Introducción a la Criptografía: tipos de algoritmos

Este artículo ofrece una breve descripción de los algoritmos criptográficos más utilizados en la actualidad para proteger información en formato electrónico. Sin entrar en detalles matemáticos, se presentan las diferentes familias de algoritmos y las aplicaciones actuales de los mismos. Se describen fundamentalmente las aplicaciones de cifrado y firma electrónica. También se introduce el concepto de certificado digital y se especifica su importancia para garantizar la seguridad y las aplicaciones que mayor beneficio obtienen. Los algoritmos van acompañados de ejemplos prácticos basados en OpenSSL, programa de libre distribución que está disponible para diversas arquitecturas.

Vera Delgado

Estudiante de quinto curso de la Titulación de Ingeniería Informática de la Escuela Técnica Superior de Ingeniería (ICAI), Universidad Pontificia Comillas. Previsiblemente obtendrá el título oficial en junio de 2006.

Rafael Palacios

Ingeniero Industrial del ICAI (1990) y Doctor Ingeniero (1998). Es investigador del Instituto de Investigación Tecnológica y profesor del Departamento de Sistemas Informáticos de la Escuela Técnica Superior de Ingeniería (ICAI). Imparte clases de seguridad informática en la titulación de Ingeniero en Informática del ICAI.

Comentarios a:
comentarios@icai.es

¿Qué es la criptografía?

Es la ciencia que estudia los métodos y procedimientos para modificar los datos, con objeto de alcanzar las características de seguridad.

Las principales características que un sistema de seguridad quiere obtener son:

- **Confidencialidad**. Consiste en garantizar que sólo las personas autorizadas tienen acceso a la información.
- **Integridad**. Consiste en garantizar que el documento original no ha sido modificado. El documento puede ser tanto público como confidencial.
- **Autenticación**. Permite garantizar la identidad del autor de la información.

Existen diversos algoritmos matemáticos que intentan cubrir una o varias de estas características básicas de seguridad. El nivel de cumplimiento de sus objetivos es difícil de evaluar, ya que diversos algoritmos pueden ser vulnerables ante técnicas de ataque diferentes, además la mayoría de los algoritmos pueden trabajar con claves de distinta longitud lo cual afecta directamente a la robustez. Por otro lado, existen otras características, a parte de la robustez del algoritmo, que también influyen en el proceso de selección del algoritmo más apropiado para una determinada aplicación. Algunas de estas características son: el tiempo de cálculo del proceso de cifrado, el tiempo de cálculo del proceso de descifrado, la relación de tamaño entre el documento original y el documento cifrado, etc. Este artículo no pretende entrar en detalles de la implementación de los algoritmos ni analiza características técnicas de rendimiento de los mismos, ni mostrar técnicas concretas que puedan romperlos El objetivo fundamental del artículo es presentar las familias de algoritmos que existen y sus características principales.

Existen infinidad de algoritmos criptográficos que, partiendo de un documento original, obtienen otro documento o conjunto de información. Los algoritmos más conocidos son los que obtienen un documento a partir de un documento original al aplicar un algoritmo

que utiliza una clave secreta como argumento. En general los algoritmos criptográficos se pueden clasificar en tres grandes familias.

- **Criptografía de clave secreta** o criptografía simétrica.
- **Criptografía de clave pública** o criptografía asimétrica.
- **Algoritmos HASH** o de resumen.

A continuación se describe cada una de estas familias de algoritmos, así como los algoritmos más utilizados dentro de cada familia.

Criptografía de clave secreta

Se incluyen en esta familia el conjunto de algoritmos diseñados para cifrar un mensaje utilizando una única clave conocida por los dos interlocutores, de manera que el documento cifrado sólo pueda descifrarse conociendo dicha clave secreta. Algunas de las características más destacadas de este tipo de algoritmos son las siguientes:

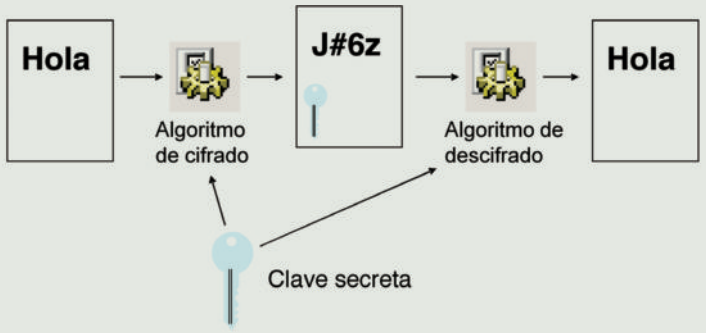
- A partir del mensaje cifrado no se puede obtener el mensaje original ni la clave que se ha utilizando, aunque se conozcan todos los detalles del algoritmo criptográfico utilizado[1].
- Se utiliza la misma clave para cifrar el mensaje original que para descifrar el mensaje codificado.
- Emisor y receptor deben haber acordado una clave común por medio de un canal de comunicación confidencial antes de poder intercambiar información confidencial por un canal de comunicación inseguro.

El esquema general de cifrado y descifrado mediante algoritmos de clave privada se muestra en la Figura 1. A partir de un documento original se obtiene un documento cifrado al aplicar una clave secreta; esa misma clave secreta se utiliza posteriormente para volver a obtener el documento original.

Los algoritmos simétricos más conocidos son: DES, 3DES, RC2, RC4, RC5, IDEA, Blowfish y AES.

- El algoritmo **DES** [1], basado en Lucifer de IBM (1975), fue seleccionado como algoritmo estándar de cifrado en 1977 por NIST (National Institute of Standards and Technology, USA). Utiliza claves de cifrado bastante cortas (56 bits, de los cuales sólo se utilizan 48 bits) y hoy en día se considera poco robusto, sobre todo desde que en 1998 la Electronica Frontier Foundation hizo público un crackeador de código DES capaz de descifrar mensajes DES en menos de 3 días.

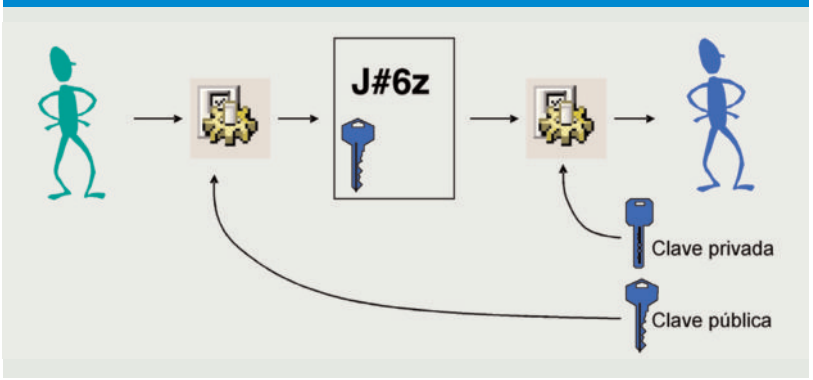
Figura 1. Esquema de los algoritmos de clave secreta



- El algoritmo **3DES** [2], desarrollado por Tuchman en 1978, es una manera de mejorar la robustez del algoritmo DES que consiste en aplicarlo tres veces consecutivas. Se puede aplicar con la misma clave cada vez, o con claves distintas y combinando el algoritmo de cifrado con el de descifrado, lo cual da lugar a DES-EEE3, DES-EDE3, DES-EEE2 y DES-EDE2. El resultado es un algoritmo seguro y que se utiliza en la actualidad, aunque resulta muy lento comparado con otros algoritmos más modernos que también son seguros.
- En 1989, Ron Rivest desarrolló el algoritmo **RC2** (Rivest's Cipher) para RSA Data Security, Inc. [3]. Se trata de un algoritmo de cifrado por bloques, que utiliza una clave de tamaño variable. Es de dos a tres veces más rápido que el algoritmo DES, siendo más seguro (http://theory.lcs.mit.edu/~rivest/).
- Ron Rivest desarrolló el algoritmo **RC4** en 1987 para RSA Data Security, que se hizo público en 1994 [4]. Se considera inmune al criptoanálisis diferencial y lineal. Es el algoritmo utilizado en el cifrado WEP de la mayoría de los puntos de acceso WiFi. Es necesario comentar que el protocolo WEP se considera vulnerable, pero no por problemas con el algoritmo de cifrado RC4 sino por otros aspectos del propio protocolo que permiten determinar la clave de cifrado en un tiempo corto (pocas horas en una red con mucho tráfico).
- Otro algoritmo de Ron Rivest es **RC5**, publicado en 1994 [5]. Se trata de un algoritmo de cifrado por bloques, que utiliza claves de tamaño variable. Se caracteriza por la sencillez

[1] Hay que tener en cuenta que la mayoría de los algoritmos de cifrado son públicos, ya que la seguridad reside en el diseño del propio algoritmo. En algunas aplicaciones, por ejemplo la telefonía GSM, se han utilizado algoritmos secretos que, al cabo de un tiempo, han sido descubiertos y la seguridad ha visto comprometida.

Figura 2. Esquema de los algoritmos de clave pública



del algoritmo, que lo hacen muy rápido y fácil de implementar tanto en software como en hardware.

- International Data Encription Algorithm, **IDEA**, diseñado por Xuejia Lai y James L. Massey de ETH-Zürich. Es un algoritmo de cifrado por bloques de 64 bits que emplea claves de 128 bits, que se presentó por primera vez en 1991 [6]. Es dos veces más rápido que DES, a pesar de utilizar claves mucho más largas. Es un algoritmo patentado en algunos países (entre ellos España), salvo para usos no comerciales.
- **Blowfish** es un algoritmo de cifrado por bloques de 64 bits diseñado por Bruce Schneier en 1993 [7]. Utiliza claves de longitud variable entre 32 y 448 bits. A pesar de utilizar un tamaño de bloque pequeño, que podría facilitar su vulnerabilidad al procesar textos largos, se considera un algoritmo seguro y es más rápido que DES.
- **Twofish** es una variante de Blowfish que utiliza bloques de 128 bits y claves de 256 bits [8]. También diseñado por Bruce Schneier, en colaboración con John Kelsey, Doug Whiting, David Wagner, Chris Hall, y Niels Ferguson, fue uno de los cinco finalistas en el proceso de selección de NIST para sustituir a DES como algoritmo estándar.
- **Advanced Encryption Standard, AES**, es el estándar para cifrado simétrico del NIST desde el 26 de mayo de 2002 en sustitución de DES. AES también es conocido por Rijndael, nombre original del algoritmo propuesto en 1999 [9] que cambió al ser seleccionado por NIST y convertirse en el estándar. Fue desarrollado por dos criptógrafos Belgas, Joan Daemen y Vincent Rijmen, Es un algoritmo de cifrado por bloques con longitud de bloque y longitud de clave variables. Los valores adoptados para el estándar son bloques de 128 bits, y claves de longitud 128, 192 ó 256 bits [10].

La recomendación general es utilizar el algoritmo AES, ya que fue cuidadosamente analizado durante el proceso de selección desarrollado por NIST y se considera muy bueno. Además, al convertirse en el estándar se ha extendido muchísimo y seguramente se ha convertido en el algoritmo que más criptoanálisis ha experimentado lo que demuestra su robustez.

Criptografía de clave pública

Esta categoría incluye un conjunto de algoritmos criptográficos que utilizan dos claves distintas para cifrar y para descifrar el mensaje. Ambas claves tienen una relación matemática entre sí, pero la seguridad de esta técnica se basa en que el conocimiento de una de las claves no permite descubrir cuál es la otra clave. En realidad sería necesario conocer todos los números primos grandes para ser capaz de deducir una clave a partir de otra, pero está demostrado que en la práctica se tardarían demasiados años sólo en el proceso de obtención de los número primos grandes.

Cada usuario cuenta con una pareja de claves, una la mantiene en secreto y se denomina **clave privada** y otra la distribuye libremente y se denomina clave pública. Para enviar un mensaje confidencial sólo hace falta conocer la **clave pública** del destinatario y cifrar en mensaje utilizando dicha clave. En este caso los algoritmos asimétricos garantizan que el mensaje original sólo puede volver a recuperarse utilizando la clave privada del destinatario (ver el esquema de la Figura 2). Dado que la clave privada se mantiene en secreto, sólo el destinatario podrá descifrar el mensaje.

Estos algoritmos pueden trabajar indistintamente con cualquiera de las claves, de manera que un mensaje cifrado con la clave pública sólo puede descifrarse con la clave privada, pero cualquier mensaje cifrado con la clave privada sólo puede ser descifrado con la clave pública. Esta característica permite utilizar este método para otras aplicaciones además de las que sólo requieren confidencialidad, como es el caso de la firma electrónica (como se verá en el artículo de criptografía del próximo número de *Anales*).

Algunas de las características más destacadas de este tipo de algoritmos son las siguientes:

- Se utilizan una pareja de claves denominadas clave pública y clave privada, pero a

partir de la clave pública no es posible descubrir la clave privada.

- A partir del mensaje cifrado no se puede obtener el mensaje original, aunque se conozcan todos los detalles del algoritmo criptográfico utilizado y aunque se conozca la clave pública utilizada para cifrarlo.
- Emisor y receptor no requieren establecer ningún acuerdo sobre la clave a utilizar. El emisor se limita a obtener una copia de la clave pública del receptor, lo cual se puede realizar, en principio, por cualquier medio de comunicación aunque sea inseguro.

A diferencia de los algoritmos de clave secreta, que existen desde los tiempos de los romanos, los métodos asimétricos son muy recientes. En 1976, Whitfield Diffie y Martin Hellman crearon un método con la ayuda de Ralph Merkle para iniciar una comunicación segura sin haber acordado previamente una clave secreta. El método se conoce como Diffie-Hellman Key Exchange [13]. Poco más tarde se publicó el primer algoritmo asimétrico completo, denominado RSA, que sigue siendo el más utilizado en la actualidad.

RSA fue desarrollado en 1977 por Ron Rivest, Adi Shamir y Len Adleman. El nombre RSA proviene de las iniciales de los apellidos de sus inventores [11]. El algoritmo fue patentado en 1983 por MIT, pero la patente expiró el 21 de septiembre de 2000 y desde entonces se utiliza libremente. La seguridad de este algoritmo reside en la dificultad que supone la factorización de un número compuesto por factores primos muy grandes. Si un criptoanalista fuera capaz de encontrar los factores primos sería capaz también de determinar la clave privada y, por lo tanto, descifrar el mensaje. Sin embargo el problema de factorización se considera imposible de resolver en la práctica, y cuanto más grande sean los números utilizados, es decir las longitudes de las claves, mayor dificultad se alcanza.

Los algoritmos asimétricos se pueden utilizar para **cifrado de documentos secretos** o para **firma electrónica** tanto de documentos privados como públicos. Para garantizar la confidencialidad, el documento se cifra con la clave pública del destinatario y por lo tanto sólo el destinatario puede volver a recuperar el documento original. Sin embargo, para firmar electrónicamente un documento, el autor lo cifra con su propia clave privada y por lo tanto cualquier usuario puede ver el contenido original descifrando el documento mediante la clave pública correspondiente. En este último caso el documento es público porque cualquier usuario puede verlo, pero sólo el autor podría hacer modificaciones y volver a cifrarlo porque sólo él tiene su clave privada. Por lo tanto, no se tiene confidencialidad pero sí integridad, y es una herramienta fundamental para generar documentos "oficiales" que sean públicos, pero que vayan debidamente firmados por la autoridad pertinente.

Algoritmos HASH o de resumen

Los algoritmos HASH, parten de una información de entrada de longitud indeterminada y obtienen como salida un código, que en cierto modo se puede considerar único para cada entrada. La función de estos algoritmos es determinista, es decir que partiendo de una misma entrada siempre se obtiene la misma salida. Sin embargo, el interés de estos algoritmos reside en que partiendo de entradas distintas se obtienen salidas distintas.

Unos ejemplos muy sencillos, aunque muy vulnerables, son los dígitos de control y los CRC (Cyclic Redundancy Code) que se utilizan para detectar errores de trascripción o de comunicación. Estos algoritmos en particular garantizan que el código generado cambia ante una mínima modificación de la entrada y tienen aplicaciones muy concretas de control de integridad en procesos con perturbaciones fortuitas y poco probables. Sin embargo son poco robustos y está demostrado que se pueden realizar pequeños conjuntos de modificaciones en un documento de manera que el CRC resulte inalterado. En un buen algoritmo HASH es inadmisible que un conjunto reducido de modificaciones no altere el código resultante, ya que se podrían realizar retoques en el documento sin que fuesen detectados, y por lo tanto no se garantiza la integridad.

Dado que el tamaño del código que se genera como salida es de tamaño limitado, (típicamente 128, 256 ó 512 bits) mientras que el tamaño del documento de entrada es ilimitado (típicamente un archivo), es evidente que se cumplen dos propiedades:

- El algoritmo es irreversible, es decir, no es posible obtener el documento original a partir del código generado.
- Existen varios documentos que dan lugar a un mismo código.

La segunda propiedad es debida a que el número de combinaciones de los códigos de tamaño limitado es menor al número de

combinaciones de cualquier archivo grande. Sin embargo los buenos algoritmos consiguen que los documentos que dan lugar al mismo código, sean completamente diferentes y por lo tanto sólo uno de ellos será legible. Los algoritmos más utilizados son MD5 y SHA1, pero nunca se utilizan códigos CRC en aplicaciones de seguridad. Por ejemplo los certificados incluyen un campo llamado fingerprint con el resultado de los algoritmos MD5 y SHA1.

- **MD5 (Message-Digest Algorithm 5)** es un algoritmo que produce un código de 128 bits. Fue creado por Ron Rivest en 1991 y se convirtió en el estándar de Internet RFC 1321. Recientemente se han encontrado pequeñas vulnerabilidades en este algoritmo que sugieren un movimiento hacia SHA1. La descripción del algoritmo y de las vulnerabilidades recientes puede encontrarse en www.wikipedia.org.
- NIST presentó en 1993 un algoritmo basado en las mismas técnicas que MD5 y denominado **SHA (Secure Hash Algorithm)**. Este algoritmo fue declarado estándar Federal Information Processing Standard PUB 180 en 1993, pero en 1995 la Agencia de Seguridad Nacional (NSA) lo sustituyó por una versión mejorada que actualmente se conoce como SHA-1 y que se considera más seguro que MD5. Produce un código hash de 160 bits para mensajes de longitud máxima 264 bits, aunque existen otras variantes poco utilizadas todavía que producen códigos de mayor longitud.

En general, SHA1 se considera el mejor algoritmo de esta familia y es el que se aplica en la mayoría de las aplicaciones de firma electrónica. Por lo tanto es muy habitual aplicar SHA1 seguido de RSA para realizar una firma electrónica de un documento, o bien el algoritmo DSA específico para firma electrónica que también utiliza SHA1 internamente.

- **Digital Signature Algorithm (DSA)** es el estándar de United States Federal Government para firma digital. Su desarrollo se atribuye a David W. Kravitz, de la National Security Agency. Fue presentado por NIST en agosto de 1991, adoptado como estándar en 1993, y con última revisión de 2000. [12]. Es un algoritmo exclusivo de firma electrónica basado en clave pública, pero no vale para comunicaciones confidenciales.

El procedimiento completo de firma electrónica y la utilización prácticas de las funciones HASH se verá en el artículo de criptografía del próximo número de *Anales*.

Conclusiones

En este artículo se han descrito las familias de algoritmos criptográficos y se han presentados los algoritmos más utilizados dentro de cada tipo, indicando algunas de sus características principales. Además se han indicado cuáles son aplicaciones de cada tipo de algoritmo atendiendo a las características básicas de seguridad que se alcanzan. Un artículo posterior describirá aplicaciones prácticas en las que se utilizan los algoritmos. ▮

Referencias

[1] National Bureau of Standards, Data Encryption Standard, FIPS-Pub.46. National Bureau of Standards, U.S. Department of Commerce, Washington D.C., Jan 1977.

[2] National Bureau of Standards, Data Encryption Standard, FIPS-Pub.46-3. National Bureau of Standards, U.S. Department of Commerce, Washington D.C., Oct 1999.

[3] Lars R. Knudsen, Vincent Rijmen, Ronald L. Rivest, and M.J.B. Robshaw: "On the Design and Security of RC2", Proceedings Fifth Fast Software Encryption Workshop FSE'98, pages 206—221, (1998).

[4] R.L. Rivest: "The RC4 Encryption Algorithm". RSA Data. Security, Inc., Mar 12th, 1992.

[5] Ronald L. Rivest: "The RC5 Encryption Algorithm", Proceedings of the 1994 Leuven Workshop on Fast Software Encryption, pages 86-96. (Springer 1995).

[6] Xuejia Lai, James L. Massey: "A Proposal for a New Block Encryption Standard", EUROCRYPT 1990. Workshop on the Theory and Application of Cryptographic Techniques, Aarhus, Denmark, May 1990, pp389–404.

[7] B. Schneier: "Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish)", Fast Software Encryption, Cambridge Security Workshop Proceedings (December 1993), Springer-Verlag, 1994, pp. 191-204.

[8] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson: "Twofish: A 128-Bit Block Cipher", Jun 1998. http://www.schneier.com/paper-twofish-paper.html.

[9] Joan Daemen, Vincent Rijmen: "The Rijndael Block Cipher", sep 1999. http://csrc.nist.gov/CryptoToolkit/aes/rijndael/Rijndael.pdf.

[10] National Bureau of Standards, Data Encryption Standard, FIPS-Pub.197. National Bureau of Standards, U.S. Department of Commerce, Washington D.C., Nov. 2001.

[11] R. Rivest, A. Shamir, L. Adleman: "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". Communications of the ACM, Vol. 21 (2), pp. 120–126. 1978. Previously released as an MIT Technical Memo in April 1977.

[12] National Bureau of Standards, Data Encryption Standard, FIPS-Pub.186-2. National Bureau of Standards, U.S. Department of Commerce, Washington D.C., 2000.

[13] W. Diffie and M. E. Hellman: "New Directions in Cryptography", IEEE Transactions on Information Theory, vol. IT-22, Nov. 1976, pp: 644-654.