Seguridad en Redes de Ordenadores

Sistemas Telemáticos para Medios Audiovisuales

GSyC
Departamento de Teoría de la Señal y Comunicaciones y
Sistemas Telemáticos y Computación

Noviembre 2018

Las transparencias contienen en parte material adaptado del libro:
Redes de Computadores: un enfoque descendente
James F. Kurose, Keith W. Ross

Contenidos

Contenidos

Propiedades de una comunicación segura (I)

Confidencialidad

Únicamente el emisor y el receptor de un mensaje "entienden" su contenido.

- El emisor cifra el mensaje
- El receptor descifra el mensaje

Autenticación de los extremos

El emisor y el receptor quieren confirmar la identidad de su interlocutor.

- Cada uno quiere estar seguro de que el otro es quien dice ser.

Propiedades de una comunicación segura (II)

Integridad del mensaje (o "autenticación" del mensaje)

El emisor y el receptor quieren estar seguros de que el mensaje no ha sido alterado en tránsito.

- Lo que envió el emisor es exactamente lo que recibió el receptor

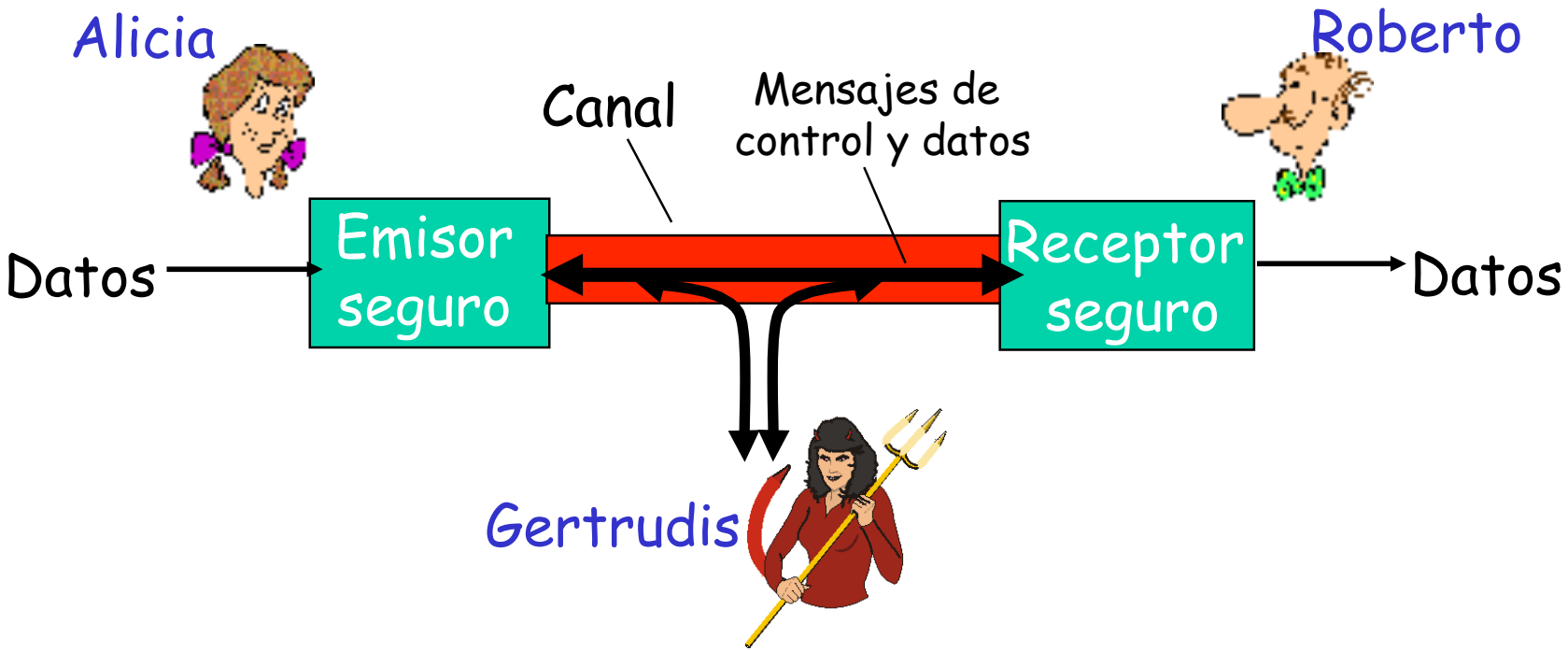
Junto con la autenticación, posibilita el no repudio: El emisor no puede negar que envió ese mensaje.

Disponibilidad y acceso

Los servicios deben estar accesibles y disponibles para los usuarios.

Alicia, Roberto y Gertrudis

- Personajes que aparecerán en los ejemplos.
- Roberto y Alicia quieren comunicarse de forma segura.
- Gertrudis es una intrusa que puede:
  - interceptar los mensajes que intercambian Roberto y Alicia, y ver lo que contienen
  - modificar los mensajes
  - insertar o eliminar mensajes



¿Cuáles son los equivalentes de Alicia y Roberto?

- Dos usuarios de la red que chatean o se envían correos electrónicos.
- Un navegador y un servidor de HTTP que efectúan transacciones electrónicas (p. ej.: compras por Internet).
- Un cliente y un servidor de banco online.
- Dos servidores de DNS que intercambian consultas y respuestas.
- Dos *routers* que intercambian información de sus tablas de encaminamiento.

Ataques a la seguridad

- Descubrimiento de servicios en una red
  - IPs, nombres de máquinas, routers
  - Escaneado de puertos (en los que hay servidores escuchando)
  - Ej: `nmap`.
- Escuchar a escondidas: interceptar mensajes *(packet sniffing)*
- Insertar activamente mensajes en una conexión.
- Modificar cualquier campo de un paquete.
  - Suplantación: falsear la dirección origen de un paquete *(IP spoofing, MAC spoofing)*
  - Secuestro: "apoderarse" de una conexión eliminando al receptor o al emisor e insertándose en su lugar.
- Denegación de servicio *(Denial of Service, DOS)*: generar paquetes masivamente para inundar a un servidor: impide que el servicio sea utilizado por otros.
  - Denegación de servicio distribuida *(Distributed Denial of Service, DDOS)*: Los paquetes que inundan los generan diferentes máquinas (o, al menos, tienen diferentes direcciones IP de origen).

Contenidos

Definición

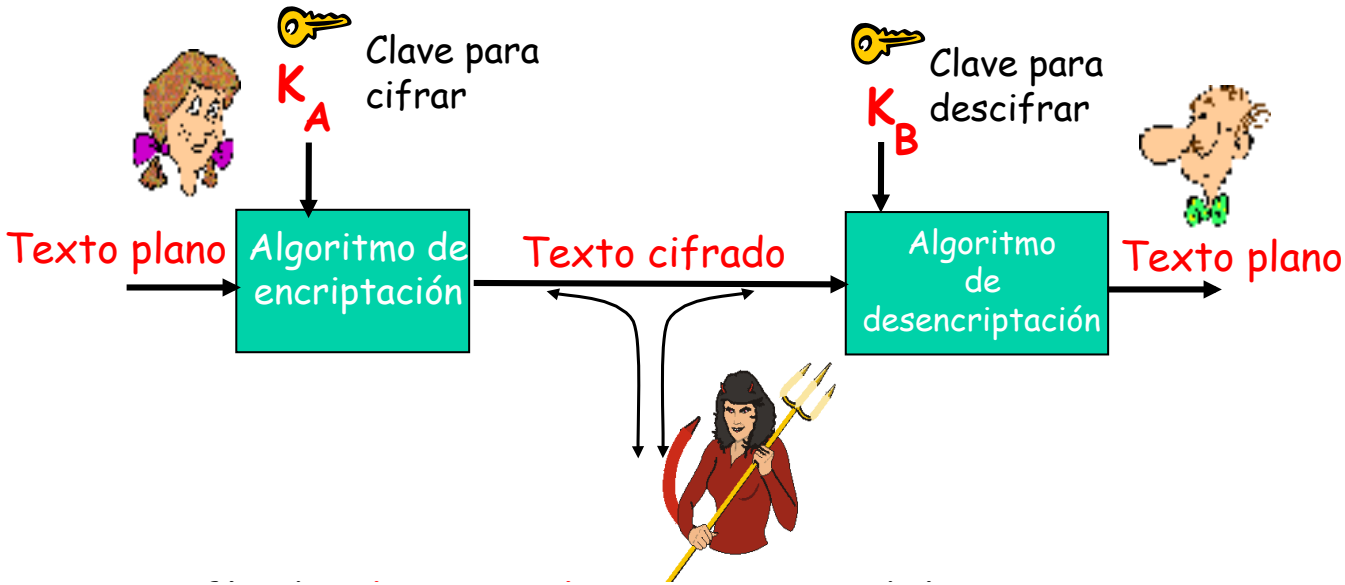
Criptografía

Del griego *krypto* («oculto») y *graphos*, («escribir»): literalmente «escritura oculta»

Práctica y estudio de la ocultación de información

- Sus orígenes suelen situarse en las civilizaciones de Grecia y Roma.
- Tradicionalmente **criptografía** se ha identificado con **cifrado**, pero hoy el término tiene un sentido más general.

Tipos de criptografía



- Criptografía de clave simétrica: Se usa la misma clave para cifrar y para descifrar: $K_A = K_B$
  - Alicia y Roberto deben conocer esa clave para poder comunicarse
- Criptografía de clave pública (o asimétrica): Las claves para cifrar y descifrar son diferentes:
  - Alicia cifra con una clave K_A que Roberto proporciona públicamente a todo el que quiera comunicarse con él.
  - Roberto descifra con una clave K_B que mantiene en secreto.

Criptografía de clave simétrica: Cifrado de sustitución

- Cifrado de sustitución: Sustituir una cosa por otra.
  - **Cifrado César** (atribuido a Julio César): sustituye cada letra del alfabeto por otra desplazada un número fijo de posiciones
  - **Cifrado monoalfabético**: sustituye una letra del alfabeto por otra arbitraria, sin ningún tipo de patrón
- Ejemplo de cifrado monoalfabético:

```
Alfabeto:           abcdefghijklmnopqrstuvwxyz
Alfabeto cifrado:   mnbvcxzasdfghjklpoiuytrewq
```

```
Mensaje:            roberto, te quiero. alicia
Mensaje cifrado:    okncouk, uc pyscok. mgsbsm
```

- Problema: El ataque por fuerza bruta (probar todas las combinaciones) sería muy lento, pero la frecuencia estadística de aparición de letras en un idioma puede ayudar mucho.

Criptografía de clave simétrica: Cifrado por bloques

- Cifrado por bloques (ECB, *Electronic Codebook*)
  - El texto plano (en binario) se trocea en bloques de bits
  - Cada patrón de bits se sustituye por otro diferente
- Ejemplo (con trozos de 3 bits, en realidad se hace con trozos de 64 o 128):

```
Patrones de texto plano:   000 001 010 011 100 101 110 111
Patrones de texto cifrado: 110 111 101 100 011 010 000 001
```

```
Mensaje:            010110001111...
Mensaje cifrado:    101000111001...
```

- Ahora el ataque de fuerza bruta es mucho más difícil, y las frecuencias estadísticas no ayudan
- Problema: el mismo trozo de texto claro genera el mismo trozo de texto cifrado
- Solución: Cifrado por bloques en cadena (CBC, *Cipher-Block Chaining*)
  - una vez cifrado un bloque, se hace un XOR de él con el siguiente bloque sin cifrar, y ése es el siguiente bloque a cifrar, y así consecutivamente
  - de esta manera el resultado de un bloque cifrado depende no sólo del bloque sin cifrar, sino de los bloques precedentes

Criptografía de clave simétrica: DES *(Data Encryption Standard)*

- Estándar de Cifrado de EE.UU. [NIST 1993, National Institute os Standards and Technology].
- Cifrado por bloques ECB.
- Clave simétrica de 56 bits, para bloques de texto plano de 64 bits.
- ¿Qué seguridad tiene el DES?
  - Desafío DES: En 1997 Se propuso descifrar una frase cifrada con DES. Por fuerza bruta se consiguió descifrar en 4 meses. La frase era "la criptografía fuerte hace del mundo un lugar más seguro".
  - Hoy se descifra en horas.
- Un DES más robusto:
  - Utilizar tres claves secuencialmente (3-DES) en cada dato.
  - Utilizar CBC en vez de ECB.

Criptografía de clave simétrica: AES *(Advanced Encryption Standard)*

- Nuevo sistema de clave simétrica (Nov. 2001, estándar NIST), que reemplaza a DES.
- Procesa datos en bloques de 128 bits.
- Claves de 128, 192, ó 256 bits.
- Mucho más pesado el descifrado por fuerza bruta:
  - Un ordenador que descifrara DES con clave de 56 bits en 1 segundo, tardaría 149 billones de años en descifrar AES con clave de 128 bits.

Criptografía de clave simétrica: Problema

- Criptografía de clave simétrica: Roberto y Alicia comparten y conocen la misma clave (simétrica): K
- Ejemplo: la clave K es un patrón de sustitución conocido en un cifrado de sustitución monoalfabético.
- En todos los sistemas de criptografía de clave simétrica, el problema es: ¿cómo se pondrán de acuerdo Roberto y Alicia en la clave que usarán? (por ejemplo, ¿cómo se podrán de acuerdo en el patrón de sustitución que usarán?)

Criptografía de Clave simétrica vs de Clave pública

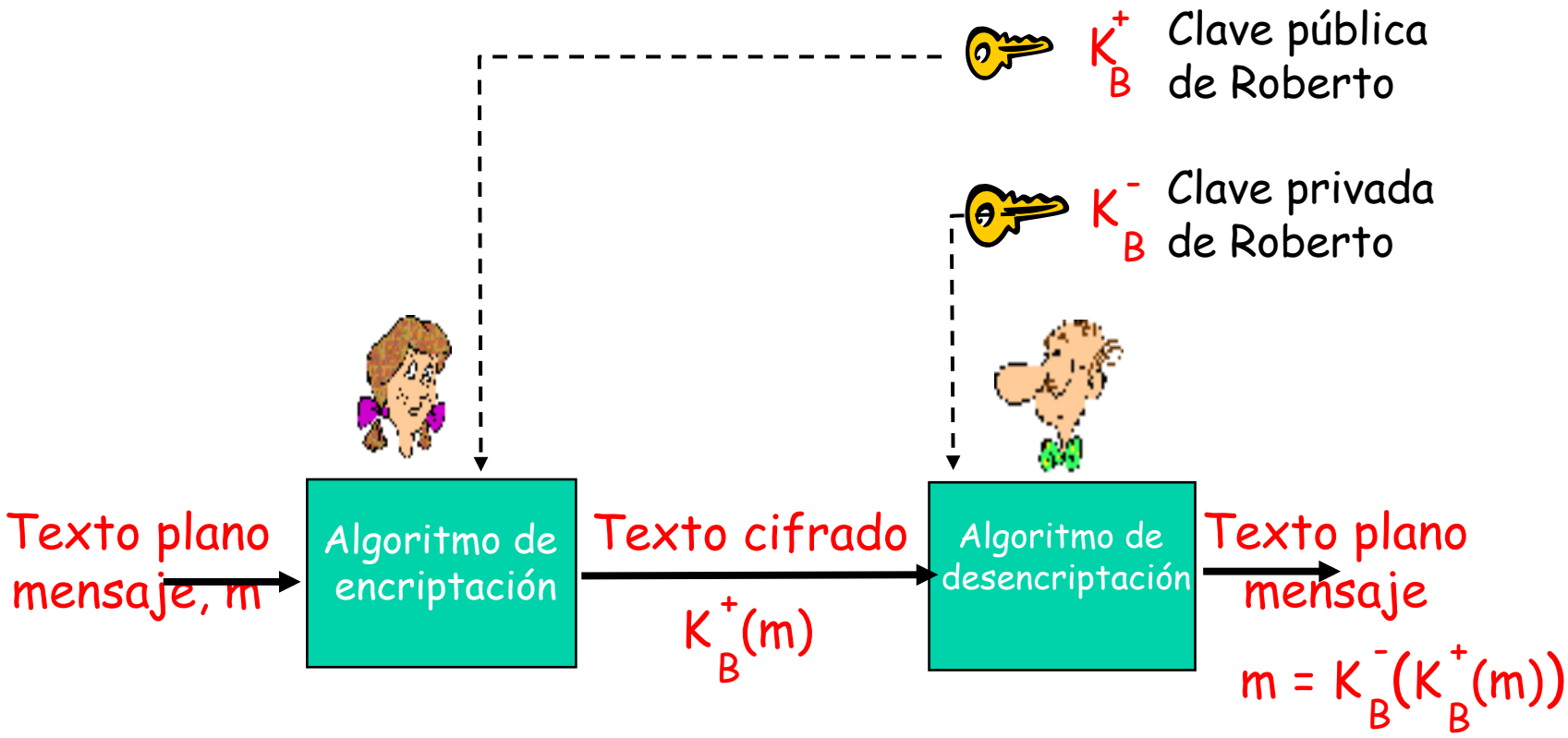
Criptografía de clave simétrica

- Requiere que emisor y receptor conozcan **la clave secreta compartida**.
- ¿Cómo ponerse de acuerdo en la clave, especialmente si nunca se han visto?

Criptografía de clave pública

- Ideado por Diffie y Hellman en 1976.
- Emisor y receptor no comparten una clave secreta:
  - Clave de cifrado **pública** conocida por **todos**.
  - Clave de descifrado **privada**, conocida sólo por **el receptor**.

Criptografía de clave pública



Cada usuario X del sistema tiene 2 claves:

- Una **clave pública** K_X^+ con la que otros cifran los mensajes que le envían a X
- Una **clave privada** K_X^- con la que X descifra los mensajes que le envían a él

Algoritmos de cifrado de clave pública

Requisitos:

1. Se necesita $K_B^+(.)$ y $K_B^-(.)$ de manera que:

$$K_B^-(K_B^+(m)) = m$$

2. Dada una clave pública K_B^+, debe ser imposible obtener su clave privada K_B^-.

RSA: Algoritmo de Rivest, Shamir y Adleman (1978).

RSA: elegir claves

1. Elegir dos números primos grandes, p y q. (por ejemplo, de 1024 bits cada uno).
2. Calcular n y z:
$$n = pq$$
$$z = (p-1)(q-1)$$
3. Elegir e (con $e < n$) que no tenga factores comunes con z (e y z son primos entre sí).
4. Encontrar un número d , tal que $ed - 1$ sea divisible de forma exacta entre z ($ed \mod z = 1$)
5. La clave pública es $\overbrace{(n, e)}^{K_B^+}$. La clave privada es $\overbrace{(n, d)}^{K_B^-}$

RSA: cifrado/descifrado

- Dados $\overbrace{(n, e)}^{K_B^+}$ y $\overbrace{(n, d)}^{K_B^-}$ calculados anteriormente:
  1. Para cifrar un patrón de bits m, se calcula:

     $c = m^e \mod n$

     (es decir, el resto cuando m^e se divide por n)
  2. Para descifrar el patrón de bits recibidos, c, calcular:

     $m = c^d \mod n$

     (es decir, el resto cuando c^d se divide por n)

- Tal como se han elegido los números, se demuestra que:

$$m = \underbrace{(m^e \mod n)}_{c}^d \mod n$$

Ejemplo RSA

- Roberto elige p=5, q=7. Entonces n=p*q=35 y z=(p-1)*(q-1)=24.
  - e=5 (entonces e y z son primos entre sí)
  - d=29 (entonces ed-1 es divisible de forma exacta por z).

1. Cifrado

letra	m	m^e	$c = m^e \mod n$
l	12	1524832	17

2. Transmisión de c

3. Descifrado

c	c^d	$m = c^d \mod n$	letra
17	48196857210675091509141182522307169 7	12	l

Cuando p y q son suficientemente grandes, encontrar p y q a partir de n (público) es muy costoso

Criptografía de clave pública: Propiedad

- La siguiente propiedad resulta muy útil: Es lo mismo aplicar primero la clave pública y luego la privada que hacerlo al revés

$$\underbrace{K_B^-(K_B^+(m))}_{\substack{\text{Usar primero clave pública, seguida de clave privada}}} = m = \underbrace{K_B^+(K_B^-(m))}_{\substack{\text{Usar primero clave privada, seguida de clave pública}}}$$

¡El resultado es el mismo!

Combinando clave simétrica y clave pública

- Lo peor de la criptografía de clave simétrica es **la distribución de la clave secreta** entre los interlocutores.
- Lo peor de la criptografía de clave pública es **que resulta mucho más lento cifrar/descifrar** que con criptografía de clave simétrica (RSA es unas 10.000 veces más lento que DES).
- Solución: Combinar los dos sistemas, lo que se conoce como Criptográfia Híbrida.

Criptografía Híbrida

- Alicia quiere enviar a Roberto un mensaje muy largo
- Alicia elige una clave de simétrica al azar (clave de sesión): K_S
- Alicia envía a Roberto la clave de sesión cifrada con la clave pública de Roberto: $K_B^+(K_S)$
- Roberto descifra la clave de sesión con su clave privada: $K_B^-(K_B^+(K_S)) = K_S$
- Alicia y Roberto se comunican cifrando y descifrando con K_S, clave simétrica.

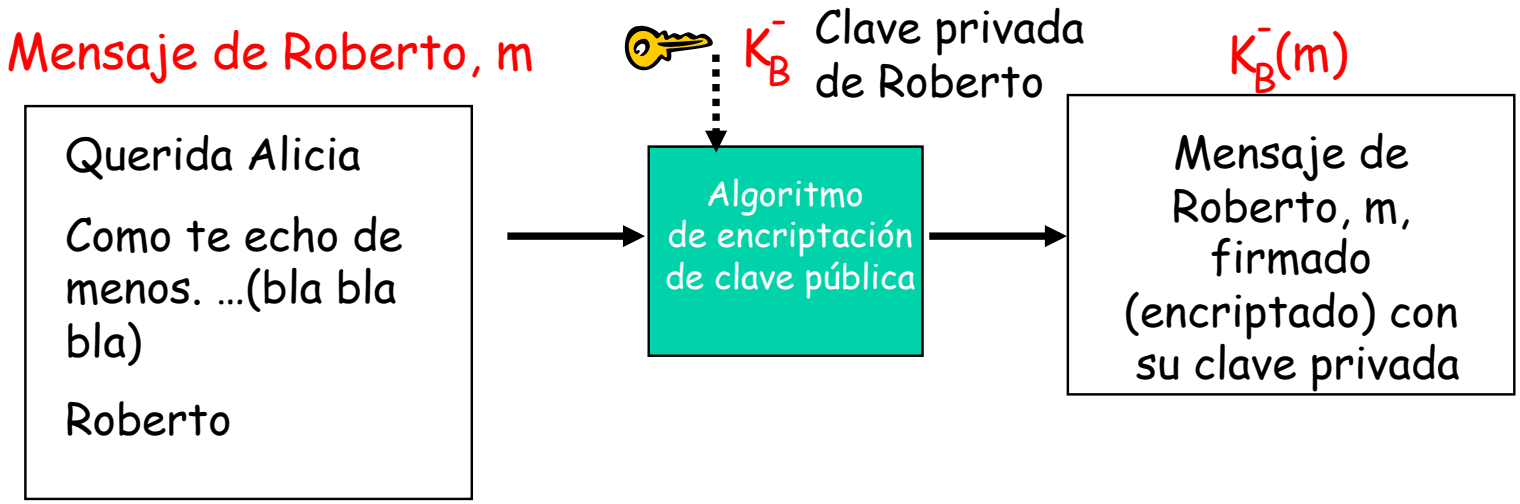
Contenidos

Firma digital

- Técnica criptográfica análoga a las firmas manuscritas.
- Garantiza que un mensaje no ha sido alterado en tránsito, y que viene de quien dice venir (Autenticación + Integridad)
- Emisor (Roberto) firma digitalmente un documento y establece que es su propietario/creador.
- **Verificable, no falsificable, no repudiable**

Firma digital simple: en emisión

- Roberto firma m cifrándolo con su clave privada K_B^-, creando un mensaje "firmado", $K_B^-(m)$.



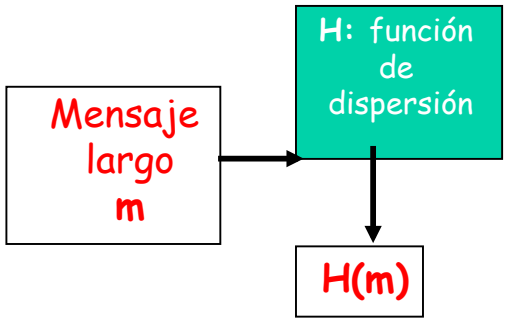
Firma digital simple: en recepción

- Supongamos que Alicia recibe el mensaje m, con firma digital $K_B^-(m)$.
- Alicia verifica m firmado por Roberto aplicando la clave pública de Roberto K_B^+ a $K_B^-(m)$ y comprueba que $K_B^+(K_B^-(m)) = m$.
- Si $K_B^+(K_B^-(m)) = m$, cualquiera que haya firmado m debe haber usado la clave privada de Roberto.
  - *Alicia verifica que:*
    - *Roberto ha firmado m*
    - *Nadie más ha firmado m*
    - *Roberto ha firmado m y no m'.*
  - *Roberto no puede repudiar ese mensaje:*
    - *Alicia puede tomar m y la firma $K_B^-(m)$ para demostrar ante un tercero (juzgado) que Roberto ha firmado m.*

Resumen (*Hash*) de un mensaje

- Ya sabemos que es lento cifrar con clave pública mensajes largos.
- Objetivo: en vez de utilizar el mensaje completo, utilizar algo de longitud fija, fácil de computar: la "huella dactilar".
  - Aplicar **función de resumen *(hash)* H a m** para obtener resumen del mensaje de tamaño fijo, H(m).

- Propiedades de la función *hash*:
  - Produce resumen de mensaje de tamaño fijo (huella dactilar).
  - Varios mensajes diferentes pueden tener el mismo resumen, pero dado el resumen x del mensaje m, es computacionalmente inviable hallar m' para que x = H(m'). Por tanto, es computacionalmente inviable dar el cambiazo.



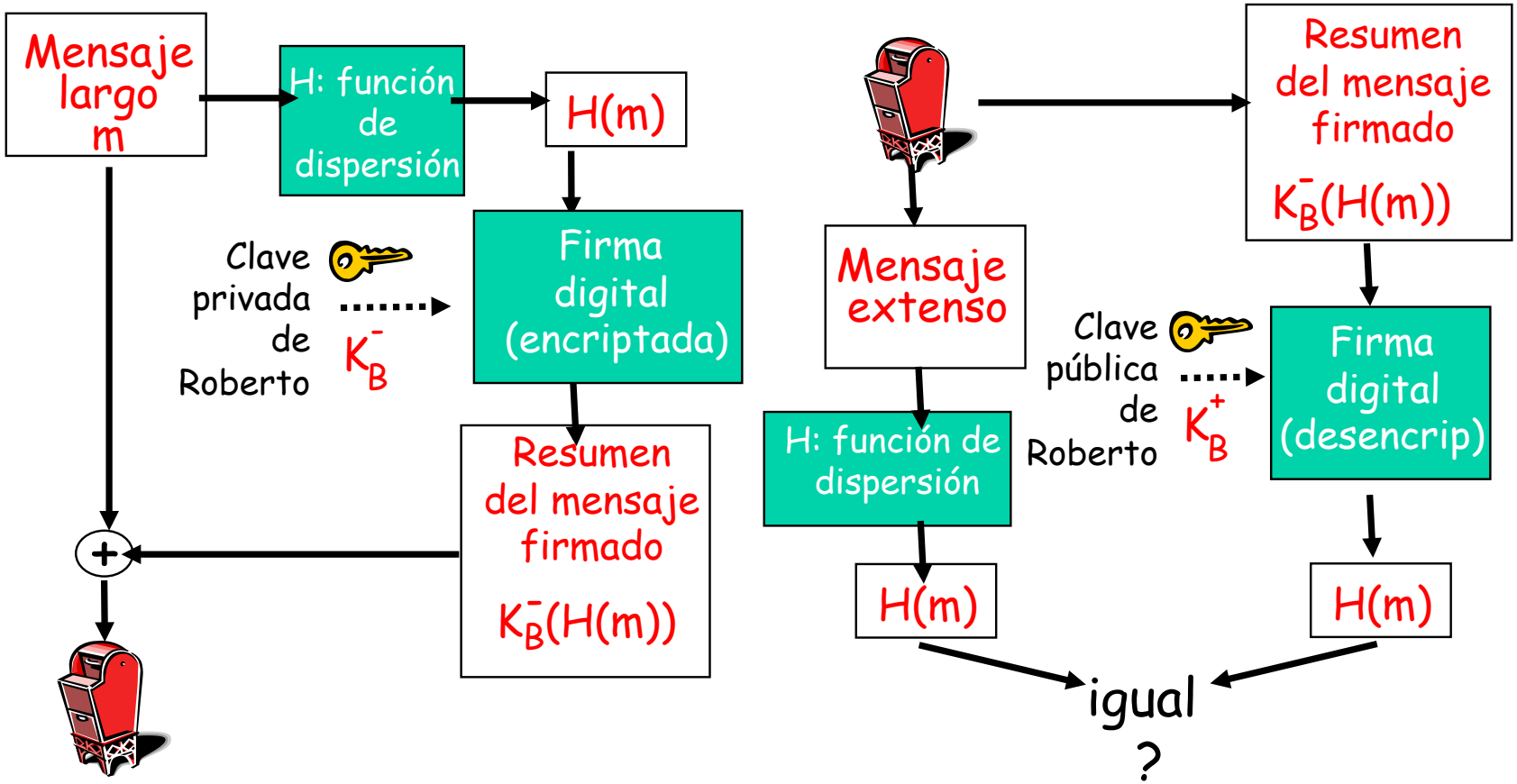
Firma digital del resumen de un mensaje (I)

- Que Roberto enviara cifrado con K_B^- todo el mensaje para que Alicia pudiera comprobar su integridad era lento
- Ahora sabemos hacer un resumen $H(m)$ de un mensaje m utilizando una función *hash*.
- Roberto enviará el mensaje junto con su resumen, pero sólo cifrará el resumen.
- Al ser el resumen corto y de tamaño fijo, este cifrado no resultará lento.
- Normalmente se llama firma digital al resumen de un mensaje cifrado con la clave privada del usuario que lo envía.

Firma digital del resumen de un mensaje (II)

Roberto envía mensaje firmado digitalmente

Alicia verifica la firma y la integridad del mensaje firmado digitalmente



Firma digital del resumen de un mensaje (III)

- MD5: función *hash* muy utilizada (RFC 1321):
  - Calcula un resumen de mensaje de 128 bits en un proceso de cuatro pasos.
  - A partir de una cadena x arbitraria de 128 bits, resulta muy difícil construir un mensaje m cuya función *hash* MD5 sea igual a x.
- También se utiliza SHA-1:
  - Estándar de EE.UU. [NIST, FIPS PUB 180-1].
  - Resumen de mensaje de 160 bits.

Contenidos

Intermediario de confianza

Problema de clave simétrica

- ¿Cómo pueden dos entidades tener una clave secreta compartida a través de la red?

Solución:

- **Centro de distribución de claves (KDC) que actúa como intermediario entre las entidades.**

Problema de clave pública:

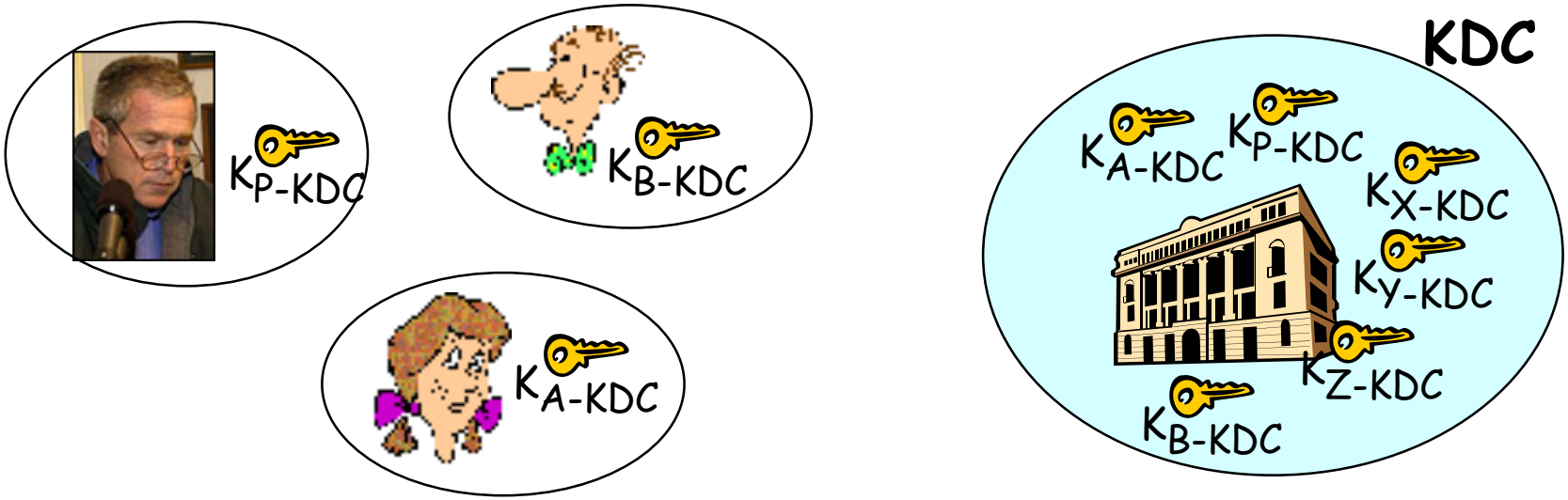
- Cuando Alicia obtiene la clave pública de Roberto (de un sitio web, por correo electrónico, en una memoria USB), ¿cómo puede saber que es la clave pública de Roberto y no la de Gertrudis?

Solución:

- **Autoridad de certificación de confianza (CA).**

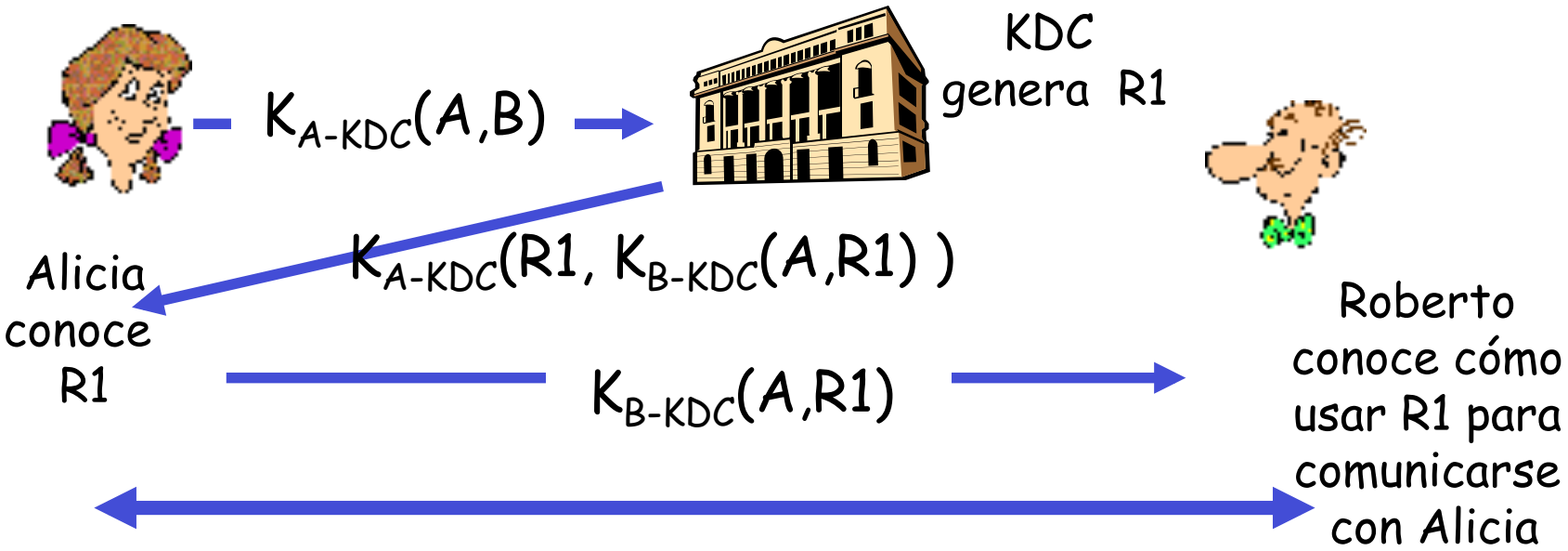
Centro de distribución de claves (KDC)

- Alicia y Roberto necesitan una clave simétrica compartida.
- KDC: servidor que comparte diferentes claves secretas con cada usuario registrado (muchos usuarios).
  - Alicia y Roberto conocen sus claves simétricas, K_{A-KDC}, K_{B-KDC}, para comunicarse con KDC.



Centro de distribución de claves (KDC)

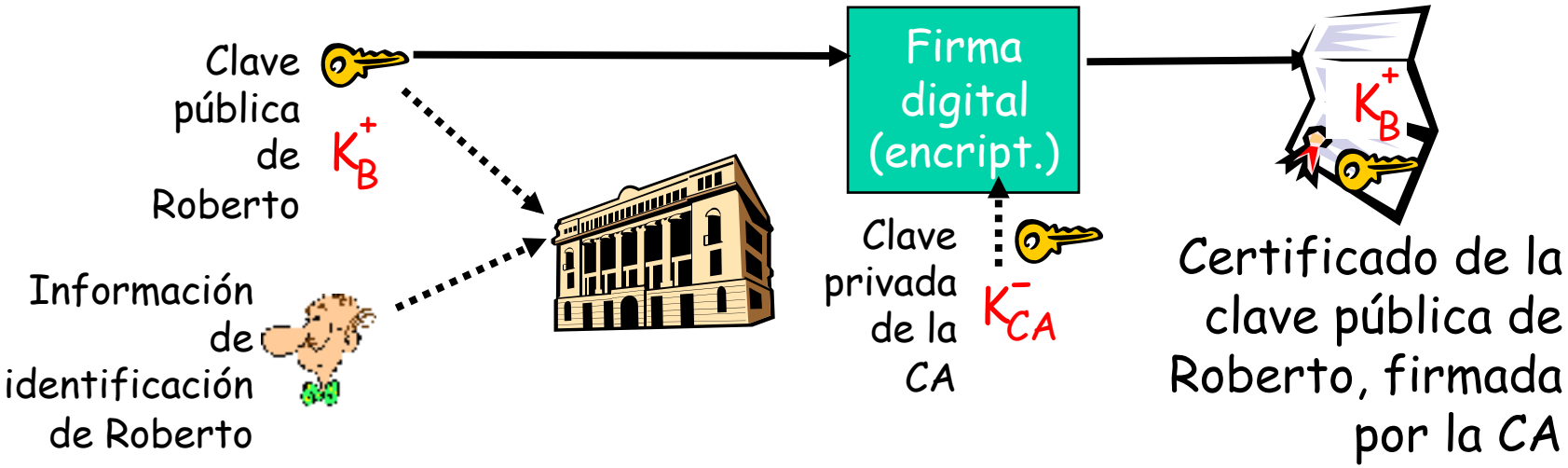
- ¿Cómo permite el KDC a Roberto que Alicia determine la clave simétrica compartida para comunicarse entre sí?
  - Alicia le dice al KDC que quiere comunicarse con Roberto
  - El KDC le genera una clave (R1) para comunicarse con Roberto, y le pasa también esa clave cifrada con la clave con la que Roberto se comunica con el KDC.



Alicia y Roberto se comunican: utilizan R1 como clave de sesión compartida para el cifrado simétrico.

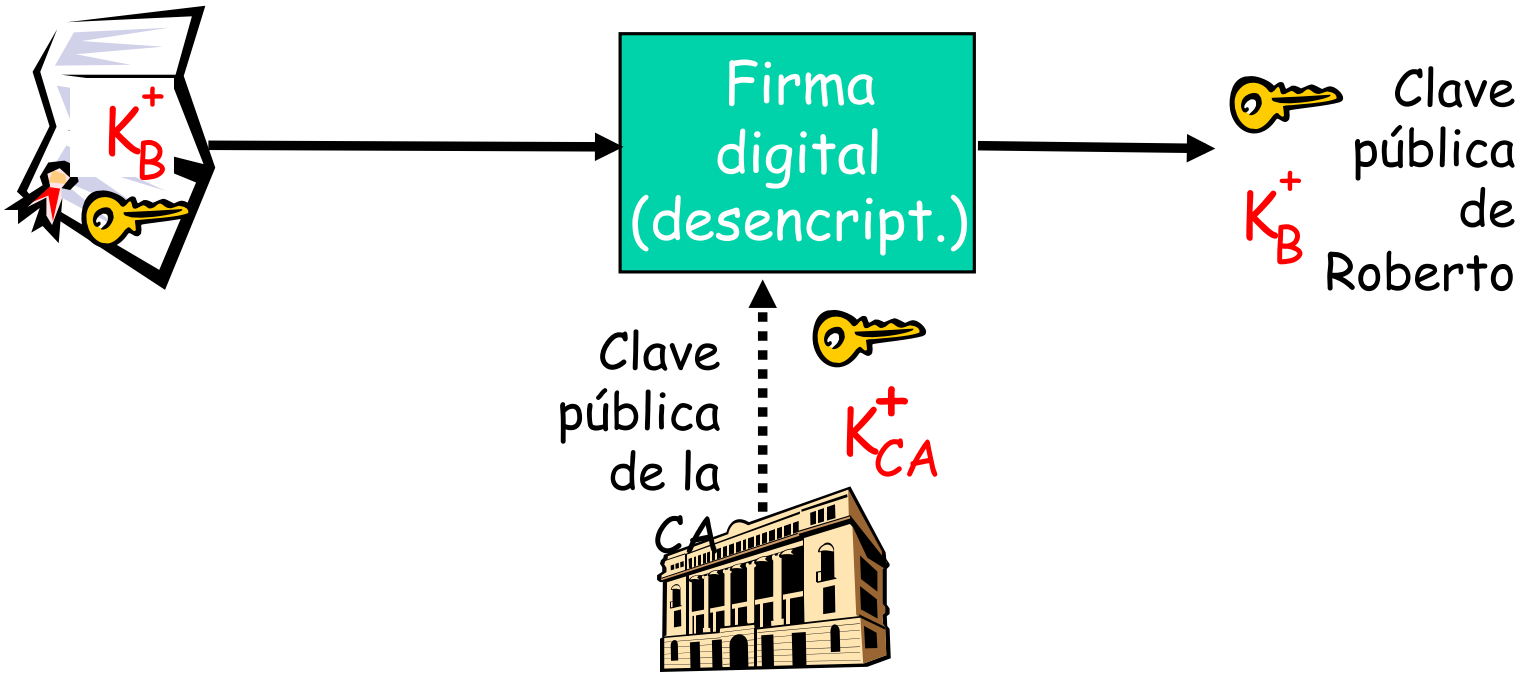
Autoridades de certificación

- Autoridad de certificación(CA): vincula una clave pública a una entidad particular, E.
- E (persona, *router*,...) registra su clave pública con CA:
  - E proporciona una "prueba de identidad" a CA.
  - CA crea un certificado que vincula a E con su clave pública.
  - El certificado contiene la clave pública de E firmada digitalmente por CA. CA dice "Ésta es la clave pública de E".

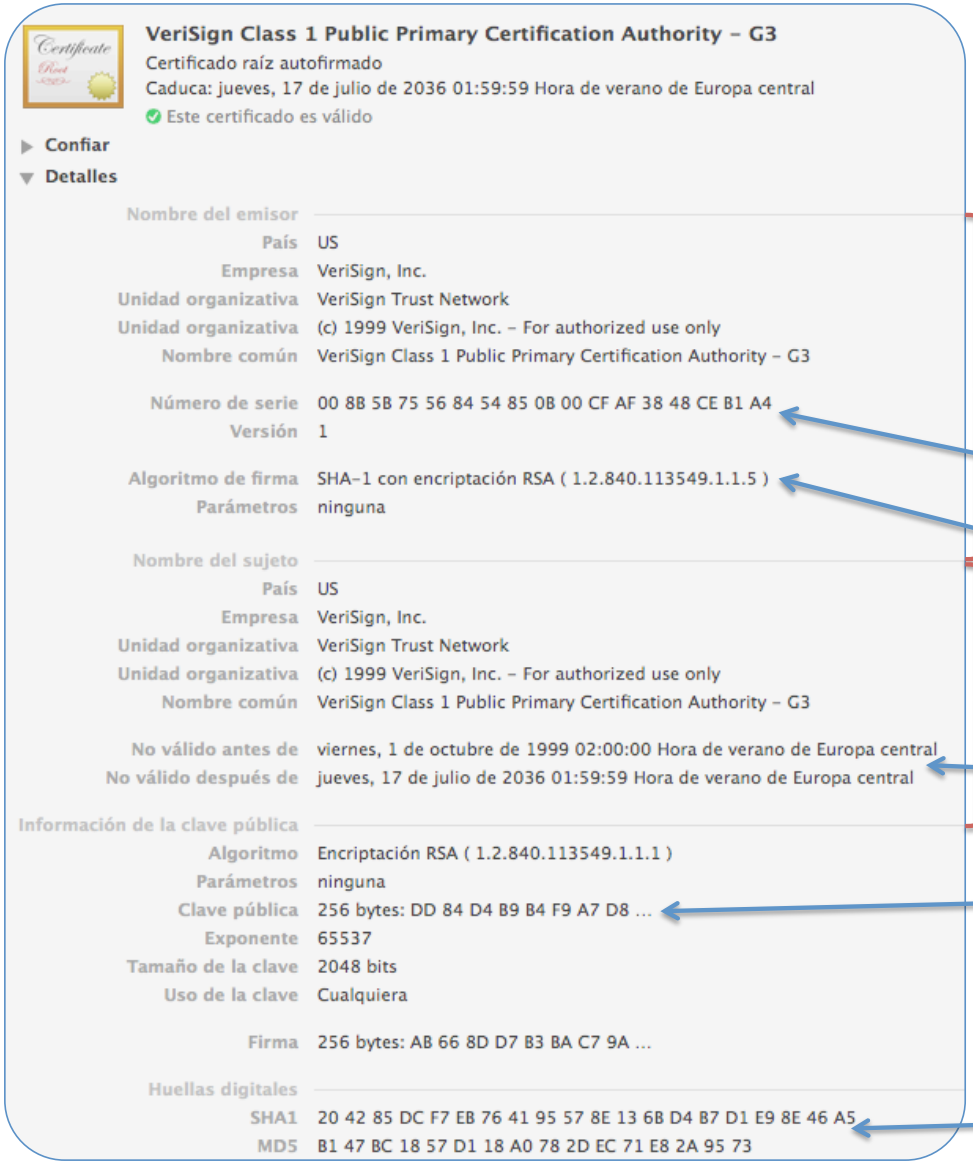


Autoridades de certificación

- Cuando Alicia quiere la clave pública de Roberto:
  - Obtiene el certificado de Roberto (de Roberto o de cualquiera).
  - Aplica la clave pública de CA al certificado de Roberto, obteniendo la clave pública de Roberto.



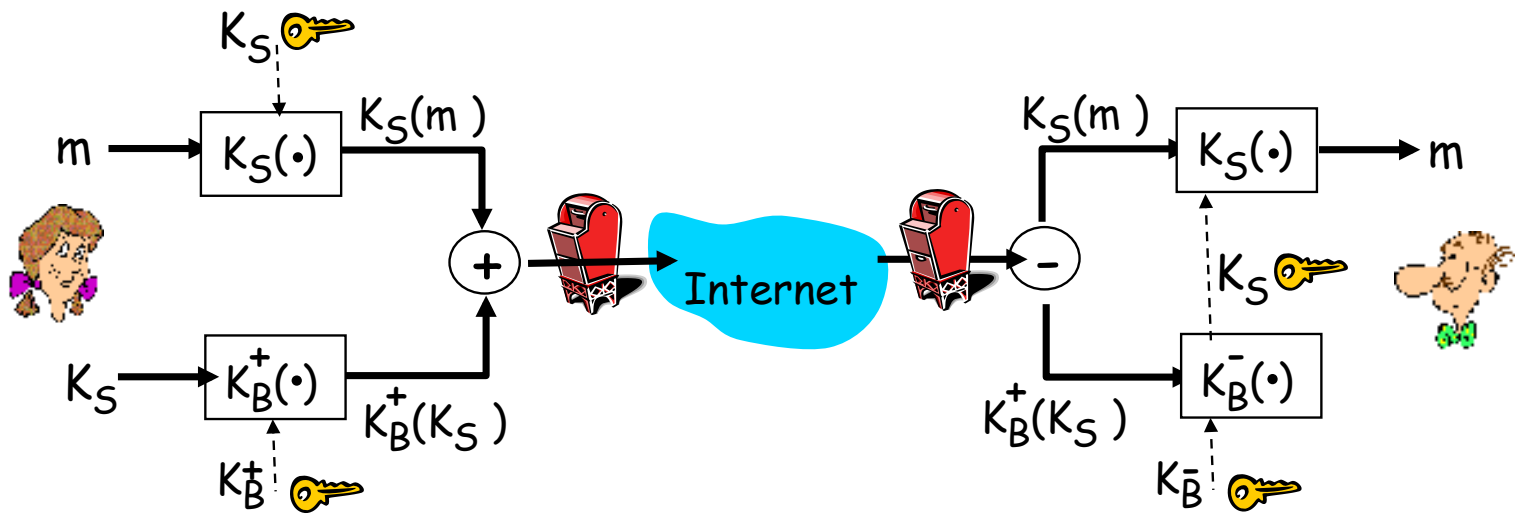
Certificado digital



Contenidos

Correo electrónico confidencial: en emisión

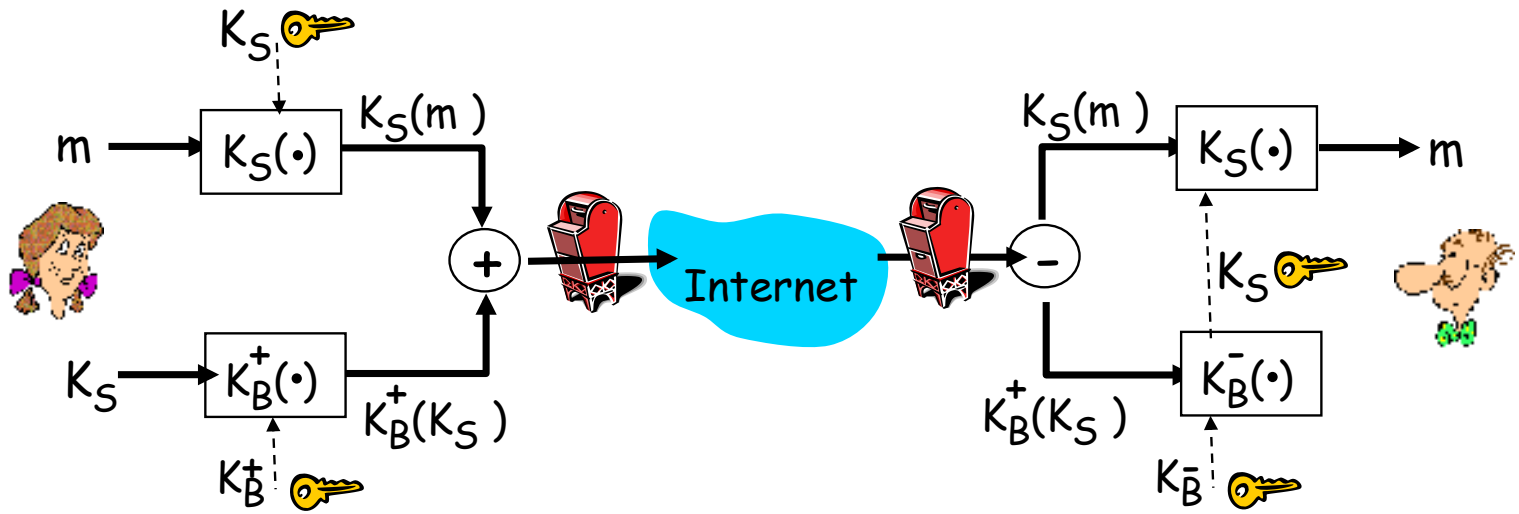
- Alicia quiere enviar un correo confidencial, m, a Roberto.



- Alicia:
  - Genera una clave de sesión aleatoria (clave simétrica), K_S.
  - Cifra el mensaje con K_S (por eficiencia)
  - Cifra K_S con la clave pública de Roberto.
  - Envía juntos $K_S(m)$ y $K_B^+(K_S)$ a Roberto.

Correo electrónico confidencial: en recepción

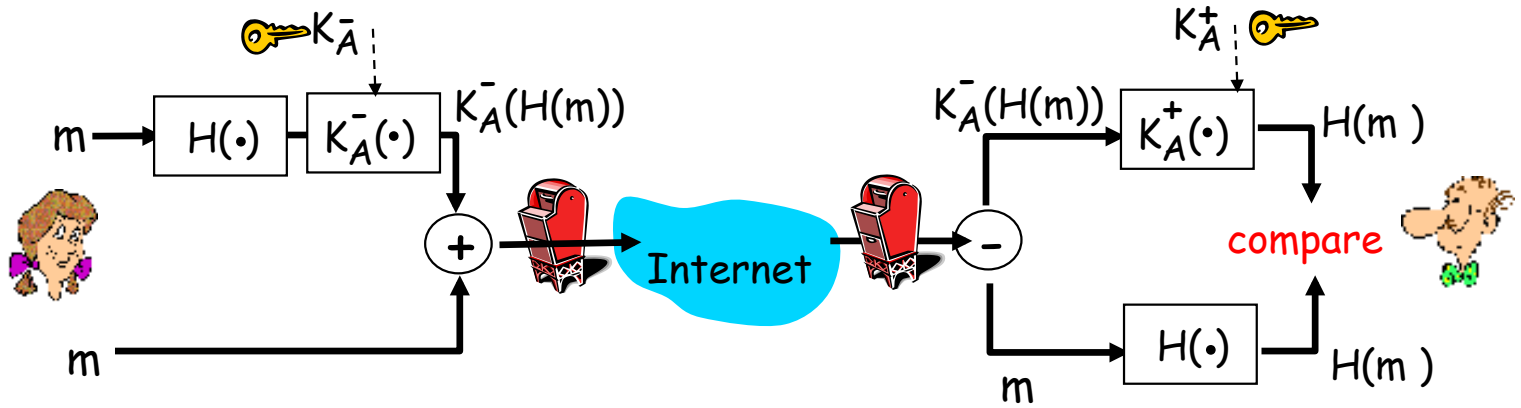
- Alicia quiere enviar un correo confidencial, m, a Roberto.



- Roberto:
  - Utiliza su clave privada para descifrar y recuperar K_S.
  - Utiliza K_S para descifrar $K_S(m)$ y recuperar m.

Correo electrónico íntegro y auténtico

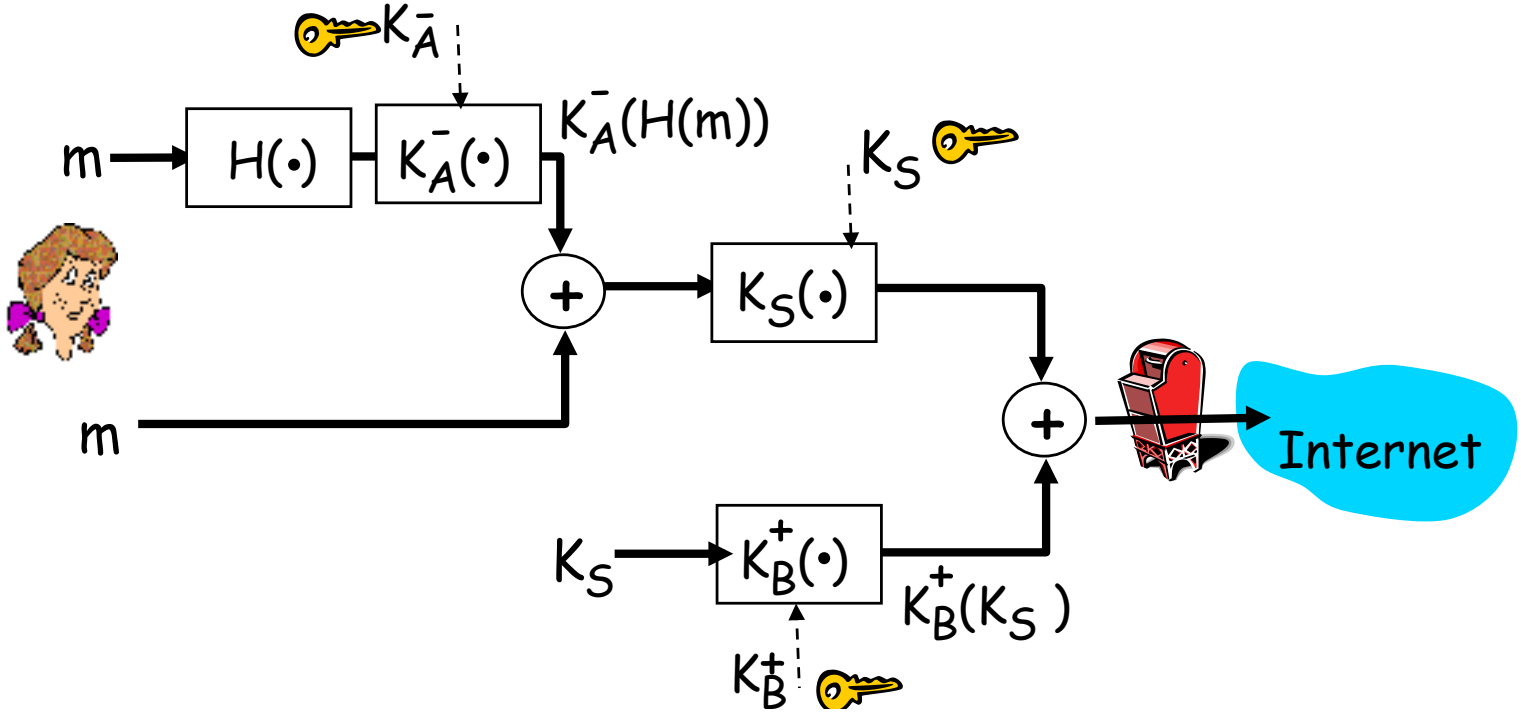
- Alicia quiere proporcionar integridad y autenticación del mensaje, m, a Roberto.



- Alicia:
  - Firma el mensaje digitalmente (es decir, obtiene un resumen del mensaje $H(m)$, y cifra ese resumen con su clave privada)
  - Envía juntos el mensaje en claro y la firma digital.

Correo electrónico confidencial, íntegro y auténtico

- Alicia quiere proporcionar confidencialidad, integridad y autenticación del mensaje, m, a Roberto.



- Alicia:
  - Primero firma digitalmente y luego cifra mensaje y firma con K_S.
  - Envía juntos el mensaje firmado cifrado y la clave de sesión cifrada.
- Alcia utiliza 3 claves: clave privada de Alicia, clave pública de Roberto y clave simétrica.

Pretty Good Privacy (PGP) (I)

- PGP es un sistema de cifrado para el correo electrónico que se ha convertido en el estándar de facto.
- Proporciona confidencialidad, autenticación del emisor e integridad.
- Su inventor, Phil Zimmerman, fue investigado por el gobierno de EE.UU por exportar algoritmos criptográficos fuera del país.

Un mensaje PGP firmado:

```
---BEGIN PGP SIGNED MESSAGE---
Hash: SHA1

Roberto:Puedo verte esta noche?.
Apasionadamente tuya, Alicia

---BEGIN PGP SIGNATURE---
Version: PGP 5.0
Charset: noconv
yhHJRHhGJGhgg/12EpJ+lo8gE4vB3mqJhFEvZP9t6n7G6m5Gw2
---END PGP SIGNATURE--
```

Un mensaje PGP cifrado y firmado:

```
---BEGIN PGP MESSAGE---
Version: PGP 5.0
hQE0AycLXbY/jTXaEAP+00X2Fdko5YzAa094x949b1pPw4rHG/
yd2M7+oNH86acvya749Lwf3p+ylIh1EYKsGvFS5G9/9Wm5hPA1
FG+S10RQbi0RJCaskT
=Ky7U
---END PGP MESSAGE---
```

Pretty Good Privacy (PGP) (II)

- PGP utiliza:
  - compresión del mensaje a enviar
  - criptografía de clave simétrica con claves de sesión para cifrar los mensajes
  - criptografía de clave pública para autenticación y para cifrar las claves de sesión
  - función *hash* (firma digital) para autenticación e integridad
- PGP puede usarse también para cifrar ficheros y guardarlos cifrados en el sistema de ficheros, aunque no se vayan a enviar por correo electrónico.
- OpenPGP: Estándar internacional en desarrollo por la IETF (actualmente: RFC 4880), intentando evitar los algoritmos sometidos a restricciones de licencias y/o exportación.
- GnuPG *(GNU Privacy Guard)*: Implementación libre de OpenPGP. Es un paquete instalable en la mayoría de sistemas operativos que permite integrarse con los programas de correo más habituales.

Contenidos

SSL (I)

- Proporciona seguridad en la capa de transporte a cualquier aplicación basada en TCP
- Utilizada sobre todo entre navegadores de Internet y servidores de comercio electrónico (https://)
- También puede usarse para que un cliente de correo electrónico recupere/envíe mensajes de correo del/al servidor de correo por IMAP en una sesión cifrada.
- Servicios de seguridad:
  - Autenticación de servidor.
  - Cifrado de datos.
  - Autenticación de cliente (opcional).
- Desarrollado originalmente por Netscape.
- TLS *(Transport Layer Security)*: Protocolo estandarizado por IETF basado en SSL (RFC 4346).

SSL (II)

- Autenticación de servidor:
  - El navegador tiene almacenadas las claves públicas de autoridades de certificación de confianza: K^+_{CA1}, K^+_{CA2}...
  - El navegador solicita el certificado del servidor (cifrado con K^-_{CA1}, p.ej.) emitido por las autoridades de certificación de confianza.
  - El navegador utiliza la clave pública del CA (K^+_{CA1}) para extraer del certificado la clave pública del servidor (K^+_{server}).
- Autenticación de cliente:
  - El cliente presenta un certificado que el servidor comprueba que ha sido emitido por una CA.
  - Es opcional, se utiliza sólo en algunos casos (ej: IRPF, tramitaciones ante las Administraciones Públicas).
- En algunos navegadores puedes ver en el menú de seguridad los CA de confianza.

SSL(III)

Sesión SSL cifrada:

- El navegador genera clave de sesión simétrica, la cifra con la clave pública del servidor, envía la clave cifrada al servidor.
- Con su clave privada el servidor descifra la clave de sesión.
- El navegador y el servidor conocen la clave de sesión:
  - Todos los datos enviados al socket TCP (por cliente o servidor) son cifrados con la clave de sesión.

Contenidos

SSH (I)

- Protocolo para intercambiar datos a través de un canal seguro entre dos ordenadores conectados a una red.
- Modelo cliente/servidor sobre TCP. El servidor normalmente espera conexiones TCP en el puerto 22.
- Se utiliza, entre otras cosas, para:
  - Entrar en una máquina remota, reemplazando a `telnet` o `rlogin`.
  - Ejecutar comandos de *shell* en una máquina remota (reemplazando a `rsh`).
  - Transmitir de forma segura la salida una aplicación gráfica X Window entre ordenadores
  - Transmitir ficheros de forma segura, usando uno de estos protocolos asociados:
    - SCP *(Secure Copy)*: interfaz de comandos, sólo permite copiar ficheros.
    - SFTP *(SSH File Transfer Protocol)*: permite también hacer listados de directorios, ver y cambiar permisos de archivos... por lo que suele permitir realizar clientes gráficos.

SSH (II)

- Actualmente se utiliza la versión 2 del protocolo (RFC 4251).
- OpenSSH: Implementación libre disponible para la mayoría de sistemas operativos.
- Sesión en una máquina remota:
  1. Se trata de autenticar que la máquina remota no ha sido suplantada:
     - el usuario almacena en su cuenta las claves públicas de las máquinas a las que ya se ha conectado alguna vez
     - cuando vuelve a conectarse a una máquina, se comprueba que no haya sido suplantada (a veces simplemente ha sido reinstalada)
  2. Se autentica al usuario que accede:
     - Tiene que tener cuenta en la máquina remota
     - Puede autenticarse de varias formas, las más habituales son mediante su **contraseña** o mediante un **clave pública**.
  3. Todo el tráfico después del acceso se cifra con una clave de sesión simétrica.

Contenidos

Referencias

- James F. Kurose y Keith W. Ross, **Redes de Computadores: un enfoque descendente**, Pearson Educación, 5ª edición, capítulo 8