Componentes de seguridad

Las condiciones que debe reunir una comunicación segura a través de Internet (o de otras redes) son en general las siguientes:

- Confidencialidad: evita que un tercero pueda acceder a la información enviada.
- Integridad: evita que un tercero pueda modificar la información enviada sin que lo advierta el destinatario.
- Autenticación: permite a cada lado de la comunicación asegurarse de que el otro lado es realmente quien dice ser.
- No repudio: Permite a cada lado de la comunicación probar fehacientemente que el otro lado ha participado en la comunicación. En el caso de no repudio de origen, el remitente del mensaje no puede negar haberlo enviado. En el caso de no repudio de destino, el destinatario del mensaje no puede negar haberlo recibido.

La herramienta básica para cumplir las condiciones anteriores son las técnicas criptográficas, en particular los métodos de cifrado simétrico (usan una misma clave secreta para cifrar y descifrar) o asimétrico (cada usuario tiene una pareja de claves, una pública y otra privada, con la propiedad de que lo que se cifra con una de las claves sólo se puede descifrar con la otra).

Criptografía simétrica

Utiliza una misma clave para cifrar y para descifrar mensajes. Las dos partes que se comunican se ponen de acuerdo de antemano sobre la clave a usar. Una vez que ambas tienen acceso a esta clave, el remitente cifra un mensaje usándola, lo envía al destinatario, y éste lo descifra con la misma.

Un buen sistema de cifrado pone toda la seguridad en la clave y ninguna en el algoritmo. En otras palabras, no debería ser de ninguna ayuda para un hacker o cracker conocer el algoritmo que se está usando. Sólo si el atacante obtuviera la clave, le serviría conocer el algoritmo. Los algoritmos de cifrado usados por ejemplo en el sistema GNU, GnuPG tienen estas propiedades.

Dado que toda la seguridad está en la clave, es importante que sea muy difícil descifrar el tipo de clave. Esto quiere decir que el abanico de claves posibles, es decir, el espacio de posibilidades de claves, debe ser amplio.

Actualmente los computadores y servidores pueden adivinar claves con extrema rapidez, y ésta es la razón por la cual el tamaño de la clave es importante en los criptosistemas modernos. El algoritmo de cifrado DES usa una clave de 56 bits, lo que significa que hay 2 elevado a 56 claves posibles. 2 elevado a 56 son 72.057.594.037.927.936 claves. Esto representa un número muy alto de claves, pero un PC de uso general puede comprobar todo el espacio posible de claves en cuestión de días. Una máquina especializada lo puede hacer en horas. Por otra parte, algoritmos de cifrado de diseño como 3DES, Blowfish e IDEA usan todos claves de 128 bits, lo que significa que existen 2 elevado a 128 claves posibles.

Esto representa muchas más claves, y aun en el caso de que todos los PCs del planeta estuvieran cooperando, todavía tardarían más tiempo que la misma edad del universo en encontrar la clave. Incluso en la actualidad se pueden encontrar en el mercado claves a 256 bits, 512 bits y más.

El principal problema con los sistemas de cifrado simétrico no está ligado a su seguridad, sino al intercambio de claves. Una vez que el remitente y el destinatario hayan intercambiado las claves pueden usarlas para comunicarse con seguridad, pero ¿qué canal de comunicación que sea seguro han usado

para transmitirse la clave entre sí? Sería mucho más fácil para un atacante intentar interceptar una clave que probar las posibles combinaciones del espacio de claves. Es aquí donde entran la criptografía asimétrica y la criptografía híbrida.

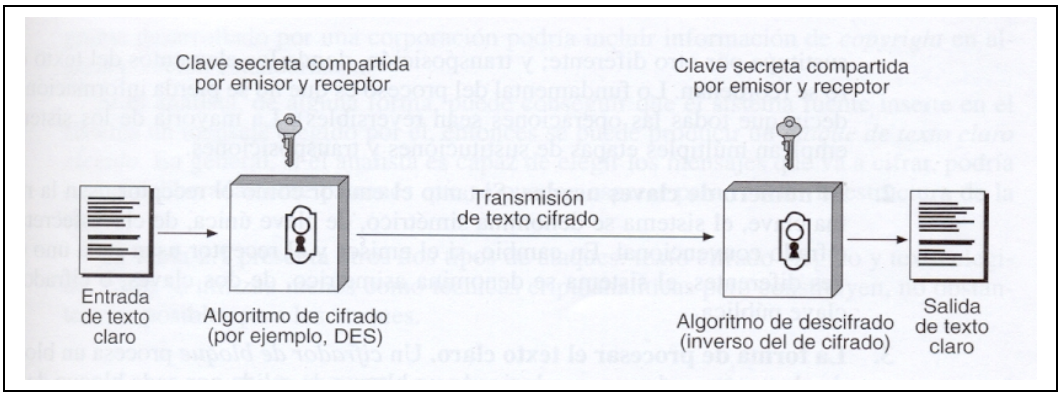

Figura 1

Modelo simplificado del cifrado convencional

Algoritmos de cifrado simétrico

Los algoritmos de cifrado simétrico más comúnmente usados son los cifradores de bloques. Un cifrador de bloques procesa la entrada de texto claro en bloques de tamaño fijo y genera un bloque cifrado del mismo tamaño para cada texto claro.

DES (Data Encryption Standard)

El esquema de cifrado más extendido se basa en el DES (Data Encryption Standard) adoptado en 1977 por el Nacional Bureau of Standards, ahora el NIST (Nacional Institute of Standards and Technology), como Federal Information Processing Standard 46.

- Descripción del algoritmo

El texto claro tiene una longitud de 64 bits y la clave, de 56; si el texto en claro es más largo se procesa en cloques de 64 bits. La estructura del DES consiste en una pequeña variación de la red de Feistel, que se muestra en la figura 2. Hay 16 etapas de proceso. Se generan 16 subclaves partiendo de la clave original de 56 bits, una para cada etapa.

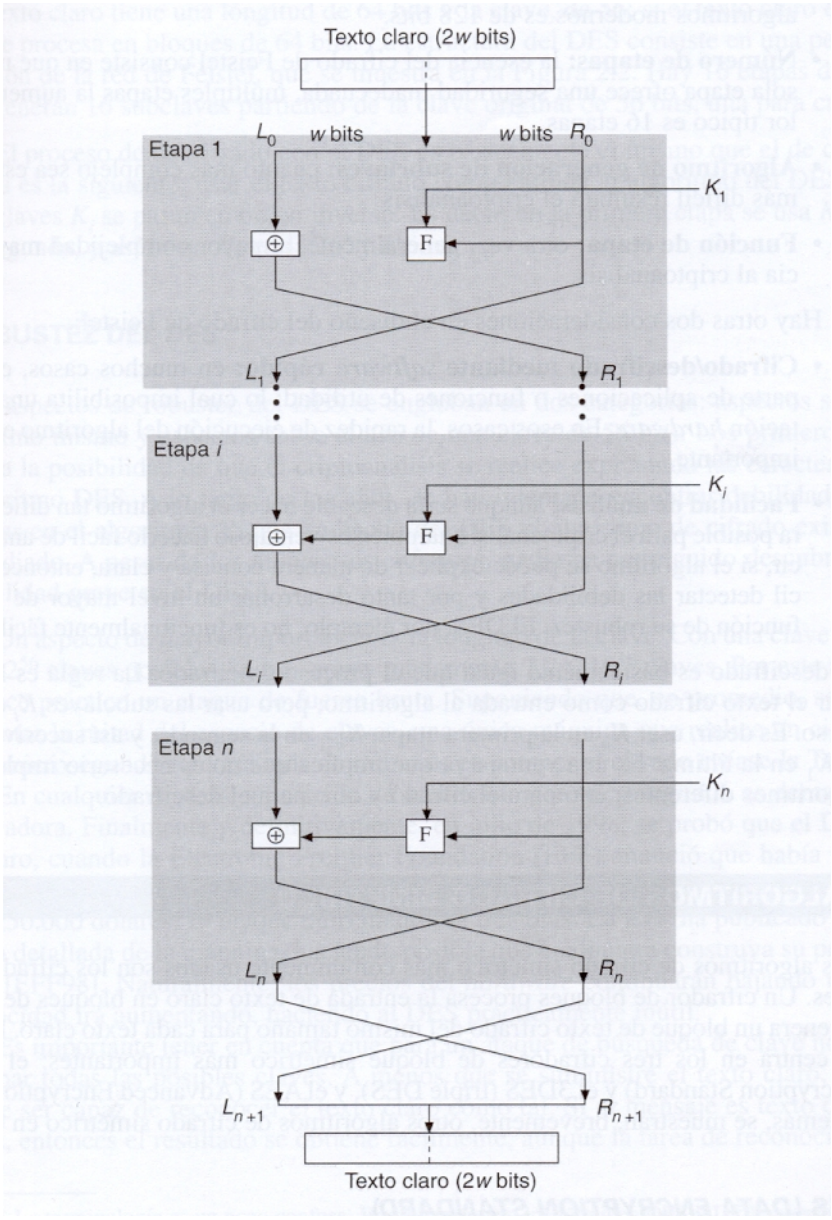

Figura 2. Red clásica de Feistel

El proceso de descifrado con el DES es básicamente el mismo que el de cifrado. La regla es la siguiente: usar el texto cifrado como entrada al algoritmo del DES, pero las subclaves K_i se pasan en orden inverso. Es decir, en la primera etapa se usa K_{16}, K_{15} en la segunda y así sucesivamente hasta K_1 en la 16ª y última.

- Robustez del DES

Los aspectos de robustez del DES se engloban en dos categorías: aspectos sobre el algoritmo mismo y aspectos sobre el uso de una clave de 56 bits. Los primeros se refieren a la posibilidad de que el criptoanálisis se realice explotando las características del algoritmo DES. A lo largo de los años, se han intentado encontrar debilidades que explotar en el algoritmo, lo que ha hecho del DES el algoritmo de

cifrado existente más estudiado. A pesar de los numerosos enfoques, nadie ha conseguido descubrir ninguna debilidad grave en el DES.

Un aspecto de mayor importancia es la longitud de la clave. Con una clave de 56 bits, hay 2^{56} claves posibles, que es aproximadamente 7,2 X 10^{16} claves. Por este motivo, no parece práctico un ataque de fuerza bruta. Suponiendo que, en promedio, se tiene que intentar la mitad del espacio de claves, una única máquina que realice un cifrado DES por microsegundo tardaría más de mil años en romper el cifrado.

En cualquier caso, la suposición de un cifrado por microsegundo es demasiado conservadora. Finalmente y definitivamente, en julio de 1998, se probó que el DES no era seguro, cuando la Electronic Frontier Foundation (EFF) anunció que había roto un cifrado DES utilizando una máquina especializada <<DES craker>>, construida por menos de 250.000 dólares. El ataque duró menos de tres días. La EFF ha publicado la descripción detallada de la máquina, haciendo posible que cualquiera construya su propia craker. Naturalmente, los precios del hardware continuarán bajando mientras la velocidad irá aumentando, haciendo al DES prácticamente inútil.

Es importante tener en cuenta que para que un ataque de búsqueda de clave no basta con probar todas las posibles claves. A menos que se suministre el texto claro, el analista debe ser capaz de reconocer el texto claro como tal. Si el mensaje es texto claro en inglés, entonces el resultado se obtiene fácilmente, aunque la tarea de reconocimiento del inglés tendría que estar automatizada. Si el mensaje de texto se ha comprimido antes del cifrado, entonces el reconocimiento es más difícil. Y si el mensaje es de un tipo más general de datos, como un fichero numérico, y ha sido comprimido, el problema es aún más difícil de automatizar. Pero eso, para complementar el enfoque de fuerza bruta, se necesita algún grado de conocimiento sobre el texto claro esperado y alguna forma de distinguir automáticamente el texto claro de lo que no lo es. El enfoque de la EFF trata también este tema, e introduce algunas técnicas automatizadas que serían efectivas en muchos contextos.

Como punto final, si la única forma de ataque a un algoritmo de cifrado es la fuerza bruta, entonces la manera de contrarrestar este ataque es obvia: usar claves más largas. Para tener una idea del tamaño de clave necesario, usaremos el craker de la EFF como base de nuestras estimaciones. El craker de la EFF era un prototipo, y se puede suponer que con la tecnología actual es rentable construir una máquina más rápida. Si asumimos que un dispositivo como el craker puede realizar un millón de descifrados por us, entonces se tardaría alrededor de diez horas en descifrar un código DES. Esto constituye un incremento de velocidad aproximadamente un factor de siete comparado con los resultados de la EFF. Usando este ratio, la figura 3 muestra cuanto tardaría en romper un algoritmo del estilo del DES en función del tamaño de la clave. Por ejemplo, para una clave de 128 bits, que es común entre los algoritmos actuales, se tardaría 10^{18} años en romper el código usando el craker de la EFF. Incluso, aunque se aumentara la velocidad del craker en un factor de un trillón (10^{12}), todavía se tardaría un millón de años en romper el código. Así que una clave de 128 bits garantiza que el algoritmo es inexpugnable por la fuerza bruta.

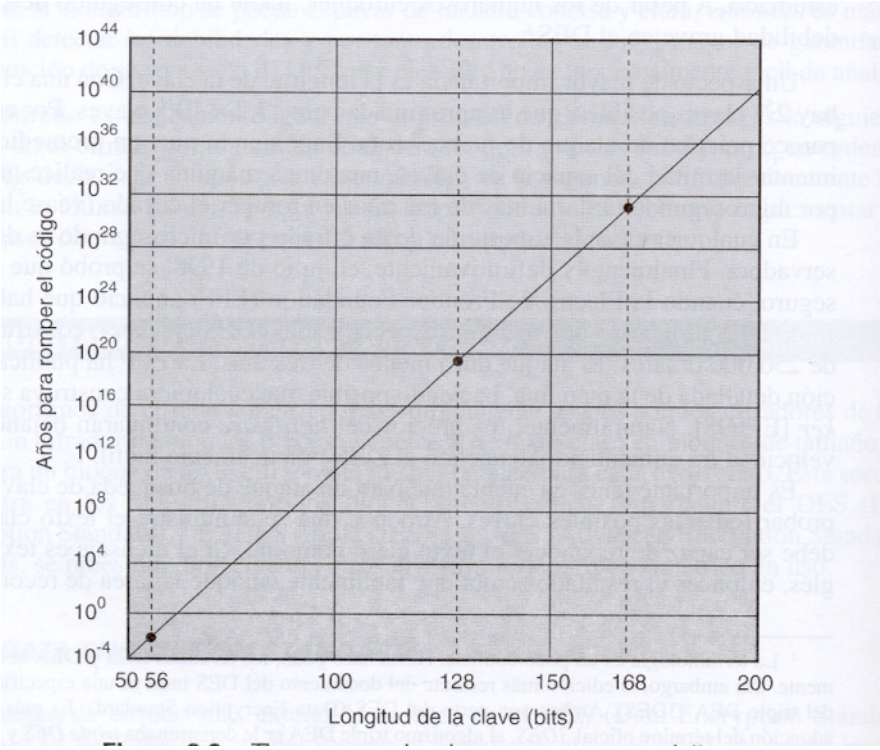

Figura 3

Tiempo empleado en romper un código

(Suponiendo 10^6 descifrados/us)

Triple DES

El triple DES (3DES) se estandarizó inicialmente para aplicaciones financieras en el estándar ANSI X9.17 en 1985. el 3DES se incorporó como parte del DES en 1999, con la publicación de FIPS PUB 46-3.

El 3DES usa tres claves y tres ejecuciones del algoritmo DES. La función sigue la secuencia cifrar-descifrar-cifrar (EDE: encrypt-decrypt-encrypt) figura 4a:

$$C = E_{K3} [D_{K2} [E_{K1} [P]]]$$

Donde

C = texto cifrado

P = texto claro

$E_K [X]$ = cifrado de X usando la clave K

$D_K [Y]$ = descifrado de Y usando la clave K

El descifrado es simplemente la misma operación con las claves en orden inverso (figura 4b):

$$P = D_{K1}[\,E_{K2}[D_{K3}[C]]]$$

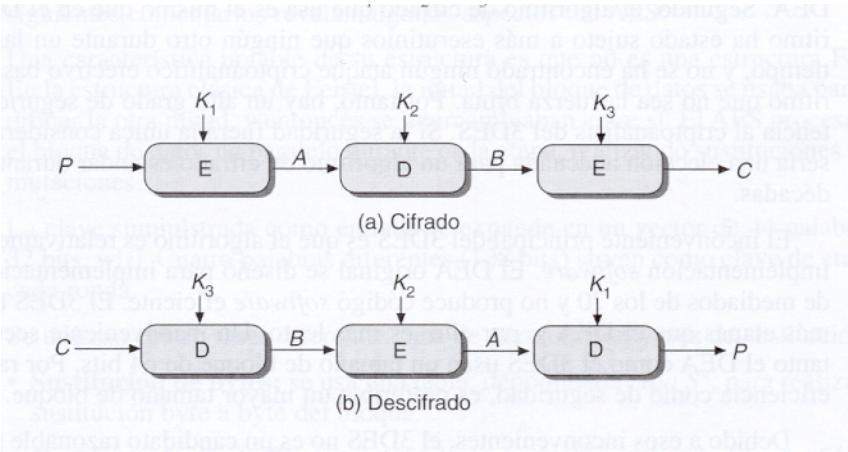

Figura 4

Triple DES

El descifrado del segundo paso no es significativo en términos criptográficos. Su única ventaja es que permite a los usuarios del 3DES descifrar datos cifrados por usuarios del DES:

$$C = E_{K1}[\,D_{K1}[E_{K1}[P]]] = E_{K1}[P]$$

Con tres claves diferentes, el 3DES tiene una longitude efectiva de clave de 168 bits. El FIPS 46-3 también permite el uso de dos claves, con $K_1 = K_3$, lo que proporciona una longitud de clave de 112 bits. El FIPS 46-3 incluye las siguientes directrices para el 3DES:

- El 3DES es el algoritmo de cifrado simétrico oficial del FIPS.
- El DES original, que usa una única clave de 56 bits, se mantiene sólo para los sistemas existentes. Las nuevas adquisiciones deberían admitir 3DES.
- Se apremia a las organizaciones gubernamentales con sistemas que usan DES a migrar a 3DES.
- Se prevé que el 3DES y el AES (Advanced Encryption Standard) coexistirán como algoritmos oficiales del FIPS, permitiendo una transición gradual hacia el AES.

Es fácil observar que el 3DES es un algoritmo robusto. Debido a que el algoritmo criptográfico que lo sustenta es el DES, el 3DES resulta igual de resistente al criptoanálisis basado en el algoritmo que el DES. Es más, con una clave de 168 bits de longitud, los ataques de fuerza bruta son efectivamente imposibles.

AES (Advanced Encryption Standard)

El 3DES tiene dos atractivos que aseguran su uso durante los próximos años. Primero, con su longitud de clave de 168 bits evita la vulnerabilidad al ataque de fuerza bruta del DES. Segundo, el algoritmo de cifrado que usa es el mismo que en el DES. Este algoritmo ha estado sujeto a más escrutinios que ningún

otro durante un largo periodo de tiempo, y no se ha encontrado ningún ataque criptoanalítico efectivo basado en el algoritmo que no sea la fuerza bruta. Por lo tanto, hay un alto grado de seguridad en la resistencia al criptoanálisis del 3DES. Si la seguridad fuera la única consideración, el 3DES sería una elección adecuada para un algoritmo de cifrado estándar durante las primeras décadas.

El inconveniente principal del 3DES es que el algoritmo es relativamente lento en su implementación software. El DES original se diseñó para implementaciones hardware de mediados de los 70 y no produce código software eficiente. El 3DES tiene tres veces más etapas que el DES y, por ello es más lento.

Debido a este inconveniente, el 3DES no es un candidato razonable para usarlo durante mucho tiempo. Para reemplazarlo, el NIST realizó en 1997 un concurso de propuestas para el desarrollo de un nuevo estándar de cifrado avanzado (AES), que debería ser tan robusto o más que el 3DES y que mejoraría significativamente la eficiencia. Además de esos requisitos generales, el NIST especificó que el AES debía ser un cifrador simétrico de bloque con una longitud de bloque de 128 bits y permitir longitudes de clave de 128, 192 y 256 bits.

- Descripción del algoritmo

El AES usa una longitud de bloque de 128 bits y la longitud de la clave puede ser de 128, 192 o 256 bits. En la descripción de esta sección se asume una longitud de clave de 128 bits, que posiblemente es la más implementada.

La figura 5 muestra la estructura genera del AES. La entrada a los algoritmos de cifrado y descifrado es un solo bloque de 128 bits. En el FIPS PUB 197, este bloque se representa como una matriz cuadrada de bytes. Este bloque se copia en el vector Estado, que se modifica en cada etapa del cifrado o descifrado. Después de la última etapa, Estado se copia en una matriz de salida. De igual manera, la clave de 128 bits se representa como una matriz cuadrada de bytes. Esta clave luego se expande en un vector de palabras para la generación de claves; cada palabra tiene cuatro bytes, y el número total de palabras para generar claves es de 44 para la clave de 128 bits. El orden de los bytes dentro de una matriz se establece por columnas. Así, por ejemplo, los primeros cuatro bytes de una entrada de texto claro de 128 bits al cifrador ocupan la primera columna de la matriz in, los segundos cuatro bytes la segunda columna, y así sucesivamente. De igual forma, los primeros cuatro bytes de la clave expandida, que forman una palabra, ocupan la primera columna de la matriz w.

Los siguientes comentarios revelan algunos aspectos del AES:

1.- Una característica notable de su estructura es que no es una estructura Feistel. En la estructura clásica de Feistel, la mitad del bloque de datos se usaba para modificar la otra mitad, y entonces se intercambiaban entre sí. El AES procesa todo el bloque de datos en paralelo durante cada etapa, realizando sustituciones y permutaciones.

2.- La clave suministrada como entrada se expande en un vector de 44 palabras de 32 bits, w[i]. Cuatro palabras diferentes (128 bits) sirven como clave de etapa en cada ronda.

3.- se utilizan cuatro fases diferentes, una de permutación y tres de sustitución:

* Sustitución de bytes: se usa una tabla, denominada caja S^4, para realizar una sustitución byte a byte del bloque.

* Desplazamiento de filas: una simple permutación realizada fila por fila.

* Mezcla de columnas: una sustitución que altera cada byte de una columna en función de todos los bytes de la columna.

*Suma de la clave de etapa: una simple operación XOR bit a bit del bloque actual con una porción de la clave expandida.

4.- La estructura es muy simple. Tanto para el cifrado como para el descifrado, se comienza con una fase de suma de clave de etapa, seguido de nueve etapas de cuatro fases cada una, y acaba con una décima etapa de tres fases. La figura 6 muestra la estructura de una etapa completa de cifrado.

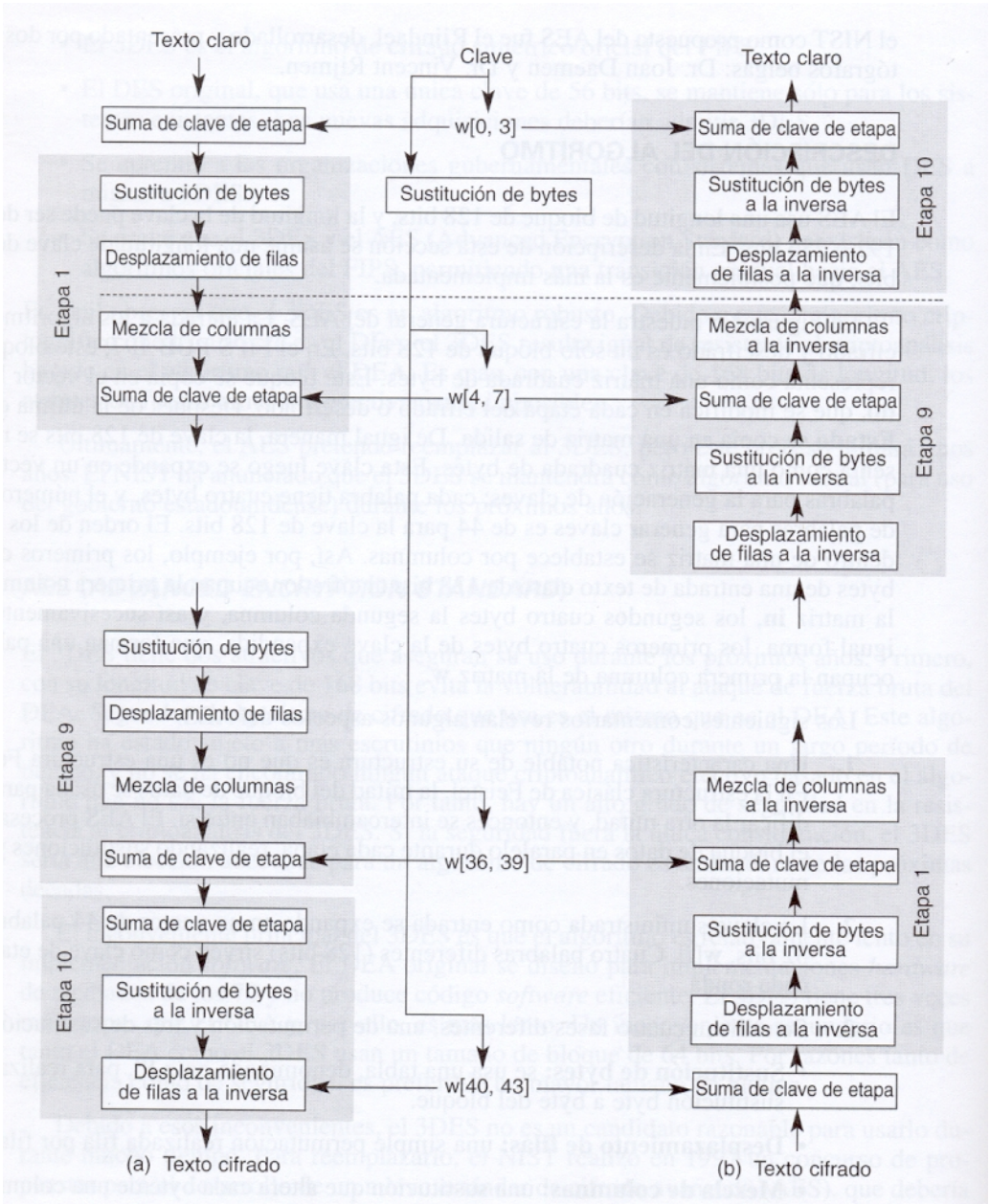

Figura 5

Cifrado y descifrado AES

5.- Solamente la fase de suma de la clave de etapa utiliza la clave. Por esta razón el cifrador comienza y termina con una suma de clave de etapa. Cualquier otra fase, aplicada al comienzo o al final, sería reversible sin conocer la clave y por tanto añadiría inseguridad.

6.- La fase de suma de la clave de etapa no funcionaría por sí misma. Las otras tres fases juntas desordenan los bits, pero no proporcionan seguridad por sí mismas, porque no usan la clave. Se puede ver el cifrador como una secuencia alternativa de operaciones de cifrador XOR (suma de clave de etapa) de un bloque, seguida por un desordenamiento del bloque (las otras tres fases), seguida por un cifrado XOR, y así sucesivamente. Este esquema es eficiente y muy seguro.

7.- cada fase es fácilmente reversible. Para las fases de sustitución de byte, desplazamiento de fila y mezcla de columnas, se usa una función inversa en el algoritmo de descifrado. Para la fase de suma de clave de etapa, la inversa se consigue con un XOR entre la misma clave de etapa y el bloque, usando la propiedad de que A $\ominus$ A $\ominus$ B = B.

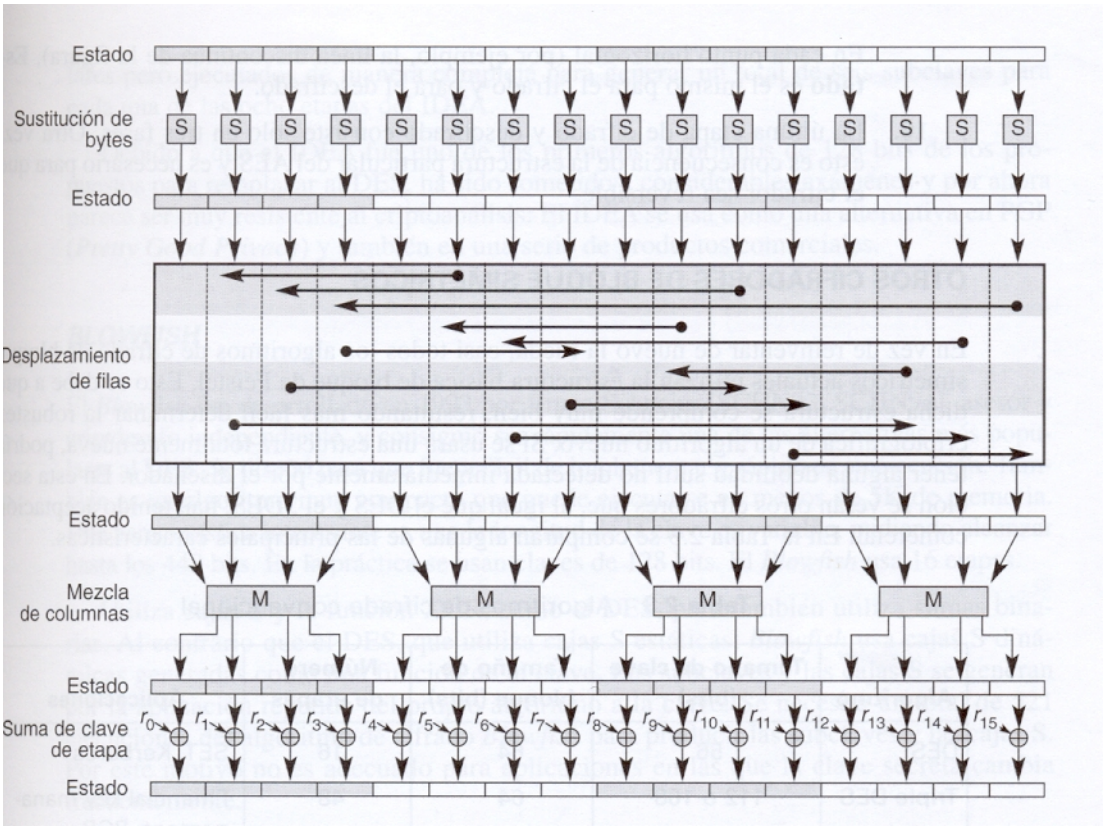

Figura 6

Etapa de cifrado del AES

8.- Como con la mayoría de los cifradores de bloque, el algoritmo de descifrado hace uso de la clave expandida en orden inverso. De todas formas, como consecuencia de la estructura particular del AES, el algoritmo de descifrado no es idéntico al de cifrado.

9.- Una vez se ha establecido que las cuatro fases de cada etapa son reversibles, es fácil verificar que el descifrador recupera el texto claro. La figura 5 muestra el cifrado y el descifrado desplazándose en direcciones verticalmente opuestas. En cada punto horizontal (por ejemplo, la línea discontinua la figura), Estado es el mismo para el cifrado y para el descifrado.

10.- La última etapa de cifrado y descifrado consiste sólo en tres fases. Otra vez, esto es consecuencia de la estructura particular del AES y es necesario para que el cifrador sea reversible.

Otros cifradores de bloque simétricos

En vez de reinventar de nuevo la rueda, casi todos los algoritmos de cifrado de bloque simétricos actuales utilizan la estructura básica de bloque de Feistel. Esto se debe a que dicha estructura se comprende muy bien, resultado más fácil determinar la robustez criptográfica de un algoritmo nuevo. Si se usara una estructura totalmente nueva, podría tener alguna debilidad sutil no detectada inmediatamente por el diseñador. En la tabla 1 se comparan algunas de las principales características.

Algoritmo	Tamaño de clave (bits)	Tamaño de bloque (bits)	Número de etapas	Aplicaciones
DES	56	64	16	SET, Kerberos
Triple DES	112 o 168	64	48	Financial Key management, PGP, S/MIME
AES	128, 192 o 256	128	10, 12 o 14	Destinados a sustituir DES y 3DES
IDEA	128	64	8	PGP
Blowfish	Variable hasta 448	64	16	Varios paquetes de software
RC5	Variable hasta 2048	64	Variable hasta 255	Varios paquetes de software

Tabla 1

Algoritmos de cifrado convencional

IDEA

El IDEA usa una clave de 128 bits. Difiere notablemente del DES en la función de etapa así como en la función de generación de subclaves. Para la función de etapa el IDEA no usa cajas S, sino que cuenta con tres operaciones matemáticas diferentes: XOR, suma binaria de enteros de 16 bits, y multiplicación binaria de enteros de 16 bits. Esas funciones se combinan de tal forma que producen una transformación compleja muy difícil de analizar, y por ende muy difícil para el criptoanálisis. El algoritmo de generación de subclaves se basa solamente en el uso de desplazamientos circulares pero ejecutados de manera compleja para generar un total de seis subclaves para cada una de las ocho etapas del IDEA.

Debido a que el IDEA fue uno de los primeros algoritmos de 128 bits de los propuestos para remplazar al DES, ha sido sometido a considerables exámenes y por ahora parece ser muy resistente al criptoanálisis. El IDEA se usa como una alternativa en PGP (Pretty Good Privacy) y también en una serie de productos comerciales.

Blowfish

El Blowfish consiguió ser rápidamente una de las alternativas más populares al DES. Se diseñó para que fuera fácil de implementar y rápido en su ejecución. También es un algoritmo muy compacto que puede ejecutarse en menos de 5K de memoria. Una característica interesante es que la longitud de la clave es

variable, pudiendo alcanzar hasta los 448 bits. En la práctica se usan claves de 128 bits. El Blowfish usa 16 etapas.

Utiliza cajas S y la función XOR, como el DES, pero también utiliza sumas binarias. Al contrario que el DES, que utiliza cajas S estáticas, Blowfish usa cajas S dinámicas generadas como una función de la clave. Las subclaves y las cajas S se generan por la aplicación repetida del propio algoritmo a la clave. Se necesita un total de 521 ejecuciones del algoritmo de cifrado Blowfish para producir las subclaves y las cajas S. Por este motivo no es adecuado para aplicaciones en las que la clave secreta cambia frecuentemente.

Este es uno de los algoritmos de cifrado simétrico más robusto hasta la fecha, porque tanto las subclaves como las cajas S se generan por un proceso de aplicaciones repetidas del propio algoritmo, lo cual modifica totalmente los bits haciendo muy difícil el criptoanálisis. Hasta ahora, se han publicado algunos artículos sobre Blowfish, sin que se hayan encontrado debilidades.

RC5

El RC5 se diseñó para tener las siguientes características:

- *Adecuado para hardware y software*: sólo usa operaciones computacionales primitivas que se encuentran comúnmente en los microprocesadores.
- *Rápido*: para conseguir esto, el RC5 es un algoritmo simple y orientado a palabras. Las operaciones básicas procesan palabras enteras de datos cada vez.
- *Adaptable a procesadores con diferentes tamaños de palabra*: el número de bits en una palabra es un parámetro del RC5; diferentes longitudes de palabra producen algoritmos diferentes.
- *Número variable de etapas*: el número de etapas es un segundo parámetro. Esto permite alcanzar un compromiso entre mayor rapidez y mayor seguridad.
- *Longitud de clave variable*: la longitud de la clave es un tercer parámetro. Otra vez, posibilita un acuerdo entre velocidad y seguridad.
- *Simple*: la estructura simple del RC5 es fácil de implementar y facilita la tarea de determinar la robustez del algoritmo.
- *Bajo consumo de memoria*: la poca necesidad de memoria hace que el RC5 sea adecuado para tarjetas inteligentes y otros dispositivos con restricciones de memoria.
- *Alta seguridad*: proporciona alta seguridad con los parámetros adecuados.
- *Rotaciones dependientes de los datos*: incorpora rotaciones (desplazamientos circulares de bits) cuya cantidad depende de los datos. Esto parece fortalecer el algoritmo contra el criptoanálisis.

Principios de criptografía de clave pública

De igual importancia que el cifrado convencional es el cifrado de clave pública, que se emplea en autenticación de mensajes y en distribución de claves.

Estructura del cifrado de clave pública

El cifrado de clave pública, propuesto por primera vez por Diffie y Hellman en 1976, es el primer avance realmente revolucionario en el cifrado en miles de años. El motivo es que los algoritmos de clave pública están basados en funciones matemáticas y no en simples operaciones sobre los patrones de bits. Además, la criptografía de clave pública es asimétrica, lo que implica el uso de dos claves separadas, a diferencia del cifrado simétrico convencional, que emplea sólo una clave. El uso de dos claves tiene importantes consecuencias en el terreno de la confidencialidad, la distribución de claves y la autenticación.

Antes de continuar, deberíamos mencionar algunas confusiones comunes en lo relativo al cifrado de clave pública. La primera es la creencia de que el cifrado de clave pública es más seguro ante el criptoanálisis que el cifrado convencional. De hecho, la seguridad de cualquier esquema de cifrado depende de la longitud de la clave y el coste computacional necesario para romper un cifrado. No hay nada sobre el cifrado convencional ni de clave pública que haga a uno superior al otro en lo que respecta a la resistencia al criptoanálisis. Otra equivocación la hallamos en la idea de que el cifrado de clave pública es una técnica con propósitos generales que ha dejado desfasado el cifrado convencional. Por el contrario, debido al coste computacional de los esquemas actuales de cifrado de clave pública, no parece que el cifrado convencional vaya a abandonarse. Por último, se piensa que la distribución de claves no es importante cuando se usa el cifrado de clave pública, en comparación con los incómodos acuerdos previos que tienen lugar con los centros de distribución de claves para el cifrado convencional. De hecho, se necesita alguna forma de protocolo, que a menudo implica a un agente central, y los procedimientos que tienen lugar no son más sencillos ni más eficientes que los que se requieren para el cifrado convencional.

Un esquema de cifrado de clave pública tiene seis componentes (figura 7a):

- Texto claro: consiste en el mensaje o los datos legibles que se introducen en el algoritmo como entrada.
- Algoritmo de cifrado: el algoritmo de cifrad realiza diferentes transformaciones en el texto claro.
- Clave pública y privada: es una pareja de claves que han sido seleccionadas, de las cuales una se usa para el cifrado y la otra para el descifrado. Las transformaciones exactas llevadas a cabo por el algoritmo de cifrado dependen de la clave pública o privada que se proporciona como entrada.
- Texto cifrado: es el mensaje desordenado producido como salida. Depende del texto claro y de la clave. Para un mensaje dado, dos claves diferentes producirán dos textos cifrados diferentes.
- Algoritmo de descifrado: este algoritmo acepta el texto cifrado y la clave correspondiente y produce el texto claro original.

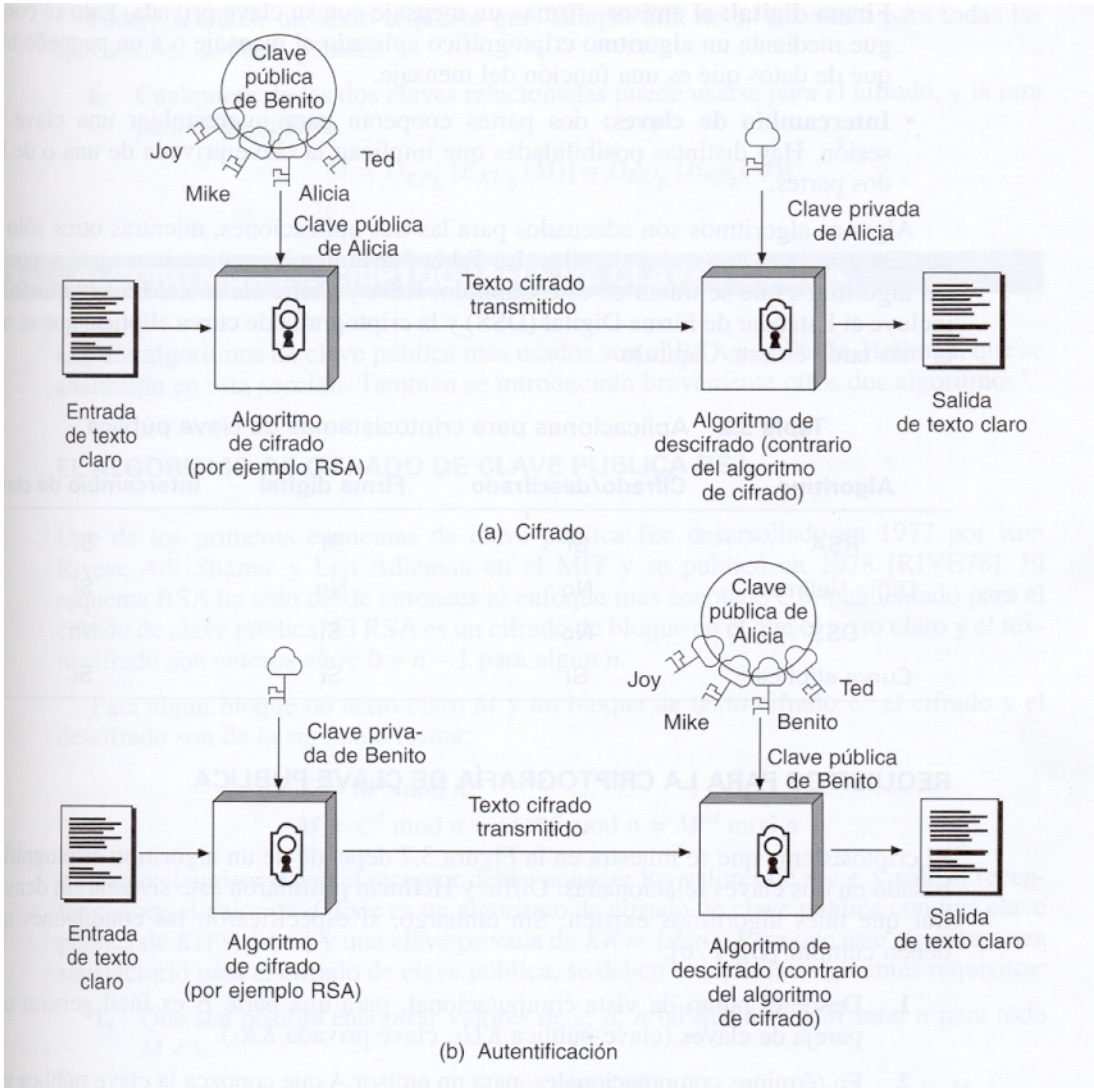

Figura 7

Criptografía de clave pública

Como sugieren los nombres, la clave pública de dicha pareja de claves se hace pública para que otros la usen, mientras que la clave privada sólo es conocida por su propietario. Un algoritmo criptográfico de clave pública con propósito general se basa en una clave para el cifrado y otra diferente, aunque relacionada, para el descifrado.

Los pasos fundamentales son los siguientes:

1.- Cada usuario genera una pareja de claves para el cifrado y el descifrado de mensajes.

2.- Cada usuario localiza una de las dos claves en un registro público u otro archivo accesible. Esta es la clave pública. La otra clave no se revela. Como sugiere la figura 7ª, cada usuario mantiene un grupo de claves públicas que han obtenido de otros.

3.- Si Benito quiere enviar un mensaje privado a Alicia, cifra el mensaje usando la clave pública de Alicia.

4.- Cuando Alicia recibe el mensaje lo descifra usando su clave privada. Ningún otro receptor puede descifrar el mensaje porque sólo Alicia conoce su clave privada.

En este enfoque, todos los participantes tienen acceso a las claves públicas, y las claves privadas las genera cada participante de forma local y, por tanto, nunca necesitan ser distribuidas.

Mientras el usuario proteja su clave privada, la comunicación entrante es segura. En cualquier momento un usuario puede cambiar la clave privada y publicar la clave pública que la acompaña para sustituir la clave pública antigua.

Aplicaciones para criptosistemas de clave pública

Los sistemas de clave pública se caracterízan por el uso de un tipo de algoritmo criptográfico con dos claves, una no se revela y la otra sí. Dependiendo de la aplicación, el emisor usa su clave privada o la clave pública del receptor, o las dos, para realizar algún tipo de función criptográfica. En términos generales, podemos clasificar el uso de criptosistemas de clave pública en tres categorías:

- Cifrado/descifrado: el emisor cifra un mensaje con la clave pública del receptor.
- Firma digital: el emisor firma un mensaje con su clave privada. Esto se consigue mediante un algoritmo criptográfico aplicado al mensaje o a un pequeño bloque de datos que es una función del mensaje
- Intercambio de llaves: dos partes cooperan para intercambiar una clave de sesión. Hay distintas posibilidades que implican la clave privada de una o de las dos partes.

Algunos algoritmos son adecuados para las tres aplicaciones, mientras otros sólo se pueden usar para una o dos de ellas. La tabla 2 indica las aplicaciones que soportan los algoritmos, la tabla también inclute el Estándar de Firma Digital (DSS) y la criptografía de curva eliptica.

Algoritmo	Cifrado/descifrado	Firma digital	Intercambio de clave
RSA	Sí	Sí	Sí
Diffie-Hellman	No	No	Sí
DSS	No	Sí	No
Curva elíptica	Sí	Sí	Sí

Tabla 2

Aplicaciones para criptosistemas de clave pública

Requisitos para la criptografía de clave pública

El criptosistema que se muestra en la figura 7 depende de un algoritmo criptográfico basado en dos claves relacionadas. Diffie y Hellman postularon este sistema sin demostrar que tales algoritmos existen. Sin embargo, sí especificaron las condiciones que deben cumplir:

1.- Desde el punto de vista computacional, para una parte B es fácil generar una pareja de claves (clave pública KU_b, clave privada KR_b).

2.- En términos computacionales, para un emisor A que conozca la clave pública y el mensaje que ha de cifrarse, M, es fácil generar el texto cifrado correspondiente:

$$C = E_{KUb}(M)$$

3.- En términos computacionales, para un receptor B es fácil descifrar el texto cifrado resultante usando la clave privada para recuperar el mensaje original:

$$M = D_{KRb}(C) = D_{KRb}[E_{KUb}(M)]$$

4.- Desde el punto de vista computacional, es imposible que un openente, conociendo la clave pública, KU_b, determine la clave privada KR_b.

5.- Desde el punto de vista computacional, es imposible que un oponente, conociendo la clave pública, KU_b, y un texto cifrado, C, recupere el mensaje original, M.

Podemos añadir un sexto requisito que, aunque útil, no es necesario para todas las aplicaciones de clave pública:

6.- Cualquiera de las dos claves relacionadas puede usarse para el cifrado, y la otra para el descifrado.

$$M = D_{KRb}[E_{KUb}(M)] = D_{KUb}[E_{KRb}(M)]$$

Algoritmos de criptografía de clave pública

Los dos algoritmos de clave pública más usados son el RSA y el Diffie-Hellman, los cuales, se analizarán a continuación.

El algoritmo de cifrado de clave pública RSA

El RSA es un cifrado de bloque en el que el texto claro y el texto cifrado son enteros entre 0 y n – 1 para algún n.

Para algún bloque de texto claro M y un bloque de texto cifrado C, el cifrado y el descifrado son de la siguiente forma:

$$C = M^e \bmod n$$

$$M = C^e \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

Tanto el emisor como el receptor deben conocer los valores de n y e, y sólo el receptor conoce el valor de d. este es un algoritmo de cifrado de clave pública con una clave pública de KU = {e, n} y una clave privada de KR = {d, n}. Para que este algoritmo sea satisfactorio para el cifrado de clave pública, se deben cumplir los siguientes requisitos:

1.- Que sea posible encontrar valores de e, d, n tal que $M^{ed} = M \bmod n$ para todo $M < n$.

2.- Que sea relativamente fácil calcular M^e y C^d para todos los valores de $M < n$.

3.- Que sea imposible determinar d dados e y n.

Los dos primeros requisitos se cumplen fácilmente. El tercero se puede cumplir para valores grandes de e y n.

La figura 8 resume el algoritmo RSA. Se empieza seleccionando dos números primos, p y q, y calculando su producto n, que es el módulo para cifrado y descifrado. A continuación, se necesita la cantidad Φ(n), conocida como función totient de Euler de n, que es el número de enteros positivos menor que n y primo relativo de n. Luego, se selecciona un entero e que sea primo relativo de Φ(n) [el mayor común divisor de e y Φ(n) es 1]. Finalmente, se calcula d como el inverso multiplicativo de e, módulo Φ(n). Se puede demostrar que d y e tienen las propiedades deseadas.

Generación clave

Seleccionar p, q	p y q primos, $p \neq q$
Calcular $n = p \times q$	
Calcular $\phi(n) = (p - 1)(q - 1)$	
Seleccionar entero e	$gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calcular d	$de \bmod \phi(n) = 1$
Clave pública	$KU = \{e, n\}$
Clave privada	$KR = \{d, n\}$

Cifrado

Texto claro:	$M < n$
Texto cifrado:	$C = M^e \pmod n$

Descifrado

Texto cifrado:	C
Texto claro:	$M = C^d \pmod n$

Figura 8

El algoritmo RSA

Hay dos enfoques posibles para romper el algoritmo RSA. El primero es el enfoque de fuerza bruta: intentar todas las claves privadas posibles. Así, cuanto mayor sea el número de bits en e y d, más seguro será el algoritmo. Sin embargo, debido a que los cálculos que tienen lugar tanto en la generación de la clave como en el cifrado/descifrado son complejos, cuanto mayor sea el tamaño de la clave, más lento irá el sistema.

La mayoría de las discusiones sobre ciptoanálisis del RSA se han centrado en la tarea de factorizar n en sus dos factores primos. Un número n producto de dos números primos grandes es difícil factorizar,

aunque no tanto como solía ser. Una ilustración llamativa de ello ocurrió en 1977; los tres inventores de RSA lectores de Scientific American a descodificar un cifrado que publicaron en la columna de juegos matemáticos de Martin Gardner. Ofrecieron una recompensa de 100 dólares por la recuperación de una frase de texto claro, algo que, según predijeron, no ocurriría durante unos 40 cuatrillones de años. En abril de 1994, un grupo que trabajaba en Internet y que usaba unos 1.600 computadores consiguió el premio después de ocho meses de trabajo. En este reto se usó un tamaño de clave pública (longitud de n) de 129 dígitos decimales, alrededor de 428 bits. Este resultado no invalida al RSA; simplemente significa que debe usarse un tamaño de clave mayor. Actualmente, un tamaño de clave de 1024 bits (300 dígitos decimales aproximadamente) se considera lo suficientemente robusto para casi todas las aplicaciones.

Intercambio de clave Diffie – Hellman

El primer algoritmo de clave pública apareció en el artículo de Diffie y Hellman que definía la criptografía de clave pública y que se conoce por el intercambio de clave Diffie – Hellman. Una serie de productos comerciales empleados emplearon ésta técnica de intercambio de claves.

La finalidad del algoritmo es hacer posible que los usuarios intercambien de forma segura una clave secreta que luego pueda ser usada para el cifrado posterior de mensajes. El algoritmo está limitado al intercambio de claves.

El algoritmo de Diffie – Hellman depende para su efectividad de la dificultad de computar logaritmos discretos. En resumen, podemos definir el logaritmo discreto de la siguiente forma: primero definimos una raíz primitiva de un número primo p cuyas potencias generan todos los enteros desde 1 a p – 1. Es decir, si a es una raíz primitiva del número primo p, entonces los números

$$a \bmod p,\ a^2 \bmod p \ldots,\ a^{p-1} \bmod p$$

son distintos y consisten en los enteros desde 1 hasta p – 1 en alguna de sus permutaciones

Para cualquier entero b menor que p y una raíz primitiva a del número primo p, se puede encontrar un único exponente i tal que

$$b = a^i \bmod p \qquad \text{donde } 0 \le i \le (p - 1)$$

El exponente i se conoce como el logaritmo discreto o índice de b para la base a, mod p. Este valor se representa como $\text{ind}_{a,p}(b)$.

Con toda esta información se puede definir el intercambio de clave de Diffie – Hellman, que se resume en la figura 9. Para este esquema, hay dos números conocidos públicamente: un número primo q y un entero α que es la raíz primitiva de q.

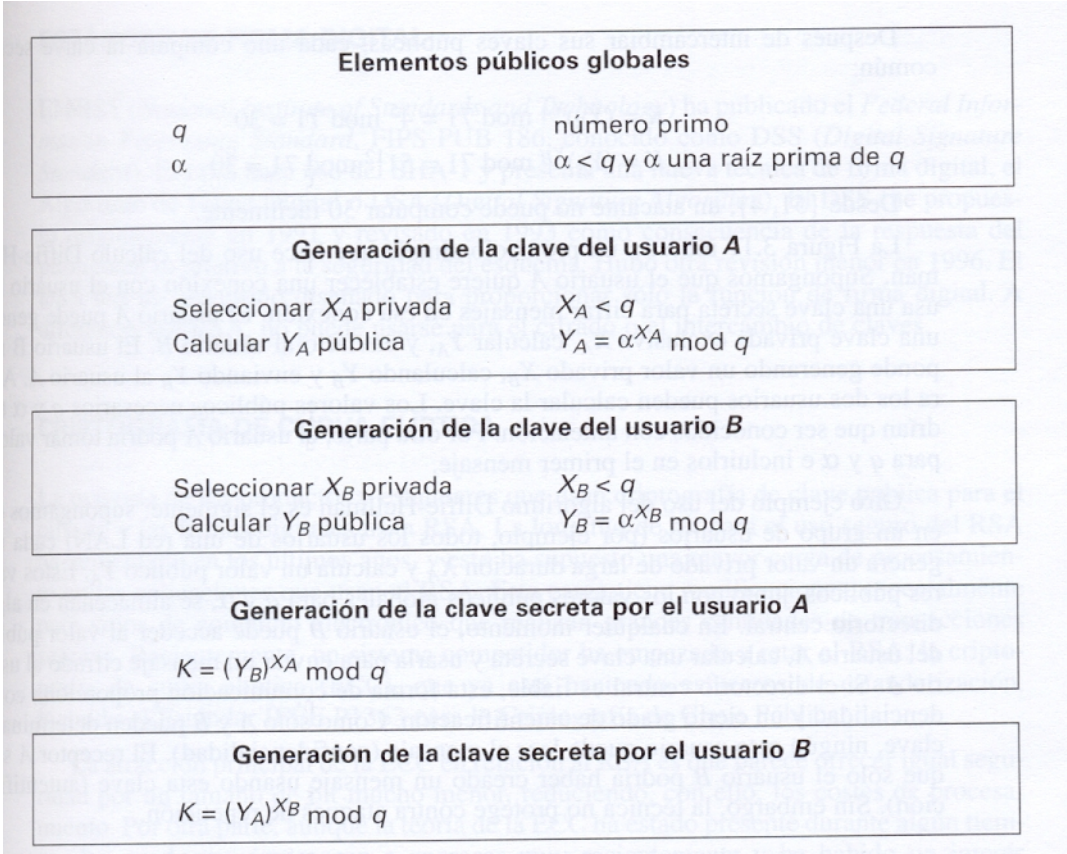

Figura 9

Algoritmo de intercambio de claves de Diffie – Hellman

La seguridad del intercambio de claves de Diffie – Hellman se basa en el hecho de que, aunque es relativamente fácil calcular exponenciales módulo un número primo, es muy difícil calcular logaritmos discretos. Para números primos grandes, la última tarea se considera imposible.

La figura 10 muestra un protocolo simple que hace uso del cálculo Diffie – Hellman. Supongamos que el usuario A quiere establecer una conexión con el usuario B y usa una clave secreta para cifrar mensajes en esa conexión. El usuario A puede generar una clave privada exclusiva X_A, calcular Y_A, y enviarlo al usuario B. El usuario B responde generando un valor privado X_B, calculando Y_B y enviando Y_B al usuario A. Ahora los dos usuarios pueden calcular la clave. Los valores públicos necesarios q y α tendrían que ser conocidos con antelación. Por otra parte, el usuario A podría tomar valores para q y α e incluirlos en el primer mensaje.

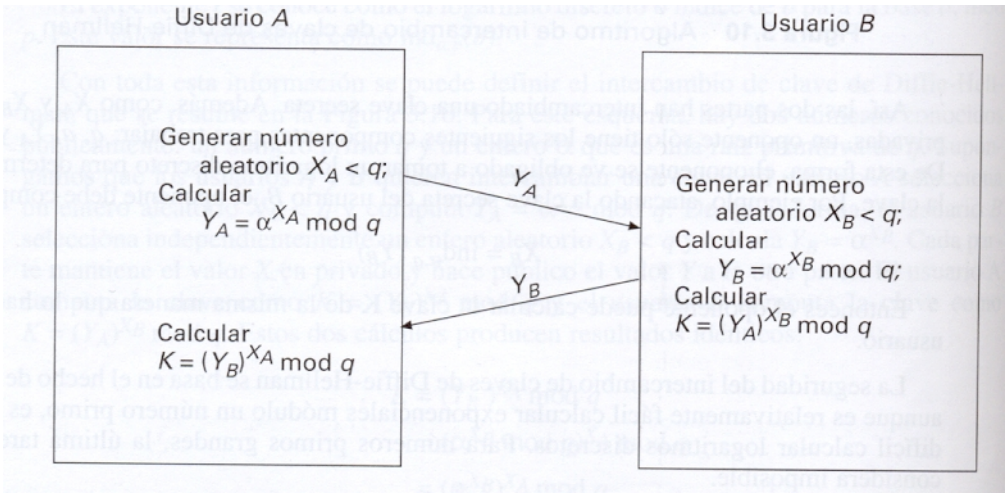

Figura 10

Intercambio de claves Diffie – Hellman

Otro ejemplo del uso del algoritmo Diffie – Hellman es el siguiente: supongamos que en un grupo de usuarios (por ejemplo, todos los usuarios de una red LAN) cada uno genera un valor privado de larga duración X_A y calcula un valor público Y_A. Éstos valores públicos, junto con los valores públicos globales para q y α , se almacenan en algún directorio central. En cualquier momento, el usuario B puede acceder al valor público del usuario A, calcular una clave secreta y usarla para enviar un mensaje cifrado al usuario A. Si el directorio central es confiable, esta forma de comunicación proporciona confidencialidad y un cierto grado de autenticación. Como solo A y B pueden determinar la clave, ningún otro usuario puede leer el mensaje (confidencialidad). El receptor A sabe que sólo el usuario B podría haber creado un mensaje usando esta clave (autenticación). Sin embargo, la técnica no protege contra ataques de repetición.

Firmas Digitales

El cifrado de clave pública se puede usar de otra forma, como ilustra la figura 7b. Supongamos que Benito quiere enviar un mensaje a Alicia y, aunque no es necesario que el mensaje se mantenga en secreto, quiere que Alice se asegure de que el mensaje, efectivamente, proviene de él. En este caso Benito usa su propia clave privada para cifrar el mensaje. Cuando Alice recibe el texto cifrado, se encuentra con que puede descifrarlo con la clave pública de Benito, demostrando así, que el mensaje ha debido ser cifrado por él. Nadie más tiene la clave privada de Benito y, por lo tanto, nadie más ha podido crear un texto cifrado que pueda ser descifrado con su clave pública. Por consiguiente, sin acceso a la clave privada de Benito, así que el mensaje queda autenticado tanto en lo que respecta a la fuente como a la integridad de los datos.

En el esquema anterior, se ha cifrado todo el mensaje, que, aunque valide el autor y los contenidos, necesita una gran cantidad de almacenamiento. También se debería almacenar una copia en texto cifrado para que el origen y el contenido se puedan verificar en caso de desacuerdo. Una forma más efectiva de lograr los mismos resultados es cifrar un pequeño bloque de bits que sea una función de un documento. Este bloque, llamado autentificador, debe tener la propiedad por la cual es imposible cambiar el documento sin cambiar el autentificador. Si el autentificador se cifra con la clave privada del emisor, sirve como firma que verifica el origen, el contenido y la secuencia.

Es importante resaltar que el proceso de cifrado que se acaba de describir no proporciona confidencialidad. Es decir, el mensaje que se está enviando es seguro contra alteraciones pero no contra escuchas, lo que es obvio en el caso de una firma basada en una parte del mensaje, porque el resto del mensaje se transmite en claro. Incluso en el caso del cifrado completo, no hay protección de confidencialidad ya que cualquier observador puede descifrar el mensaje usando la clave pública del emisor.

Gestión de claves

Una de las funciones principales del cifrado de clave pública es la de tratar el problema de la distribución de claves. Hay dos aspectos fundamentales sobre el uso del cifrado de clave pública en este sentido:

- La distribución de claves públicas
- El uso de cifrado de clave pública para distribuir claves secretas.

Certificados de clave pública

A la vista de todo esto, la base del cifrado de clave pública se encuentra en el hecho de que la clave pública es pública. Así, si hay un algoritmo de clave pública aceptado, como el RSA, cualquier participante puede enviar su clave pública a cualquier otro o difundir la clave a la comunidad en general. Aunque este enfoque es conveniente, presenta una debilidad fundamental: cualquiera puede falsificar ese dato público. Es decir un usuario podría hacerse pasar por el usuario A y enviar una clave pública a otro participante o difundirla. Hasta el momento que A descubre la falsificación y alerta a los otros participantes, el falsificador puede leer todos los mensajes cifrados enviados a A y puede usar las claves falsificadas para la autenticación.

La solución a este problema es el certificado de clave pública. Básicamente, un certificado consiste en una clave pública y un identificador o nombre de usuario del dueño de la clave, con todo el bloque firmado por una tercera parte confiable. Comúnmente, la tercera parte es una autoridad de certificación (CA, Certificate Authority) en la que confía la comunidad de usuarios, que podría ser una agencia gubernamental o una institución financiera. Un usuario puede presentar su clave pública a la autoridad de forma segura, obtener un certificado y luego publicarlo. Cualquiera que necesite la clave pública de este usuario puede obtener el certificado y verificar que es válida por medio de la firma fiable adjunta. La figura 11 ilustra este proceso

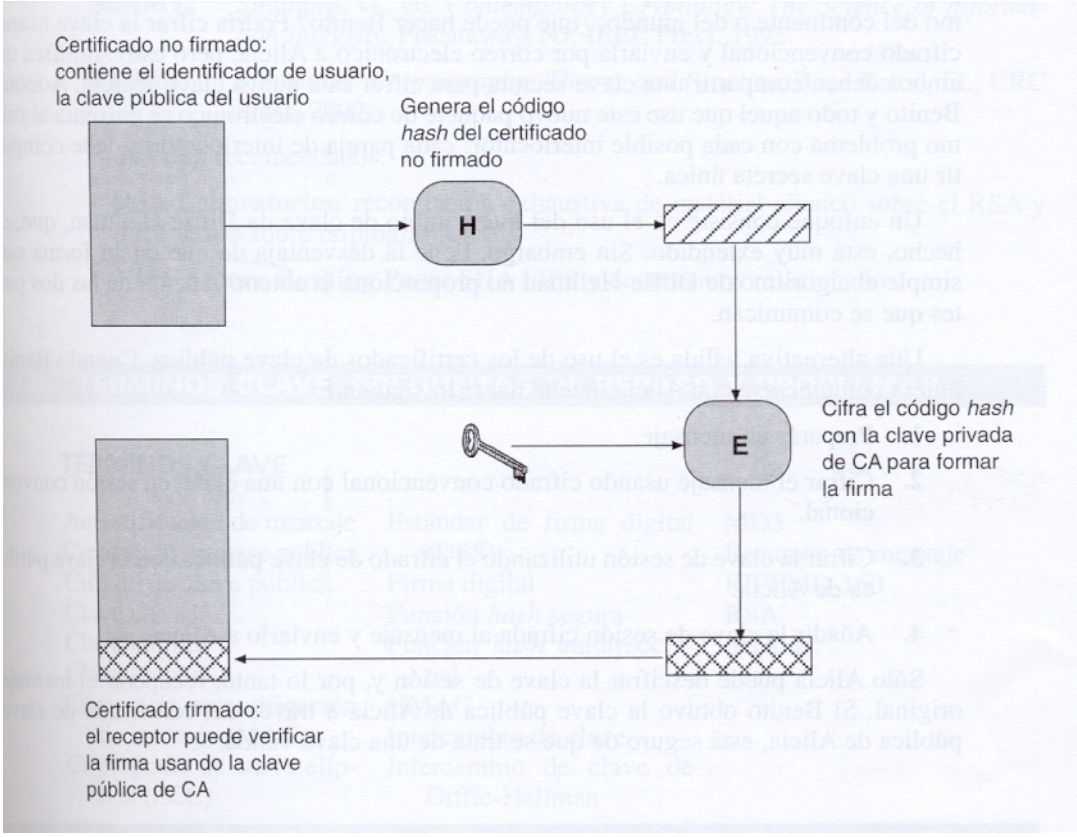

Figura 11

Uso del certificado de clave pública

El esquema que se ha aceptado mundialmente para el formateado de los certificados de clave pública es el estándar X.509, cuyos certificados se emplean en la mayoría de las aplicaciones de seguridad de redes.

Distribución de claves secretas mediante criptografía de clave pública

Con el cifrado convencional, un requisito fundamental para que las dos partes se comuniquen de forma segura es que compartan la clave secreta. Supongamos que Benito quiere crear una aplicación de mensajes que le permita intercambiar correo electrónico de forma segura con alguien que tiene acceso a Internet o a otra red que ambos comparten. Supongamos además, que quiere hacerlo usando cifrado convencional. Con el cifrado convencional, Benito y su interlocutor, Alicia, deben acordar una forma de compartir una clave secreta que nadie más conozca. ¿cómo van a hacerlo? Si Alicia se encuentra en la habitación continua a Benito, éste podría generar una clave y anotarla en un papel o guardarla en un disquete y entregarla a Alicia. Pero si Alicia esta en el otro extremo del continente o del mundo, ¿qué puede hacer Benito? Podría cifrar la clave usando cifrado convencional y enviarla por correo electrónico a Alicia, pero esto significa que ambos deben compartir una clave secreta para cifrar esta nueva clave secreta. Además, Benito y todo aquel que use este nuevo paquete de correo electrónico se enfrenta al mismo problema con cada posible interlocutor: cada pareja de interlocutores debe compartir una clave secreta única.

Un enfoque consiste en el uso del intercambio de clave de Diffie – Hellman, que, de hecho, está muy extendido. Sin embargo, tiene la desventaja de que su forma más simple el algoritmo de Diffie – Hellman no proporciona la autenticación de las dos partes que se comunican.

Una alternativa válida es el uso de los certificados de clave pública. Cuando Benito quiera comunicarse con Alicia, puede hacer lo siguiente:

1.- Preparar un mensaje

2.- Cifrar el mensaje usando cifrado convencional con una clave de sesión convencional.

3.- Cifrar la clave de sesión utilizando el cifrado de clave pública con la clave pública de Alicia.

4.- Añadir la clave de sesión cifrada al mensaje y enviarlo a Alicia.

Sólo Alicia puede descifrar la clave de sesión y, por lo tanto, recuperar el mensaje original. Si Benito obtuvo la clave pública de Alicia a través del certificado de clave pública de Alicia, está seguro de que se trata de una clave válida.

Referencias:

- Handbook of Applied Cryptography
    - Alfred J. Menezes
    - Paul C. Van Oorschot
    - Scout A. Vanstone
- Seguridad en redes
    - William Stallings (Prentice Hall)