Sistemas de claves públicas y privadas

Noemí P. Kisbye

Resumen

La criptografía consiste en transformar un texto claro en otro texto cifrado, de modo que este último sea ilegible a menos que se conozca la clave para descifrarlo. A través del tiempo se han utilizado diversos métodos: claves de sustitución, claves de desplazamiento, y otros.

Una de las desventajas de estos métodos tradicionales es que se utiliza la misma clave para cifrar y descifrar. Por lo tanto, es necesario que las partes intervinientes en la comunicación acuerden previamente la clave a utilizar. Se corre el riesgo entonces que un tercero pueda interceptar la clave y sea capaz de descifrar el mensaje, y por lo tanto no pueda usarse una misma clave reiteradamente.

En 1976, Hellman y Diffie idearon un sistema de criptografía llamado de *clave pública* o *asimétrica* que permite cifrar mensajes con una clave de cifrado conocida públicamente, pero tal que la clave para descifrar sea sólo conocida por el destinatario. En 1978, los matemáticos Ron Rivest, Adi Shamir y Len Adleman publicaron un algoritmo de generación de claves públicas, conocido como el algoritmo RSA, cuya eficiencia está basada en la dificultad de factorizar números enteros relativamente grandes y convenientemente elegidos como producto de números primos.

1. Criptografía de clave pública

El método de *clave pública o asimétrica*, ideado por Whitfield Diffie y Martin Hellman (1976) propone que cada parte que interviene en una comunicación posea dos claves diferentes: una pública y otra privada. El emisor de un mensaje M cifra el mismo utilizando la *clave pública* del destinatario, éste recibe el texto cifrado C y lo descifra con su propia *clave privada* obteniendo nuevamente M.

Así, cada miembro de la comunicación posee un par de estas claves, manteniendo en secreto la clave privada y dando a conocer públicamente la clave pública.

Estas dos claves son inversas una de la otra, es decir, lo que es cifrado por una de ellas es descifrado por la otra. Ahora bien, aún conociendo conociendo el texto cifrado C y la clave pública que se utilizó para cifrarlo, no es posible recuperar el mensaje M. Es por ello que la clave pública puede ser dada a conocer. Ya no hay riesgo de que un tercero intercepte la comunicación ya que sólo podrá descifrar el mensaje con la clave privada del destinatario.

Una descripción matemática de la idea de Diffie y Hellman es la siguiente: cada parte de una comunicación genera una función que sirve para cifrar mensajes y que tiene las siguientes propiedades:

1. la función transforma un *texto claro* o texto a enviar M en un *texto cifrado* C,
2. existe la inversa de esta función que a su vez permite transformar el texto C en M,
3. existen algoritmos eficientes que permiten calcular tanto la función como su inversa,
4. conocida la función es computacionalmente imposible conocer su inversa.

Supongamos entonces que en un grupo de usuarios desean enviarse mensajes secretos. Cada uno de ellos elige una función o algoritmo de cifrado y su correspondiente función inversa o algoritmo de decodificación. El algoritmo de cifrado es comunicado al resto, y pasa a ser la *clave pública* y se mantiene en secreto el algoritmo de decodificación o *clave privada*. Así, si A desea enviarle un mensaje a B, utilizará la clave pública de B para cifrar de modo que sólo B pueda descifrarlo, dado que sólo B conoce su propia clave privada. Si ahora B le responde a A, entonces B utiliza la clave pública de A para que sólo A pueda descifrar el mensaje con su propia clave privada.

Este sistema puede ser además utilizado para enviar firmas digitales. Así por ejemplo, si A envía su firma a B lo hace cifrándola con su clave privada. B des-

cifra el mensaje con la clave pública de A y si recibe la firma de A sabe que sólo A pudo haberla enviado.

Ahora bien, ¿es posible hallar funciones de tales características? ¿Existe algún método relativamente simple para generarlas? Y siendo así, ¿existen infinitas funciones de este tipo?

Los matemáticos Ron Rivest, Adi Shamir y Len Adleman publicaron en 1978 un algoritmo que permite generar claves públicas conocido como el algoritmo RSA. La efectividad de este algoritmo se basa en la dificultad de factorizar números enteros muy grandes como producto de números primos. Si bien esto no es "matemáticamente imposible", en la práctica una computadora tardaría miles de años en factorizar un entero relativamente grande y convenientemente elegido.

El sistema de cifrado y descifrado ideado por Rivest, Shamir y Adleman consiste en lo siguiente. El mensaje a enviar o texto claro está dividido en bloques, donde cada bloque es un número menor que un cierto entero positivo n. Tengamos en cuenta que los mensajes se transmiten actualmente en forma electrónica, por lo que cada carácter es representado como un número, y el mensaje resulta ser en realidad un número entero con muchísimas cifras. La idea es dividir este gran número en varios bloques de números, cada uno menor que un cierto natural fijo n.

Cada bloque M se transmite en forma cifrada como un número C, también menor que n, y con la propiedad que, para ciertos naturales d y e se verifica:

$$M^d \equiv C \mod (n), \qquad \text{y} \qquad C^e \equiv M \mod (n). \tag{1}$$

Aquí, el símbolo $\equiv$ indica *congruencia* módulo n. Para quien no está familiarizado con este concepto, el mismo ha sido brevemente desarrollado en la §2.

Notemos que las fórmulas dadas en (1) implican que

$$(M^d)^e \equiv C^e \equiv M \mod (n), \qquad \text{o bien} \qquad M^{d\cdot e} \equiv M \mod (n),$$

es decir que los números e y d están especialmente elegidos en función del número n.

Así, cada terna de números n, e y d relacionados según la fórmula (1) produce un par de claves: $\{e, n\}$ y $\{d, n\}$. Dando a conocer los números d y n, éstos constituyen la clave pública del usuario y $\{e, n\}$ la clave privada.

Para que este sistema sea factible como método de cifrado y descifrado deben darse ciertas condiciones, a saber:

1. que sea posible y computacionalmente fácil hallar enteros e, d y n tales que $M^{d \cdot e} \equiv M \mod (n)$, para todo entero $M < n$,

2. que sea relativamente simple calcular M^d y C^e, para todos los valores $M < n$,

3. que sea imposible calcular e conocidos n y d.

El algoritmo RSA describe precisamente la forma de encontrar ternas de números n, d y e con las características dadas anteriormente, y se basa en un importante resultado de la Teoría de Números: El Teorema de Euler-Fermat. Este teorema afirma que si n es el producto de dos números primos distintos p y q, entonces para todo entero $M < n$ se cumple que

$$M^{k \cdot (p-1)(q-1)+1} \equiv M \mod (n),$$

cualquiera sea $k \in \mathbb{Z}$. La idea es entonces determinar un par de enteros e y d tales que

$$e \cdot d = k \cdot (p-1)(q-1) + 1, \qquad \text{para algún entero } k,$$

o lo que es lo mismo,

$$e \cdot d \equiv 1 \mod \big((p-1)(q-1)\big). \tag{2}$$

Para esto basta elegir un entero d que sea coprimo con $(p-1)(q-1)$ y su correspondiente inverso[1] e.

Ahora bien, si n es el producto de dos primos p y q relativamente grandes, del orden de 40 o más cifras, resulta prácticamente imposible factorizar el número n para

[1] Ver Definicion 2.6

quien no conoce p o q. En consecuencia, conocidos n y e, tampoco es posible calcular el valor de d en la fórmula (2), ya que para ello hace falta conocer los valores de p y q. Este hecho es precisamente lo que da eficiencia a este algoritmo.

Damos a continuación una sección en la que repasaremos algunas propiedades básicas relativas a la congruencia de números enteros. Al finalizar la misma enunciaremos dos importantes resultados: El Pequeño Teorema de Fermat (Teorema 2.9) y el Teorema de Euler-Fermat (Teorema 2.12). Ambos teoremas justifican el mecanismo de cifrado y descifrado propuesto por Rivest, Shamir y Adleman y que explicaremos más detalladamente en la §3.

2. Congruencias

Denotaremos con $\mathbb{N}$ y $\mathbb{Z}$ a los números naturales y enteros, respectivamente. Si m y n son números enteros, y $n \neq 0$ diremos que *m divide a n* o que *n es un múltiplo de m* si $n = q \cdot m$ para algún entero q. Equivalentemente, si el resto de la división de n por m es 0.

Se dice que un número entero p es *primo* si p es distinto de 1 y de -1 y sus únicos divisores son p, $-p$, 1 y -1. Por ejemplo, 2, 3, -11 y 53 son números primos. Se dice que dos números enteros son *coprimos* entre sí si no tienen ningún divisor común, excepto el 1 y el -1. Por ejemplo, 10 y 21 son coprimos entre sí, ya que 2, 5 y 10 no son divisores de 21. También podemos decir que dos números enteros a y b son coprimos si los primos que aparecen en la factorización de a no aparecen en la factorización de b.

Definición 2.1. Dado un número natural n, decimos que dos números enteros a y b son *congruentes módulo n* si $(a - b)$ es divisible por n y se escribe

$$a \equiv b \mod (n).$$

Por ejemplo,

$$27 \equiv 12 \mod 5 \quad \text{y} \quad -2 \equiv 16 \mod 3$$

puesto que $27 - 12 = 3 \cdot 5$, y $-2 - 16 = -18 = (-6) \cdot 3$.

Notemos que si el resto de la división por n de un entero a es r, entonces a es congruente a r módulo n. Por lo tanto, podemos decir que dos enteros son congruentes módulo n si tienen el mismo resto en la división por n.

Por ejemplo, 27 y 12 tienen resto 2 en la división por 5, mientras que -2 y 16 tienen resto 1 en la división por 3 (notar que $-2 = 3 \cdot (-1) + 1$).

Lema 2.2. *Si* $a \equiv b \mod (n)$ *y* $c \equiv d \mod (n)$, *entonces*

$$a + c \equiv b + d \mod (n) \quad y \quad a \cdot c \equiv b \cdot d \mod (n).$$

Prueba. Según la Definición 2.1 debemos mostrar que $(a+c)-(b+d)$ y $(a \cdot c)-(b \cdot d)$ son enteros divisibles por n. En efecto,

$$(a + c) - (b + d) = (a - b) + (c - d)$$
$$(a \cdot c) - (b \cdot d) = (a - b) \cdot c + b \cdot (c - d).$$

En ambos casos el miembro derecho es una suma de múltiplos de n, y por lo tanto es divisible por n. □

Una consecuencia de este lema es que si $a \equiv b \mod (n)$, entonces podemos multiplicar miembro a miembro k congruencias de éstas y obtener

$$a^k \equiv b^k \mod (n),$$

donde a^k y b^k indican el producto repetido k veces de a y b respectivamente.

Dado que además todo número es congruente a sí mismo, es decir $c \equiv c \mod (n)$, tenemos también que si $a \equiv b \mod (n)$ entonces

$$a \cdot c \equiv b \cdot c \mod (n),$$

cualquiera sea el entero c. Pero en general no es cierto que si $a \cdot c \equiv b \cdot c \mod (n)$ entonces $a \equiv b \mod (n)$, es decir no siempre es posible "simplificar". Veamos esto en un ejemplo:

$$6 \cdot 5 \equiv 4 \cdot 5 \mod (10),$$

pero no es cierto que $6 \equiv 4 \mod (10)$. Precisamos esto en el siguiente lema:

Lema 2.3. *Sean $a, b, c \in \mathbb{Z}$, $n \in \mathbb{N}$ tales que $a \cdot c \equiv b \cdot c \mod (n)$. Si c y n son coprimos entonces $a \equiv b \mod (n)$.*

En efecto, puesto que n divide a $a \cdot c - b \cdot c = (a - b) \cdot c$, entonces los divisores primos de n deben dividir a $(a - b)$ o a c. Al ser n y c coprimos entre sí esto implica que todo divisor primo de n divide a $a - c$, y por lo tanto n divide a $a - c$. Equivalentemente, $a \equiv c \mod (n)$.

2.1. El Pequeño Teorema de Fermat

Fijemos ahora un número natural n y consideremos el conjunto $\Phi(n)$ formado por todos los números naturales coprimos con n y menores que n, esto es

$$\Phi(n) = \{r \in \mathbb{N} \mid r < n \text{ y } x \text{ es coprimo con } n\}.$$

Si $r \in \Phi(n)$ ningún primo que divide a r es divisor de n. Por lo tanto, si multiplicamos dos elementos de $\Phi(n)$ obtenemos a su vez otro número coprimo con n que a su vez es congruente con un número menor que n y coprimo con n. Por ejemplo,

$$\Phi(8) = \{1, 3, 5, 7\}.$$

El producto $3 \cdot 7 = 21$ es congruente con 5 módulo 8, y $5 \in \Phi(8)$.

El siguiente lema afirma que si a es cualquier número entero coprimo con n, entonces al multiplicarlo por todos los coprimos menores que n se obtienen nuevamente, módulo n, todos estos números.

Veamos esto con un ejemplo. Sea $n = 10$ y $a = 13$. Entonces $\Phi(10) = \{1, 3, 7, 9\}$. Si multiplicamos a 13 por cada uno de estos números obtenemos:

$$13 \cdot 1 \equiv 3 \mod (10), \quad 13 \cdot 3 \equiv 9 \mod (10),$$

$$13 \cdot 7 \equiv 1 \mod (10), \quad 13 \cdot 9 \equiv 7 \mod (10).$$

Los resultados obtenidos son todos los elementos de $\Phi(10)$: 3, 9, 1 y 7.

Nota 2.4. Si a es un entero y n un natural, denotaremos $a \mod (n)$ al resto de la división de a por n. Así por ejemplo, $15 \mod (4) = 3$, $100 \mod 9 = 1$ y $27 \mod 3 = 0$.

Lema 2.5. *Si a es coprimo con n y $\Phi(n) = \{r_1, r_2, \ldots, r_k\}$, entonces*

$$\{a \cdot r_1 \mod (n), a \cdot r_2 \mod (n), \ldots, a \cdot r_k \mod (n)\} = \{r_1, r_2, \ldots, r_k\} = \Phi(n).$$

Prueba. Notemos que al multiplicar a a por cada uno de estos números se obtienen a lo sumo k números diferentes coprimos con n y menores que n.

Para mostrar que son exactamente k, veamos que dos de estos productos no pueden ser congruentes módulo n, es decir que todos son congruentes a un resto distinto módulo n. En efecto, sean x e y dos números coprimos con n y tales que

$$a \cdot x \equiv a \cdot y \mod (n).$$

Entonces por Lema 2.3 debe ser que $x \equiv y \mod (n)$. Si además se tiene la hipótesis de que x e y son menores que n esto puede darse sólo si $x = y$.

□

Definición 2.6. Sea a entero y n natural. Si existe r entero tal que

$$a \cdot r \equiv 1 \mod (n),$$

entonces a se dice *inversible* y r es un *inverso* de a módulo n.

Notemos que si r es un inverso de a, entonces $r + k \cdot n$ también lo es, cualquiera sea el entero k. Por ello se dice que r es *un* inverso.

Corolario 2.7. *(del Lema 2.5): Si $a \in \Phi(n)$ entones a es inversible y existe un único natural r inverso de a tal que $r < n$. Además, $r \in \Phi(n)$.*

Prueba. Puesto que $1 \in \Phi(n)$, entonces, en la notación del Lema 2.5, existe $r_j \in \Phi(n)$ tal que $a \cdot r_j \equiv 1 \mod (n)$. Entonces r_j es inverso de a.

La unicidad se sigue del Lema 2.3. □

Nota 2.8. Notemos que si p es un número primo, entonces todos los números naturales menores que p son coprimos con p, y por lo tanto inversibles. Es decir, si p es primo entonces

$$\Phi(p) = \{1, 2, \ldots, (p-1)\}.$$

Teorema 2.9 (Pequeño Teorema de Fermat). *Si a es un entero y p es un primo positivo, entonces*

$$a^p \equiv a \mod (p).$$

Prueba. Si p divide a a entonces p divide a a^p, por lo que el resultado es obvio. Si p es coprimo con a, por el Lema 2.5 y la Nota 2.8 tenemos que en las $p-1$ congruencias:

$$1 \cdot a \equiv r_1 \mod (p) \tag{3}$$

$$2 \cdot a \equiv r_2 \mod (p) \tag{4}$$

$$\vdots \tag{5}$$

$$(p-1) \cdot a \equiv r_{p-1} \mod (p), \tag{6}$$

los números $r_1, r_2, \ldots, r_{p-1}$ constituyen una permutación de los números $1, 2, \ldots, p-1$.

Si multiplicamos miembro a miembro todas estas congruencias obtenemos que

$$1 \cdot 2 \cdots (p-1) \cdot a^{p-1} \equiv 1 \cdot 2 \cdots (p-1) \mod (p),$$

y dado que p es coprimo con todos los números menores que él y por lo tanto con su producto, concluimos que

$$a^{p-1} \equiv 1 \mod (p).$$

Multiplicando ambos miembros por a,

$$a^p \equiv a \mod (p),$$

como queríamos probar. □

Corolario 2.10. *Si p es un primo positivo y a es coprimo con p, entonces* $a^{p-1} \equiv 1 \mod (p)$.

2.2. El Teorema de Euler-Fermat

Consideremos ahora un número n natural arbitrario, y denotemos con $\phi(n)$ al número de elementos de $\Phi(n)$. Esto es:

$$\phi(n) := |\{r \in \mathbb{N} \mid r < n \text{ y } r \text{ es coprimo con } n\}|,$$

donde $|A|$ denota la cantidad de elementos o *cardinal* del conjunto A.

Tenemos entonces los siguientes ejemplos:

$$\phi(10) = |\{1, 3, 7, 9\}| = 4, \qquad \phi(14) = |\{1, 3, 5, 9, 11, 13\}| = 6,$$

$$\phi(5) = |\{1, 2, 3, 4\}| = 4.$$

Si p es un número primo entonces $\phi(p) = p - 1$.

Lema 2.11. *Si $n = p \cdot q$, con p y q primos positivos distintos, entonces*

$$\phi(n) = (p - 1) \cdot (q - 1).$$

Prueba. En efecto, de los $n - 1$ números menores que n debemos descontar aquellos no coprimos con n. Estos son precisamente los múltiplos de p y los múltiplos de q, es decir:

$$p,\ 2p,\ \ldots, (q-1)p, \qquad \text{y} \qquad q,\ 2q,\ \ldots, (p-1)q.$$

Luego hay $(q - 1) + (p - 1) = (p + q - 2)$ números menores que n no coprimos con n y por lo tanto

$$\phi(n) = (n - 1) - (p + q - 2) = p \cdot q - p - q + 1 = (p - 1) \cdot (q - 1),$$

como habíamos afirmado. □

Estamos entonces en condiciones de enunciar y demostrar el Teorema de Euler-Fermat, teorema sobre el cual se basa el algoritmo RSA.

Teorema 2.12 (Teorema de Euler-Fermat). *Sea $n = p \cdot q$, con p y q primos distintos, y sea M un natural menor que n. Entonces para todo k entero se cumple que*

$$M^{k \cdot \phi(n)+1} \equiv M \mod (n).$$

Prueba. Para probar este teorema, consideraremos dos casos: i) M y n coprimos y ii) M y n no coprimos.

En el caso i) procedemos de una manera análoga a la demostración del Teorema 2.9. Consideremos $a_1, a_2, \ldots, a_{\phi(n)}$ todos los números menores que n coprimos con n, y las $\phi(n)$ congruencias

$$a_1 \cdot M \equiv x_1 \mod (n) \tag{7}$$

$$a_2 \cdot M \equiv x_2 \mod (n) \tag{8}$$

$$\vdots \tag{9}$$

$$a_{\phi(n)} \cdot M \equiv x_{\phi(n)} \mod (n). \tag{10}$$

En este caso, los números $x_1, x_2, \ldots, x_{\phi(n)}$ se eligen todos menores que n y coprimos con n, por lo que resultan ser una permutación de los números $a_1, a_2, \ldots, a_{\phi(n)}$. Multiplicando miembro a miembro todas estas congruencias obtenemos

$$a_1 \cdot a_2 \cdots a_{\phi(n)} M^{\phi(n)} \equiv a_1 \cdot a_2 \cdots a_{\phi(n)} \mod (n),$$

de donde concluimos que

$$M^{\phi(n)} \equiv 1 \mod (n).$$

Luego, para cualquier k natural, $(M^{\phi(n)})^k = M^{k \cdot \phi(n)} \equiv 1 \mod (n)$, y multiplicando ambos miembros por M resulta

$$M^{k \cdot \phi(n)+1} \equiv M \mod (n).$$

En el caso ii), M es divisible por p o por q (pero no por ambos). Podemos asumir que es divisible por q y coprimo con p. Entonces $M = c \cdot q$ para algún natural c. Por Corolario 2.10 podemos afirmar entonces que

$$M^{p-1} \equiv 1 \mod (p).$$

Elevando ambos miembros al exponente $q-1$ tenemos que

$$M^{(p-1)(q-1)} \equiv 1 \mod (p), \qquad \text{es decir} \qquad M^{\phi(n)} \equiv 1 \mod (p),$$

y por lo tanto, para cualquier natural k

$$M^{k\cdot\phi(n)} \equiv 1 \mod (p).$$

Esto significa que $M^{k\cdot\phi(n)} - 1$ es un múltiplo de p, y multiplicando por $M = c \cdot q$ obtenemos que $(M^{k\cdot\phi(n)} - 1) \cdot M$ es un múltiplo de $p \cdot q$, es decir un múltiplo de n, esto es

$$\left(M^{k\cdot\phi(n)} - 1\right) \cdot M = (\lambda \cdot p) \cdot (c \cdot q) = (\lambda \cdot c) \cdot n.$$

Se sigue entonces que

$$M^{k\cdot\phi(n)} M - M = M^{k\cdot\phi(n)+1} - M$$

es un múltiplo de n, o lo que es lo mismo

$$M^{k\cdot\phi(n)+1} \equiv M \mod (n).$$

□

3. El algoritmo RSA

Estamos en condiciones entonces de entender el algoritmo RSA de generación de claves públicas ideado por Rivest, Shamir y Adleman. El algoritmo RSA consta de los siguientes pasos:

1. elegir dos números primos p y q,
2. calcular $n = p \cdot q$,
3. calcular $\phi(n) = (p-1) \cdot (q-1)$,
4. determinar un entero e, $e < \phi(n)$, coprimo con $\phi(n)$,
5. calcular d, el inverso de e módulo $\phi(n)$.

El paso 1 es sólo conocido por quien genera la clave, y la idea es elegir dos números primos distintos muy grandes, del orden de 40 cifras o más. El producto de estos dos primos, $n = p \cdot q$, sí es comunicado públicamente. El cálculo de $\phi(n)$ y la elección de e en el paso 4 también son privados y el par $\{e, n\}$ constituye la clave privada, mientras que el inverso de e junto con n forman la clave pública $\{d, n\}$.

Puesto que existen infinidad de números primos, esto garantiza la existencia de infinitos pares de claves públicas y privadas generadas mediante este método. Claramente, si sólo hubiera una cantidad finita, podrían conocerse todas las claves y este algoritmo sería ineficiente.

Supongamos entonces que el señor A ha publicado su clave pública $\{d, n\}$ y que el señor B desea enviar a A un mensaje M. Entonces B calcula $C \equiv M^e \mod (n)$ y envía a A el texto cifrado C. A recibe entonces el mensaje cifrado C y lo descifra calculando $M \equiv C^e \mod (n)$.

Veamos esto con un ejemplo. Para generar una clave pública el señor A sigue los siguientes pasos:

1. elige dos números primos: $p = 13$ y $q = 7$,
2. calcula $n = p \cdot q = 13 \cdot 7 = 91$,
3. calcula $\phi(n) = \phi(91) = 12 \cdot 6 = 72$,
4. elige e coprimo con 72, por ejemplo $e = 5$,
5. calcula el entero d, $d < 72$ tal que $d \cdot 5 \equiv 1 \mod 72$. En este caso, $d = 29$, puesto que $29 \cdot 5 = 145 = 72 \cdot 2 + 1$.

Las claves resultantes son entonces la clave pública $\{29, 91\}$ y la clave privada $\{5, 91\}$. Si el usuario B desea enviarle a A el mensaje $M = 61$ entonces utiliza la clave pública de A y calcula $61^{29} \mod 91$. Para realizar este cálculo seguiremos los siguientes pasos:

Puesto que $61 \equiv -30 \mod 91$, se sigue que

$$61^{29} \equiv (-30)^{29} \equiv -30^{29} \mod 91.$$

Puesto que

$$30^2 = 900 = 910 - 10 \equiv -10 \mod 91$$

se tiene

$$30^4 \equiv (30^2)^2 \equiv (-10)^2 \equiv 100 \equiv 9 \mod 91.$$

Dado que $30^6 = 30^4 \cdot 30^2$, llegamos a que

$$30^6 \equiv (-10) \cdot 9 \equiv 1 \mod 91.$$

Entonces, siendo $29 = 6 \cdot 4 + 5$, tenemos que

$$30^{29} \equiv 30^{6 \cdot 4 + 5} \equiv (30^6)^4 \cdot 30^5 \equiv 30^4 \cdot 30 \equiv 9 \cdot 30 \equiv 270 \equiv -3 \mod 91.$$

Puesto que $61 \equiv -30 \mod 91$ y que $(-30)^{29} \equiv 3 \mod 91$, concluimos que

$$61^{29} \equiv 3 \mod 91.$$

Entonces A recibe de B el mensaje cifrado $C = 3$.

Ahora A descifra este mensaje utilizando su clave privada $\{5, 91\}$ y calcula

$$M \equiv 3^5 \mod 91.$$

Puesto que $3^5 = 243 = 91 \cdot 2 + 61$, entonces recibe $M = 61$, es decir, el mensaje enviado por B.

En este ejemplo hemos utilizado dos números primos pequeños para facilitar los cálculos posteriores. En la aplicación del algoritmo RSA los primos elegidos deben ser del orden de 40 cifras o más, y los cálculos son realizados por computadoras.

Si bien no se conocen métodos para generar números primos grandes, sí existen pruebas que permiten determinar, con un alto nivel de probabilidad, si un número es

o no un número primo. Estas pruebas se conocen por el nombre de *tests de primalidad* pero no las desarrollaremos en el presente artículo.

Referencias:

Network and internetwork security, William Stallings, Ed. Prentice Hall, 1995.

Patricia N. Kisbye. CIEM-FaMAF. Universidad Nacional de Córdoba. (5000).
E-mail: kisbye@mate.uncor.edu