Algoritmos Asimétricos

David Tito Joaquin Bernal Cortes
Corporación Universitaria Minuto de Dios
dbernal9@uniminuto.edu.co
Facultad de Ingeniería

1 Introducción

Los algoritmos asimétricos utilizan diferentes claves para cifrar y descifrar datos. Un ejemplo de cifrado asimétrico es la criptografía de clave pública. La criptografía de clave pública utiliza dos claves que forman un par de claves llamadas clave pública y clave privada. La clave que cifra el texto sin formato no se puede utilizar para descifrar el texto cifrado. La clave pública cifra el texto plano y la clave privada descifra el texto cifrado.

Clave pública: Proporcionada a quienes le envían datos cifrados.
Clave privada: Una clave en posesión exclusiva del usuario. Cuando un mensaje de texto sin formato se cifra con la clave pública, solo el poseedor de la clave privada puede descifrar el texto cifrado. Cuando un mensaje de texto sin formato se cifra con la clave privada, cualquiera que tenga la clave pública puede descifrarlo. Existe una certeza absoluta de que el mensaje de texto sin formato se originó con el poseedor de la clave privada. Las claves asimétricas proporcionan autenticación, integridad y no repudio. También pueden respaldar la confidencialidad cuando se utilizan para la gestión de claves.

2 Algoritmos Asimétrico existente:

A. Diffie-Hellman

El algoritmo de intercambio de claves Diffie-Hellman fue publicado por primera vez en 1976 por Whitfield Diffie y Martin Hellman, aunque el algoritmo había sido inventado unos años antes por la agencia de inteligencia del gobierno británico GCHQ pero se mantuvo clasificado. En 2002, Martin Hellman sugirió que el algoritmo fuera renombrado como "El intercambio de claves Diffie-Hellman-Merkle" en reconocimiento de la contribución de Ralph Merkle a la criptografía de clave pública.

El algoritmo de intercambio de claves Diffie-Hellman resuelve el siguiente problema: Alice y Bob quieren compartir una clave secreta para, por ejemplo, un algoritmo de clave simétrica como DES o AES , pero solo pueden comunicarse a través de un canal inseguro que es escuchado por su adversario Eva. Es decir, todos los mensajes enviados entre Alice y Bob son observados por Eve.
En la figura siguiente se muestra un ejemplo de funcionamiento del protocolo Diffie-Hellman.

Los valores de “p” y “g” son públicos y cualquier atacante puede conocerlos, pero esto no supone una vulnerabilidad. Aunque un atacante conociese dichos valores y

capturara los dos mensajes enviados entre las máquinas A y B, no sería capaz de averiguar la clave secreta. A continuación se muestra la información capturada por un atacante en el escenario de la Figura 46:

(ga mod p) = 8 → (5a mod 23) = 8
(gb mod p) = 19 → (5b mod 23) = 19

A partir de las ecuaciones anteriores, intentar calcular los valores de “a” y “b” es lo que se conoce como el problema del algoritmo discreto, un problema que se cree computacionalmente intratable y cuya notación es la siguiente:

a = log discg (ga mod p) = log disc 5 (8)
b = log discg (gb mod p) = log disc 5 (19)

Con los valores del ejemplo sí que es posible encontrar la solución, ya que se ha escogido un número primo “p” muy pequeño (p = 23), y se sabe que “a” y “b” son menores que “p”. Por lo tanto, para obtener los valores secretos en este ejemplo, un atacante tendría que probar sólo 22 posibles valores.

Por suerte, las implementaciones actuales del protocolo Diffie-Hellman utilizan números primos muy grandes, lo que impide a un atacante calcular los valores de “a” y “b”. El valor “g” no necesita ser grande, y en la práctica su valor es 2 ó 5. En el RFC 3526 aparecen publicados los números primos que deben utilizarse. A modo de ejemplo, se facilita aquí el número primo de 1024 bytes propuesto. El valor “g” utilizado es 2:

p = 28192 – 28128 – 1 + 264 x ((28062 pi) + 4743158)

B. DSA

(Digital Signature Algorithm en español Algoritmo de Firma Digital) es un estándar del Gobierno Federal de los Estados Unidos de América o FIPS para firmas digitales. Fue un algoritmo propuesto por el Instituto Nacional de Normas y Tecnología de los Estados Unidos para su uso en su Estándar de Firma Digital (DSS), especificado en el FIPS 186. DSA se hizo publico el 30 de Agosto de 1991, este algoritmo como su nombre lo indica, sirve para firmar y no para cifrar informacion. Una desventaja de este algoritmo es que requiere mucho mas tiempo de computo que RSA.

El esquema de la firma está correcto en el sentido que el verificador aceptará siempre firmas genuinas. Esto puede ser demostrado como sigue:

De $g = h^z \pmod p$ sigue $g^q \equiv h^{qz} \equiv h^{v-1} \equiv 1 \pmod p$ por Pequeño teorema de Fermat. Ya que $g>1$ y q es primo sigue que g tiene orden q.

El firmante computa

$$s = k^{-1}(\text{SHA-1}(m) + xr) \mod q.$$

Entonces

$$\begin{aligned} k &\equiv \text{SHA-1}(m)s^{-1} + xrs^{-1} \\ &\equiv \text{SHA-1}(m)w + xrw \pmod q. \end{aligned}$$

Ya que g tiene orden q tenemos que

$$\begin{aligned} g^k &\equiv g^{\text{SHA}-1(m)w} g^{xrw} \\ &\equiv g^{\text{SHA}-1(m)w} y^{rw} \\ &\equiv g^{u1} y^{u2} \pmod p. \end{aligned}$$

Finalmente, la correctitud de DSA surge de

$$r = (g^k \mod p) \mod q = (g^{u1} y^{u2} \mod p) \mod q = v.$$

C. Cifrado El Gamal

El procedimiento de cifrado/descifrado ElGamal se refiere a un esquema de cifrado basado en el problema matemático del logaritmo discreto. Es un algoritmo de criptografía asimétrica basado en la idea de Diffie-Hellman y que funciona de una forma parecida a este algoritmo discreto.

El algoritmo de ElGamal puede ser utilizado tanto para generar firmas digitales como para cifrar o descifrar.

Fue descrito por Taher Elgamal en 19841 y se usa en software GNU Privacy Guard, versiones recientes de PGP, y otros sistemas criptográficos. Este algoritmo no está bajo ninguna patente lo que lo hace de uso libre.

La seguridad del algoritmo se basa en la suposición que la función utilizada es de un solo sentido debido a la dificultad de calcular un logaritmo discreto.

Cifrado [editar]

Supongamos que Bruce tiene un texto claro que quiere enviar cifrado a Alicia. Lo primero por hacer es convertir este texto en un entero m entre 1 y $p-1$ ($m \in Z_p$). Esto no es parte del cifrado, sino que es una manera de codificar estándar, conocida por todos. Luego Bruce escoge arbitrariamente un número $b \in \{2, \ldots, p-1\}$ (que mantendrá secreto) para finalmente calcular:

$$y_1 = g^b \pmod{p}$$
$$y_2 = K^b m \pmod{p}$$

El mensaje cifrado final corresponde a la tupla $C_b(m, b) = (y_1, y_2)$

Descifrado [editar]

Para descifrar aprovechamos que:

$$y_1^{-a} y_2 \pmod{p} = (g^b)^{-a} K^b m \pmod{p} = g^{-ab} (g^a)^b m \pmod{p} = (g^a)^{-b} (g^a)^b m \pmod{p} = m \pmod{p}$$

Por tanto para descifrar se tiene que realizar el siguiente cálculo:

$$y_1^{-a} y_2 \pmod{p}$$

donde y_1^{-a} representa el inverso de y_1^a módulo p lo que indica que para calcular el descifrado, es necesario conocer a, que es la clave privada de Alicia.

Por el pequeño teorema de Fermat se demuestra que $y_1^{-a} = y_1^{p-1-a}$ y lo podemos aplicar.

Demostración: El pequeño teorema de Fermat indica que $x^{p-1} \pmod{p} = 1$ con p primo y x>0 coprimo con p. Elevando a a la expresión obtenemos que $x^{p-1^a} \pmod{p} = 1^a = 1$. Por tanto $x^{(p-1)\cdot a} \pmod{p} = x^{ap-a} \pmod{p} = x^a \cdot x^{p-1-a} \pmod{p} = 1$. Despejando $x^{p-1-a} \pmod{p} = x^{-a} \pmod{p}$

D. Criptografía de curva elíptica

Es una variante de la criptografía asimétrica o de clave pública basada en las matemáticas de las curvas elípticas. Sus autores argumentan que la CCE puede ser más rápida y usar claves más cortas que los métodos antiguos —como RSA— al tiempo que proporcionan un nivel de seguridad equivalente. La utilización de curvas elípticas en criptografía fue propuesta de forma independiente por Neal Koblitz y Victor Miller en 1985.

Sea $p > 3$ primo. La curva elíptica E: $y^2 = x^3 + ax + b$ sobre $\mathbb{Z}_p$ es el conjunto de soluciones $(x, y) \in \mathbb{Z}_p \times \mathbb{Z}_p$ en la congruencia

$$y^2 = x^3 \dotplus ax \dotplus b \pmod{p},$$

donde $a, b \in \mathbb{Z}_p$ son constantes tal que $4a^3 \dotplus 27b^2 \neq 0 \pmod{p}$

Se define una operación aditiva como sigue: Considerando que

$$P = (x_1, y_1)$$

y

$$Q = (x_2, y_2)$$

son puntos en E y $\mathcal{O}$ es un punto en el infinito. Si $x_2 = x_1$ e $y_2 = -y_1$, entonces $P + Q = \mathcal{O}$; de lo contrario $P + Q = (x_3, y_3)$, donde

$$x_3 = \lambda^2 - x_1 - x_2,$$
$$y_3 = \lambda(x_1 - x_3) - y_1,$$

y

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1}, & \text{si } P \neq Q \\ \dfrac{3x_1^2 + a}{2y_1}, & \text{si } P = Q \end{cases}.$$

Finalmente, definimos

$$P + \mathcal{O} = \mathcal{O} + P = P \, \forall P \in E.$$

Con esto se puede mostrar que *E* es un grupo abeliano con elemento identidad $\mathcal{O}$. Cabe notar que la inversa de (*x*, *y*) (que se escribe como -(*x*, *y*) ya que la operación es aditiva) es (*x*, -*y*), para todo $(x, y) \in E$.

De acuerdo al teorema de Hasse, el número de puntos #*E* que contiene *E* es cercano a *p*. Más precisamente se satisface la siguiente desigualdad

$$p + 1 - 2\sqrt{p} \leq \#E \leq p + 1 + 2\sqrt{p}.$$

Como se sabe que cualquier grupo de orden primo es cíclico, lo que se requiere es encontrar un subgrupo de *E* de orden *q* (*q* primo) para tener un isomorfismo con $\mathbb{Z}_q$ donde el problema del logaritmo discreto sea intratable. En este caso, siendo α un generador del grupo cíclico (el cual puede ser cualquier elemento del grupo distinto de $\mathcal{O}$, la identidad), se pueden calcular las «potencias» de α (las que se escriben como múltiplos de α, debido a que la operación del grupo es aditiva).

E. Criptosistema de Merkle-Hellman

Fue uno de los primeros criptosistemas de llave pública y fue inventado por Ralph Merkle y Martin Hellman en 1978.1 Aunque sus ideas eran elegantes, y mucho más simples que RSA, no tuvo el mismo éxito que este último, debido a que MH ya fue roto,2 y además no ofrece funcionalidades para firmar.

Generación de las claves [editar]

Para cifrar un mensaje de *n*-bits, elegir una secuencia supercreciente :

$$w = (w_1, w_2, \ldots, w_n) \text{ tal que } w_{i+1} > \sum_{j=1}^{i} w_j$$

de *n* números naturales (distintos de cero). Elegir un número *q* (preferiblemente al azar), tal que

$$q > \sum_{i=1}^{n} w_i$$

y otro número entero, *r* tal que mcd(r,q) = 1.

q es escogido de esta forma para asegurar la unicidad del texto cifrado. Si fuera menor, podría haber varios textos claros que resultarían en el mismo texto cifrado. *r* debe ser coprimo con *q* puesto que de otra forma podría no tener inverso en $\pmod{q}$. La existencia del inverso de *r* es necesaria para que se pueda realizar el descifrado.

A continuación, se calcula la secuencia:

$$\beta = (\beta_1, \beta_2, \ldots, \beta_n) \text{ tal que } \beta_i = r w_i \pmod{q}$$

La clave pública es β , mientras que la llave privada es (w, q, r) .

Descifrado [editar]

Para descifrar el criptograma *c*, el receptor tiene que encontrar los bits del mensaje α_i tales que satisfacen

$$c = \sum_{i=1}^{n} \alpha_i \beta_i.$$

Este problema sería difícil de resolver si los β_i fueran valores aleatorios, debido a que el receptor tendría que resolver una instancia del problema de la mochila, el cual se sabe que es NP-hard. Sin embargo, los valores β_i fueron elegidos de forma que el descifrado sea fácil si la clave privada (w, q, r) es conocida.

Para el descifrado se debe encontrar un entero *s* tal que es el inverso de *r* módulo *q*. Esto es, *s* satisface la ecuación :

$$rs \equiv 1 \pmod{q}$$

o equivalentemente, existe un entero *k* tal que *sr* = *kq* + 1. Dado que *r* fue escogido como un coprimo de *q* es posible encontrar *s* y *k* usando el Algoritmo de Euclides extendido. Luego el receptor del criptograma *c* calcula:

$$c' \equiv cs \pmod{q}.$$

Por tanto

$$c' \equiv cs \equiv \sum_{i=1}^{n} \alpha_i \beta_i s \pmod{q}.$$

Ya que $rs \equiv 1 \pmod{q}$ y $\beta_i \equiv r w_i \pmod{q}$ entonces

$$\beta_i s \equiv w_i r s \equiv w_i \pmod{q}.$$

Con esto

$$c' \equiv \sum_{i=1}^{n} \alpha_i w_i \pmod{q}.$$

La suma de todos los valores w_i es menor que *q* y por ende $\sum_{i=1}^{n} \alpha_i w_i \mod q$ también está en el intervalo $[0, q-1]$.

De este modo el receptor tiene que resolver el siguiente problema de la mochila.

$$c' = \sum_{i=1}^{n} \alpha_i w_i.$$

Este problema es fácil debido a que la secuencia *w* es supercreciente

F. RSA

3 Objectivos de los algoritmos Asimétricos

es suministrar la dificultad máxima al proceso de desencriptar los datos sin utilizar la llave exacta garantías de seguridad de la información en el proceso que se implemente para

asegurar la información que circula diariamente por ella, algo que es de suma importancia para los desarrolladores de sistemas pues de ésto depende la confiabilidad que se le ofrezca a los usuarios.

4 Conclusion:

Los Algoritmos Asimétricos son uno de los métodos para poder proteger tu información, forma parte de la seguridad informática que cada usuario puede tener, en la información anterior podemos observar que la Algoritmos Asimétricos tiene su historia, y mediante esta podemos observar las diferentes opciones que esta nos otorga para poder cuidar nuestra información, sin embargo, en este laboratorio, me intereso el tema de cifrar archivos con algoritmos complejos ya que logra una mayor confidencialidad.