Teoría de Números y Criptografía*

Claudio Gutiérrez

Desde siempre los números han tenido su encanto. K. F. Gauss, probablemente el matemático más destacado de los tiempos modernos, hace un par de siglos decía que las matemáticas son la reina de las ciencias y la teoría de números la reina de las matemáticas.

Tales afirmaciones no son gratuitas. En pocas disciplinas es posible hacer tanto con tan poco material, o dicho de otra forma, se pueden plantear tantos interesantes problemas conociendo sólo unas cuantas definiciones básicas. Por otro lado, los problemas más bellos (uno podría agregar también los más difíciles) de las matemáticas se encuentran ligados de una u otra forma a la teoría de números. El ejemplo clásico es seguramente la conjectura más famosa de las matemáticas, los ceros de la función Zeta de Riemann[1].

Ahora, dentro de la teoría de números, los números primos son una suerte de elegidos. Recordemos que un número p es *primo* si sus únicos divisores son él mismo y 1. Por ejemplo los números 2, 3, 5, 7, 11, 13, 17, 19, 23 son primos, mientras que 4, 6, 8, 9, 10, 12, 14, 15, 16, 18 *no* son primos. Inmediatamente surge la pregunta: ¿hay infinitos números primos? Euclides demostró que sí. Otro problema es cómo encontrarlos, o cómo saber si un número es primo (como veremos más adelante, éstos son dos problemas que han pasado a tener gran importancia hoy en día).

*Versión con modificaciones menores del artículo aparecido en la Revista del Profesor, Sociedad Matemática de Chile, Año 3, No. 2, 1998.

[1]La formulación de esta conjetura, cuya solución pareciera no estar cercana, dice así: los números complejos z con parte real $0 \leq \mathrm{re}(z) \leq 1$ en los cuales se anula la función compleja definida por la serie infinita

$$\zeta(z) = \sum_{n=0}^{\infty} \frac{1}{n^z}$$

están en la línea $\mathrm{re}(z) = \frac{1}{2}$. Las profundas relaciones de esta conjetura con la teoría de números escapan los límites de este artículo.

Es interesante hacer notar que sabiendo sólo la definición de número primo, uno ya puede formular problemas complejos, por ejemplo la famosa conjetura que Goldbach formuló en 1742:

> Todo número par mayor que 6 se puede expresar como suma de dos números primos mayores que 2.

Algunos ejemplos: $6 = 3 + 3$, $8 = 3 + 5$, $20 = 3 + 17$, $100 = 43 + 47$. Hasta hoy aún se desconoce si esto es verdad en general o no, lo que dice bastante de la dificultad de este simple y lindo problema.

Desgraciadamente estos despliegues de belleza "pura" parecieran no impresionar a aquellos que hoy día deciden los presupuestos educacionales y científicos, donde pareciera primar un criterio de utilidad inmediata. El objetivo de este artículo es presentar la sorpresas que la teoría de números le tenía reservada a estos modernos traficantes que, en palabras de Oscar Wilde, "saben los precios de todo pero el valor de nada".

1 La "utilidad" de la Teoría de Números

Es claro que cualquier respuesta a esta pregunta depende de qué se entienda por "utilidad". Por ejemplo, creo que todos estaríamos de acuerdo en que encontrar números primos grandes no pasa el famoso test de Gandhi:

> Recuerda la cara del hombre más pobre y desamparado que hayas visto, y pregúntate a tí mismo si el paso que contemplas dar será de algún uso para él. ¿Ganará algo con ello? ¿Le devolverá el control sobre su vida y destino?

Aunque aquí el problema pareciera ser la excesiva exigencia del test para imperfectos mortales como nosotros, ¿no queda realmente la impresión de que la teoría de números no pasaría ningún test de utilidad?

Curiosamente hasta hace poco cualquier matemático habría concordado en ello; es decir, que los tópicos de teoría de números poco o nada influyen en los asuntos mundanos. El famoso matemático G. Hardy, dedicado a esta disciplina, afirmaba provocativamente que estaba orgulloso de que lo que él hacia no servía para nada:

> ...tanto Gauss y otros matemáticos estaban en lo cierto cuando se regocijaban que hay al menos una ciencia [la teoría de números], su propia ciencia,

> cuya lejanía de las actividades humanas ordinarias la mantendría elegante y limpia.

Sólo recientemente, estos pronósticos han dejado de ser ciertos, y de la manera más inesperada: la teoría de números se ha convertido en herramienta central de una de las más delicadas áreas de cualquier empresa comercial como es la seguridad de la información.

Clásicamente nadie más que los militares (y uno que otro banquero) estaban interesados en mantener el secreto de la información que poseían o transmitían. Por ello es que la *criptografía* era mirada esencialmente como una curiosidad bélica y su utilidad estaba principalmente restringida a academias militares y ... a alguno que otro "servicio" lateral como el siguiente, relatado en las memorias de J. Casanova:

> Cinco o seis semanas después, Madame dÚrfé me preguntó si yo había decifrado su manuscrito. Le dije que sí.
> – ¿Sin la clave señor?, excúseme si le digo que creo que es imposible
> – ¿Desea que le diga la clave Madame?
> – Si usted lo desea.
> Cuando le dije la palabra clave, que no pertenecía a ningún lenguage, ví en su cara la sorpresa. Me dijo que era imposible. Le pude haber dicho la verdad (...) pero por capricho le dije que un genio me la había revelado. Esta falsedad dejó a Madame dÚrfé a mis pies. Ese día llegué a ser el dueño de su alma, y abusé de mi poder ...

Hoy por hoy las cosas han cambiado bastante debido esencialmente al vertiginoso avance en el procesamiento y transmisión de la información producto de la computación. Aunque seguramente a más de alguien le sigue prestando servicios como el anterior, la criptografía actualmente (aparte de las clásicas aplicaciones militares) está orientada a proveer *seguridad en la información.* Hay ciertos servicios que podríamos considerar los básicos:

1. Confidencialidad de la información: cómo codificar la información de tal forma que terceros no puedan leerla.

2. Integridad de la información: cómo asegurarse que terceros no desvirtuaron parte de la información para darle otro sentido. Esto es particularmente sensible en situaciones militares, donde un bando no sólo desea interceptar la información, sino tambien *desinformar* al enemigo.

3. Sistemas de identificación o autentificación: En general, como saber que cierto documento fué escrito por tal persona (firmas). Ejemplos: para sacar dinero de un cajero automático, para llamados telefónicos usando tarjetas telefónicas o de crédito, para conectarse a un computador en una red. En todos estos casos uno necesita una contraseña, generalmente un número, que debe ser enviado a una central para ser verificado.

4. Protocolos de Cero-Conocimiento: ¿Cómo convencer a alguien de que se tiene la información requerida sin mostrársela completamente?

Todas estas cuestiones están relacionadas. Aunque parezca increíble, hoy en día la mayoría de las soluciones a estos problemas y sus implementaciones (reales) tienen que ver *directamente* con los números primos. ¿Cómo? Vamos por parte.

2 Criptografía

La criptografía clásica trata esencialmente del envío y recepción de mensajes secretos, es decir, sin que terceros –que pudieran interceptarlos– puedan leerlos. Hay muchas técnicas para lograr esto, la gran mayoría basada en la simple idea de reemplazar cada letra o grupo de letras por otro símbolo, y así cifrar el mensaje. Por ejemplo consideremos una codificación del abecedario castellano de 27 letras por su número correlativo: el espacio blanco por 00, "a" por 01, "b" por 02, etc. La frase "linda guagua" quedaría transformada en el número

$$120914040100072201072201$$

Para quien no conoce nuestra convención, es prácticamente imposible descifrar este mensaje[2].

Esta idea clásica de cifrado tiene un problema esencial (aparte de otros técnicos menores): si uno piensa en aplicaciones que *no* son militares, como las descritas más arriba, uno se da cuenta que el sistema clásico sirve muy poco. Para visualizar realmente las dimensiones del problema, imaginemos que dos profesores, A de Arica y B de Buin (que no necesariamente se conocen de antes) quieren empezar a intercambiar documentos secretos por fax, digamos de ciertos escándalos que descubrieron sobre los sueldos de los profesores en sus respectivos Liceos.

[2] Peor aún, en general este texto cifrado se descompone en pequeñas piezas para transmitirlo. Sin embargo, si el texto cifrado es más largo, hay buenas posibilidades que con mucho tiempo un buen criptoanalista logre descifrarlo. Para ello hay varias técnicas a disposición: análisis de frecuencia, etc. Con este tipo de ideas Champollion descifró los jeroglíficos egipcios.

Supongamos además que A y B no tienen ningún otro medio de comunicación. Por supuesto que las autoridades leen todo lo que llega y todo lo que sale en el fax y los profesores A y B desean evitar malos ratos.

¿Cómo hacerlo? Cualquier intento de acuerdo entre A y B respecto de claves o códigos a usar ("te mandaré el texto en quechua", "codificaré según el orden de aparición de letras en *Ternura* de G. Mistral") sería inmediatamente conocido por las autoridades del Liceo también. ¿Podrán entonces A y B burlar la vigilancia? No es mala idea pensar por un momento algún artificio. Una conclusión es clara: la criptografía clásica no nos ayuda.

3 Criptografía de clave pública

Clásicamente, el cifrado de un texto y su desciframiento fué siempre considerado un mismo proceso: si uno tiene la clave, puede cifrar y descifrar. Y viceversa: cifrar y descifrar pasa por tener un acuerdo bilateral previo respecto de la clave. Así, todo el secreto está en la clave (sino, pregúntenle a Casanova ...).

El descubrimiento de que ésto no tiene por qué ser así es el fundamento de lo que se conoce como *criptografía de clave pública.*

La idea es muy sencilla (aunque luego su implementación se complique un poco). Lo mejor para entenderla es quizás ilustrarla con una figura imaginaria.

Supongamos que la naturaleza nos proveyera de parejas de líquidos transparentes (A_p, A_s), (B_p, B_s), (C_p, C_s), etc. con la propiedad que al mezclar pares correspondientes se forman colores. Supongamos además que es prácticamente imposible detectar cualquiera de ellos por separado, sin conocer su exacta contraparte. Entonces tendríamos un sistema de cifrado muy eficiente *abierto al público*: dejo el tintero con el líquido C_p en una mesa abierta al público, y guardo en secreto su pareja C_s. Quien desea mandarme cartas secretas, va a la mesa, y escribe con la tinta invisible C_p que está allí. Nadie –ni el mismo que la escribió– las puede leer, salvo yo, que tengo el compuesto C_s, que me permite –al mojar la carta con él– transformar las letras invisibles a color rojo digamos. Nótese bien la diferencia con el antiguo sistema de cifrado: quien desee escribirme confidencialmente *no* necesita acuerdo previo conmigo: parte de mi clave es pública (la tinta C_p a disposición de todo el mundo).

Si cada persona pudiera tener un par de compuestos de este tipo, hay muchas cosas interesantes que se pueden hacer: por ejemplo, autentificar un documento. ¿Como sé yo que fué Amanda quien me envió un mensaje pidiéndome una cita el sábado y no fué su marido el que fue a la guía pública de claves, y escribió la carta simulando ser ella y me estará esperando el sábado revólver en mano? Sencillo: le pido a Amanda que me mande un documento cifrado y *firmado.* ¿Qué significa esto? Que Amanda va a la guía pública de claves (donde están todas las tintas con subíndices A_p, B_p, C_p, etc.), escribe con mi tinta pública C_p (cosa que nadie –incluído su marido que es amigo del cartero– la pueda leer) y finalmente escribe su nombre con *su* tinta secreta A_s. Me llega la carta, le aplico mi tinta secreta C_s, aparecen las letras rojas invitándome para el sábado. Aún no estoy seguro del remitente, aunque diga Amanda. Pero voy a la guía pública de claves, consigo la tinta pública A_p de Amanda, la aplico a la carta, y aparece –en letras verdes– su firma: al menos este sábado no me matan ...

El único problema para formalizar estas ideas es encontrar los famosos líquidos, y peor aún, enviarlos a través de ondas electromagnéticas (que es como generalmente se dan las comunicaciones hoy en día). La idea básica en la figura es la existencia de procesos llamados de *vía única*: es fácil ir en una dirección, pero muy difícil en la otra si no se posee la información adecuada; es fácil escribir con la tinta A_p, pero imposible leer (si no se tiene la tinta A_s). Y aquí es donde entra en gloria y majestad la teoría de números: hoy en día los principales procesos de vía única usados en sistemas criptográficos de clave pública provienen de esta disciplina. He aquí uno de los más populares:

> Es fácil multiplicar dos números primos.
> Es dificilísimo encontrar los factores de un número.
> (Por ejemplo intentar –sin calculadora– encontrar los factores de 1457 o de 9271. Con calculadora intentar con éste: 1042387).

En este problema se fundamenta el sistema de cifrado público conocido con la sigla RSA que veremos en detalle a continuación.

4 El sistema criptográfico RSA y la teoría de números

Antes de entrar en materia, es bueno insistir que el problema de enviar textos cifrados se reduce al problema de enviar números (basta hacer una pre-codificación de letras a números usando cualquier sistema como el ejemplo que dimos al principio). Luego, el asunto se trata de codificar números con ... otros números.

Hace casi veinte años[3] que apareció uno de los más populares sistemas de criptografía de clave pública existente hoy día. El sistema RSA funciona de la siguiente forma: cada usuario tiene que

1. Elegir dos números primos p y q.
2. Calcular $n = pq$.
3. Calcular $f = (p-1)(q-1)$.
4. Elegir aleatoriamente un número e entre 1 y f, que no tenga factores comunes con f.
5. Calcular el número $(e^{-1} \bmod f)$, es decir, encontrar d tal que $(ed \bmod f) = 1$. ($a \bmod b$ se lee "a módulo b")[4]

Finalmente publica el par (n, e) como su clave pública y guarda d como la parte secreta de su clave. Por ejemplo si $p = 3$ y $q = 5$, entonces $n = 15$ y $f = 8$. Podemos elegir por ejemplo $e = 5$ y entonces $d = 5$ (pues $ed = 25 = 3 \times 8 + 1$, es decir $(25 \bmod 8) = 1$). Luego publicamos $(15, 5)$ y guardamos 5 como la parte secreta de nuestra clave. Es importante hacer notar que los pasos (1) hasta (5) son "fáciles" de calcular, computacionalmente hablando, es decir, hay algoritmos *eficientes* para hacerlo (ver detalles sobre esto en el Apéndice 5.1).

Veamos ahora cómo proceder si uno quiere mandarle un texto[5] m a Amanda. Primero, se consigue las páginas amarillas de claves públicas, busca la clave

[3] En 1978 fué publicado el artículo "A method for obtaining digital signatures and public-key cryptosystem" por Rivest, Shamir y Adleman, de ahí la sigla RSA. La idea de criptografía de clave pública no es mucho más antigua: data de 1976 y se le acredita a D. Diffie y M. Hellman.

[4] Sumar, restar, multiplicar, exponenciar *módulo n* significa hacer la correspondiente operación como siempre, luego dividir el resultado final por n y quedarse con el resto. Esta última operación se denota por $(a \bmod n)$. Formalmente $(a \bmod n)$ es el número $0 < r < n$ tal que $a = nq + r$ para cierto entero q. Por ejemplo, $(35 \bmod 6) = 5$, $(16 \bmod 4) = 0$, $(-12 \bmod 5) = 3$ (no -2)..

Nótese la similitud y la diferencia con la notación matemática usual $a \equiv b \pmod n$, que señala que a y b son *equivalentes módulo n*, es decir, $(a - b)$ es divisible por n. Sucintamente, digamos que $(a \bmod n)$ es un número, mientras que $a \equiv b \pmod n$ expresa una *relación* entre los números a y b.

[5] Recordemos que el texto es un número. Más aún, es conveniente que este número sea menor que n: si es más grande, lo descomponemos en pequeños trozos y enviamos estos trozos por partes. Por ejemplo si $n = 10000$, descomponemos 930982341982939849 en 9309 8234 1982 9398 49.

pública de ella, que supongamos es (n, e), y luego calcula la codificación (que denotaremos M) del texto m según esa clave de acuerdo a la fórmula

$$M = (m^e \bmod n). \tag{1}$$

Finalmente envía el número M, que es el texto cifrado.

Ahora bien, una vez recibido, ¿cómo Amanda decodifica el texto cifrado M? Muy sencillo, simplemente calculando

$$(M^d \bmod n). \tag{2}$$

No es difícil demostrar que se obtiene de vuelta m, es decir el texto original que se envió (ver Apéndice 5.2 para los detalles).

Un ejemplo (un poco más real) para ilustrar las ideas: supongamos que Amanda usa $p = 101$, $q = 113$. Entonces $n = 11413$, y $f = 100 \times 112 = 11200$. Además supongamos que Amanda elige $e = 3533$ (nótese que 3533 no tiene factores comunes con $f = 11200$). Haciendo unos pequeños cálculos, obtenemos que $d = (6597 \bmod 11200)$. Entonces Amanda publica $(11413, 3533)$ como su clave pública. Supongamos ahora que queremos mandarle el texto 9726. Entonces tenemos que calcular

$$9726^{3533} \bmod 11413$$

lo cual da 5761, que es el texto cifrado que enviamos. Cuando Amanda recibe el texto cifrado, calcula

$$5761^{6597} \bmod 11413$$

que es 9726, es decir, el texto original que se envió.

El sistema RSA está pensado de tal forma que si los números primos p, q son suficientemente grandes (alrededor de 100 dígitos cada uno es suficientemente seguro para la tecnología computacional de hoy en día) y bien eligidos[6] es virtualmente imposible que nadie más que Amanda (que se quedó con el número secreto d) pueda decodificar el texto, es decir obtener el número original.

¿Por qué? Porque la única forma posible (actualmente conocida) de *quebrar* el código es encontrar formas eficientes de factorizar números. Observemos que si logramos factorizar $n = pq$, entonces podemos calcular $f = (p-1)(q-1)$ y luego encontrar la parte secreta de la clave $d = (e^{-1} \bmod n)$ usando el algoritmo de división común (ver el paso (5) en Apéndice 5.1). Pero, hasta hoy día nadie sabe cómo hacer esto eficientemente. El lector puede "sentir" esta dificultad intentando factorizar los números que dejamos de tarea ...

[6]Hay ciertas condiciones –que uno debiera evitar– que hacen más fácil el trabajo de quien quiere factorizar n, por ejemplo que p, q estén muy cercanos el uno del otro, que tengan alguna forma conocida, etc.

5 Apéndice

5.1 RSA es factible

Decíamos que el proceso de calcular la clave pública (el cálculo de los pasos (1) a (5)) debe ser *eficiente* desde el punto de vista computacional, es decir, debemos ser capaces de calcularla en tiempo "razonable".

Es bueno señalar que "eficiente" tiene un significado matemático muy preciso a cuyo estudio se dedica la disciplina llamada *teoría de complejidad computacional.* Aplicando estas ideas a la Teoría de Números surgió la *Teoría de Números Computacional*, disciplina que se ha desarrollado vertiginosamente en los últimos años y que se ocupa esencialmente de la eficiencia de los algoritmos en esta área: algoritmos de división, algoritmos de factorización, tests para determinar si un número es primo, etc.

Pero vamos a lo concreto. Es evidente que los pasos (2), (3) y (4) son simples de calcular. Para el paso (1), si uno quiere elegir primos grandes (y ésa es la idea), lo mejor es buscar un número entero grande en forma aleatoria, y luego verificar si es primo. Hoy en día hay tests eficientes para verificar si un número es primo.

Finalmente, para el paso (5), el cálculo de d es "barato": como e y f no tienen factores comunes, se sabe que, usando el algoritmo de división común, uno puede encontrar números u, v tales que $ue + vf = 1$. Entonces basta tomar $d = (u \bmod f)$ y tendremos que $(ed \bmod f) = 1$, es decir d es el inverso de e módulo f. Se puede demostrar detalladamente que todo esto se puede hacer eficientemente.

5.2 RSA es correcto

Debemos demostrar que si uno decodifica el mensaje M usando la ecuación (2), entonces obtiene de vuelta el mensaje original.

El pequeño Teorema de Fermat dice que que si p es primo, entonces para todo número a con $(a, p) = 1$

$$a^{p-1} \equiv 1 \pmod{p}.$$

(Aquí usamos la notación $a \equiv b \pmod{n}$ usual para seguir la notación de los libros de teoría de números. Ver nota 3). Recordemos que $ed \equiv 1 \pmod{f}$, luego por definición $ed = kf + 1$ para cierto número k. Es decir $ed = kf + 1 =$

$k(p-1)(q-1)+1$. Ahora calculemos M^d:

$$M^d = (m^e)^d = m^{ed} = m^{kf+1} = (m^{p-1})^{k(q-1)}m.$$

Las siguientes dos congruencias valen para todo m.

$$(m^{p-1})^{k(q-1)}m \equiv m \pmod p \tag{3}$$

$$(m^{q-1})^{k(p-1)}m \equiv m \pmod q \tag{4}$$

Veamos (3): si $(m,p) \neq 1$, entonces como p es primo, p divide a m y luego ambos lados son cero. Si $(m,p)=1$ se deduce fácilmente del pequeño teorema de Fermat. El argumento es similar para la congruencia (4).

Luego usando estas congruencias y el hecho que $n = pq$ y $0 \leq m < n$, tenemos $M^d \equiv m \pmod n$, es decir $(M^d \bmod n) = m$.

Referencias

[1] M. Kiwi, *Curso Criptografía, Facultad de Ciencias Físicas y Matemáticas, Universidad de Chile.*
`http://www.dim.uchile.cl/~mkiwi/cripto_98.html`
Apuntes de clase y materiales.

[2] *A course in Number Theory and Cryptography, Second Edition* por Neal Koblitz. Springer Verlag, 1994. (Excelente introduccion al tema. Hay muchas indicaciones para diferentes líneas de trabajo actuales, ejemplos y bibliografía).

[3] *Cryptography, Theory and Practice* por D. R. Stinson. Un libro general muy completo de teoría de criptografía.

[4] D. Kahn, *The Code-Breakers, Revised Ed.*, Scribner, 1996. Un libro muy completo sobre la historia de la criptografía hasta los tiempos actuales.

Dirección del Autor: Depto. de Ciencias de la Computación, Universidad de Chile, Blanco Encalada 2120, Santiago, Chile. E-mail: `cgutierr@dcc.uchile.cl`