NOTA: *Dada la inseguridad de las personas a pasar información por vía digital, debido a la falta de conocimiento y a las noticias que se escuchan en los medios de comunicación y la necesidad que conlleva la práctica de los modelos de negocios de realizar ciertas operaciones como pueden ser transacciones, pagos se ha creído interesante realizar este estudio detallado de las medidas de seguridad que se conocen y aplican para pasar información por la red.*

I.I. INTRODUCCIÓN

Internet ha evolucionado desde su uso fundamentalmente científico a un uso mayoritariamente comercial. Para facilitar el intercambio comercial, se han desarrollado una serie de técnicas que hacen de Internet un medio relativamente seguro para las transacciones comerciales. Posibles ocasiones en las que se hace necesario recibir o enviar información personal a través de un servidor de la web son:

- Existen sitios web de la **Administración** con información confidencial sobre becas, datos de personal, nóminas, etc. Resulta evidente el interés de que esa información no sea accesible a toda la web, sino a un pequeño número autorizado de usuarios. Por lo tanto, no vale con restringir el acceso mediante claves de acceso o procedimientos similares, además la información que viaje hacia los usuarios debe ir cifrada, para evitar escuchas.

- Otro caso se produce cuando se envía a un servidor información confidencial sobre **nuestra persona**, por ejemplo en las fórmulas CGI. Interesa que el servidor conozca los datos, pero no el resto de la web, especialmente si se está realizando una transacción comercial electrónica y se revelan el número de tarjeta de crédito o simplemente la dirección postal.

Como es sabido al realizar B2B se tiene necesidad de asegurarse de la autenticidad de las transacciones que se realizan.

Aspectos de seguridad en Internet

Las cuestiones de seguridad a tener en cuenta en Internet son las siguientes:

- Protección del sistema (cortafuegos)
- Codificación de la información que se envía (enviar números de tarjeta de crédito)
- Autenticación de los usuarios (firmas digitales)

I.II. ENTIDADES CERTIFICADORAS

Cabe pensar que al enviar los datos cifrados por la web ya no existen problemas de piratas informáticos, ahora bien, si se les envía o recibe un impostor, la situación es igual de peligrosa. Por ello, surge la necesidad de contar con un mecanismo que dé fe de si un servidor seguro es quien se cree que es y se puede confiar en él a la hora de transmitirle la información. Actualmente esto se hace mediante las Autoridades de Certificación (AC), conocidas informalmente como notarios electrónicos, encargados de dar autenticidad a los participantes en transacciones y comunicaciones a través de la web. Su función es emitir certificados a los usuarios, de manera que se pueda estar seguro de que el interlocutor (cliente o servidor) es quien pretende ser, garantizando así la seguridad de las transacciones.

El certificado de seguridad se concede a una entidad después de comprobar una serie de referencias, para asegurar la identidad del receptor de datos cifrados. Se construye a partir de la clave pública del servidor solicitante, junto con algunos datos básicos del mismo y es firmado por la autoridad de certificación correspondiente con su clave privada.

En la práctica, se sabe que el servidor es seguro porque en el navegador se ve una llave o un candado cerrado en la parte izquierda, si se usa Netscape, o bien un candado cerrado en la parte derecha, si se usa Explorer. Cuidado porque la posición y la forma del icono son importantes y modificaciones de ellos no garantizan la seguridad.

Autoridades de certificación

- Con las firmas digitales, el único problema de seguridad se refiere a la identidad del usuario
  - Ej: el que alguien genere una clave pública (y una privada) diciendo que es el Banco de España no le convierte en el Banco de España
  - Lo que sí es cierto es que nadie que no sea él podrá emplear esa firma digital, ya que sólo él conoce la clave privada
- Para evitar este problema, existen empresas de autorización independientes (autoridades de certificación)
  - Estas empresas expiden un certificado digital que asegura que la empresa que tiene la clave pública de la empresa X es realmente la empresa X
  - Los certificados digitales tienen una fecha de caducidad, por lo que deben ser renovados cada cierto tiempo

I.III. PROTOCOLOS DE SEGURIDAD

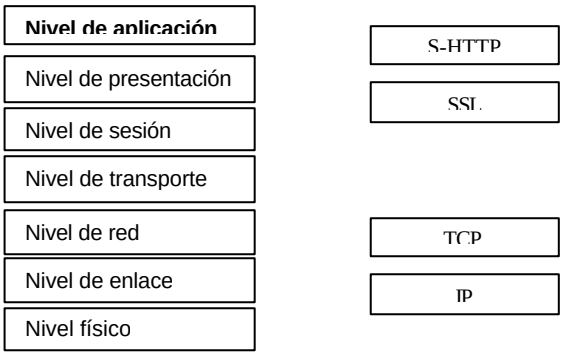
Secure socket layer (SSL)

El protocolo SSL es un sistema diseñado y propuesto por Netscape Communications Corporation. Se encuentra en la pila OSI entre los niveles de TCP/IP y de los protocolos HTTP, FTP, SMTP, etc. Proporciona sus servicios de seguridad cifrando los datos intercambiados entre el servidor y el cliente con un algoritmo de cifrado simétrico, típicamente el RC4 o IDEA, y cifrando la clave de sesión de RC4 o IDEA mediante un algoritmo de cifrado de clave pública, típicamente el RSA. La clave de sesión es la que se utiliza para cifrar los datos que vienen del cliente y van al servidor seguro. Se genera una clave de sesión distinta para cada transacción dada, lo cual permite que aunque sea reventada por un atacante en una transacción dada, no sirva para descifrar futuras transacciones.

Cuando el cliente pide al servidor seguro una comunicación segura, el servidor abre un puerto cifrado, gestionado por un software llamado Protocolo SSL Record, situado encima de TCP. Será el software de alto nivel, Protocolo SSL Handshake, quien utilice el Protocolo SSL Record y el puerto abierto para comunicarse de forma segura con el cliente.

Codificación a nivel de presentación - SSL

- SSL responde a las iniciales de Socket Security Layer (Capa de puertos de seguridad)
- Se trata de codificar a nivel de presentación todos los mensajes que entran y salen de un ordenador
  - Utiliza un mecanismo de claves privadas y públicas similar al de S-HTTP
  - Puede utilizarse en combinación con S-HTTP o cualquier otro medio de codificación al nivel de aplicación



El Protocolo SSL Handshake

Con el protocolo SSL Handshake, el cliente y el servidor intercambian una serie de mensajes para negociar las mejoras de seguridad. Este protocolo sigue las seis fases que a continuación se describen:

- La fase **hola,** usada para ponerse de acuerdo sobre el conjunto de algoritmos para mantener la intimidad y para la autenticación.
- La fase de **intercambio de claves,** en la que intercambian información sobre las claves, de modo que al final ambas partes comparten una clave maestra.
- La fase de **producción de clave de sesión,** que será la usada para cifrar los datos intercambiados.
- La fase de **verificación del servidor,** presente sólo cuando se usa RSA como algoritmo de intercambio de claves, y sirve para que el cliente autentifique al servidor.
- La fase de **autentificación del cliente,** en la que el servidor solicita al cliente un certificado X.509 (si es necesaria la autenticación de cliente).
- Por último, la fase de **fin,** que indica que ya se puede comenzar la sesión segura.

El Protocolo SSL Record

El protocolo SSL Record especifica la forma de encapsular los datos transmitidos y recibidos. La porción de datos del protocolo tiene tres componentes:

- MAC-DATA, el código de autenticación del mensaje.
- ACTUAL-DATA, los datos de aplicación a transmitir.
- PADDING-DATA, los datos requeridos para rellenar el mensaje cuando se usa cifrado en bloque.

Secure hypertext transfer protocol (S-HTTP)

El protocolo S-HTTP fue desarrollado por Enterprise Integration Technologies (EIT). Al igual que SSL, permite tanto el cifrado como la autenticación digital. Sin embargo, a diferencia de SSL, S-HTTP es un protocolo de nivel de aplicación, es decir, que extiende el protocolo HTTP por debajo.

Usando GET, un cliente solicita un documento, le dice al servidor qué tipo de cifrado puede manejar y le dice también dónde puede encontrar su clave pública. Si el usuario con esa clave está autorizado a acceder al documento, el servidor responde cifrando el documento y enviándoselo al cliente, que usará su clave secreta para descifrarlo y mostrárselo al usuario.

Las negociaciones entre el cliente y el servidor tienen lugar intercambiando datos formateados. Estos datos incluyen una variedad de oposiciones de seguridad y algoritmos a utilizar. Las líneas usadas en las cabeceras incluyen:

- Dominios privados S-HTTP, que especifica la clase de algoritmos de cifrado así como la forma de encapsulamiento de los datos (PEM o PKCS-7).
- Tipos de certificado S-HTTP, que especifica el formato de certificado aceptable, actualmente X.509.

- Algoritmo de intercambio de clave S-HTTP, que indica los algoritmos que se usarán para el intercambio de claves (RSA, fuera de banda, dentro de banda y Krb).
- Algoritmos de firmas S-HTTP, que especifica el algoritmo para la firma digital (RSA o NITS-DSS).
- Algoritmos de resumen de mensaje S-HTTP, que identifica el algoritmo para proporcionar la integridad de los datos usando funciones de hash (RSA-MD2, RSA-MD5 o NITS-SHS).
- Algoritmos de contenido simétrico S-HTTP, que especifican el algoritmo simétrico de cifrado en bloque usado para cifrar datos (DES-CBC, DES-EDE-CBC, DES-EDE3-CBC,DESX-CBC,IDEA-CFB, RC2-CBC, RC4, CDMF).
- Algoritmos de cabecera simétrica de S-HTTP, que proporcionan una lista del cifrado de clave simétrica utilizada para cifrar las cabeceras (DES-ECB, DES-EDE-ECB, IDEA-ECB, RC2-ECB, CDMF-ECB).
- Mejoras de la intimidad de S-HTTP, que especifican las mejoras en la intimidad asociadas con los mensajes, como firmar, cifrar o autenticar.

Uno de los métodos de cifrado disponible en S-HTTP es el popular PGP.

Servidores "seguros" – protocolo S-HTTP

- S-HTTP contiene toda la funcionalidad de HTTP con funciones de seguridad añadidas
  - S-HTTP manda la información entre el cliente y el servidor codificada (permite el envío de password y números de tarjeta de crédito de forma segura)
- El proceso resumido es el siguiente:
  - El navegador (cliente) envía su clave pública al servidor
  - El servidor devuelve una nueva clave (clave de sesión) codificada con la clave pública que le ha enviado el cliente
  - El cliente obtiene la clave de sesión utilizando su clave privada
  - El cliente codifica y descodifica todas las siguientes comunicaciones utilizando la clave de sesión, igual que el servidor (ahora utilizan un mecanismo de clave única)

SSL frente a S-HTTP

S-HTTP y SSL utilizan aproximaciones distintas con el fin de proporcionar servicios de seguridad a los usuarios de la web. SSL ejecuta un protocolo de negociación para establecer una conexión segura al nivel de socket (nombre de máquina más puerto). Los servicios de seguridad de SSL son transparentes al usuario y a la aplicación.

Por su parte, los protocolos S-HTTP están integrados con HTTP. Aquí, los servicios de seguridad se negocian a través de las cabeceras y atributos de la página. Por lo tanto, los servicios de S-HTTP están disponibles sólo para las conexiones de HTTP.

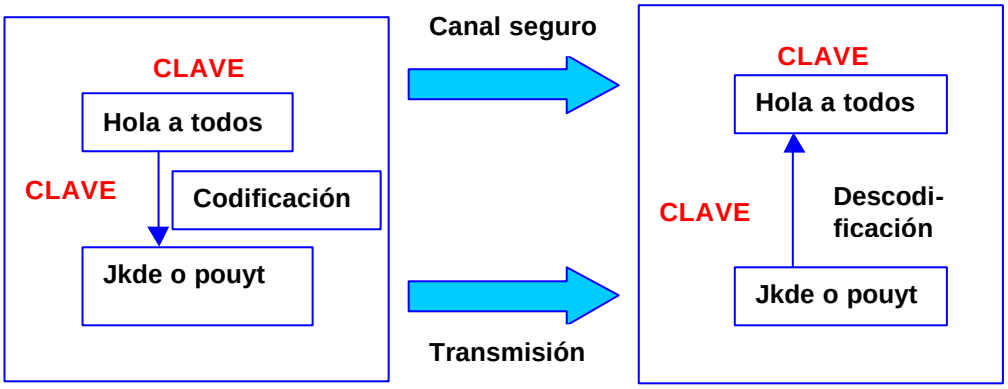
Dado que SSL se integra en la capa de sockets, también permite ser usado por otros protocolos además del HTTP, mientras que el S-HTTP está concebido para ser usado exclusivamente en comunicaciones HTTP.

I.IV. CODIFICACIÓN

A continuación se pasa a analizar de forma gráfica distintas formas de codificar y resaltar alguna de sus características:

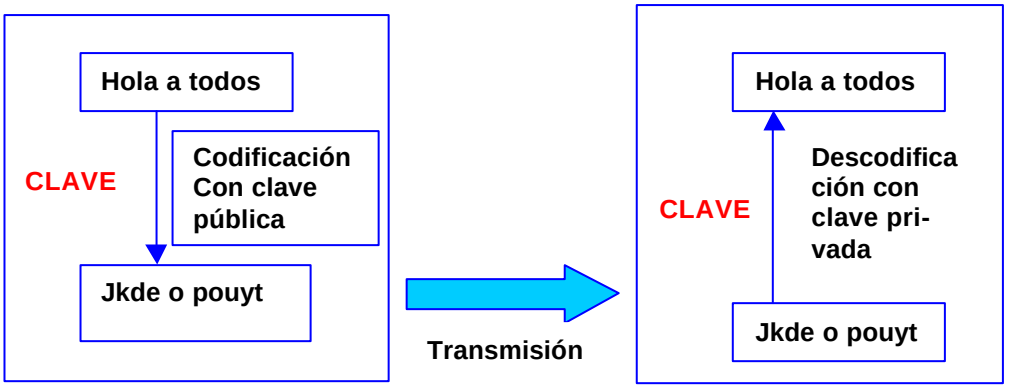
1. Codificación: Clave única

- Ambas partes deben conocer la clave antes de cada transmisión
- Se deberán crear distintas claves para cada individuo o grupo de individuos que realicen transmisiones



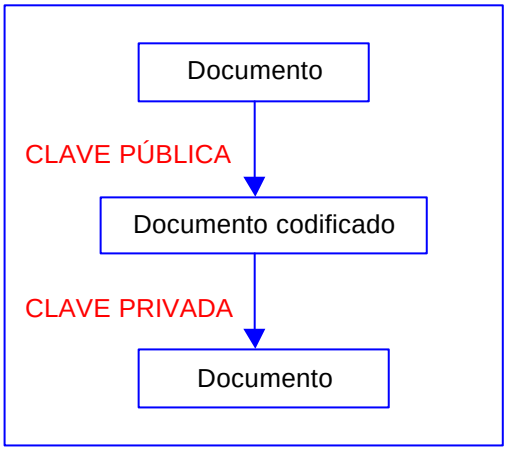
2. Codificación: Clave pública

- La clave pública se puede distribuir libremente, de forma que cualquiera puede codificar un mensaje con esa clave.
- La base de funcionamiento de este sistema de codificación es que no se puede deducir la clave privada a partir de la clave pública.
- Hay fundamentalmente dos tipos de algoritmos de creación de claves privadas y públicas:
  - RSA (Rivest, Shamir, Adleman)
  - Diffie y Hellmam



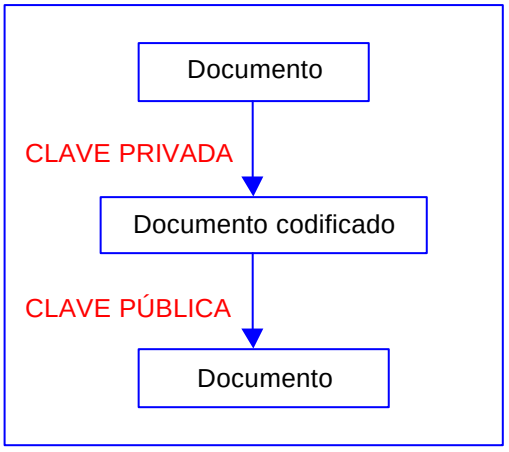
- La codificación con clave pública permite transmitir mensajes que sólo el poseedor de la clave privada puede decodificar.

  - Permite enviar mensajes de forma segura.



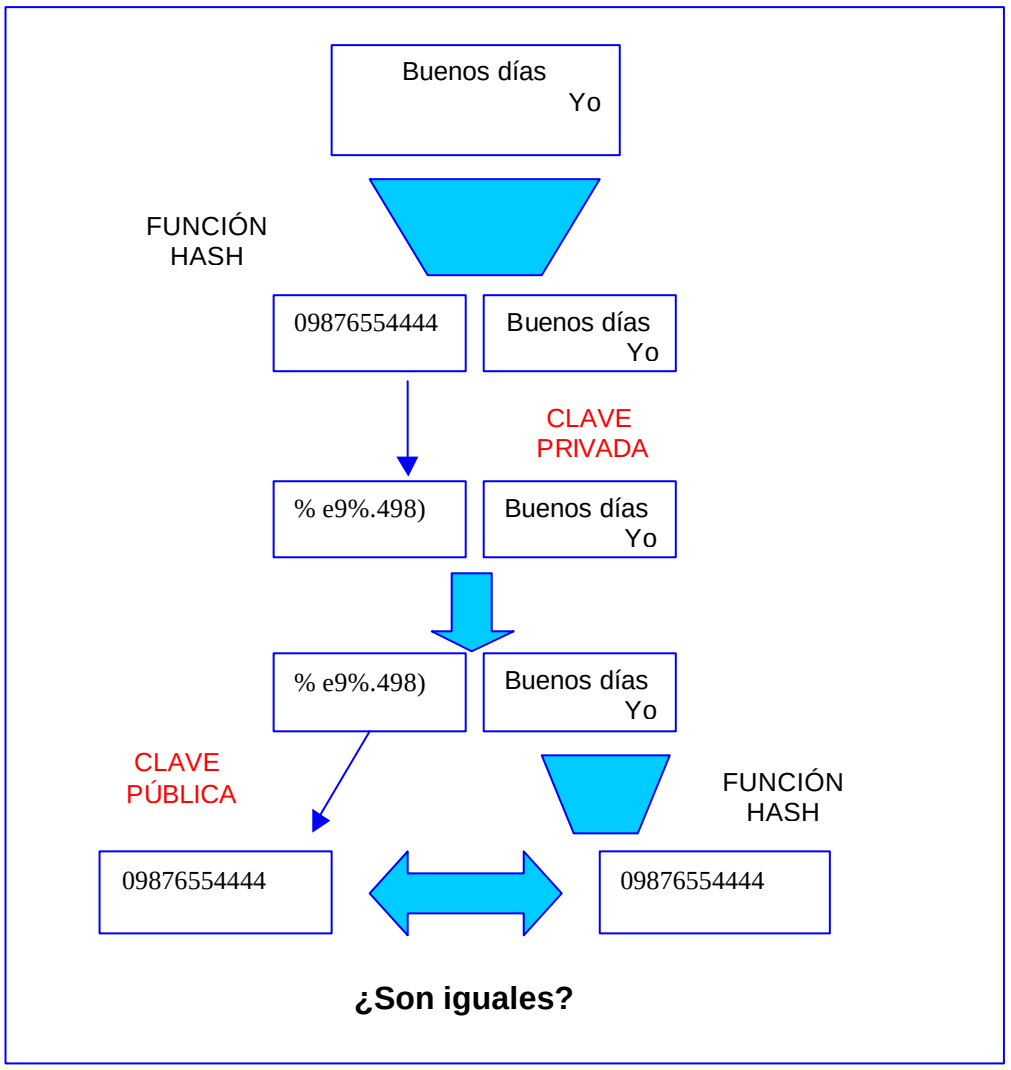
- La codificación con clave privada permite crear mensajes que sólo ha podido codificar el poseedor de la clave privada.

- Permite crear firmas digitales



3. Firmas digitales

Asegura que se ha enviado el mensaje y que no se ha modificado por el camino.



Los pasos que sigue el proceso son:

- En el primer paso se le aplica la **función hash** y ésta hace un resumen del mensaje.
- En el segundo paso se le aplica la **clave privada** al mensaje resumido.

- En el tercer paso se produce la **transmisión** por Internet.
- En el cuarto paso se le aplica la **clave pública** y se obtiene el mensaje resumido.
- Finalmente se **compara** este mensaje con el que se obtiene de aplicar la función fash, si son iguales el mensaje ha sido correctamente enviado.