Seguridad matemática en la Sociedad de la Información

Pino T. Caballero Gil

Profesora Titular de Ciencias de la Computación e Inteligencia Artificial
Departamento de Estadística, Investigación Operativa y Computación
Universidad de La Laguna

Introducción

La ciencia de la Criptografía estudia los sistemas de cifrado y descifrado, la gestión de claves y toda una variedad de aplicaciones útiles en el medio hostil y vulnerable de las comunicaciones digitales electrónicas. Uno de los hitos más importantes de la larga y fascinante historia de la Criptografía ocurrió en 1976 cuando se introdujo el revolucionario concepto de la Clave Pública, que marcó el comienzo de la moderna época de la Criptografía de Clave Pública, frente a la época de la Criptografía de Clave Secreta. Mientras en este clásico tipo de sistemas, las operaciones de cifrado y descifrado son imposibles sin la clave secreta compartida por emisor y receptor, en la criptografía de clave pública existen dos piezas de información (claves privada y pública) tales que una de ellas (clave privada de descifrado) es imposible de obtener a partir de la otra (clave pública de cifrado). De esta sencilla manera se elimina la compleja tarea de la distribución de claves secretas.

La revolucionaria idea de la clave pública se puede llevar a cabo de forma sencilla gracias al análogo matemático de las calles de un sólo sentido, que son las funciones unidireccionales las cuales resultan fáciles de aplicar pero imposibles o difíciles de invertir. Así, la seguridad de los cifrados de clave pública reside en la dificultad o intratabilidad computacional de varios problemas matemáticos, por lo que se puede afirmar que toda la criptografía moderna se sustenta en el pilar de la Teoría de la Complejidad Computacional. La mayoría de cifrados de clave pública en la actualidad, y concretamente el que es el más difundido, el RSA, se apoya en la estructura matemática más socorrida en Criptografía que es el grupo multiplicativo de los enteros módulo un gran número primo, y se basa en la hipótesis de que factorizar números grandes es computacionalmente intratable. A este respecto merece la pena mencionar la computación cuántica cuyo estudio está en auge actualmente, ya que se ha probado que con ordenadores cuánticos es posible factorizar números grandes mucho más rápido que con ordenadores digitales convencionales. También merecen ser resaltados los sistemas basados en la dificultad del problema del logaritmo discreto en el grupo abeliano de una curva elíptica en un campo finito, ya que representan uno de los cifrados de clave pública más prometedores debido a su eficiencia y seguridad.

Firma Digital y Control de Accesos

Para el uso seguro de criptografía de clave pública es importante garantizar la autenticidad de las claves públicas. Una elegante solución a este problema viene de la mano de la firma digital, que puede verse como la analogía electrónica de la firma manual. Este tipo de esquemas puede basarse de forma sencilla en la criptografía de clave pública sin más que considerar que una usuaria A (*Alice*) siempre puede firmar un documento mediante su propia clave secreta, de forma que otro usuario B (*Bob*) siempre puede validar dicha firma usando la

clave pública de A. De esta forma, la aplicación de la firma digital en la certificación de claves públicas se basa en unas Terceras Partes de Confianza conocidas como Autoridades de Certificación cuya misión es firmar digitalmente certificados de autenticidad de las claves públicas de los usuarios.

En la práctica, cuando se pretende firmar un mensaje digital hay que considerar que normalmente éste tiene una longitud variable mientras que la entrada al algoritmo de firma suele ser de longitud fija, y por lo general más corta que el mensaje. Para resolver este problema se recurre al uso de funciones *hash*, que se han utilizado tradicionalmente en Ciencias de la Computación sobre todo para procedimientos de búsqueda. Esta solución consiste en aplicar una función hash al mensaje antes de aplicar el algoritmo de cifrado. Dicha función hash debe ser una función unidireccional, simple y eficiente que convierta cualquier mensaje de longitud arbitraria en uno de longitud fija, sea computacionalmente imposible encontrar dos mensajes con igual resumen, y tal que pequeños cambios en el mensaje producen cambios significativos en el resumen. El ataque por búsqueda exhaustiva contra las funciones hash es inabordable para las que producen resúmenes de tamaño 128, ya que por término medio habría que probar 2^{64} mensajes antes de conseguir uno que genere un resumen predeterminado. Sin embargo estas funciones sí son sensibles al ataque del cumpleaños (llamado así por la conocida paradoja probabilística del cumpleaños) ya que sólo se necesita generar $2^{n/2}$ mensajes para obtener dos que generen el mismo resumen.

En cuanto a la identificación para el control de accesos a sistemas, los métodos existentes tienen grados de seguridad muy diferentes. El método más simple es el habitual sistema de contraseñas fijas, que constituye el método de identificación más débil. Dado que el sistema realmente no necesita almacenar las contraseñas, sino sólo diferenciar las válidas de las inválidas, puede mejorarse un poco el esquema sin más que aplicar directamente una función unidireccional sobre las contraseñas introducidas de manera que el sistema sólo necesite almacenar las imágenes para compararlas. Por otra parte están los métodos de identificación fuerte basados en un protocolo criptográfico conocido como Demostración de Conocimiento Nulo. Este tipo de protocolos impiden posibles ataques de suplantación ya que permiten a los usuarios demostrar su conocimiento de la contraseña secreta sin revelar ninguna información sobre ella. Se trata de protocolos interactivos en los que se realiza un intercambio de mensajes (retos aleatorios y respuestas), tras el cual el sistema acepta o rechaza la demostración de conocimiento con una cierta probabilidad de acierto. Las dos demostraciones de conocimiento nulo más estudiadas se basan en dos de los problemas más utilizados en criptografía, la residuosidad cuadrática y los logaritmos discretos.

Protocolos Criptográficos

En general los protocolos criptográficos se pueden definir como algoritmos utilizados por dos o más participantes con una meta común, e implementados en entornos distribuidos inseguros. Formalmente un protocolo k-partito es una función pública $f: (\{0,1\}^*)^k \rightarrow (\{0,1\}^*)^k$ computable en tiempo polinomial, tal que la entrada es información privada de cada uno de los k participantes, y la salida es un vector de los valores que requieren los participantes para resolver el problema. A continuación describimos algunos de los algoritmos más representativos de cada uno de los protocolos estudiados, observando la frecuencia con que se utilizan como base problemas matemáticos como el de la factorización, la residuosidad cuadrática o el logaritmo discreto, todos ellos provenientes de la Teoría de Números.

En los protocolos básicos conocidos como de Transferencia Inconsciente la situación a resolver es la siguiente: A quiere transferir un secreto a B de forma que dicha información se

transfiera con probabilidad ½, y que al final B sepa con certeza si la ha recibido, pero A no. Uno de los algoritmos de Transferencia Inconsciente más sencillos y conocidos es el Protocolo de Rabin, en el que el secreto que se pretende transferir inconscientemente es la factorización del producto de dos grandes números primos, y que se basa en el problema de la residuosidad cuadrática. Dicho protocolo se puede describir como sigue:

1. A escoge al azar dos grandes números primos p y q, y calcula y envía a B, $N=p\cdot q$.
2. B escoge un entero x al azar entre 1 y N-1 y primo con N, y envía $x^2 \pmod N$ a A.
3. A calcula gracias a su conocimiento de p y q, las cuatro raíces cuadradas diferentes de $x^2 \pmod N$, {x, N-x, y, N-y}, escoge una al azar y se la envía a B.
4. Si B recibe y o N-y, entonces puede calcular p y q gracias a que m.c.d.((x+y), N) es p o q. Si B recibe x o N-x, entonces no puede.

Otro algoritmo de Transferencia Inconsciente, esta vez basado en un criptosistema de clave pública es el Protocolo de Pfleeger, que se define de la forma siguiente:

1. A genera dos parejas de claves pública y privada.
2. B escoge una clave secreta propia k_B y una de las dos claves públicas de A al azar, cifrando con ésta su clave k_B y enviando el resultado a A.
3. A escoge al azar una de sus dos claves privadas, descifra con ella el mensaje recibido, cifra con el resultado de este descifrado un mensaje secreto y envía lo que obtiene a B.
4. Si la clave elegida por A en el paso 3 se corresponde con la clave pública escogida por B en el paso 2, B podrá descifrar el mensaje secreto de A. Si no, no podrá.

Los protocolos de Compromiso de Bits configuran la segunda piedra angular del área de los protocolos criptográficos. El objetivo de los esquemas de Compromiso de Bits es el siguiente: A se quiere comprometer frente a B con un valor, de forma que A no pueda cambiarlo, y B no pueda descubrir el valor hasta que A abra el compromiso. En la definición de estos esquemas es habitual plantear como analogía un sobre cerrado, que resulta a la vez inalterable e ilegible. Además, la apertura del compromiso puede verse como una correspondencia definida desde un gran dominio sobre el conjunto binario {0,1}, de forma que un bit se considera comprometido cuando el origen de la correspondencia para ese valor de salida puede ser cualquier elemento aleatorio. El esquema definido a partir de dicha correspondencia es inalterable cuando la correspondencia es función, es decir cuando cada elemento posee una única imagen, y es ilegible cuando las distribuciones del conjunto origen del 0 y las del conjunto origen del 1 son indistinguibles. Existen muchos algoritmos interesantes de Compromiso de Bits basados en diversas herramientas criptográficas típicas tales como cifrados de clave secreta, logaritmos discretos, o residuos cuadráticos. El primer esquema analizado a continuación es una propuesta de Compromiso de Bits basada en la otra primitiva criptográfica de Transferencia Inconsciente. En todos los algoritmos se repiten los pasos de compromiso, apertura y verificación, y están garantizadas las propiedades de ilegibilidad e inalterabilidad. A continuación vemos brevemente la definición de cada uno de los esquemas con el objetivo de comprometer en cada caso un bit b.

En el Compromiso de Bits basado en Transferencia Inconsciente que se describe a continuación la probabilidad de fraude de A es $\leq ½$, pero esta cota puede bajarse a 2^{-m} si se ejecuta m veces el algoritmo.

1. Compromiso: A escoge n bits aleatorios b_i, tales que $b_1+b_2+...+b_n=b$, y envía cada b_i por orden mediante Transferencia Inconsciente.
2. Apertura: A envía a B los bits b_i
3. Verificación: B compara los bits b_i recibidos en el paso de compromiso con los correspondientes de la apertura.

El siguiente esquema de Compromiso de Bits se basa en un Cifrado de Clave Secreta en el que en primer lugar A y B acuerdan una secuencia seudoaleatoria R:

1. Compromiso: A cifra con su clave secreta k, la secuencia R y el bit b, y envía a B el resultado $E_k(R,b)$.
2. Apertura: A envía a B la clave k.
3. Verificación: B descifra el mensaje, comprueba la secuencia R, y descubre el bit b.

El siguiente esquema de Compromiso de Bits está basado en el problema de los Logaritmos Discretos. En primer lugar A y B acuerdan un número primo p, un elemento a generador de Z_p y un elemento aleatorio s de Z_p:

1. Compromiso: A escoge al azar un entero y entre 0 y p-2, y calcula y envía a B el número $x=s^b a^y (\text{mod } p)$.
2. Apertura: A envía a B el entero y.
3. Verificación: B obtiene b y comprueba x.

El último esquema descrito se basa en el problema de la Residuosidad Cuadrática. Primero A y B acuerdan n=p*q, siendo p=q=3 (mod 4) información secreta de A:

1. Compromiso: A genera y envía a B:
   a. un residuo cuadrático mod n, si b=0.
   b. un no residuo cuadrático mod n, si b=1.
2. Apertura: A envía a B la factorización (p,q) de n.
3. Verificación: B obtiene b y comprueba si el número enviado es residuo o no.

Tal y como se puede observar el primer esquema descrito tiene la ventaja de requerir únicamente una comunicación unidireccional, mientras que en el resto se hace necesario un contacto previo entre las partes para acordar determinadas piezas de información que se usan durante el desarrollo del correspondiente protocolo.

Los protocolos bipartitos conocidos como Firmas de Contratos se caracterizan por que A y B quieren firmar simultáneamente un contrato a través de una red de comunicaciones de forma que ninguno pueda obtener la firma del otro sin haber firmado el contrato, garantizando además que ninguno pueda repudiar su propia firma. En un intento simple de solución al problema, A y B se podrían intercambiar alternativamente bits de sus firmas digitales del contrato, de forma que si uno interrumpe el proceso, ambos tienen prácticamente la misma cantidad de firma del otro. El inconveniente de esta solución es que si uno envía basura en lugar de su firma, el otro no lo detecta hasta el final. De hecho, en general se ha demostrado que es imposible diseñar un protocolo determinístico que no requiera la participación de una tercera parte de confianza. Por tanto, el diseño de una Firmas de Contratos independiente de terceras partes de confianza ha de basarse en un proceso de aleatorización. En general las Firmas de Contratos aleatorizadas usan esquemas de Compromiso de Bits y se basan en Transferencias

Inconscientes. De hecho, concretamente cualquier protocolo de Transferencia Inconsciente puede ser adaptado a una Firma de Contratos sin más que aplicar la Transferencia Inconsciente sucesivamente, y considerar el contrato firmado al final de la ejecución si ambos usuarios logran conocer la información secreta del otro.

Los Lanzamientos de Monedas son protocolos bipartitos en los que se trata de generar una secuencia aleatoria común a dos usuarios de forma que el que actúa como generador de la secuencia no pueda elegir una secuencia no aleatoria particular, y a la vez el otro usuario no pueda anticipar el resultado del lanzamiento. La aplicación práctica más inmediata de este tipo de protocolos se encuentra en la generación de claves secretas seudoaleatorias compartidas entre dos usuarios, imprescindibles en los sistemas de clave secreta; aunque también los Lanzamientos de Monedas resultan útiles a la hora de diseñar protocolos multipartitos tales como el Póquer Mental. Se puede diseñar un Lanzamiento de Monedas a partir de cualquier Transferencia Inconsciente de manera que el usuario B gana si logra recibir el secreto transferido. También es posible definir un Lanzamiento de Monedas basándose en un esquema cualquiera de Compromiso de Bits. En este caso A y B escogen al azar sendos bits aleatorios *a* y *b,* y se los intercambian mediante Compromiso de Bits. Tras las fases de apertura y verificación de los compromisos, ambos participantes toman como resultado del lanzamiento *a+b.* Aparte de los esquemas generales mencionados, basados en Transferencia Inconsciente y Compromiso de Bits, existen otras muchas propuestas basadas en herramientas criptográficas típicas como funciones hash, criptografía de clave secreta, o el problema de la residuosidad cuadrática, por ejemplo.

En los esquemas multipartitos de Compartición de Secretos un secreto se divide en partes (sombras) que se distribuyen entre varios participantes de forma que sólo cuando un número de ellas (no necesariamente todas) se reúnen es posible reconstruir el secreto. Si P es el conjunto de sombras, se llaman estructuras de acceso ÍG2^{P} a los distintos subconjuntos de sombras que permiten calcular el secreto, y agrupación autorizada a cada uno de esos subconjuntos. Un esquema perfecto es el que permite a las agrupaciones autorizadas obtener el secreto, y no permite a ninguna agrupación no autorizada conseguir ninguna información sobre el secreto. Se ha demostrado que siempre se puede diseñar un esquema perfecto a partir de cualquier estructura de acceso. La base del esquema es el subconjunto de agrupaciones autorizadas minimales, y la tasa de información mide la cantidad de información que tienen los participantes. El esquema de Compartición de Secretos más conocido es el llamado esquema umbral *(t, w)* (con $t \leq w$) en el que un secreto *k* puede reconstruirse a partir de cualesquiera *t* de las *w* sombras, y *k* no puede reconstruirse mediante ningún subconjunto de *t-1* o menos sombras. El esquema umbral más conocido es el Esquema de Shamir, que se basa en la interpolación polinomial para reconstruir una curva de grado *t-1* a partir de *t* puntos. El siguiente esquema más conocido se conoce como Esquema Umbral de las Sombras Congruentes, y se basa en el Teorema de los Restos Chinos Congruentes. Una de las principales aplicaciones de la Compartición de Secretos es la Criptografía Visual, usada con el propósito de cifrar una imagen que podrá ser reconstruida mediante la superposición de un número mínimo de sombras.

Conclusiones

Hemos analizado aquí diversas aplicaciones criptográficas que permiten resolver en el mundo digital varios problemas habituales en el mundo real, llegando a mejorar en muchos casos las condiciones de seguridad. Hemos estudiado varios protocolos diseñados a partir de herramientas matemáticas para tareas específicas diferentes, aunque íntimamente relacionados

en su diseño. De hecho, se han investigado tanto las relaciones generales como las provenientes de diseños particulares entre los diferentes protocolos analizados.

Bibliografía

P. Caballero Gil, C. Hernández Goya, C. Bruno Castañeda: *Cryptographic Applications.* Computational Mathematics, Narosha Publishing House, New Delhi, India, 2002.

B. Schneier: *Applied Cryptography*. J. Wiley & Sons, 1993.

En Internet

http://www.kriptopolis.com

Kriptópolis

Revista independiente sobre privacidad y seguridad en Internet.

http://www.iec.csic.es/criptonomicon

Criptonomicón

Página del Instituto de Física Aplicada del CSIC dedicada a la criptografía y la seguridad en Internet.

http://www.criptored.upm.es

CriptoRed

Red Temática Iberoamericana de Criptografía y Seguridad de la Información.