SEGURIDAD EN REDES Y SISTEMAS INFORMÁTICOS

Ana González (Kao)

UNIDAD 4. Técnicas para la seguridad de los datos

Ana González (Kao)

ÍNDICE UNIDAD 4

4.1 Introducción a la Criptografía

La palabra criptografía proviene del griego Crypto, que significa Oculto y grafia, escritura. La criptografía se ha utilizado desde tiempos inmemoriales tanto por faraones en el antiguo egipto como por emperadores emperatrices, reyes e ilustrados matemáticos, u hombres de ciencia para preservar secretos que no debían ser desvelados, se considera tan antigua como la escritura misma. Uno de los primeros métodos criptográficos si no el primero se le atribuye al general Julio Cesar, el libro más antiguo que se conoce sobre criptografía data del siglo XIV.

El considerado por muchos "*padre de la criptografía*" es Leon Battista Alberti matemático, arquitecto e intelectual del siglo XV, por sus obras "Tratado de cifras", publicada en 1470 y "Poligrafía" publicada en 1530.

A comienzos del siglo XX el uso de la criptografía en las transmisiones de mensajes cobra una importancia inusitada por los tiempos que corrían (Primera y Segunda Guerras Mundiales), originando esto un gran auge tanto de las técnicas como de las máquinas de cifrar, recuérdese la popular enigma y su contrapartida Bombes de donde provienen nuestros pcs actuales, como sistemas de estas máquinas.

Un gran ejemplo de cómo nos ha afectado la criptografía a lo largo de la historia es el que se produjo el 17 de enero de 1917, cuando William Montgomery, criptoanalista de la sección diplomática de la famosa Habitación 40 del Almirantazgo de la Marina Británica en Londres, intercepta un telegrama lleno de códigos que el Ministro de Relaciones Exteriores alemán Arthur Zimmermann envía a su embajador en los Estados Unidos.

Tras romper los códigos, descubren atónitos que entre otras cosas el mensaje anunciaba la guerra con los Estados Unidos. Con ello los Estados Unidos entran en la confrontación mundial y ayudan a los aliados a ganar la guerra. Según palabras de David Khan, autor de la obra más completa sobre historia de la criptografía: "Nunca un único criptoanálisis ha tenido tan enormes consecuencias". De hecho, el descubrimiento de este secreto cambió el rumbo de la historia. Y no es el único caso.

Según el pergamino virtual http://www.pergaminovirtual.com.ar/definicion/Criptografia.html la criptografía es: *“”””El procedimiento que permite asegurar la transmisión de informaciones privadas por las redes públicas desordenándola matemáticamente, (encriptándola), de manera que sea ilisible para cualquiera excepto para la persona que posea la "llave" que puede ordenar (desencriptar) la información. “””*

Otra buena definición sacada de las diapositivas Criptografía Clásica de Jorge Ramió Aguirre es: *“””Rama inicial de las Matemáticas y en la actualidad también de la Informática y la Telemática, que hace uso de métodos y técnicas con el objeto principal de cifrar, y por tanto proteger, un mensaje o archivo por medio de un algoritmo, usando una o más claves.*

Esto dará lugar a diferentes tipos de sistemas de cifra, denominados criptosistemas, que nos permiten asegurar al menos tres de los cuatro aspectos básicos de la seguridad informática: la confidencialidad o secreto del mensaje, la integridad del mensaje y autenticidad del emisor, así como el no repudio mutuo entre emisor (cliente) y receptor (servidor). ”””

La criptografía cumple las siguientes **finalidades**:

- Garantizar la privacidad o secreto en la comunicación entre dos entidades.
- Asegurar el no repudio tal como vimos en el tema uno. Al firmar el mensaje y utilizar una clave compartida garantizamos que quien nos lo envía es quien dice ser.
- Impedir que el contenido del mensaje o criptograma sea modificado en su tránsito.

La criptología abarca dos grandes ramas, el criptoanálisis y la criptografía, definiremos el criptoanálisis ya que de criptografía ya hablamos anteriormente: **Criptoanálisis**: Es la ciencia que estudia las técnicas y sistemas de descifrado de la información.

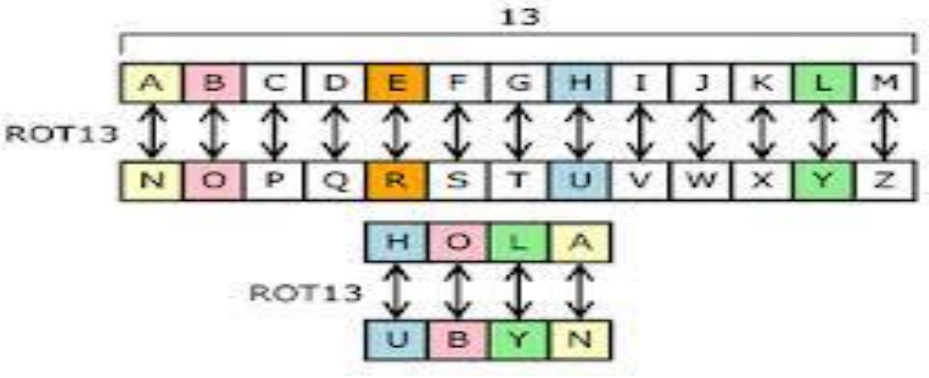
Criptografía clásica

Es la criptografía tradicional. Es muy utilizada en el estudio para hacer entender los métodos de cifrado. Las **dos técnicas** más sencillas de *cifrado*, en la criptografía clásica son la **sustitución**, que supone el cambio de significado de los elementos alfanuméricos del mensaje y la **trasposición,** que supone una reordenación de los mismos. La gran mayoría de los **cifrados** clásicos son combinaciones de estas dos operaciones básicas. Algunos de los algoritmos criptográficos más tradicionales son:

Algoritmo de Cesar: Llamado así por es el cifrado que utilizaba Julio César para enviar mensajes secretos, es uno de los algoritmos más simples que hay:
Por ejemplo a la letra A le corresponde la D, a la B, la letra E y así sucesivamente.

Algoritmo ROT13:

A partir de la treceava posición se empieza a contar el alfabeto para construir el cifrado de este modo la A pasa a ser la N la O pasa a ser la B y así sucesivamente. Como podéis ver no resulta tan complicado una vez se conoce la clave.



Algoritmo ROT47

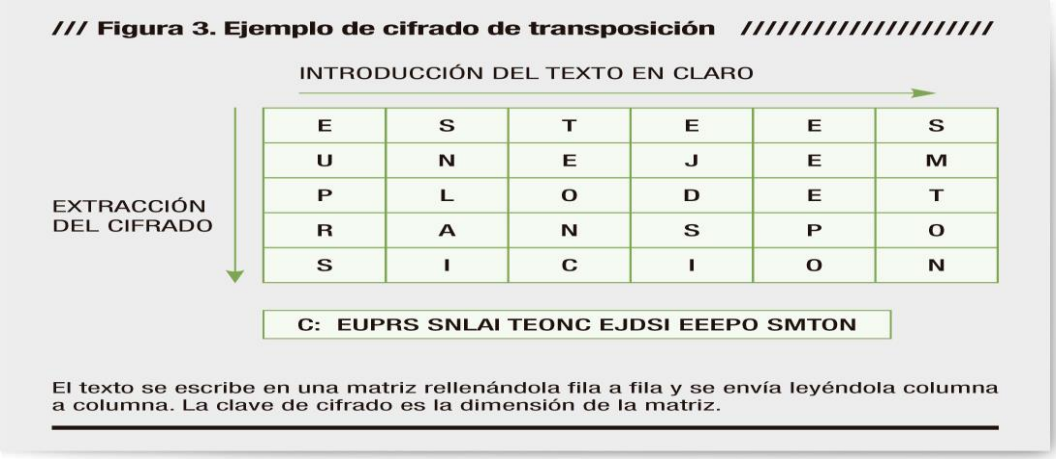
A diferencia del algoritmo Rot13, ROT47 toma el carácter que está 47 caracteres antes del carácter original, ya que toma en cuenta todos los caracteres imprimibles distinguiendo mayúsculas y minúsculas.

Sustitución por clave (Vigénere ,"Blaisede Vigénere ").

Se establece una correspondencia entre el alfabeto en claro y el cifrado, después de lo cual la asignación de caracteres se realiza teniendo en cuenta la posición del carácter en el mensaje y el dígito que le corresponde según la clave.

Trasposición

En este método no se sustituyen unos símbolos por otros si no que se cambia su orden dentro del texto.



4.2 Criptografía simétrica

Alejandro Coletti nos cuenta sobre la criptografía simétrica: *“”””Este tipo de algoritmo, emplea la misma clave para cifrar que para descifrar. Este algoritmo es el primero de todos, y se empleo casi con exclusividad hasta principios de los ochenta. Los ejemplos más conocidos son el DES, Triple DES, CAST, RC 4 y RC 5, Blowfish, IDEA, y CAST.*

Como ejemplo simple, se desea encriptar el siguiente:

Código ASCII: Mje fuente = {77, 69, 83, 65} = {MESA}
El algoritmo consiste en sumarle siete a cada símbolo (Clave = 7).
Mensaje Cripto = {84, 76, 90, 72} = {T, L, Z, H}
Para desencriptarlo, el algoritmo será la resta, y la Clave será la misma = 7.

Este ejemplo que a simple vista parece trivial, de hecho no lo es tanto pues fue utilizado durante muchos siglos. Se puede hacer la prueba de redactar un texto cualquiera y emplear distintas claves, comprobando que no es tan simple decodificarlo, también se puede incrementar la complejidad del algoritmo con distintas operaciones simples e inclusive con combinaciones de ellas, logrando paso a paso un grado de dificultad cada vez mayor.

Como conclusión, se puede apreciar que se emplea la misma clave para encriptar que para desencriptar, y el algoritmo es la inversa. Si se desea incrementar el grado de dificultad, se realiza fácilmente, incrementando la cantidad de operaciones y la longitud de la clave.

El gran problema radica en la forma en la cual se hace llegar la clave pues debería ser por otro medio de comunicaciones debido a que justamente este no es seguro. Este método posee la gran debilidad que esta clave no puede ser difundida pues a medida que más de dos personas conocen un secreto, este poco a poco va dejando de serlo. El gran inconveniente radica en que se está creyendo que la información es confidencial y sobre este concepto se fundamenta la

toma de decisiones, siendo esto más peligroso que si se es consciente que la información puede ser escuchada y se opera al respecto. """

Con lo cual, tal como hemos visto la criptografía simétrica consiste en cifrar un mensaje utilizando una clave compartida, clave que sirve tanto para cifrar como descifrar el mensaje. Como ya nos advierte Alejandro Coletti hay varias desventajas respecto a este tipo de cifrado:

Imaginemos que Alice y Bob quieren transmitirse un mensaje sin que Eve pueda llegar a saberlo:

No hay problema, eligen un tipo de cifrado y comparten su secreto entre ellos dos, sólo ellos conocen cual es la forma de descifrarlo por lo que de momento es seguro y Eve no puede leerlo.

Ahora llega Pedro que también desea comunicarse con Alice y Bob sin que Eve sepa lo que hablan, Alice y Bob deciden no compartir su secreto con Pedro porque entonces dejaría de ser secreto y los tres deciden otro tipo de cifrado, pero para intercambiarse la clave utilizan un canal no seguro, con lo cual Eve intercepta el mensaje y lo puede leer con tranquilidad, el problema es que Alice, Bob y Pedro siguen pensando que su mensaje está cifrado.

Alice, Bob y Pedro se dan cuenta y cambian su secreto pero Pedro resulta ser un chivato por lo que comparte su secreto con Marco, aunque la comunicación sigue estando cifrada ya no es ningún problema leerla.

Digamos que Alice y Bob se dan cuenta y vuelven a su clave original la cual no compartieron con Pedro por lo cual vuelven a hablar en privado entre ellos.

Con lo cual resumimos en que:

- Debemos utilizar un canal seguro en la criptografía simétrica para la transmisión del mensaje.
- Cuantas más personas conozcan el secreto, más inseguro se vuelve el cifrado.
- El trasmitir un mensaje cifrado con una clave simétrica garantiza la confidencialidad el mensaje

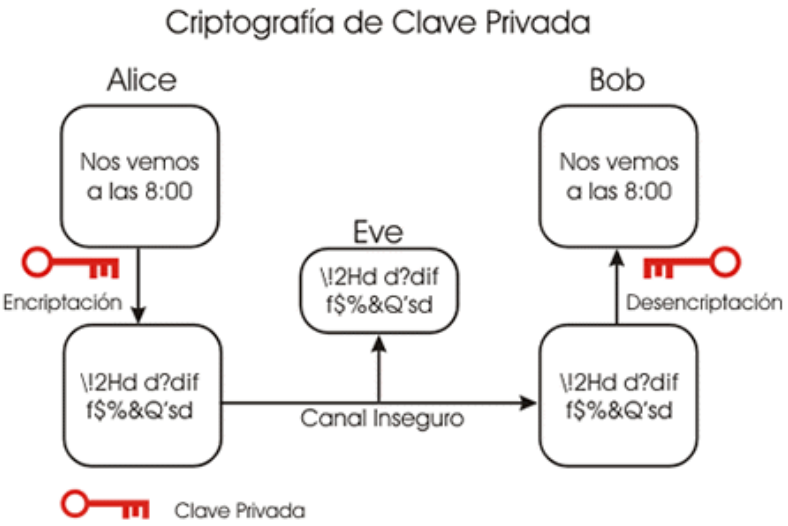
Los métodos de encriptación pueden dividirse en dos grandes grupos:

- Clave secreta (simétrica). utilizan un secreto compartido
- Clave pública (asimétrica): utilizan una clave pública y una privada.

El transmitir un mensaje con clave asimétrica garantiza autenticidad, integridad, confidencialidad en el envío y no repudio si van asociados a una firma digital. Los modernos algoritmos de encriptación simétricos mezclan la trasposición y la permutación, mientras que los de clave pública se basan más en complejas operaciones matemáticas, los vamos estudiar un poco más adelante.

Clave secreta (simétrica)

Utiliza una clave para la encriptación y desencriptación del mensaje. Esta clave se debe intercambiar entre los equipos por medio de un canal seguro. Ambos extremos deben tener la misma clave para cumplir con el proceso



Para que un algoritmo de este tipo sea considerado fiable debe cumplir algunos requisitos básicos:

- Conocido el criptograma (texto cifrado) no se pueden obtener de él ni el texto en claro ni la clave.
- Conocidos el texto en claro y el texto cifrado debe resultar más caro en tiempo o dinero descifrar la clave que el valor posible de la información obtenida por terceros.

Todos los sistemas criptográficos clásicos se pueden considerar simétricos, y los principales algoritmos simétricos actuales son DES, Triple DES, CAST, RC 4 y RC 5, Blowfish, IDEA, y CAST, **AES**.

Las principales desventajas de los métodos simétricos son la distribución de las claves, el peligro de que muchas personas deban conocer una misma clave y la dificultad de almacenar y proteger muchas claves diferentes.

Veamos algunos de ellos:

DES

El Algoritmo de encriptación DES trabaja con claves simétrica, fue desarrollado en 1977 por la empresa IBM, se basa en un sistema monoalfabético, con un algoritmo de cifrado consistente en la aplicación sucesiva de varias permutaciones y sustituciones.

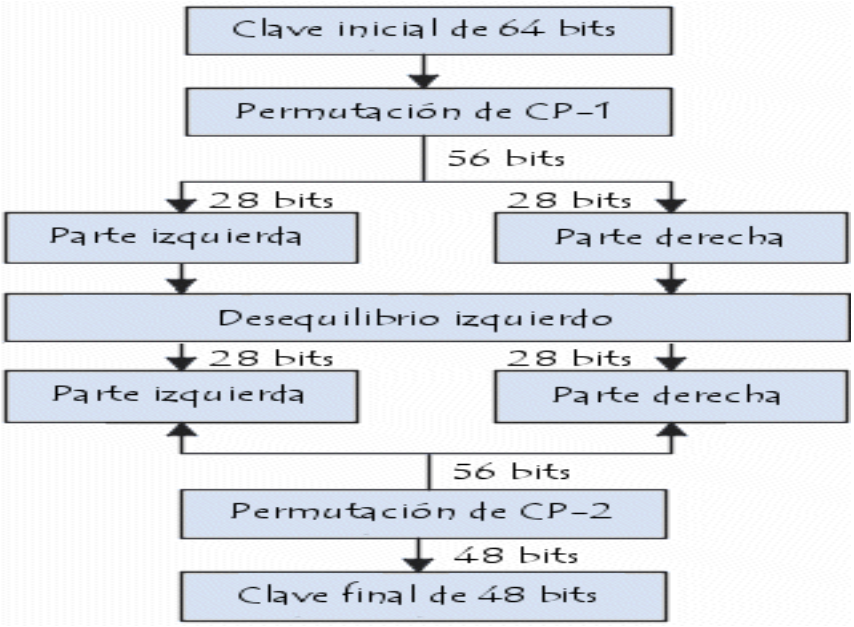
Inicialmente el texto a cifrar se somete a una permutación, con bloque de entrada de 64 bits (o múltiplo de 64), para posteriormente ser sometido a la acción de dos funciones

principales, una función de permutación con entrada de 8 bits y otra de sustitución con entrada de 5 bits, en un proceso que consta de 16 etapas de cifrado.

En general, DES utiliza una clave simétrica de 64 bits, de los cuales 56 son usados para la encriptación, mientras que los 8 restantes son de paridad, y se usan para la detección de errores en el proceso. DES ya no es estándar y fue crackeado en Enero de 1999 con un poder de cómputo que efectuaba aproximadamente 250 mil millones de ensayos en un segundo.

Actualmente se utiliza el Triple DES con una clave de 128 bits y que es compatible con el DES visto anteriormente. Este nuevo algoritmo toma una clave de 128 bits y la divide en dos de 64 bits cada una, de la siguiente forma:

Se le aplica al documento a cifrar un primer cifrado mediante la primera clave, C1.
Al resultado (denominado ANTIDES) se le aplica un segundo cifrado con la segunda clave, C2.
Y al resultado se le vuelve a aplicar un tercer cifrado con la primera clave, C1. RC5



Este sistema es el sucesor de RC4, que consistía en hacer un XOR al mensaje con un vector que se supone aleatorio y que se desprende de la clave, mientras que RC5 usa otra operación, llamada dependencia de datos, que aplica sifths a los datos para obtener así el mensaje cifrado.

IDEA

Trabaja con bloques de texto de 64 bits, operando siempre con números de 16 bits usando operaciones como XOR y suma y multiplicación de enteros. El algoritmo de desencriptación es muy parecido al de encriptación, por lo que resulta muy fácil y rápido de programar, y hasta ahora no ha sido roto nunca, aportando su longitud de clave una seguridad fuerte ante los ataques por fuerza bruta (prueba y ensayo o diccionarios).

Este algoritmo es de libre difusión y no está sometido a ningún tipo de restricciones o permisos nacionales, por lo que se ha difundido ampliamente, utilizándose en sistemas como UNIX y en programas de cifrado de correo como PGP. Un ejemplo de criptografía simétrica es el que utilizamos con TrueCryp.

4.3 Criptografía asimétrica Clave pública (asimétrica)

Se basa en el uso de dos claves diferentes, claves que poseen una propiedad fundamental: una clave puede desencriptar lo que la otra ha encriptado. Una de las claves de la pareja, llamada clave privada, es usada por el propietario para encriptar los mensajes, mientras que la otra, llamada clave pública, es usada para desencriptar el mensaje.

Las claves pública y privada tienen características matemáticas especiales, de tal forma que se generan siempre a la vez, por parejas, estando cada una de ellas ligada intrínsecamente a la otra.

Mientras que la clave privada debe mantenerla en secreto su propietario, ya que es la base de la seguridad del sistema, la clave pública es difundida, para que esté al alcance del mayor número posible de personas, existiendo servidores que guardan, administran y difunden dichas claves.

Para que un algoritmo de clave pública sea considerado seguro debe cumplir con los siguientes puntos:

- Conocido el texto cifrado no debe ser posible encontrar el texto en claro ni la clave privada.
- Conocido el texto cifrado (criptograma) y el texto en claro, debe resultar más caro en tiempo o dinero descifrar la clave que el valor posible de la información obtenida por terceros.
- Conocida la clave pública y el texto en claro no se puede generar un criptograma correcto encriptado con la clave privada.
- Dado un texto encriptado con una clave privada sólo existe una pública capaz de desencriptarlo, y viceversa.

El primer sistema de clave pública que apareció fue el de Diffie-Hellman, en 1976, y fue la base para el desarrollo de los que después aparecieron, entre los que cabe destacar el RSA (el más utilizado en la actualidad). Se basan en tres tipos de algoritmos matemáticos:

- Logaritmos entero discretos.
- Factorización de Números primos.
- Curvas elípticas.

Cada uno crea su pareja de claves, la clave pública la difunde para que todos aquellos que deseen crear un mensaje cifrado hacia nosotros puedan hacerlo. La clave privada sólo la conoceremos nosotros y será la que permita desencriptar el mensaje cifrado con nuestra clave pública.

Como se puede suponer, ambas claves están asociadas de alguna manera, conformando un PAR DE CLAVES, causa por la cual no es imposible partiendo desde la clave pública obtener la privada, pero en la actualidad aún es excesivamente cara la inversión de tiempo, recursos y algoritmos necesarios para hacerlo, y más aún a medida que se emplean claves extensas; esta cierta complejidad es la que hace a esta técnica altamente confiable y la convierte en la más segura que se emplea en la actualidad.

La lógica es la siguiente:

Mje fuente = {M}
Emite
CLAVE PÚBLICA <Mje fuente = {M}> = Mje cripto
Recibe
CLAVE PRIVADA <Mje cripto> = Mje fuente

En este método se soluciona el problema de la distribución de claves. El primer problema que se plantea es cómo se puede estar seguro que la clave pública que dice ser, realmente es; es decir una persona puede confiar en la guía telefónica que publica anualmente la o las Empresas de telefonía local porque sabe que las mismas fueron impresas por esta Empresa y es su responsabilidad ser veraces, también se puede confiar en un teléfono o dirección que me suministra una persona conocida siempre y cuando la misma esté considerada como "confiable"; pero qué sucedería si la clave pública que se obtiene no es en realidad la que se corresponde con el remitente al cual se le desea enviar un mensaje cifrado, como ejemplo se presenta el siguiente:

PROCEDER CORRECTO
A desea enviar un mensaje a B.
A busca en la guía G la clave pública de B, la cual será KB(Pub)
Redacta el mensaje M = Mje M.
Lo criptografía con la clave pública de B,
KB(Pub) {Mje M} = Mje cripto
Lo envía a B.
B lo recibe.
B lo decriptografía con su clave privada KB(Priv) ,
KB(Priv) {Mje cripto} = Mje M
Obteniendo el mensaje original, Mje M.

PROCEDER INCORRECTO
J publica su lista falsa de claves públicas.
A desea enviar un mensaje a B.
A busca en la guía J la clave pública de B, la cual será KB(Pub falsa)

Redacta el mensaje M = Mje M.
Lo criptografía con la clave pública falsa de B,
KB(Pub falsa) {Mje M} = Mje cripto falso
Lo envía a B.
J lo escucha y lo decriptografía con la clave privada falsa de B,
KB(Priv falsa) {Mje cripto falso} = Mje M
J criptografía el mensaje con la verdadera clave pública de B,
KB(Pub) {Mje M} = Mje cripto
J lo envía a B.
B lo recibe.
B lo decriptografía con su clave privada KB(Priv) ,
KB(Priv) {Mje cripto} = Mje M
Obteniendo el mensaje original, Mje M.

Nótese como un intruso obtuvo el mensaje original, el cual desde ya que si se realiza el proceder inverso, también se tomará conocimiento de la respuesta de B hacia A. Con lo cual es de suma importancia garantizar que la clave pública pertenece a quien nosotros creemos que pertenece y es de confianza.

Para garantizar una veraz distribución lo haremos a través de listas conocidas de distribución que puedan garantizar la consistencia de sus datos o lo que es más eficiente a través de la seguridad individual de cada emisor, el cual deberá estar plenamente convencido de la confiabilidad de las fuentes de obtención, lo cual puede ser a través del propio receptor telefónicamente, vía terceros que son de confianza, dependencias dentro de la organización que garanticen, etc.

El segundo gran problema que presenta este método es la demora que introduce (llegando a ser en algunos casos hasta mil veces más lentos que las técnicas simétricas), y el mayor volumen de información que genera. Ambos problemas son naturales en todo proceso que se desee optimizar la seguridad, como regla general:

SIEMPRE QUE SE INCREMENTA LA SEGURIDAD, SE INTRODUCEN DEMORAS.

Esto da origen al cifrado híbrido pero antes de pasar a explicarla veamos con detenimiento algunos de estos algoritmos.

Diffie-Hellman

Su importancia se debe sobre todo al hecho de ser el inicio de los sistemas asimétricos, ya que en la práctica sólo es válido para el intercambio de claves simétricas, y con esta funcionalidad es muy usado en los diferentes sistemas seguros implementados en Internet, como SSL (Secure Socket Layer) y VPN (Virtual Private Network).

Matemáticamente se basa en las potencias de los números y en la función mod (módulo discreto). Uniendo estos dos conceptos se define la potencia discreta de un número como Y = Xa mod q.

Si bien el cálculo de potencias discretas es fácil, la obtención de su función inversa, el logaritmo discreto, no tiene una solución analítica para números grandes.

RSA

RSA es el más conocido y usado de los sistemas de clave pública, y también el más rápido de ellos. Presenta todas las ventajas de los sistemas asimétricos, incluyendo la firma digital, aunque resulta más útil a la hora de implementar la confidencialidad el uso de sistemas simétricos, por ser más rápidos. Se suele usar también en los sistemas mixtos para encriptar y enviar la clave simétrica que se usará posteriormente en la comunicación cifrada.

El sistema RSA se basa en el hecho matemático de la dificultad de factorizar números muy grandes. Para factorizar un número el sistema más lógico consiste en empezar a dividir sucesivamente éste entre 2, entre 3, entre 4,..., y así sucesivamente, buscando que el resultado de la división sea exacto, es decir, de resto 0, con lo que ya tendremos un divisor del número.

El cálculo de estas claves se realiza en secreto en la máquina en la que se va a guardar la clave privada, y una vez generada ésta conviene protegerla mediante un algoritmo criptográfico simétrico.

4.4 Criptografía híbrida

La criptografía híbrida es un método criptográfico que se basa en los tipos de criptografía más relevantes: el sistema simétrico y el asimétrico. Este tipo de criptografía utiliza el cifrado de clave pública para compartir una clave para el cifrado simétrico. El mensaje que se esté enviando en el momento, se cifra usando la clave y enviarlo así al destinatario. Esto es así puesto que compartir una clave simétrica no resulta seguro, la clave usada es diferente para cada sesión.

Un ejemplo de criptografía híbrida es el sistema empleado por PGP. La clave de sesión es cifrada con la clave pública, y el mensaje saliente es cifrado con la clave simétrica, todo combinado en un solo miembro. El destinatario usa su clave privada para descifrar la clave de sesión y seguidamente emplea la clave de sesión para descifrar el mensaje.

Un sistema de cifrado híbrido no es más fuerte que el sistema de cifrado asimétrico o el de cifrado simétrico. En PGP, el sistema de clave pública es probablemente la parte más débil de la combinación. Sin embargo, si un atacante pudiera descifrar una clave de sesión, sólo sería útil para poder leer un mensaje, el cifrado con esa clave de sesión. El atacante tendría que volver a empezar y descifrar otra clave de sesión para poder leer cualquier otro mensaje.

NOTA: Esta técnica en general para optimizar la seguridad se suele emplear generando en cada sesión una clave simétrica en forma aleatoria, es decir para cada mensaje se implementará una clave simétrica diferente. Esta mejora permite incrementar el algoritmo pues aún en el caso de lograr descifrar la clave simétrica, ésta sola sería de utilidad para ese mensaje y nada más.

Cifrado Irreversible:

Este procedimiento, consiste en poder criptografiar código, pero si bien puede existir un método de desencriptado, este no se difunde o no se emplea. Su aplicación más común se puede observar en la forma en que los sistemas operativos de red suelen tratar las cuentas de usuario y contraseñas, como por ejemplo Unix o Windows NT. Estos sistemas operativos, al crear una cuenta de usuario de red, la guardan en archivos dentro de alguna estructura de directorios en forma criptografiada.

Al hacerse presente este usuario en la red, solicita validarse ante un servidor, colocando su nombre de usuario y contraseña. El o los servidores que reciben esta petición, aplican el mismo algoritmo de encriptado y comparan los resultados, si son los mismos códigos, lo reconocen como usuario de red, caso contrario le niegan el acceso. Lo importante a tener en cuenta es que en ningún momento se compara texto plano, sino código criptografiado, por lo tanto no se necesita el procedimiento inverso para desencriptar.

Lamentablemente este tipo de cifrado tiene poca difusión, pues tiene la enorme potencia de no poder volver atrás. Si se tiene en cuenta este detalle, se podría implementar en discursos políticos, programas de TV, etc, en los cuales sería importantísimo que ni siquiera el que generó el discurso o diálogo pueda volver a repetirlo.

Métodos de verificación de Integridad (HMAC – SHA y MD5)

HMAC (Hashing for Message Authentication Codes) [RFC-2104]

Para proveer un modo de comprobar la integridad de la información transmitida o almacenada en un medio no confiable es necesario un mecanismo que permita compararla contra algo que se considere válido. Para esto se estandarizó un procedimiento basado en una clave secreta usualmente llamado Código de Autenticación de Mensajes (MAC).

El empleo típico de estos mecanismos es a través de dos partes que comparten una clave secreta para validar la información transmitida entre ellas.

HMAC propone el empleo de criptografía aplicada a funciones Hash (resúmenes).

En la RFC, estandariza el empleo de HMAC con las funciones Hash definidas como MD5 (Message Digest versión 5) [RFC-1321] y SHA-1 (Standard Hash Algorithm Versión 1) [FIPS 180-1]. También hace mención al algoritmo propuesto por RIPE denominado RIPEMD-128/160. Hoy ya ha salido la nueva versión SHA-256, de 256 bits.

HMAC requiere una función Hash (H) que se encargará de comprimir un texto de longitud finita por medio de iteraciones de una función de compresión básica sobre los bloques de datos (B = 64 Byte), y una clave secreta (K); y por medio de ambas se obtendrá un resumen de longitud fija (L), que será de 16 Byte para MD5 y 20 Byte para SHA-1.

Esta función Hash es llamada "One Way" pues no es posible a través del resumen de salida obtener el texto de entrada, también resultará computacionalmente imposible obtener un valor de salida igual a través de otro valor de entrada, como así tampoco desde un valor de salida ya calculado, obtener otro valor de entrada diferente al verdadero.

Se definen dos cadenas de longitud fija, diferentes una de la otra, llamadas:

Ipad = repetición del Byte 36 B veces.
Opad = repetición del Byte 5C B veces.
Luego se ejecuta: H { KB xor Opad, H (KB xor Ipad, texto)}

Para esta tarea se debe tener en cuenta:

- a)Rellenar con ceros la clave K hasta obtener una longitud de 64 Byte (llamada KB).
- b)Realizar: KB xor Ipad , (Result1).
- c)Anexar el texto completo al resultado de b, (Result2).
- d)Aplicar la función Hash (H) al resultado de c, (Result3).
- e)Realizar: KB xor Opad, (Result4).
- f)Anexar el resultado de d, (Result3). Al resultado de e, (Result4).
- g)Aplicar la función Hash (H) al resultado de f. Obteniendo el resultado, que acorde a la función Hash empleada será de 16 o 20 Byte.

The MD5 Message-Digest Algorithm Request for Comments: 1321

Este algoritmo toma un mensaje de entrada de longitud arbitraria y entrega una salida de 128 bit de longitud fija. Llamado "Huella digital" o "Recopilación de mensaje (Message Digest). Es computacionalmente imposible producir dos mensajes que posean la misma recopilación, como tampoco regenerar el mensaje a través de la recopilación. Este algoritmo puede ser empleado para aplicaciones de firma digital, donde un texto debe ser comprimido de manera segura antes de ser encriptado con sistemas de clave privada.

PGP (Pretty Good Privacy).

Y hemos hablado de pgp anteriormente pero lo veremos en detalle en la parte práctica. Pgp utiliza:
1. Clave pública y privada.
2. Verificación y validación de claves.
3. Grado de confianza.
4. Administración de claves.
5. Importación y exportación de claves.
6. Criptografía y firma electrónica.
7. PGP Net.
8. Asociaciones de seguridad.
9. Archivos autodesencriptables.
10. Borrado de archivos.

Sistema de autenticación Kerberos:
Este sistema nace en el Instituto Tecnológico de Massachusetts y su nombre se remonta a la mitología Griega donde así se denominaba el perro guardián de los Dioses. Este sistema de autenticación hoy es soportado por la masa de los sistemas operativos y componentes de red.

El sistema Kerberos está basado en un "Servidor despachador de boletos", al cual se encarga de validar la identidad de los "Principales" los cuales pueden ser:

- Usuarios.
- Servicios.

En cualquiera de los dos casos, un "Principal" queda definido por un "Trío" cuyos componentes son:

- Nombre primario: Nombre de persona o servicio.
- Instancia: Para usuarios es nula o contiene información de ésta. Para un servicio es el nombre de la máquina.
- Reino: Define distintos Dominios de autenticación.

Si un "Principal" obtiene un boleto, este tendrá un tiempo de vida limitado por el Servidor, y a partir de este poseerá una clave privada que sólo conocerán el Principal y el Servidor, por lo tanto será considerada auténtica y a través de esta podrá acceder a los recursos del sistema.

4.5 Firmas digitales

Una firma digital se logra mediante una Función Hash de Resumen. Esta función se encarga de obtener una "muestra única" del mensaje original. Dicha muestra es más pequeña y es muy difícil encontrar otro mensaje que tenga la misma firma. Suponiendo que B envía un mensaje m firmado a A, el procedimiento es:

a. B genera un resumen del mensaje r(m) y lo cifra con su clave privada.
b. B envía el criptograma.
c. A genera su propia copia de r(m) usando la clave pública de B asociada a la privada.
d. A compara su criptograma con el recibido y si coinciden el mensaje es auténtico.

Cabe destacar que:

1. Cualquiera que posea la clave pública de B puede constatar que el mensaje proviene realmente de B.
2. La firma digital es distinta en todos los documentos: si A firma dos documentos produce dos criptogramas distintos y; si A y B firman el mismo documento m también se producen dos criptogramas diferentes.

Las funciones Hash están basadas en que un mensaje de longitud arbitraria se transforma en un mensaje de longitud constante dividiendo el mensaje en partes iguales, aplicando la función de transformación a cada parte y sumando todos los resultados obtenidos. Actualmente se recomienda utilizar firmas de al menos 128 bits (38 dígitos) siendo 160 bits (48 dígitos) el valor más utilizado.

4.6 Certificados digitales

Un certificado digital es un documento digital mediante el cual un tercero confiable (una autoridad de certificación) garantiza la vinculación entre la identidad de un sujeto o entidad y su clave pública. Puede ser usado, por ejemplo, para verificar la firma digital de dicha entidad.

Un certificado contiene usualmente, entre otros datos:

- Nombre y datos burocráticos identificativos de la entidad certificada
- Número de serie
- Fecha de expiración
- Copia de la clave pública de la entidad certificada (utilizada para verificar su firma digital)
- Firma digital de la autoridad certificadora (de forma que el receptor pueda verificar que esta última ha establecido realmente la asociación)

En caso particular, por ejemplo, de un servidor web, el certificado que ofrece a los navegadores cuando un usuario visita su web garantiza que, durante la comunicación

encriptada que realice el navegador gracias a la clave pública obtenida de este servidor web, dicho servidor es realmente la que dice ser (siempre y cuando dicho certificado venga firmado por una autoridad de certificación confiable por el cliente). Los navegadores ya vienen de serie con una lista de firmas reconocidas de varias autoridades de certificación confiables (los llamados "certificados raíz")...por eso en muchas páginas seguras no salta ningún popup extraño: porque su certificado fue firmado por una autoridad reconocida por el navegador. De hecho, el navegador comprueba varias cosas para dar por válida la conexión y comenzar la comunicación encriptada:

- El certificado obtenido no haya expirado ya
- El certificado obtenido ha sido creado por el mismo servidor web al que se está accediendo
- El certificado está firmado por alguien en el que se confía

Cualquier individuo o institución puede generar certificados digitales y convertirse así en una autoridad de certificación, pero si no es reconocido por quienes interactúen con el certificado, el valor del mismo es prácticamente nulo (servirá para hacer pruebas o para un grupo de usuarios que confíen en nuestra máquina). Por ello las autoridades de certificación deben ser reconocidas como institución certificadoras por la entidad pública competente, de forma que su firma pueda ser reconocida como fiable, transmitiendo esa fiabilidad a los certificados emitidos por la citada institución.

En España, las entidades que otorgan validez a las autoridades de certificación son la Fabrica Nacional de Moneda y Timbre, el Ministerio de Industria, Turismo y Comercio, la Agencia Catalana de Certificación, etc Una autoridad de certificación gratuita es CAcert.org. La mayoría, no obstante, son comerciales: Thawte, Verisign, RSA, etc El formato más extendido para certificados digitales es el estándar UIT-T X.509 (son los ficheros con extensión .crt).

4.7 SSL/TLS La herramienta criptográfica multiusos

Secure Socket Layer es un sistema de protocolos de carácter general diseñado en 1994 por la empresa Nestcape Communcations Corporation, y está basado en la aplicación conjunta de Criptografía Simétrica, Criptografía Asimétrica (de llave pública), certificados digitales y firmas digitales para conseguir un canal o medio seguro de comunicación a través de Internet.
De los sistemas criptográficos simétricos, motor principal de la encriptación de datos transferidos en la comunicación, se aprovecha la rapidez de operación, mientras que los sistemas asimétricos se usan para el intercambio seguro de las claves simétricas, consiguiendo con ello resolver el problema de la Confidencialidad en la transmisión de datos.

Ssl es utilizado para cualquier comunicación donde deba establecerse un canal seguro (al solicitarse clave o número de tarjeta de crédito por ejemplo).

En la pila TCP/IP, se ubica entre la capa TCP (Transporte) y la de Aplicación, por lo que es muy flexible ya que puede ser utilizado en cualquier aplicación que utilice TCP/IP (Mail, HTTP, FTP, News, etc.) aunque actualmente sólo se implementa sobre HTTP.

Para diferenciar las páginas comunes HTTP de las protegidas se utiliza la denominación HTTPS conectado mediante el puerto 443. SSLv3 supera algunas limitaciones de sus versiones anteriores y ofrece estas características:

- Cifrado de datos: los datos viajan cifrados mediante algunos de los algoritmos vistos. Para el intercambio de datos entre servidor y cliente se utilizan algoritmos simétricos (DES–TDES, RC4, IDEA) y para la clave de sesión (utilizada para los algoritmos anteriores) cifrado asimétrico (típicamente RSA).
- Fragmentación de datos: en el emisor se fragmentan los datos en bloques para volver a reeensemblarlos en el receptor.
- Compresión de datos: se puede aplicar un algoritmo de compresión a los datos.
- Autentificación de servidores: el usuario puede verificar la identidad del servidor al que se conecta y al que puede mandar datos confidenciales.
- Integridad de mensajes: las modificaciones intencionales o accidentales, de la información, en el viaje por el canal inseguro son detectadas.
- Autentificación del cliente: permite al servidor conocer la identidad del usuario, con el fin de decidir si este puede acceder a cierta información protegida. Esta autentificación no siempre debe darse.

Al reunir estas características, la comunicación se realiza en **dos fases**:

- Saludo (Handshaking): los interlocutores se identifican mutuamente empleando, habitualmente, certificados X.509. Tras el intercambio de claves públicas, los dos escogen una clave de sesión simétrica para el intercambio de datos.
- Comunicación: se produce el intercambio de información propiamente dicho, que se codifica mediante las claves de sesión ya establecidas.

De aquí en adelante, durante la sesión segura abierta, SSL proporciona un canal de comunicaciones seguro entre el servidor y el cliente a través del cual se intercambiará cifrada la siguiente información:

- La URL del documento solicitado.
- El contenido del documento solicitado.
- Los contenidos de cualquier formulario enviado desde el navegador.
- Las cookies enviadas desde el navegador al servidor y viceversa.
- Los contenidos de las cabeceras HTTP.

SSL 3.0 usa los algoritmos simétricos de encriptación DES, TRIPLE DES, RC2, RC4 e IDEA, el asimétrico RSA, la función hash MD5 y el algoritmo de firma SHA-1. Los algoritmos, longitudes de clave y funciones hash de resumen, usados en SSL dependen del nivel de seguridad que se busque o se permita, siendo los más habituales los siguientes:

- **RSA + Triple DES de 168 bits + SHA-1**: soportado por las versiones 2.0 y 3.0 de SSL, es uno de los conjuntos más fuertes en cuanto a seguridad, ya que son posibles 3.7 * 1050 claves simétricas diferentes, por lo que es muy difícil de romper. Por ahora sólo está permitido su uso en Estados Unidos, aplicándose sobre todo en transacciones bancarias.
- **RSA + RC4 de 128 bits + MD5**: soportado por las versiones 2.0 y 3.0 de SSL, permite 3.4 * 10 38 claves simétricas diferentes que, aunque es un número inferior que el del caso anterior, da la misma fortaleza al sistema. Análogamente, en teoría sólo se permite su uso comercial en Estados Unidos, aunque actualmente ya es posible su implementación en los navegadores más comunes, siendo usado por organismos gubernamentales, grandes empresas y entidades bancarias.
- **RSA + RC2 de 128 bits + MD5**: soportado sólo por SSL 2.0, permite 3.4 * 10 38 claves simétricas diferentes, y es de fortaleza similar a los anteriores, aunque es más lento a la hora de operar. Sólo se permite su uso comercial en Estados Unidos, aunque actualmente ya es posible su implementación en los navegadores más comunes.
- **RSA + DES de 56 bits + SHA-1**: soportado por las versiones 2.0 y 3.0 de SSL, aunque es el caso de la versión 2.0 se suele usar MD5 en vez de SHA-1. Es un sistema menos seguro que los anteriores, permitiendo 7.2 * 10 16 claves simétricas diferentes, y es el que suelen traer por defecto los navegadores web en la actualidad (en realidad son 48 bits para clave y 8 para comprobación de errores).
- **RSA + RC4 de 40 bits + MD5**: soportado por las versiones 2.0 y 3.0 de SSL, ha sido el sistema más común permitido para exportaciones fuera de Estados Unidos. Permite aproximadamente 1.1 * 10 12 claves simétricas diferentes, y una velocidad de proceso muy elevada, aunque su seguridad es ya cuestionable con las técnicas de Criptoanálisis actuales.
- **RSA + RC2 de 40 bits + MD5**: en todo análogo al sistema anterior, aunque de velocidad de proceso bastante inferior.
- **Sólo MD5**: usado solamente para autentificar mensajes y descubrir ataques a la integridad de los mismos. Se usa cuando el navegador cliente y el servidor no tienen ningún sistema SSL común, lo que hace imposible el establecimiento de una comunicación cifrada. No es soportado por SSL 2.0, pero si por la versión 3.0.

La clave de encriptación simétrica es única y diferente para cada sesión, por lo que si la comunicación falla y se debe establecer una nueva sesión SSL, la contraseña simétrica se generará de nuevo. SSL proporciona cifrado de alto nivel de los datos intercambiados (se cifran incluso las cabeceras HTTP), autenticación del servidor (y si es necesario también del cliente) e integridad de los datos recibidos.

Durante el proceso de comunicación segura SSL existen dos estados fundamentales, el estado de sesión y el estado de conexión. A cada sesión se le asigna un número identificador arbitrario, elegido por el servidor, un método de compresión de datos, una serie de algoritmos de encriptación y funciones hash, una clave secreta maestra de 48 bytes y un flag de nuevas conexiones, que indica si desde la sesión actual se pueden establecer nuevas conexiones. Cada conexión incluye un número secreto para el cliente y otro para el servidor, usados para calcular los MAC de sus mensajes, una clave secreta de encriptación

particular para el cliente y otra para el servidor, unos vectores iniciales en el caso de cifrado de datos en bloque y unos números de secuencia asociados a cada mensaje.

¿Cómo podemos saber si una conexión se está realizando mediante SSL? Generalmente los navegadores disponen de un icono que lo indica, generalmente un candado en la parte inferior de la ventana. Si el candado está abierto se trata de una conexión normal, y si está cerrado de una conexión segura. Si hacemos deble click sobre el candado cerrado nos aparecerá el Certificado Digital del servidor web seguro.

Además, las páginas que proceden de un servidor SSL vienen implementadas mediante protocolo HTTP seguro, por lo que su dirección, que veremos en la barra de direcciones del navegador, empezará siempre por https, como por ejemplo:

https://www.evidaliahost.com

Por último, cuando estamos en una conexión segura podemos ver el certificado del servidor acudiendo al menú "Archivo" del navegador y pinchando en "Propiedades". En la parte inferior tenemos una opción "Certificados", que nos mostrará el del servidor actual.

Limitaciones y Problemas de SSL

1. Debido a la limitación de exportación del gobierno de los EE.UU. sobre los productos criptográficos, las versiones de los navegadores distribuidas legalmente más allá de sus fronteras operan con nada más que 40 bits de longitud de clave, frente a los 128 ó 256 bits de las versiones fuertes.

Claves tan cortas facilitan los ataques de fuerza bruta, dependiendo de los recursos informáticos disponibles. Este serio problema ganó notoriedad en los medios de comunicación cuando en 1995 un estudiante francés, Damien Doligez, fue capaz de descifrar un mensaje cifrado con SSL en pocos días utilizando la red de computadoras de su Universidad.

2. SSL sólo garantiza la confidencialidad e integridad de los datos en tránsito, pero nunca antes ni después. Por lo tanto, si se envían datos personales al servidor, SSL solamente asegura que no serán modificados ni espiados mientras viajan desde el navegador hasta el servidor. Lo que el servidor haga con ellos, está más allá de la competencia de este protocolo.

3. SSL no garantiza la identidad del servidor al que se conecta el usuario. Podría suceder que el servidor seguro contase con un certificado perfectamente válido y que estuviera suplantando la identidad de algún otro servidor seguro bien conocido.
Por consiguiente, es de extrema importancia que se compruebe siempre el certificado del sitio web para cerciorarse de que no se está conectando a un web falsificado.

4. El servidor identifica al navegador incluso aunque éste no se autentique mediante certificados. Cuando un usuario se conecta a un servidor, rutinariamente le comunica ciertos datos como su dirección IP, tipo y versión de navegador, sistema operativo, y otros.

5. Actualmente SSL solamente se utiliza para comunicaciones web seguras, por lo que otros servicios de Internet, como el correo electrónico, no irán cifrados a pesar de utilizar SSL para el envío de formularios o la recuperación de páginas web. Por esto, se debe usar S/MIME, PGP o algún otro software criptográfico para correo.

Ventajas de SSL

1. SSL v3.0 goza de gran popularidad y se encuentra ampliamente extendido en Internet, ya que viene soportado por los dos principales navegadores del mercado, Netscape Navigator© e Internet Explorer©,Mozilla,Chrome,Opera.

2. SSL proporciona un canal de comunicaciones seguro entre los servidores web y los clientes (los navegadores), pero su uso no se limita a la transmisión de páginas web. Al encontrarse entre los niveles de transporte y de aplicación, potencialmente SSL puede servir para securizar otros servicios, como FTP, correo, telnet, etc.

3. El usuario no necesita realizar ninguna acción especial para invocar el protocolo SSL, basta con seguir un enlace o abrir una página cuya dirección empieza por https://. El navegador se encarga del resto.

TLS

Transport Layer Security es un protocolo estandarizado por el IETF75. Está basado en SSL v3 (y es totalmente compatible) pero incorpora algunas mejoras y se destaca por no ser de una empresa privada.

4.8 Navegación segura: HTTPS

Hypertext Transfer Protocol Secure (en español: Protocolo seguro de transferencia de hipertexto). HTTPS, es un protocolo de aplicación basado en el protocolo HTTP, destinado a la transferencia segura de datos de Hipertexto, es decir, es la versión segura de HTTP.

El sistema HTTPS utiliza un cifrado basado en SSL/TLS para crear un canal cifrado (cuyo nivel de cifrado depende del servidor remoto y del navegador utilizado por el cliente) más apropiado para el tráfico de información sensible que el protocolo HTTP.

De este modo se consigue que la información sensible (usuario y claves de paso normalmente) no pueda ser usada por un atacante que haya conseguido interceptar la transferencia de datos de la conexión (man in the middle), ya que lo único que obtendrá será un flujo de datos cifrados que le resultará imposible de descifrar.

Configuración del Servidor

Para preparar un servidor web que acepte conexiones HTTPS, el administrador debe crear un Certificado de clave pública para el servidor web. Este certificado debe estar firmado por una Autoridad de certificación para que el navegador web lo acepte. La autoridad certifica que el titular del certificado es quien dice ser. Los navegadores web generalmente son distribuidos con los certificados raíz firmados por la mayoría de las Autoridades de Certificación por lo que estos pueden verificar certificados firmados por ellos.

Adquisición de certificados

Adquirir certificados puede ser gratuito1 (generalmente sólo si se paga por otros servicios), o costar entre US$132 y US$1,5003 por año.

Las organizaciones pueden también ser su propia autoridad de certificación, particularmente si son responsables de establecer acceso a navegadores de sus propios sitios (por ejemplo, sitios en una compañía intranet, o universidades mayores). Estas pueden fácilmente agregar copias de su propio certificado firmado a los certificados de confianza distribuidos con el navegador. También existen autoridades de certificación peer-to-peer.