

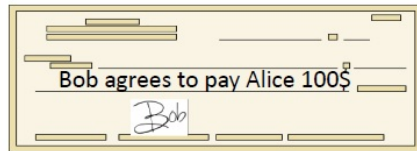
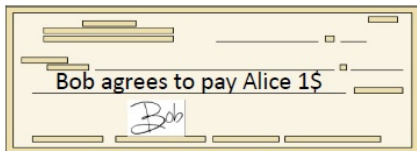
## **FIRMAS DIGITALES**

ALAN REYES-FIGUEROA

CRİPTOGRAFÍA Y CIFRADO DE INFORMACIÓN (AULA 20) 04.NOVIEMBRE.2021

# Firmas Digitales

**Objetivo:** vincular el documento a su autor.

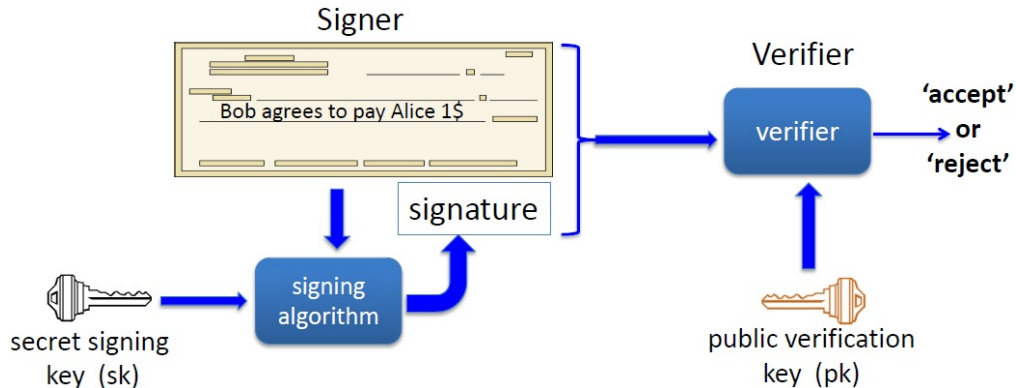


Problema en el mundo digital:

- cualquiera puede copiar la firma de Bob, y utilizarla en un documento que no corresponde (pasar la firma de un documento a otro).

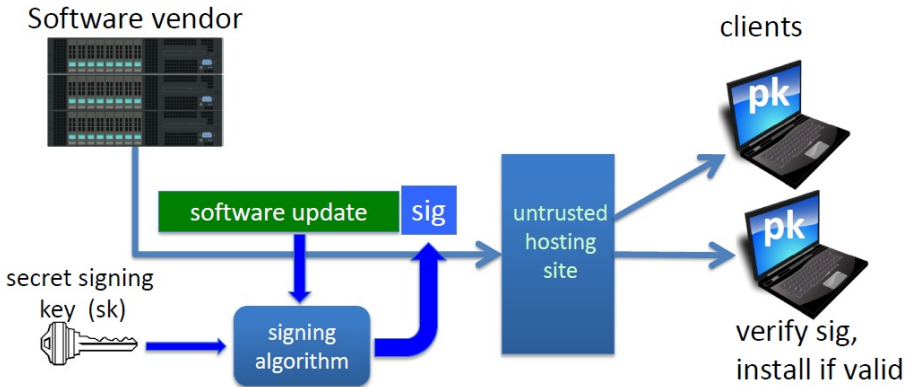
# Firmas Digitales

**Solución:** Hacer que la firma dependa de cada documento.



# Firmas Digitales

**Un Ejemplo:** Comunicación desde un servidor a diferentes clientes, a través de un sitio host que no es confiable.



## Firmas Digitales: Sintaxis.

### Definición

Un esquema de **firma digital** es una tripla  $(Gen, S, V)$  tripla de algoritmos, donde

- $Gen()$ : es un algoritmo aleatorio. Emite un par de claves  $(\mathbf{p}_k, \mathbf{s}_k)$ .
- $S : \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{S}$ , el cual recibe un par  $(\mathbf{s}_k, \mathbf{m}) \in \mathcal{K} \times \mathcal{M}$ , y produce como salida una firma  $\sigma$ .
- $V : \mathcal{K} \times \mathcal{K} \times \mathcal{S} \rightarrow \{0, 1\}$ , la cual recibe una tripla  $(\mathbf{p}_k, \mathbf{m}, \sigma)$  y genera “aceptar” o “rechazar”.

Como es usual, requerimos que se satisfaga la condición de consistencia:

$$V(\mathbf{p}_k, \mathbf{m}, S(\mathbf{s}_k, \mathbf{m})) = 1, \quad \forall \mathbf{m} \in \mathcal{M}.$$

# Seguridad en Firmas Digitales

## Seguridad en Firmas Digitales:

Para el caso de firmas digitales definimos el poder o potencial de un atacante, de la siguiente forma: (*ataque por elección*)

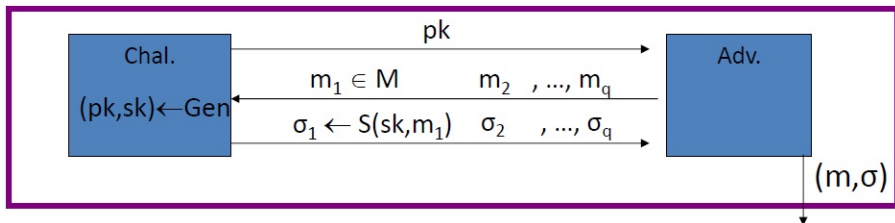
- Para  $\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_q$  mensajes, el atacante recibe  $\sigma_i = S(\mathbf{s}_k, \mathbf{m}_i)$ .
- **Objetivo del atacante:** producir una falsificación existencial (*existential forgery*). Recordemos que esto significa que el atacante puede producir un nuevo par (mensaje, signature)  $(\mathbf{m}, \sigma)$  que es válido.

$$V(\mathbf{m}, \sigma) = 1, \text{ con } \mathbf{m} \notin \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_q\}.$$

**Obs!** Queremos que el atacante no pueda producir una firma válida, para un nuevo mensaje.

# Seguridad en Firmas Digitales

Para un esquema de firma digital  $(Gen, S, V)$  y un adversario o algoritmo inteligente  $\mathcal{A}$ , definimos el siguiente experimento:



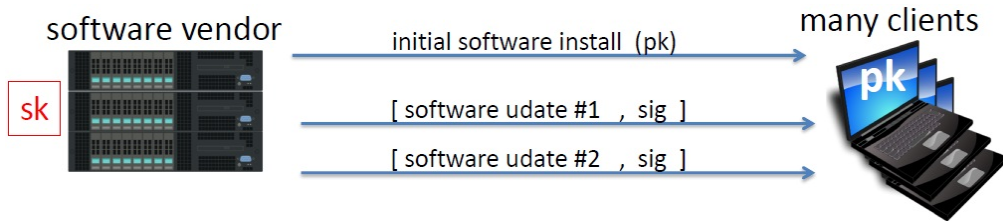
En este caso, definimos

## Definición

La firma digital  $S = (Gen, S, V)$  es **segura** si para todo algoritmo eficiente  $\mathcal{A}$ , vale

$$Adv_{Sig}(\mathcal{A}, S) = \mathbb{P}(\mathcal{A} \text{ gana}) < \varepsilon, \text{ con } \varepsilon \text{ negligible.}$$

## Firmas de código:



- El proveedor de software firma el código.
- Los clientes tienen la clave pública  $p_k$  del proveedor.
- El software se instala si se verifica la firma.

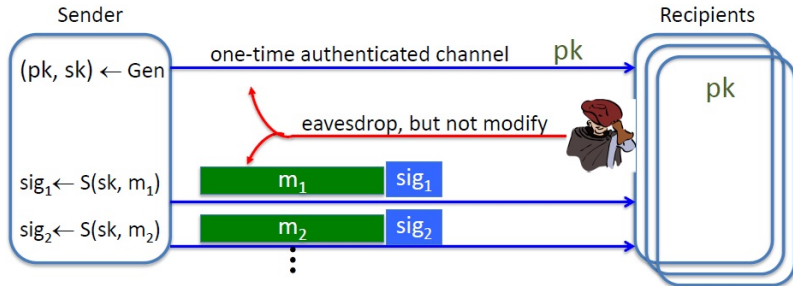


# Aplicaciones

Más generalmente:

Canal autenticado por única vez (no privado, unidireccional)

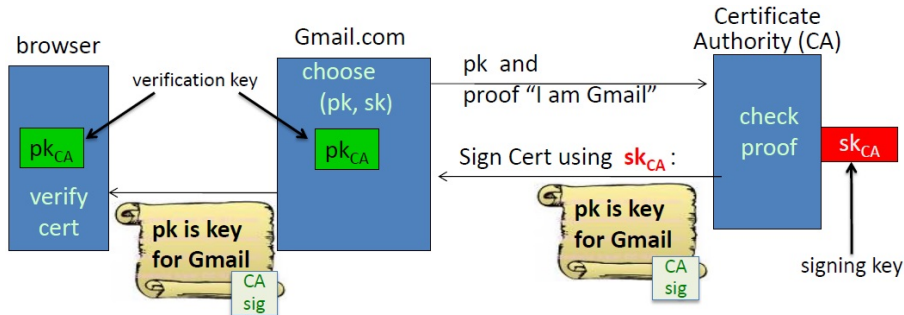
La instalación inicial del software está autenticada, pero no es privada.



# Aplicaciones

## Certificados digitales:

- Problema: el navegador necesita la clave pública  $p_k$  del servidor para configurar una clave de sesión.
- Solución: el servidor solicita a un tercero de confianza (CA) que firme su  $p_k$ .




## Important fields:

|                     |   |   |
|---------------------|---|---|
| Serial Number       | 5814744488373890497                                       | ← |
| Version             | 3   |   |
| Signature Algorithm | SHA-1 with RSA Encryption ( 1.2.840.113549.1.1.5 )        |   |
| Parameters          | none  |   |
| Not Valid Before    | Wednesday, July 31, 2013 4:59:24 AM Pacific Daylight Time |   |
| Not Valid After     | Thursday, July 31, 2014 4:59:24 AM Pacific Daylight Time  |   |
| Public Key Info     |   |   |
| Algorithm           | Elliptic Curve Public Key ( 1.2.840.10045.2.1 )           |   |
| Parameters          | Elliptic Curve secp256r1 ( 1.2.840.10045.3.1.7 )          |   |
| Public Key          | 65 bytes : 04 71 6C DD E0 0A C9 76 ...                    | ← |
| Key Size            | 256 bits  |   |
| Key Usage           | Encrypt, Verify, Derive                                   |   |
| Signature           | 256 bytes : 8A 38 FE D6 F5 E7 F6 59 ...                   | ← |

Equifax Secure Certificate Authority  
↳ GeoTrust Global CA  
↳ Google Internet Authority G2  
↳ mail.google.com

---

 **mail.google.com**  
Issued by: Google Internet Authority G2  
Expires: Thursday, July 31, 2014 4:59:24 AM Pacific Daylight Time  
✔ This certificate is valid

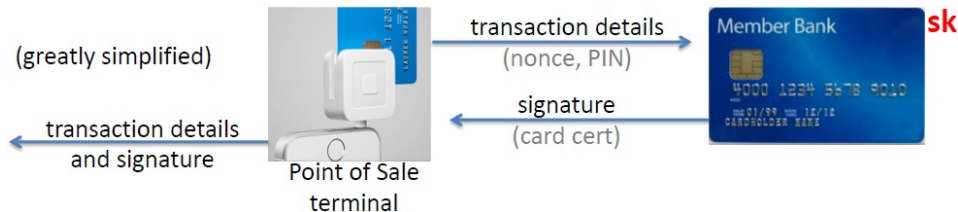
▼ Details

Subject Name  
Country US  
State/Province California  
Locality Mountain View  
Organization Google Inc  
Common Name mail.google.com ←

Issuer Name  
Country US  
Organization Google Inc  
Common Name Google Internet Authority G2

El servidor usa el certificado durante un período prolongado (por ejemplo, un año).

## Pagos Electrónicos (EMV, *EuroPay*, *Mastercard* and *Visa*):



## Correo electrónico firmado:

- El remitente firma el correo electrónico que envía a los destinatarios.
- Cada destinatario tiene la clave pública (y el certificado) del remitente.
- Un destinatario acepta el correo electrónico entrante sólo si se verifica la firma.

# Aplicaciones

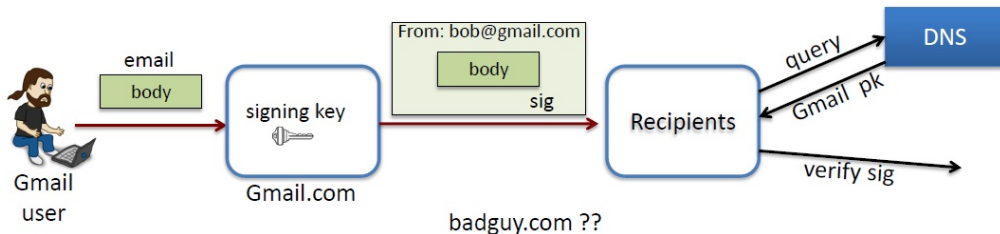
## Correo electrónico firmado: (DKIM, *Domain Key Identified Mail*)

Problema: correo electrónico incorrecto que dice ser de `someuser@gmail.com`.

Pero, en realidad, el correo proviene del dominio `badguy.com`.

incorrectamente hace que `gmail.com` parezca una mala fuente de correo electrónico.

Solución: `gmail.com` (y otros sitios) firman cada correo saliente.



**Ejemplo:** Encabezado DKIM empleado por gmail.com.

X-Google-DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;  
d=1e100.net; s=20130820; (lookup 20130820.\_domainkey.1e100.net in DNS for public key)  
h=x-gm-message-state:mime-version:in-reply-to:references:from:date:  
message-id:subject:to:content-type;  
bh=MDr/xwte+/JQSgCG+T2R2Uy+SuTK4/gxqdxMc273hPQ=; (hash of message body)

b=dOTpUVOaCrWS6AzmcPMreo09G9viS+sn1z6g+GpC/ArkfMEmcffOJ1s9u5Xa5KC+6K  
XRzwZhAWYqFr2a0ywCjbGECBPIE5ccOi9DwMjnvJRYEwNk7/sMzFfx+0L3nTqgTyd0ED  
EGWdN3upzSXwBrXo82wVcRRcNq1yUITddnHgEoEFg5WV37DRP/eq/hOB6zFNTRBwkvfS  
0tC/DNdRwftspO+UboRU2eiWaqJWPjxL/abS7xA/q1VGz0ZoI0y3/SCkxdg4H80c61DU  
jdVYhCUd+dSV5fISouLQT/q5DYEjINQbi+EcbL00liu4o623SDEeyx2isUgcvi2VxTWQ  
m80Q==

Gmail's signature on headers, including DKIM header (2048 bits)

## Resumen de aplicaciones:

- Firmado de código,
- Certificados digitales,
- Email firmado (*e.g.* DKIM),
- Pagos con tarjeta de crédito: EMV,
- ...

# Firmas Digitales vs. Códigos Autenticadores

## ¿Cuándo usar firmas digitales?:

Generalmente hablando:

- Si una de las partes firma y una de las partes verifica: **usar un MAC**.
  - A menudo requiere interacción para generar una clave compartida.
  - El destinatario puede modificar los datos y volver a firmarlos antes de pasar los datos a un tercero.
- Si una de las partes firma y muchas verifican: **usar una firma digital**.
  - Los destinatarios no pueden modificar los datos recibidos antes pasar datos a un tercero (no repudio).



# Firmas Digitales vs. Códigos Autenticadores

## Resumen de esquemas para verificar integridad:

1. Hash resistente a colisiones: Necesita un espacio público de sólo lectura.



Software  
Vendor

Small read-only  
public space

2. Firmas digitales:
  - El proveedor debe administrar una clave secreta a largo plazo,
  - La firma del proveedor en el software se envía con el software.
  - El software se puede descargar desde un sitio de distribución que no sea de confianza.
3. MAC: El proveedor debe calcular una nueva MAC de software para cada cliente, y debe administrar una clave secreta a largo plazo (para generar una clave MAC por cliente).