Seguridad de la Información

Una introducción con enfoque práctico

Ing. Mariano Aliaga

Universidad Católica de Córdoba - Facultad de Ingeniería

2021

Panorama General

1. Firma digital
   - Definición y conceptos
   - Procedimientos de firma digital
2. Certificados digitales
3. Herramientas
   - GnuPG

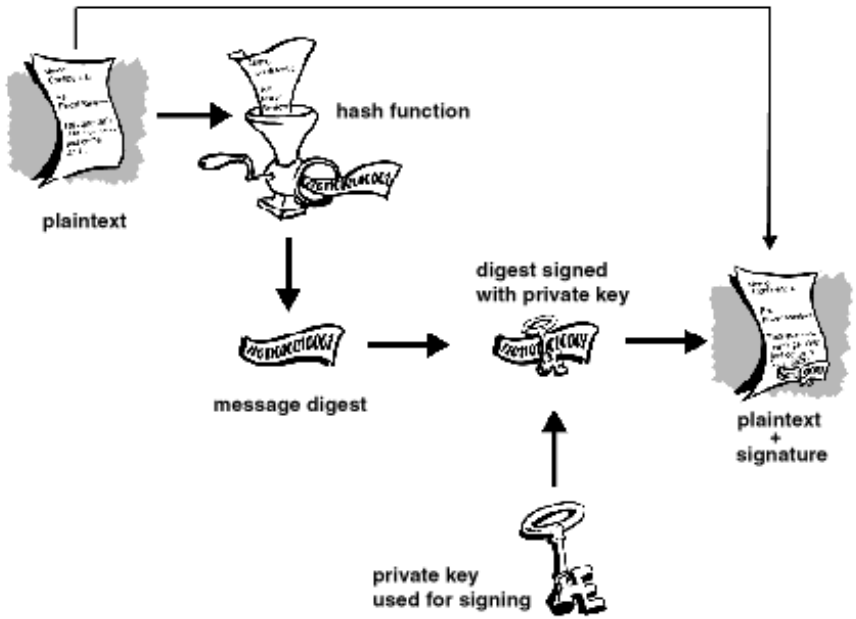
Firma Digital

Firma digital: es un esquema criptográfico que sirve para demostrar la autenticidad de un mensaje digital o documento electrónico.

- Asegura la **integridad** del documento.
- Asegura la **identidad** del emisor.
- Garantiza el **no-repudio**.

Procedimiento emisor

1. Se calcula mediante una función de hash el resumen del documento digital.
2. Este resumen se cifra con la llave privada del emisor.
3. El resultado es la Firma Digital, y ésta se adjunta al documento produciendo el documento firmado digitalmente.



Procedimiento receptor

1. Separamos el documento digital de la firma.
2. Calculamos nuevamente el resumen del documento digital con el mismo algoritmo empleado por el emisor.
3. Desciframos la firma digital empleando la llave pública del emisor para extraer el resumen calculado originalmente .
4. Comparamos ambos resúmenes.

Algoritmos

- **DSA (Digital Signature Algorithm):** basado en el esquema de firma de Elgamal, este algoritmo, junto con SHA, es el núcleo del DSS (Digital Signature Standard) y sólo se utiliza para firmar digitalmente un mensaje, no para cifrarlo. Además provee la capacidad de verificar una firma digital. Está patentado pero es de libre uso.
- **ECDSA (Elliptic Curve DSA):** es una variante del algoritmo DSA pero basado en el problema de las curvas elípticas. Esta variante requiere tamaños de llaves de menor longitud para ofrecer el mismo nivel de seguridad. El tiempo de procesamiento es similar al DSA.

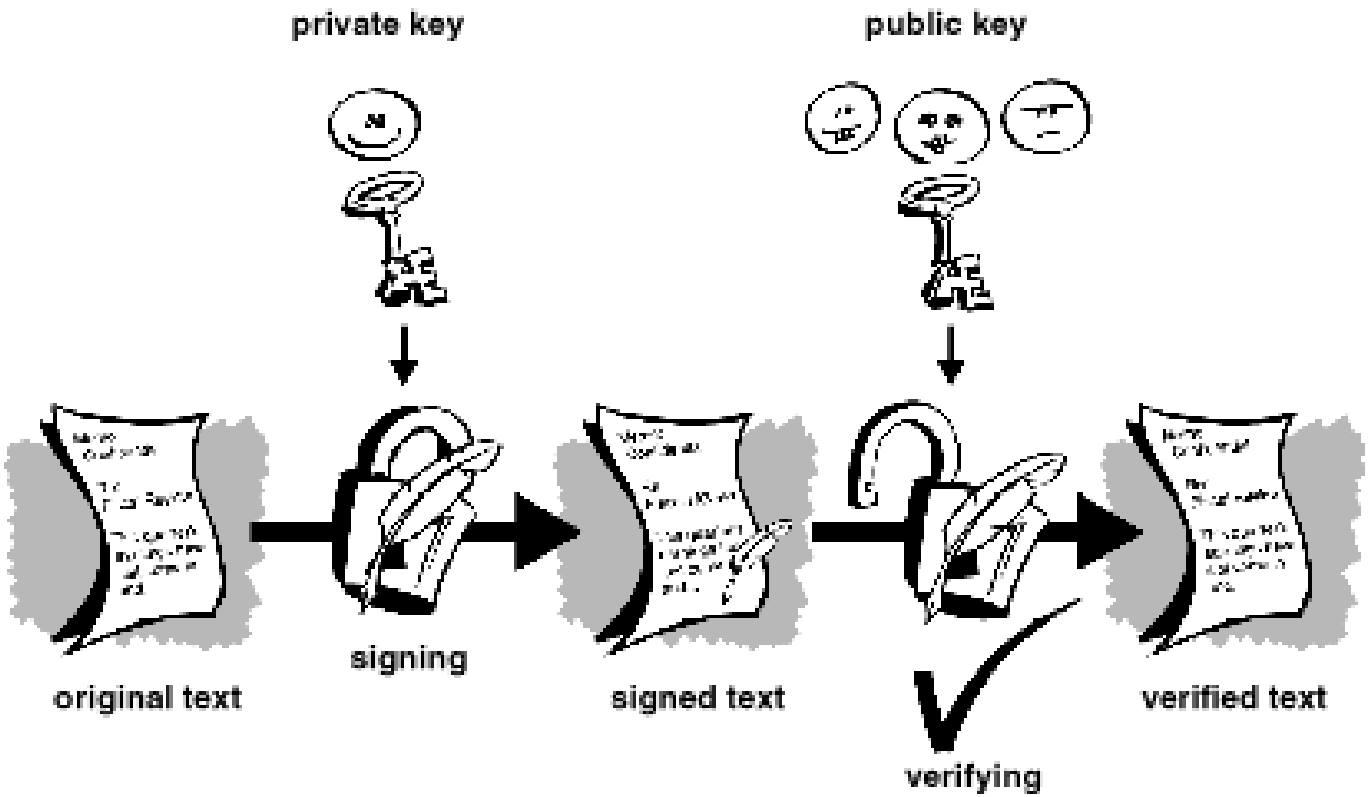
Observaciones

Firmar digitalmente NO es Cifrar

Podemos contar con los siguientes escenarios

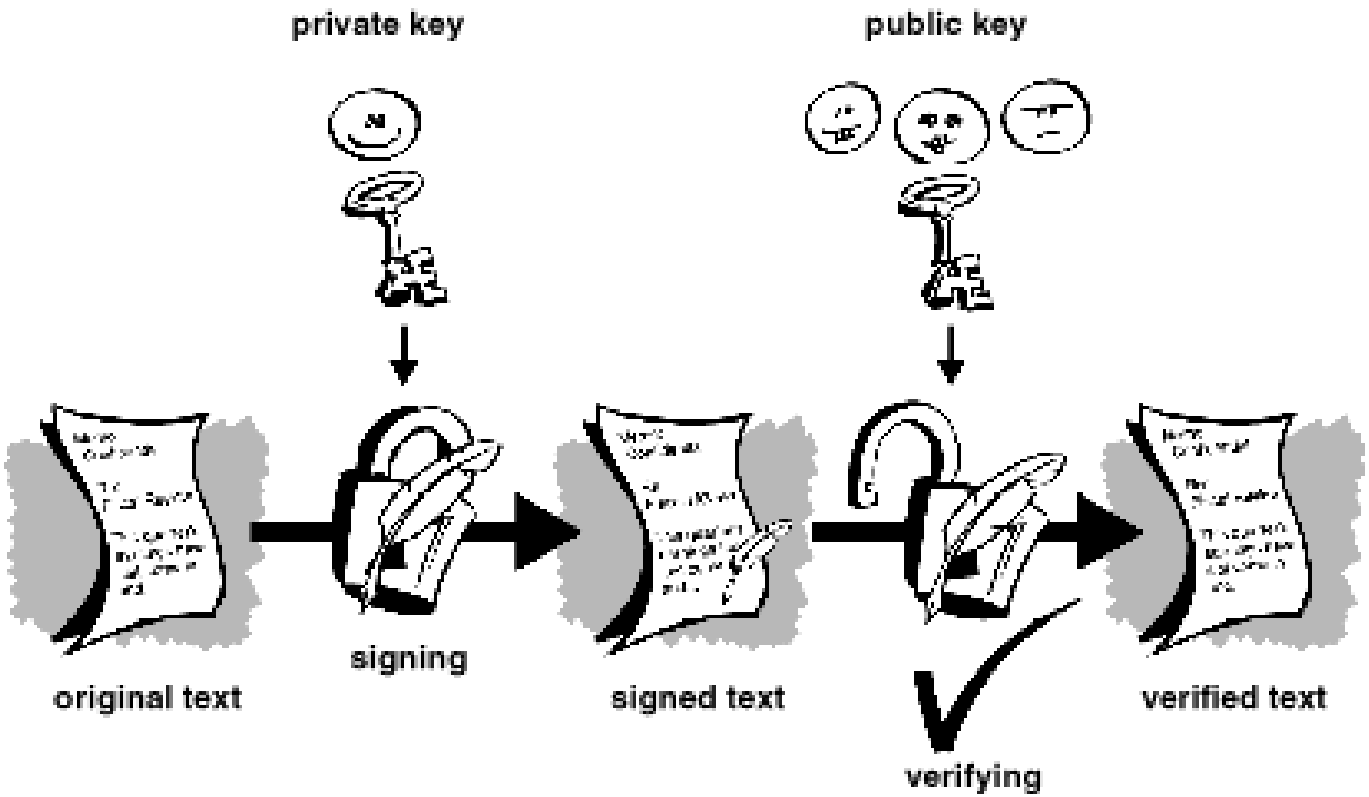
- Un documento puede estar **firmado** digitalmente y **cifrado**.
- Puede estar **firmado** y **no cifrado**.
- Puede estar **cifrado** y **no firmado**.

Problemas llave pública/privada



Susceptible a ataque de hombre en el medio (man in the middle o MITM).

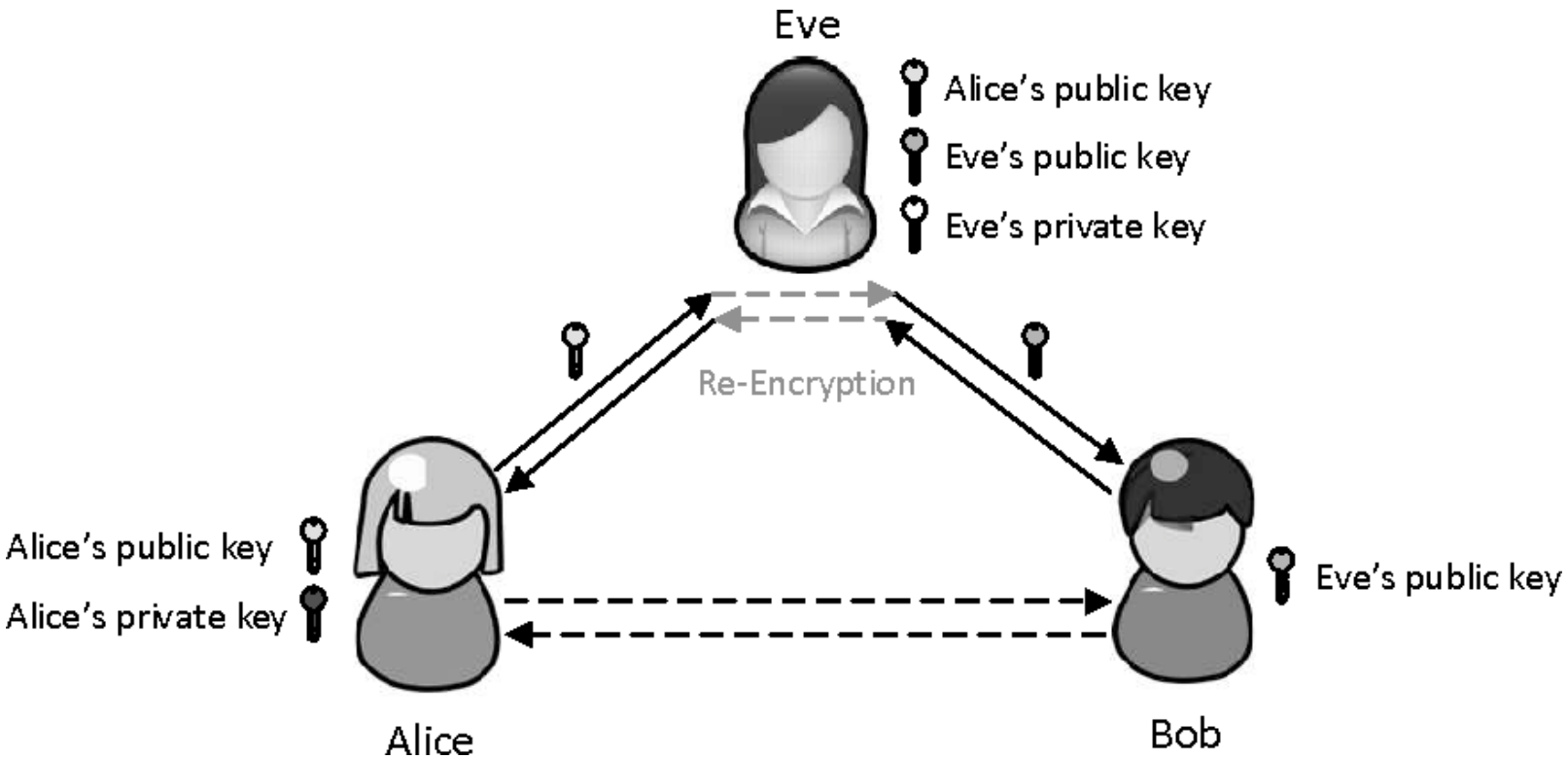
Problemas llave pública/privada



Susceptible a ataque de hombre en el medio (man in the middle o MITM).

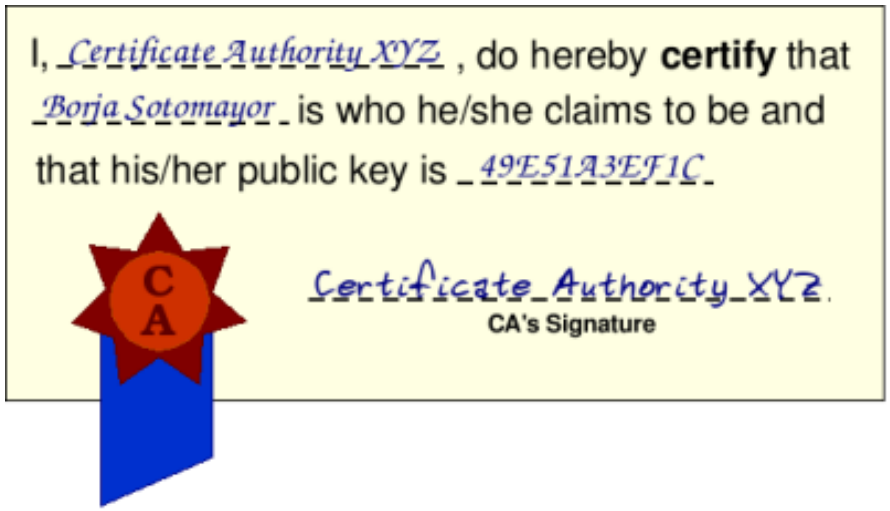
Problemas llave pública/privada

Ataque Man In The Middle (MITM)



Certificados digitales

Certificado digital: es un documento electrónico que usa una firma digital para vincular una llave pública con una identidad (el nombre de una persona, una organización, etc.). El certificado puede utilizarse para verificar que una llave pública pertenece a un individuo.



Autoridad de certificación

Autoridad de certificación (CA): es una entidad u organización que emite certificados digitales de acuerdo a determinadas políticas, procedimientos y algoritmos criptográficos, certificando así la autenticidad y validez de las llaves públicas.

- La CA es un tercero en el que confían tanto el sujeto (dueño) del certificado como quien lo utiliza luego.
- La confianza en la CA se basa en contar con su llave pública, la cual debe ser obtenida de manera segura.
- La llave pública de la CA se suele distribuir como un certificado digital autofirmado o mediante estructuras jerárquicas de autoridades de certificación.

Estructura certificados digitales

X.509: es un estándar de ITU-T para el manejo de una infraestructura de llave pública (PKI).

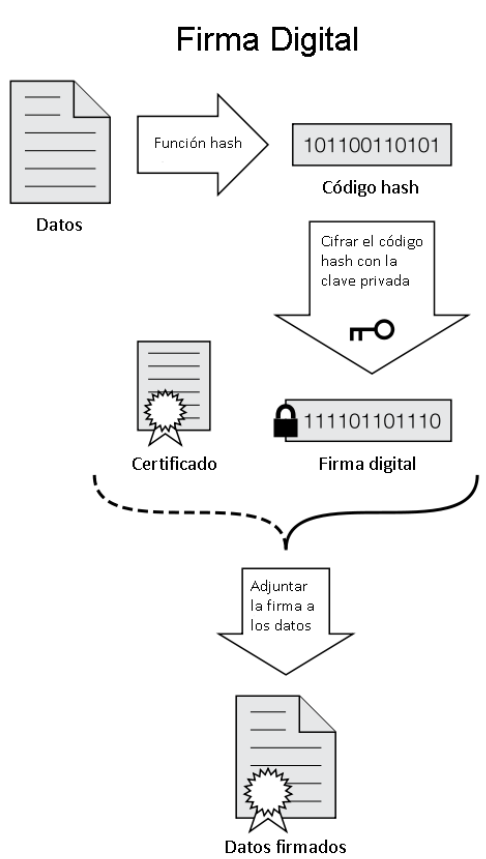
- Certificate
  - Version
  - Serial Number
  - Algorithm ID
  - Issuer
  - Validity
    - Not Before
    - Not After
  - Subject
  - Subject Public Key Info
    - Public Key Algorithm
    - Subject Public Key
  - ...
- Certificate Signature Algorithm
- Certificate Signature

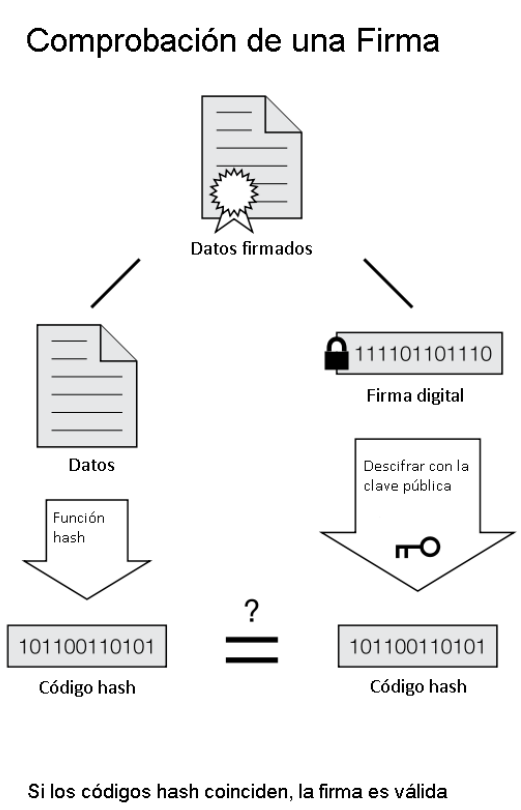
Ejemplo certificado

```
Certificate:
   Data:
       Version: 3 (0x2)
       Serial Number: 1 (0x1)
       Signature Algorithm: md5WithRSAEncryption
       Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Validity
           Not Before: Aug  1 00:00:00 1996 GMT
           Not After : Dec 31 23:59:59 2020 GMT
       Subject: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
                OU=Certification Services Division,
                CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
               Modulus (1024 bit):
                   00:d3:a4:50:6e:c8:ff:56:6b:e6:cf:5d:b6:ea:0c:
                   68:75:47:a2:aa:c2:da:84:25:fc:a8:f4:47:51:da:
                   85:b5:20:74:94:86:1e:0f:75:c9:e9:08:61:f5:06:
                   6d:30:6e:15:19:02:e9:52:c0:62:db:4d:99:9e:e2:
                   6a:0c:44:38:cd:fe:be:e3:64:09:70:c5:fe:b1:6b:
                   29:b6:2f:49:c8:3b:d4:27:04:25:10:97:2f:e7:90:
                   6d:c0:28:42:99:d7:4c:43:de:c3:f5:21:6d:54:9f:
                   5d:c3:58:e1:c0:e4:d9:5b:b0:b8:dc:b4:7b:df:36:
                   3a:c2:b5:66:22:12:d6:87:0d
               Exponent: 65537 (0x10001)
       X509v3 extensions:
           X509v3 Basic Constraints: critical
               CA:TRUE
   Signature Algorithm: md5WithRSAEncryption
       07:fa:4c:69:5c:fb:95:cc:46:ee:85:83:4d:21:30:8e:ca:d9:
       a8:6f:49:1a:e6:da:51:e3:60:70:6c:84:61:11:a1:1a:c8:48:
       3e:59:43:7d:4f:95:3d:a1:8b:b7:0b:62:98:7a:75:8a:dd:88:
```

Usos de certificados digitales

- **Verificación de la firma digital:** el receptor del mensaje verificará el certificado usando la llave pública de la CA, y, teniendo confianza en la llave pública del remitente, verificará la firma del mensaje.



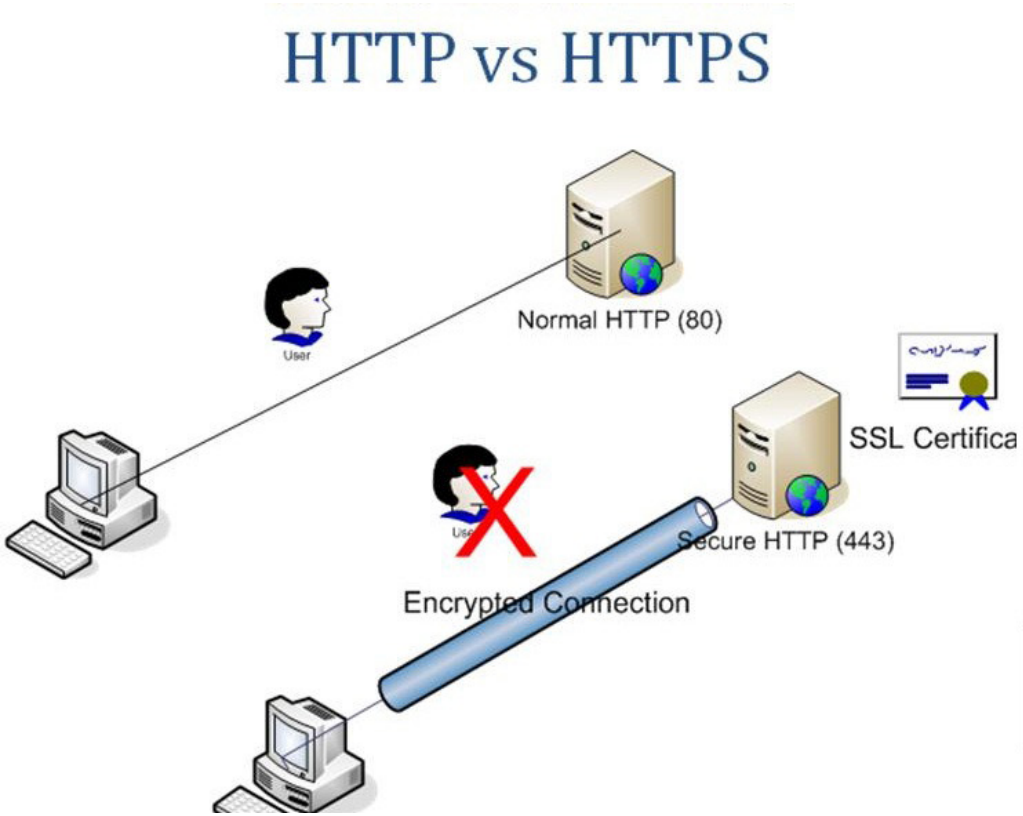


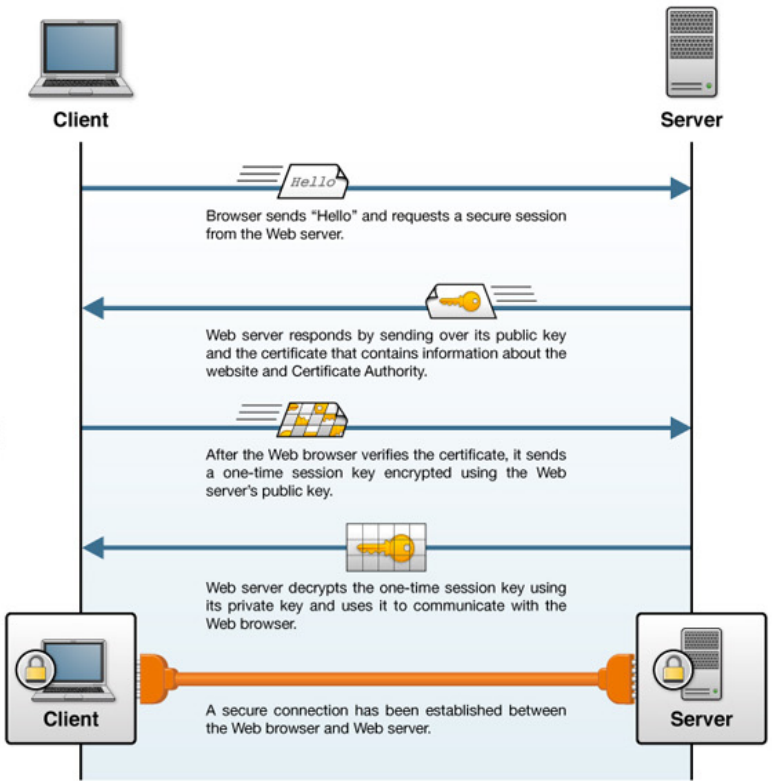
Usos de certificados digitales

- **Autenticación y confidencialidad:** por ejemplo, mediante la instalación de un certificado digital en un servidor web, los clientes que acceden al sitio pueden verificar su autenticidad y cifrar la conexión.

1. El cliente inicia la conexión.
2. El servidor responde con su certificado digital.
3. El cliente verifica la autenticidad del certificado y por ende del servidor.
4. Luego de realizada la autenticación del servidor, el cliente genera una llave de sesión aleatoria y la envía cifrada con la llave pública del servidor.
5. El servidor recibe la llave de sesión cifrada con su llave pública y la descifra empleando su llave privada.
6. Cliente y servidor pueden desde este punto intercambiar información cifrada empleando un algoritmo simétrico. Con ello se asegura la confidencialidad e integridad de la información transmitida.

Usos de certificados digitales





GnuPG

PGP: programa diseñado para proteger la información enviada a través de una red pública mediante el uso de criptografía simétrica y asimétrica.

GnuPG: reemplazo libre de PGP que puede ser utilizado sin restricciones y cumple con el estándar OpenPGP (RFC 2440).

- Proporciona facilidades para la gestión de las llaves públicas y privadas, mediante la utilización de un "anillo de llaves" o keyring.
- Soporta entre otros los siguientes algoritmos: ElGamal, DSA, RSA, AES, 3DES, Blowfish, Twofish, CAST5, MD5, SHA-1, RIPE-MD-160 y TIGER.

Usos GnuPG: manejo de llaves

- Generación de llaves pública/privada:

```
gpg --full-gen-key
```

- Listar llaves disponibles en el keyring:

```
gpg --list-public-keys
gpg --list-secret-keys
```

- Exportar llave pública:

```
gpg --armor --output usuario.asc --export \
usuario@dominio.com
```

- Importar llave pública:

```
gpg --import usuario.asc
```

- Editar keyring para confiar en llaves públicas importadas:

```
gpg --edit-key usuario@dominio.com
```

Usos GnuPG: manejo de llaves

- Buscar llave pública en servidor de llaves:

```
gpg --keyserver hkp://keys.gnupg.net:80 --search-keys 00411886
```

- Enviar llave pública a servidor de llaves:

```
gpg --keyserver hkp://keys.gnupg.net:80 --send-keys 9D2FFAC8
```

- Importar llave pública de terceros desde servidor de llaves:

```
gpg --keyserver hkp://keys.gnupg.net:80 --recv-keys 00411886
```

Usos GnuPG: cifrado / descifrado

- Cifrar un archivo utilizando nuestra llave pública:

```
gpg --recipient usuario@dominio.com --output \
archivo.txt.gpg --encrypt archivo.txt
```

- Descifrar archivo:

```
gpg --output archivo.txt --decrypt archivo.txt.gpg
```

- Cifrar archivo utilizando un algoritmo simétrico (por defecto AES o usando –cypher-algo):

```
gpg --output archivo.txt.gpg --symmetric archivo.txt
```

Usos GnuPG: cifrado / descifrado

- Cifrar un archivo para ser enviado a otro usuario en Base64:

```
gpg --armor --recipient usuario@dominio.com --output \
archivo.txt.asc --encrypt archivo.txt
```

- Cifrar un archivo para ser enviado a otro usuario en formato binario:

```
gpg --recipient usuario@dominio.com --output \
archivo.txt.gpg --encrypt archivo.txt
```

- Descifrar un archivo cifrado para nosotros:

```
gpg --decrypt-files archivo.txt.asc
```

Usos GnuPG: firma digital

- Firmar digitalmente un archivo:

```
gpg --local-user usuario@dominio.com --clearsign archivo.txt
```

- Verificar un mensaje firmado digitalmente:

```
gpg --verify archivo.txt.asc
```

- Cifrar y firmar al mismo tiempo:

```
gpg --local-user usuario@dominio.com --recipient\
maliaga@uccor.edu.ar --armor --sign --output archivo.asc\
--encrypt archivo.txt
```

Bibliografía

- **LUCENA LÓPEZ, Manuel J.**. *Criptografía y Seguridad en Computadores*. http://wwwdi.ujaen.es/~mlucena/wiki/pmwiki.php?n=Main.LCripto
- **KIDWELL, Brendan.** *A Practical Introduction to GNU Privacy Guard in Windows.* http://www.glump.net/howto/gpg_intro
- **RAMIÓ AGUIRRE, Jorge.** *Libro Electrónico de Seguridad Informática y Criptografía.* Universidad Politécnica de Madrid. http://www.criptored.upm.es/guiateoria/gt_m001a.htm
- **SOTOMAYOR, Borja.** *The Globus Toolkit 4 Programmer's Tutorial.* http://gdp.globus.org/gt4-tutorial/multiplehtml/ch09s04.html
- **Wikipedia**. Digital signature. http://en.wikipedia.org/wiki/Digital_signature
- **Wikipedia**. X.509. http://en.wikipedia.org/wiki/X.509
- **GnuPG**. GnuPG Manual. http://www.gnupg.org/gph/es/manual.html