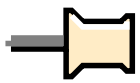
Servidores Virtuales Apache y HTTPS

Álvaro González Sotillo

Monday 11th January, 2016



1. Servidores virtuales

2. Seguridad informática
   - Criptografía
   - Cifrado simétrico
   - Cifrado asimétrico
   - Criptografía híbrida
   - Firma digital
   - PKI

Servidor virtual

- Un servidor *Apache* puede "aparentar" ser varios servidores, dependiendo de:
  - La IP y/o puerto a la que accede el cliente (conexión TCP)
  - El nombre con el que accede el cliente (cabecera *Host*)
- En estos casos, la directiva *DocumentRoot* puede cambiarse (entre otras)
  - `http://httpd.apache.org/docs/2.4/es/vhosts/`
  - `http://httpd.apache.org/docs/2.4/es/vhosts/examples.html`

Directivas

- **<VirtualHost> ... </VirtualHost>**: Agrupan directivas para hosts virtuales
- **NameVirtualHost**: Especifican IP/puertos que para un host virtual
- **ServerName**: Nombre con el que se identifica un host virtual. Se utiliza cuando el cliente envía la cabecera *Host*
- **ServerAlias**: Nombres alternativos (se admite *)
- **ServerPath**: URL alternativa al host virtual (si es incapaz de usar la cabecera *Host*)

Sitio por defecto

- El *VirtualHost* * es el sitio por defecto de *Apache*
- Las directivas que no se especifican dentro de un *VirtualHost*
  - Se aplican al sitio por defecto (*Alias...*)
  - Se aplican a todos los sitios (*Directory*)

```
<VirtualHost *:80>
  ServerAdmin webmaster@localhost
  DocumentRoot /var/www/html
  ErrorLog ${APACHE_LOG_DIR}/error.log
  CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Listado 1: 000-default.conf

- Nota: El *VirtualHost* **_default_** se aplica a las IP que no tengan otro *VirtualHost*, mientras que * se aplica a todas las IP.

Sitios

- Hasta ahora, definíamos un sitio como un fichero *conf* con directivas
  - Que afectaban al host virtual por defecto
- Pero un sitio debería ser:
  1. Un fichero *conf* con directivas
  2. Que describe un host virtual
  3. Y solo a veces, afectan a otros host virtuales definidos en otros ficheros

VirtualHost por IP

VirtualHost por IP

Añade la dirección 192.168.3.230+n a tu servidor.
Crea un sitio **alternativo.conf**, que sirva el directorio /var/www/alternativo cuando se acceda *Apache* por esa nueva IP.
Comprueba que se sirven páginas distintas en 192.168.3.230+n y en 192.168.3.200+n.

```
<VirtualHost 192.168.3.230+n:80>
  DocumentRoot /var/www/alternativo
</VirtualHost>
```

Listado 2: alternativo.conf

Seguridad

- HTTP no es un protocolo seguro.
- Intercambio de información en texto plano (*sniffing*).
- Basic y Digest no son seguros.
- No se garantiza que los equipos involucrados en la transferencia son quienes dicen ser (*spoofing* y *man-in-the-middle*).
- Robo o falsificación de cookies y/o parámetros (robo de identidad y suplantación de webs)

Criptografía

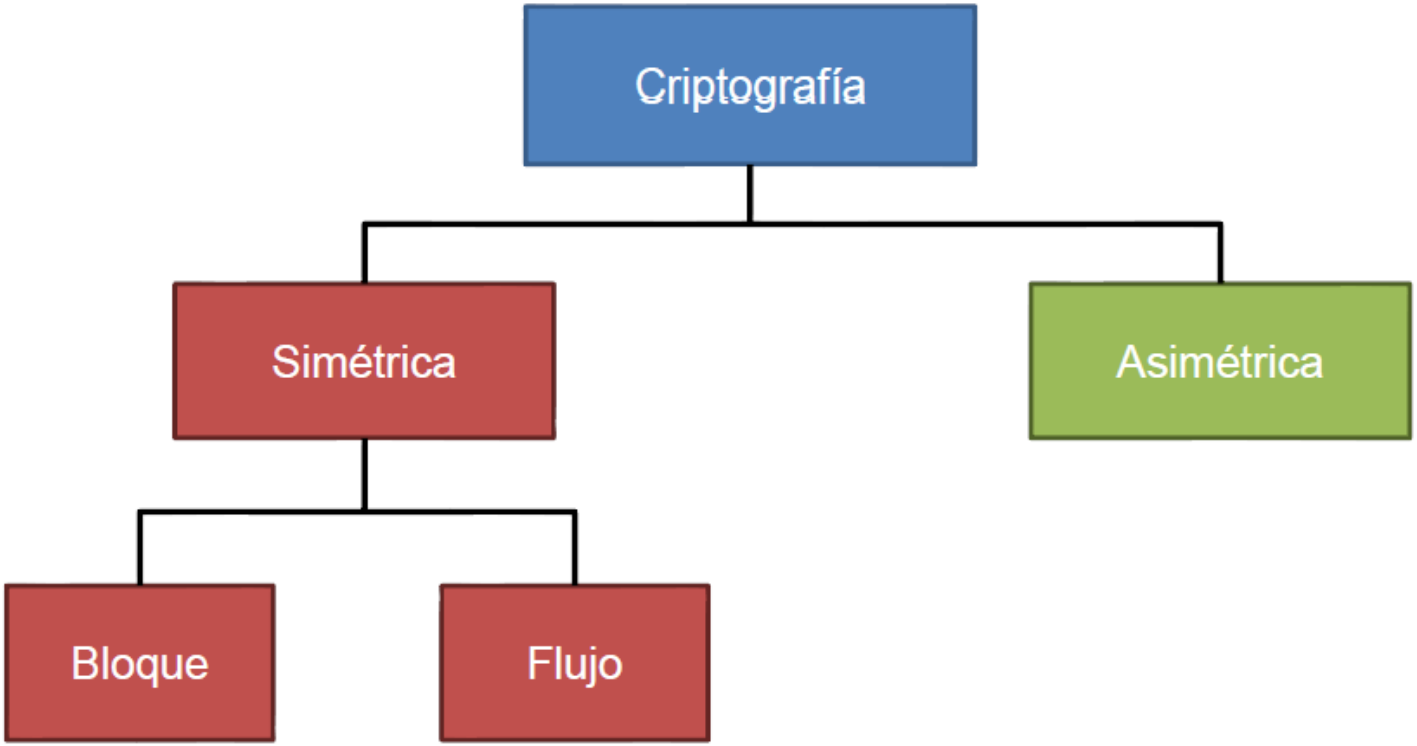
- **Criptografía** -> Cripto + Grafía (Siglo V a.C).
  - **Cripto** -> Escondido
  - **Grafía** -> Escritura
  - Ciencia que estudia la escritura oculta. Se ocupa del cifrado y el descifrado de mensajes.
- **Criptología** =criptografía + criptoanálisis (ataques).

Cifrado

- Cifrar información consiste en transformar un mensaje en claro en un mensaje ininteligible que solo puede ser descifrado por alguien autorizado.
- Se basa en la utilización de
  - Algoritmos públicos
  - Se desaconseja el uso de algoritmos supuestamente privados (criptografía por oscuridad)
  - Claves de cifrado.

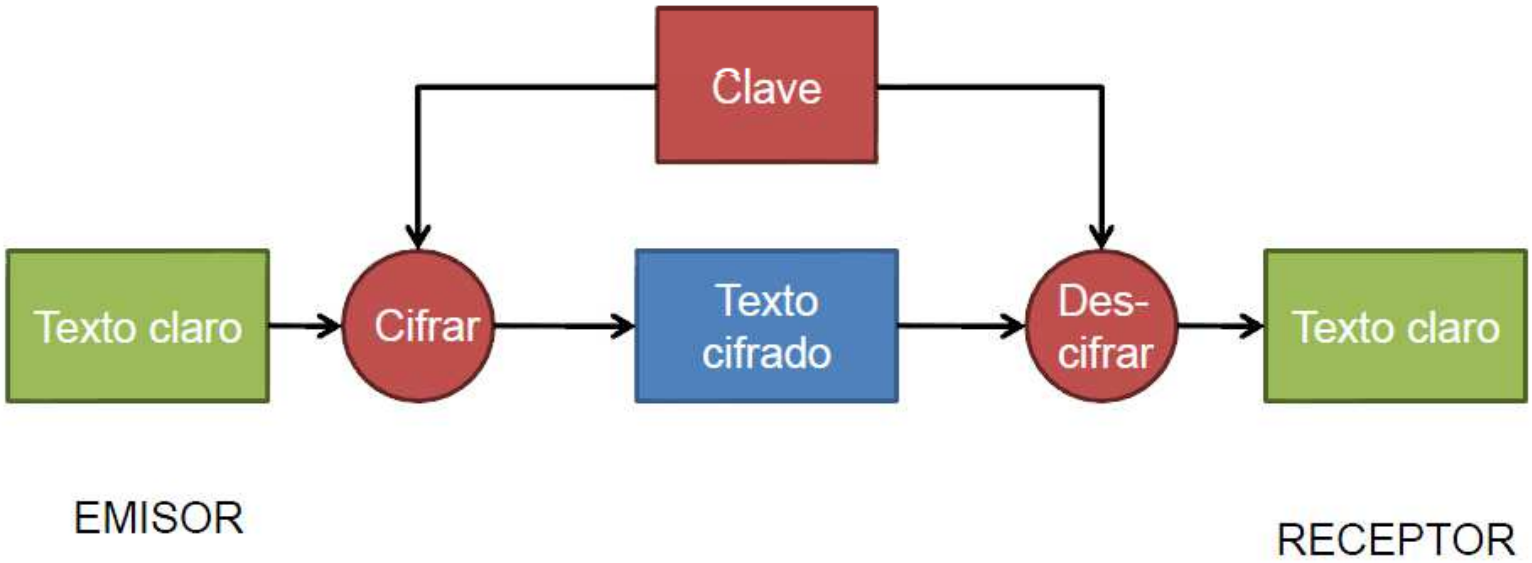
Tipos de cifrado

- Dos tipos de algoritmos de cifrado
  - Algoritmos de clave simétrica (secreta).
  - Algoritmos de clave asimétrica (pública).



Cifrado simétrico - Clave secreta

- La seguridad está en la clave, no en el algoritmo.
- Las claves hay que distribuirlas en secreto.
- Si una clave está comprometida, puede descifrarse todo el tráfico con la misma.



Cifrado simétrico - Clave secreta

Ejemplos de algoritmos de cifrado simétrico

- DES, Triple DES (3DES)
- IDEA
- AES
- BLOWFISH
- RC4, RC5

Cifrado simétrico - Clave secreta

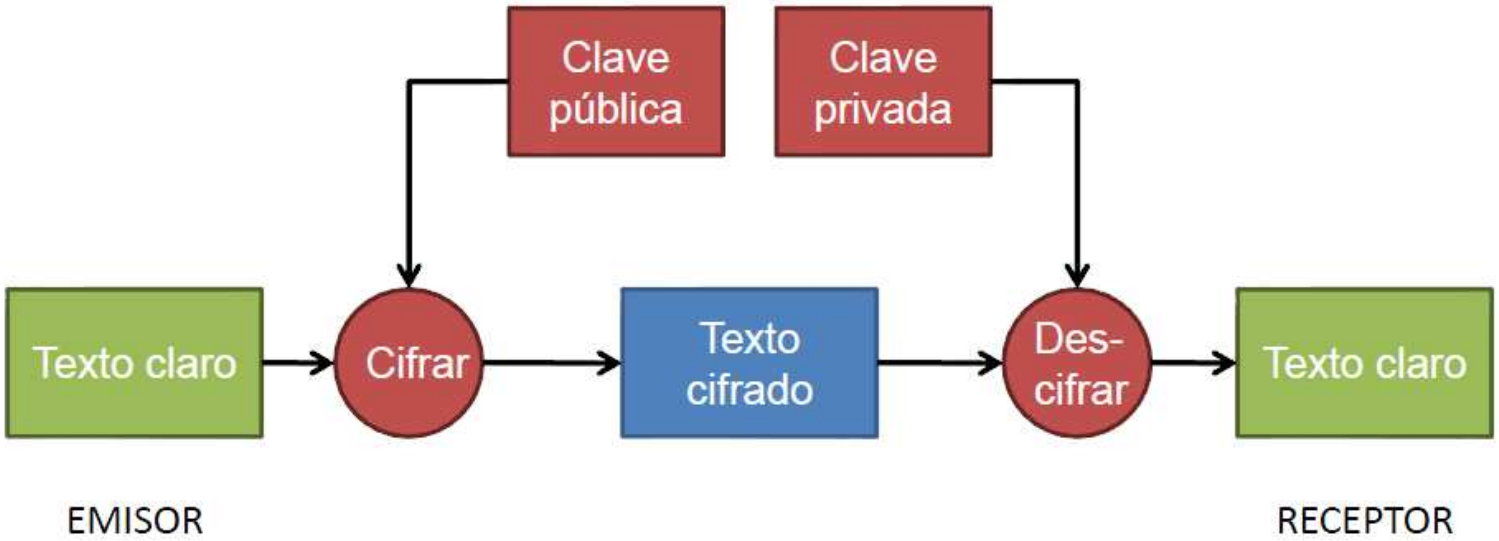
- Ventajas
  - Eficiente: Los algoritmos utilizados son muy rápidos
- Inconvenientes
  - Ambas partes deben conocer la clave
  - Muchas claves, una por cada pareja de comunicantes.
  - **Distribuir la clave secreta**

Cifrado asimétrico - Clave pública

- Se basan en el uso de dos claves: una pública y otra privada.
- Cada emisor/receptor tiene dos claves.
- La clave privada sólo la conoce el dueño de la clave, clave es decir, no se publica (no se envía por la red).
- La clave pública es conocida por otros
- Se generan al mismo tiempo dando lugar a pares biunívocos, de tal forma que la combinación pública/privada es única.

Cifrado asimétrico - Clave pública

- Lo que se cifra con la clave privada solo se puede descifrar con la pública.
- Lo que se cifra con la clave pública solo se puede descifrar con la privada.



Cifrado asimétrico - Clave pública

Ejemplos de algoritmos de cifrado asimétrico

- RSA
- DSA
- Diffie-Hellman (DH)

Cifrado asimétrico - Clave pública

- Ventajas
  - La clave privada no se transmite y es suficiente que cada usuario tenga su clave doble pública-privada
- Inconvenientes
  - No utiliza algoritmos eficientes (aunque se puede considerar una ventaja)
  - Se debe garantizar la autenticidad de las claves públicas; públicas es decir, que la clave pública de un usuario es realmente suya

Comparativa

Atributo	Clave simétrica	Clave asimétrica
Años en uso	Miles	Menos de 50
Uso principal	Cifrado de grandes volúmenes de datos	Intercambio de claves; firma digital
Estándar actual	DES, Triple DES, AES	RSA, Diffie-Hellman, DSA
Velocidad	Rápida	Lenta
Claves	Compartidas entre emisor y receptor	Privada: sólo conocida por una persona Pública: conocida por todos
Intercambio de claves	Difícil de intercambiar por un canal inseguro	La clave pública se comparte por cualquier canal La privada nunca se comparte
Longitud de claves	56 bits (vulnerable) 256 bits (seguro)	1024 – 2048 (RSA) 172 (curvas elípticas)
Servicios de seguridad	Confidencialidad Integridad Autenticación	Confidencialidad Integridad Autenticación, No repudio

Criptografía híbrida

Combinar algoritmos de clave simétrica y asimétrica en transmisión de información, para combinar sus ventajas.

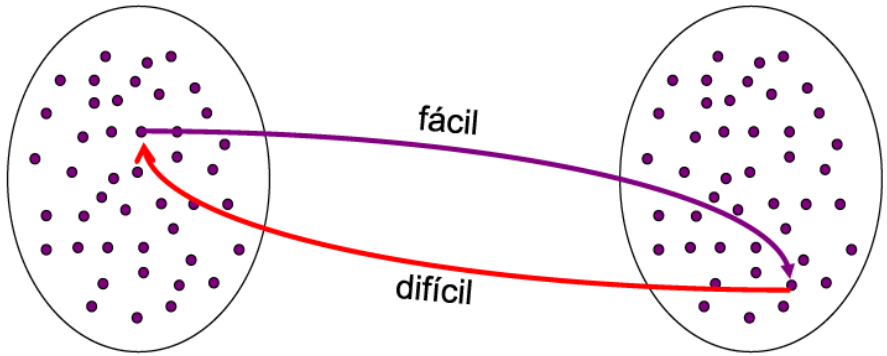
- ¿Por qué no usar únicamente criptografía simétrica?
  - Es problemático intercambiar la clave.
- ¿Por qué no usar únicamente criptografía asimétrica?
  - El cifrado y descifrado es más lento y costoso en CPU que si se usa un algoritmo de criptografía simétrica.

Criptografía híbrida

1. El cliente se conecta al servidor.
2. El servidor envía su clave pública.
3. El cliente verifica que la clave es realmente del servidor.
4. El cliente genera una clave simétrica, la cifra con la clave pública del servidor y se la envía.
5. El servidor recibe la clave simétrica y la descifra con su clave privada.
6. Los dos tienen la clave privada para intercambiar información cifrada.

Hash

- Funciones basadas en algoritmos que obtienen un resumen de fichero /mensaje
- El resumen es único para el mensaje (o por lo menos las probabilidades son muy pequeñas).
- Son funciones de un solo sentido: conocido el resumen no se puede conocer el fichero/mensaje.



Hash

Ejemplos de algoritmos de hash

- MD5
- SHA1
- WHIRLPOOL

Firma digital

- Permite firmar un documento digitalmente.
  - **Integridad**: El mensaje no ha sido modificado
  - **No repudio**: El remitente no puede negar la autoría del mensaje
- Basada en
  - Algoritmos de clave pública.
  - Funciones resumen (hash).

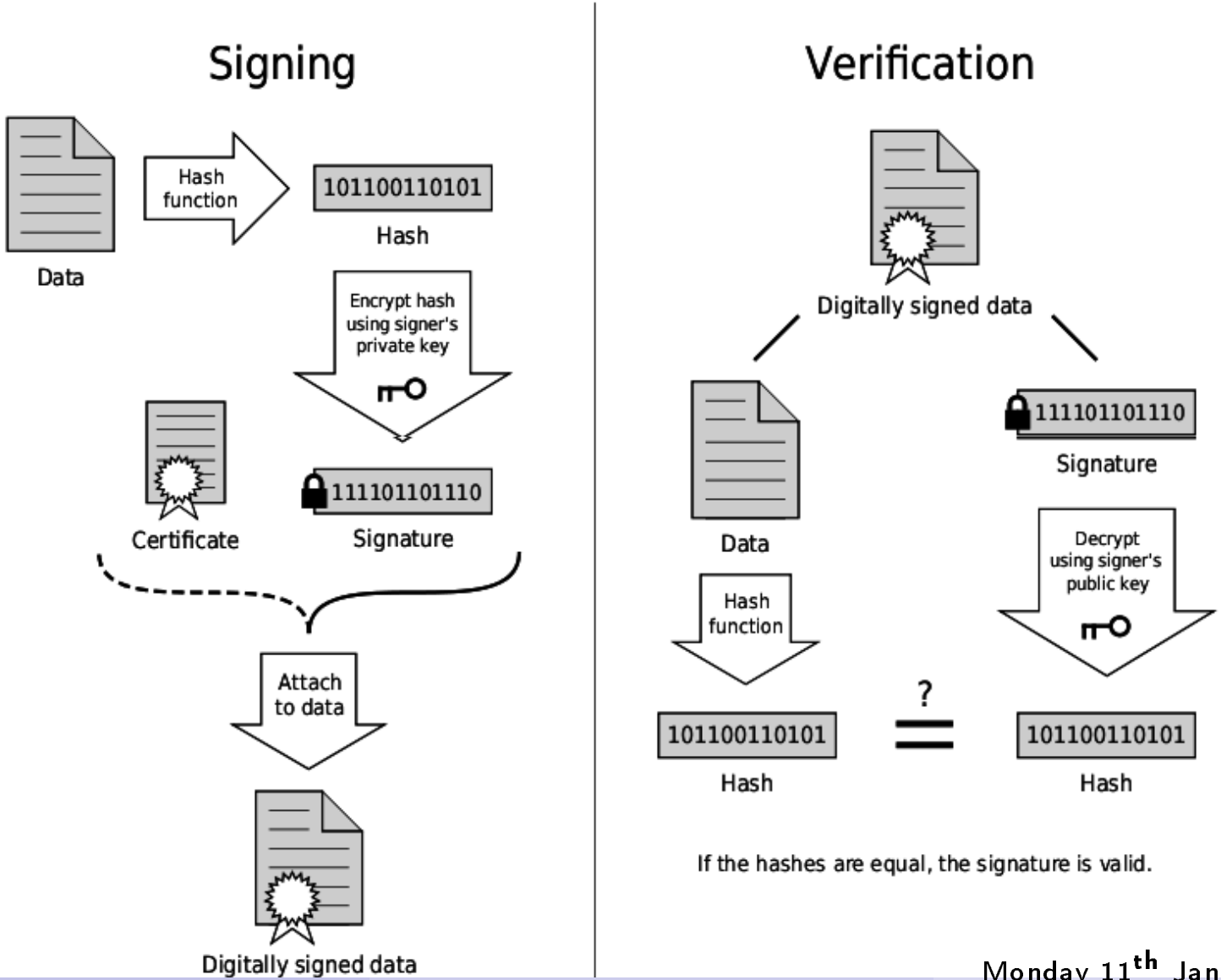
Firma digital: Firmado

1. Se calcula el resumen (hash) de un documento
2. El resumen se cifra con la clave privada delusuario
3. De esta manera se asegura que el único que ha firmado el documento es el usuario, porque es el único que conoce la clave privada.
4. El resultado es lo que se conoce como firma digital del documento.

Firma digital: Verificación

1. La firma se descifra usando la clave pública del usuario (cualquiera la puede tener, por lo tanto cualquiera puede verificar la firma del usuario)
2. Se obtienen el valor resumen del documento firmado (usando el mismo algoritmo que en el proceso de firmado)
3. Se comparan los dos resúmenes obtenidos y si coinciden la firma es válida.

Firma digital



Certificado digital

Un certificado digital documento/archivo que contiene:

- **Información sobre el propietario**: una persona, entidad, empresa, organización...
  - (nombre, dirección, email ...)
- **La clave pública** del propietario
- **La firma digital** de un organismo de confianza , una autoridad de certificación (**CA, Certificate Authority**) que garantiza que la clave púbica que contiene el certificado se corresponde con el propietario del mismo.

Un formato popular de certificado digital es **X.509**

Certificado autofirmado

- Un certificado autofirmado es el que se firma utilizando la propia clave pública/privada del propietario
- No existe ningún mecanismo automático que garantice la autenticidad del certificado

Certificado raíz

- Emitidos por las autoridades de certificación para sí mismas con su clave pública
- Son necesarios para verificar la autenticidad de los certificados emitidos por ellas
- Se convierten en certificados raíz por elección del usuario: **se decide confiar en ellas**

Certificados raíz de Chrome/Firefox/IE

Comprueba qué certificados raíz utiliza tu navegador ¿Confías en todos los emisores de certificados? ¿Cómo podrías comprobar su autenticidad?

Autoridades de certificación (CA)

- Fabrica Nacional de Moneda y Timbre: `http://www.cert.fnmt.es`
- Dirección General de la Policía (DNI Electrónico): `http://www.dnielectronico.es`
- Verisign: `http://www.verisign.com/`
- Cacert: `http://www.cacert.org`

VERISIGN®

SSL/TLS

- Protocolos de nivel transporte (aproximadamente)
  - Se pueden utilizar como una capa adicional entre TCP y HTTP,FTP,IMAP,SMTP...
- Utilizan certificados X.509
- Se basan en Autoridades de Certificación de confianza (PKI: *Public Key Infrastructure*)

HTTPS

- *Hyper Text Transfer Protocol Secure*
- Protocolo que utiliza SSL/TLS para encapsular mensajes HTTP.
- `https://` en las URIs (o URLs).
- El puerto por defecto es 443/TCP

Práctica OpenSSL

Conexión sin cifrar a servidor HTTPS

Desde *Debian*, usa el comando `nc` para conectarte a `www.microsoft.com:443`. Intenta bajar el recurso raíz.

```
alumno@debian8:~$ nc 213.0.88.85 8080
GET / HTTP/1.1
Host: www.microsoft.com:443
```

Listado 3: Petición HTTP

Nota: Por problemas con el proxy, probaremos con `192.168.3.198:443`

Práctica OpenSSL

Conexión cifrada a servidor HTTPS

Desde *Debian*, usa el comando `openssl` para conectar conectarte a `www.microsoft.com:443`. Intenta bajar el recurso raíz.

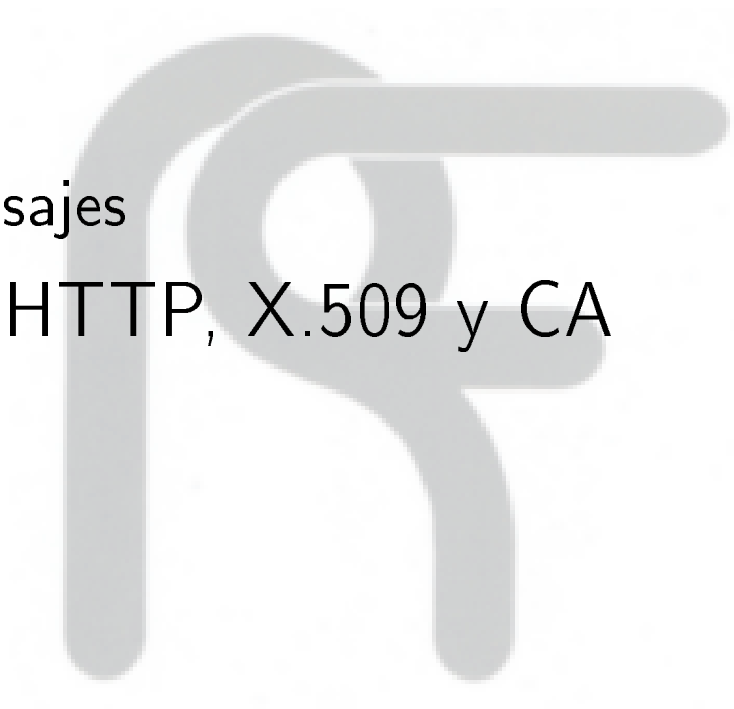
```
alumno@debian8:~$ openssl s_client -connect 213.0.88.85:8080
GET / HTTP/1.1
Host: www.microsoft.com:443
```

Listado 4: Petición HTTPS

Nota: Por problemas con el proxy, probaremos con `192.168.3.198:443`

PKI (Public Key Infrastructure)

- Sistema para la utilización de clave pública
- Define
  - Entidades de confianza
  - Protocolos de comunicación
  - Formatos de certificados y mensajes
- La más común (WWW) utiliza HTTP, X.509 y CA
- PKI en la Wikipedia



Ejercicios

Conexión a Google

Accede a `http://www.google.com`

1. ¿Cómo se llega a una conexión segura?
2. ¿Qué función/funciones de hash se utilizan en el certificado?
3. ¿Qué longitud tiene la clave simétrica utilizada?
4. ¿Qué método de encriptación se utiliza para la clave asimétrica?
5. ¿Cuál es la ruta de certificación? (camino desde el certificado de Google hasta el certificado de la CA)

Ejercicios

Cita de DNI

Inicia los trámites para conseguir una cita previa para renovación del DNI.
Explica qué problemas de seguridad encuentras.