia desea enviar un email **confidencial**,  $m$ , a Bob.



**Bob:**

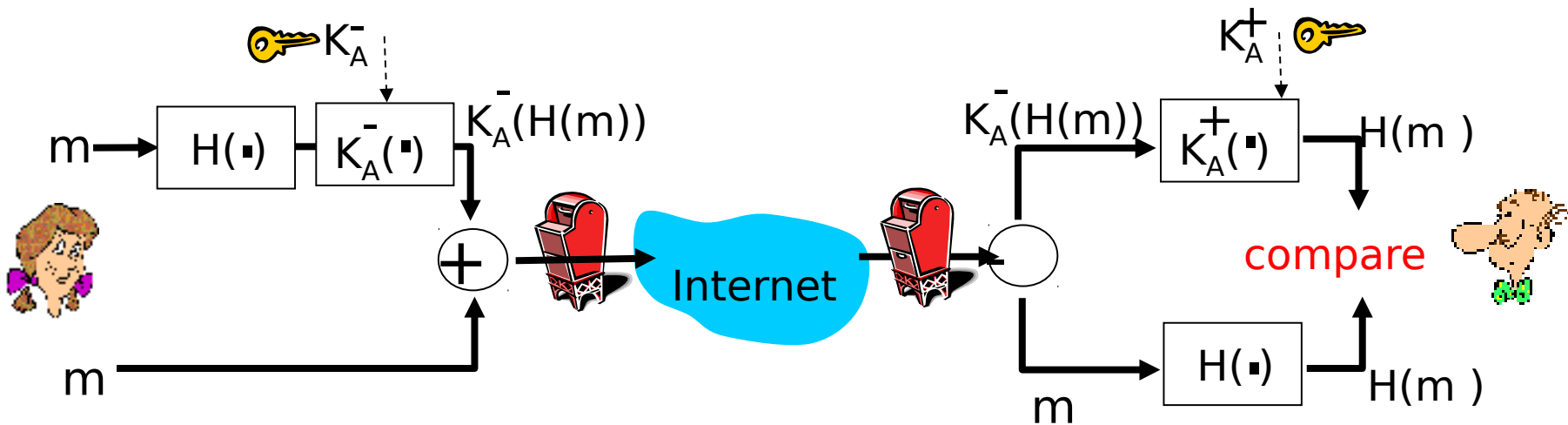
- Usa su clave privada para descifrar y recobrar  $K_S$
- usa  $K_S$  para descifrar  $K_S(m)$  y recuperar  $m$

*Se consigue integridad?*



# E-mail seguro (continuación)

- Alicia desea proveer **autenticación** de la fuente e **integridad** del mensaje.

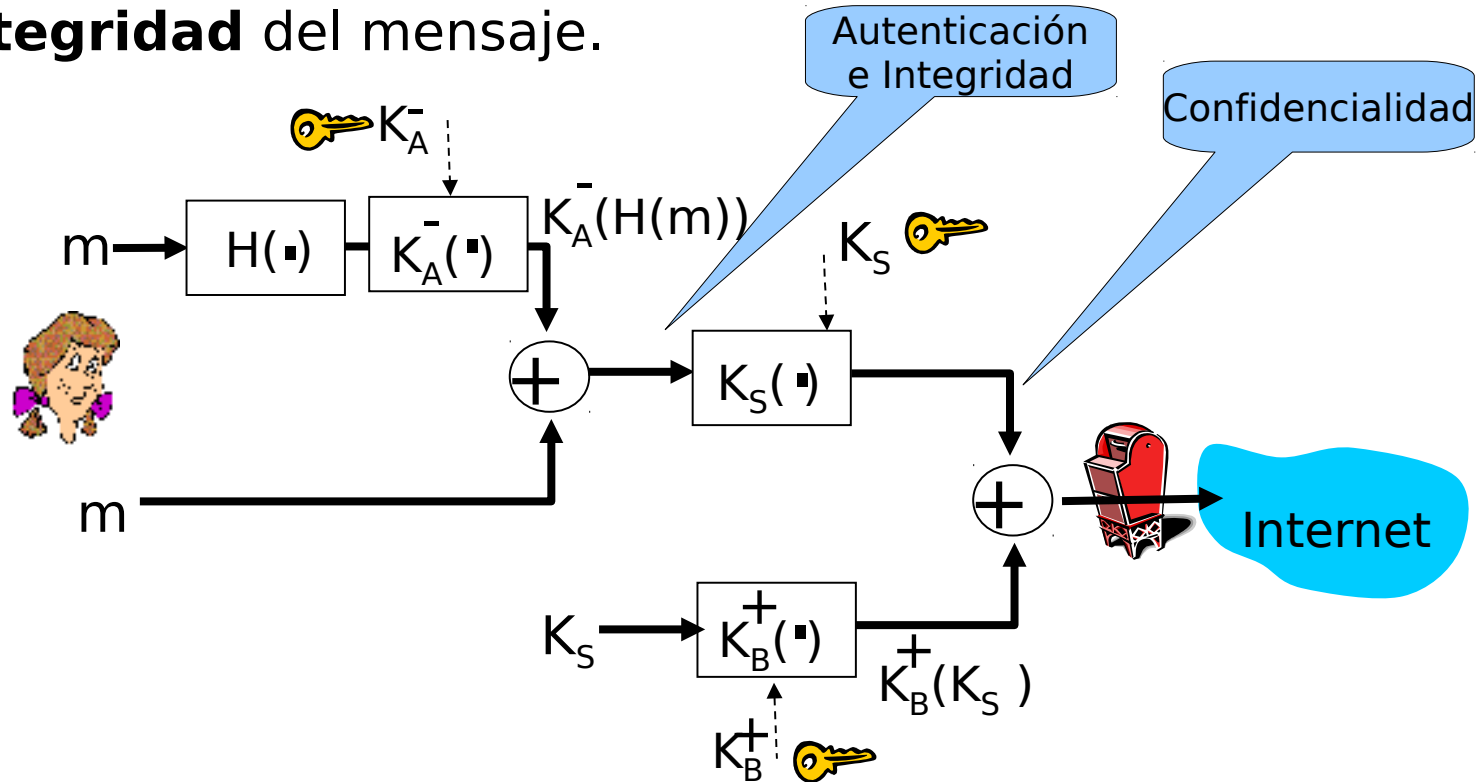


- Alicia firma digitalmente el mensaje.
- Envía tanto el mensaje (en claro) como su firma digital.

Si se tratara de una conversación, con esto no basta para asegurar integridad. Ataque de repetición.

# E-mail seguro (continuación)

- Alicia desea proveer **confidencialidad**, **autenticación** de fuente, e **integridad** del mensaje.



Alicia usa tres claves: su clave privada, la clave pública de Bob, una clave asimétrica nueva.

# Capítulo 8 contenidos

8.1 ¿Qué es la seguridad en la red?

8.2 Principios de criptografía

8.3 Integridad de mensajes

8.4 Dando seguridad a e-mail

8.5 Conexiones TCP seguras: SSL

8.6 Seguridad en capa de Red: IPsec

8.7 Seguridad en redes locales inalámbricas

8.8 Cortafuegos y Sistemas de detección de intrusión (IDS)