Primeras impresiones sobre la solicitud de la patente “Procedimiento de doble criptograma simétrico de seguridad de Shannon por codificación de información para transmisión telemática y electrónica”

Jorge Ramió Aguirre - Alfonso Muñoz Muñoz
Miembros de la Red Temática Criptored

Madrid, 21 de enero de 2013

1. Introducción

Referencias periodísticas (entre otras):
http://tecnologia.elpais.com/tecnologia/2013/01/02/actualidad/1357151945_888633.html
http://www.elmundo.es/elmundo/2013/01/17/valencia/1358439825.html

Documento de la solicitud de patente:
http://worldwide.espacenet.com/publicationDetails/originalDocument?CC=WO&NR=2012152956A1&KC=A1&FT=D&ND=3&date=20121115&DB=EPODOC&locale=es_LP

El presente escrito pretende dar luz sobre diferentes cuestiones aparecidas en la prensa nacional (España) el 3 y 17 de enero de 2013, relativas a una patente de un nuevo algoritmo criptográfico "indescifrable". Si este breve escrito sirve de guía para evaluar mejor en un futuro este tipo de sistemas, o servir de referencia a periodistas que han dado una publicidad inusual a este hecho, los firmantes del mismo se dan por satisfechos.

Este informe viene a petición de diferentes miembros de la red en cuanto a publicar una opinión, personal que no colectiva de la red, a este respecto. En los últimos días, diferentes blogs y comunidades de internautas relacionadas con la seguridad de la información se han expresado y se están expresando también en este sentido, entre ellos:

Blog de Arturo Quirantes
http://naukas.com/2013/01/20/el-sistema-perfecto-de-criptografia-o-al-menos-eso-dice-su-inventor/

Kriptópolis
http://www.kriptopolis.com/polemica-cifrado-definitivo

Nota:

Ninguno de los miembros ni los editores de la Red Temática de Criptografía y Seguridad de la Información Criptored, con base en la Universidad Politécnica de Madrid, suele realizar comentarios sobre la opinión de terceros en cuanto a la ciencia de la criptología, centrándose su trabajo fundamentalmente en la difusión, y en su caso evaluación y estudio de trabajos publicados en conferencias, congresos y seminarios científicos, que siguen por tanto un riguroso proceso de evaluación por terceras partes. No obstante, vemos importante hacer una excepción en este caso -dada la notoriedad social que esta noticia ha alcanzado- para aclarar diferentes cuestiones.

2. Comentarios sobre la solicitud de patente en cuestión

A continuación se adjunta una serie de comentarios basados en el estudio de la solicitud de patente a la que puede acceder desde la dirección ya indicada.

En primer lugar, sin entrar a valorar aún las operaciones que en dicha solicitud indica realiza el algoritmo propuesto, hay que decir que si todo se limita a operaciones matemáticas lineales -como así parece ser- (ver el estudio del algoritmo en párrafos posteriores) el sistema tiene muchas posibilidades de ser inseguro, puesto que una de las máximas en criptografía simétrica moderna para que los algoritmos sean robustos es la no linealidad de algunas de sus operaciones.

La redacción de la patente no ayuda mucho en dar credibilidad al algoritmo. Sin ser exhaustivos, comentar son un par de cuestiones:

Aunque sea muy común en Latinoamérica usar las palabras 'encriptar' y 'desencriptar' por influencia directa del inglés, lo cierto es que en lengua española esas palabras no existen y lo correcto, más aún en un documento oficial, sería utilizar cifrar y descifrar.

Toda la redacción está plagada de afirmaciones sin fundamento, errores de concepto y diversos errores básicos de lenguajes de programación.

Por ejemplo, en la página 25 (línea 25) se afirma:

"*un ordenador actual difícilmente puede operar con exactitud con números enteros de más de 15 dígitos*".

Cualquier lenguaje de programación moderno permite implementar programas que manejen números de tamaño arbitrario en ordenadores normales. Por ejemplo, en lenguaje java con la clase BigInteger.

En cualquier caso, y sólo a modo anecdótico, incluimos a continuación unas cuantas aclaraciones que el lector puede observar en el documento de solicitud de patente en cuestión:

1. Sección Descripción, página 1, línea 25:
   Afirma:
   "*... y operaciones bancarias, criptografía de obras musicales y audiovisuales, y firmas*"

   Aclaración:
   No existe la 'criptografía de obras musicales y audiovisuales'; sí algoritmos criptográficos que permiten proteger obras musicales y audiovisuales, así como diversos métodos basados en la criptografía, marcas de agua, etc.

2. Sección Descripción, página 2, líneas 13 y 14:
   Afirma:
   "*Actualmente hay diversos lenguajes de cifrado*"

   Aclaración:

No existen 'lenguajes de cifrado'; lo que existe son algoritmos de cifra que se implementan con uno u otro lenguaje de programación, en el caso de software criptográfico como lo es el invento que se propone.

3. Sección Descripción, página 2, líneas 15 y 16:
   Afirma:
   "*... o bien como sistema de cifrado asimétrico. Este último es muy seguro*"

   Aclaración:
   Si el algoritmo que se presenta es de tipo simétrico, esta afirmación no es adecuada en este entorno pues son igual de seguros (o inseguros) los sistemas de cifra simétricos que asimétricos. Hablar solamente de los asimétricos como 'muy seguros' supone excluir a los primeros, lo cual es un error.

4. Sección Descripción, página 2, líneas 16 y 17:
   Afirma:
   "*... pero no por eso resulta imposible romperlo por el procedimiento llamado fuerza bruta*"

   Aclaración:
   Los sistemas de cifra asimétricos no se rompen por fuerza bruta, aunque podría intentarse -si se desea- pero no tiene sentido con la capacidad de cómputo actual. Existen, no obstante, algoritmos específicos para romper su fortaleza, que está basada generalmente por un problema matemático tipo NP (exponencial) de costosa solución computacional cuando los números son grandes, e.g. sobre mil bits, llámese éste problema del logaritmo discreto, problema de la factorización entera, etc. Los sistemas de cifra donde es más común usar fuerza bruta -aunque existan otros procedimientos más elegantes- son los simétricos, nunca los asimétricos.

   En el mundo real los ataques por fuerza bruta son la última solución, porque son los más costosos. De hecho, en criptografía hay un dicho cuando estamos en el mundo real: "*la criptografía no se ataca, se esquiva*".

   En el mundo real es más fácil obtener una clave almacenada indebidamente, procesada incorrectamente, usar fallos de programación de los algoritmos criptográficos, de los sistemas que lo implementan, etc., que atacar al propio algoritmo de cifrado.

5. Sección Descripción, página 2, líneas 18 y 19:
   Afirma:
   "*... acaban averiguando los números primos a partir de cuyo producto se ha construido el criptograma*"

   Aclaración:
   El autor supone que como sistema asimétrico sólo existe el algoritmo RSA, que efectivamente utiliza dos primos grandes cuyo producto es el módulo o cuerpo de cifra, pero existen otros algoritmos para el intercambio de clave y/o firma digital que no usan dos o más primos sino solamente uno como es el caso de Diffie y Hellman, Elgamal, DSA (en este caso dos primos pero en sentido diferente), además de curvas elípticas, etc. Así mismo no es apropiado afirmar que "a partir de cuyo producto se ha construido el criptograma": el criptograma se construye mediante una operación

matemática (generalmente exponenciación) donde interviene ese producto, el módulo o cuerpo de cifra, pero nada más.

6. Sección Descripción, página 2, líneas 19 y 20:
   Afirma:
   "*Y no es nada fácil descubrir y almacenar números primos suficientemente grandes*"

   Aclaración:
   En tanto no especifique el autor qué entiende por 'números primos suficientemente grandes', lo cierto es que existen muchas librerías que permiten encontrar números primos grandes con total facilidad y rapidez para las necesidades actuales. Además no se entiende qué quiere decir con 'almacenar números primos'.

7. Sección Descripción, página 2, líneas 21 y 22:
   Afirma:
   "*... uno de los lenguajes más solventes continua siendo el llamado algoritmo DES*"

   Aclaración:
   Insistiendo en que en este caso el DES jamás ha sido patentado como un lenguaje, decir que continúa (el acento lo ponemos nosotros) siendo el más solvente significa remontarnos al estado del arte de la criptografía simétrica antes del año 1997. Después del DES, existen muchos algoritmos más robustos tanto en diseño como en longitud de clave, e.g. IDEA, AES. Si se desea hacer una afirmación en ese sentido en pleno siglo XXI, la elección obviamente es AES.

8. Sección Descripción, página 2, líneas 24 y 25:
   Afirma:
   "*... DES consta de 2^{56} claves, lo que significa que si se dispone de un ordenador capaz de realizar un millón de operaciones por segundo, se tardarían más de 2.200 años en probar todas las claves*"

   Aclaración:
   Aunque los cálculos son correctos, es necesario recalcar que estos ataques a sistemas simétricos permiten un divide y vencerás en red, por lo que hablar en un documento científico como éste de un solo ordenador con capacidad de cálculo de un millón de claves por segundo, resulta como poco inapropiado. Ya en el DES Challenge III cuando se rompió la clave del DES en 1999 en menos de un día, la capacidad de cómputo en red y con la máquina DES Cracker alcanzó los 250 mil millones de claves por segundo, y estamos hablando de 13 años atrás.

9. Sección Descripción, página 5, líneas 10, 11 y 12:
   Afirma:
   "*Además la conjunción de muchos ordenadores muy potentes puede romper claves asimétricas, consideradas largas, en poco tiempo*"

   Aclaración:
   Se están mezclando conceptos de ataques a criptografía simétrica con la criptografía asimétrica. El trabajo de 'la conjunción de muchos ordenadores muy potentes' se entiende como un trabajo en red, una acción en paralelo. Si el autor conoce soluciones de ataques en red al problema de la factorización entera o del logaritmo discreto, debería nombrarlos. Por lo tanto, no se sabe qué ha querido afirmar aquí el autor, y menos aún dilucidar qué entiende por 'poco tiempo'.

10. Sección Descripción, página 5, líneas 31 y 32 y página 6, líneas 1 y 2:
    Afirma:
    "*Pues bien, todos los cifradores por bloques trabajan con cadenas fijas de n bits a las que se aplican alternativamente procesos de sustitución en las llamadas S-cajas (S-boxes) y de permutación en las P-cajas (P-boxes)*"

    Aclaración:
    Esto no es cierto. No se puede generalizar las operaciones que se realizan en el algoritmo DES a todos los demás algoritmos simétricos; es más, no tienen nada que ver, e.g. IDEA no usa cajas y en AES no se llaman S-boxes.

11. Sección Descripción, página 7, línea 7:
    Afirma:
    "*... mediante la operación or exclusivo byte a byte*"

    Aclaración:
    Aunque está sacado de la referencia del libro de D. Manuel José Lucena, que cita, tal vez merecería la pena no dejar la sensación de que sólo existe ese tipo de cifra en flujo. Si bien es cierto que algunos algoritmos de flujo cifran byte a byte, otros algoritmos lo hacen bit a bit. De hecho, el principio básico de la cifra de flujo (Solomon Golomb) es la cifra bit a bit.

12. Sección Descripción, página 10, línea 1:
    Afirma:
    "*... cuando conoce las claves que han sido utilizadas para codificarlo*"

    Aclaración:
    Debería haber terminado la frase con la palabra 'cifrarlo' y no 'codificarlo'. Codificar es algo estático y siempre entrega el mismo valor (código ASCII, código Baudot, código Morse, etc.), en cambio cifrar es algo dinámico; en función de una u otra clave utilizada en la cifra, se obtiene uno u otro criptograma.

    Por esta misma razón no es adecuado usar en el título de esta solicitud de patente la frase “por codificación de la información” cuando en realidad lo que se quiere decir es “por cifrado de la información”.

En esa misma página 10 y en las línea 2 y 3, vuelve a insistir en que el único sistema de ataque a 'todos' los sistemas de cifra es la fuerza bruta.

No merece la pena profundizar en sus apreciaciones sobre lo que denomina “criptogramas seguros de Shannon”, algo inexistente, si bien existe el concepto de cifrado con secreto perfecto introducido por Shannon, pero son cosas distintas.

No haremos valoraciones sobre afirmaciones tales como que el sistema es inexpugnable 'incluso' (página 2, línea 33) para los ordenadores del futuro y otras afirmaciones que aparecen en el documento como usar la palabra 'pirata' (página 9, línea 33) como la persona que intercepta una comunicación porque sería entrar en el plano personal.

En resumidas cuentas, que no tiene interés seguir con esto. Creemos que queda claro lo que pretendemos decir; continuemos con la descripción del algoritmo para entender mejor las conclusiones de esta nota y las recomendaciones.

3. Comentarios sobre el algoritmo presentado

Si hemos entendido bien el algoritmo, lo cual no es sencillo a partir de la redacción, éste podría resumirse en lo siguiente:

1. Un residuo es un número del 1 al 9. Es posible combinarlo en grupos: 1 al 9, 11 al 99, 111 al 999, etc.

2. Tenemos una matriz alfanumérica (básicamente representamos los caracteres ASCII en matriz).

3. Tenemos una matriz base de residuos numéricos (representamos todos los residuos posibles en una matriz). Por ejemplo, si estamos trabajando con los residuos desde el 111 al 999 pues los ordenamos en una matriz.

4. Se necesita una clave de equivalencia Es una clave formada por números naturales. Se agrupan en parejas e indican qué fila "conmutar" por otra "fila".

5. Matriz base de residuos + clave equivalencia = matriz base de residuos ordenadas.

   Conmutamos la matriz en función de lo que nos indica la clave, dando lugar a una nueva matriz.

6. Construimos una tabla de equivalencias.

   a. Extraemos los valores de la matriz alfanumérica y los hacemos corresponder con uno o más valores de la matriz base de residuos ordenadas.
   b. Única condición que no se repitan los valores,

7. Construir plantilla (criptograma reducido)
   a. Ponemos el mensaje a cifrar en ASCII.
   b. Seleccionamos uno de los posibles valores de la tabla de equivalencia para cada carácter ASCII a cifrar.

8. Necesitamos una clave de protocolo. Una secuencia de números naturales, cada número indica el número de dígitos en los que se convertirá cada "carácter ASCII cifrado (residuo)".

9. Define un algoritmo “simple” para asignar valores a los dígitos indicados. No queda claro cómo es la generación de los números aleatorios indicados.

Y... eso es todo.

Emisor y receptor necesitan compartir:

1. Clave de protocolo.
2. Clave de equivalencias.
3. La matriz base de residuos (cómo ordenar los residuos en la matriz).

En el ejemplo que pone para cifrar "la reunión es mañana jueves" (23 dígitos) el resultado es de 462 dígitos (20 veces más).

4. Opinión y conclusiones

1. Primer inconveniente del algoritmo. El autor afirma que una de las bondades de este algoritmo es precisamente la "expansión" de los dígitos a cifrar, múltiples opciones entre las que elegir. Esto tiene un problema claro: el texto cifrado es mucho más grande que el texto en claro. Únicamente por esto se puede afirmar que este algoritmo no será de utilidad en la práctica. Un algoritmo de cifrado moderno no sólo debe ser seguro, sino que debe ser eficiente en múltiples aspectos. Los algoritmos modernos producen, típicamente, texto cifrados del mismo tamaño que el texto en claro (dejaremos aparte cuestiones de relleno). Un ejemplo es el actual estándar mundial AES, el cual tiene ciertas propiedades interesantes para su ejecución en terminales de bajo recursos, almacenamiento, etc.

2. Emisor y receptor necesitan compartir 3 claves. Es necesario compartir por algún procedimiento 3 claves (o una que las componga) cuyo tamaño total es mayor que las claves criptográficas actuales para proporcionar una supuesta seguridad no probada.

   El proceso en el que interviene la clave es crítico en cualquier algoritmo simétrico moderno. La patente no deja claro cómo se debería elegir la clave con unas mínimas consideraciones de seguridad, es más, en el texto recomienda que incluso emisor y receptor podrían repetir la clave una y otra vez en el proceso, recordando a algoritmos clásicos rotos hace siglos. Cómo interviene la clave en el cifrado o cómo se selecciona no es baladí. Muchos de los algoritmos simétricos utilizan generadores de subclaves basados en una clave principal que van proporcionando bits de clave según una serie de condiciones muy estudiadas. La seguridad de estos generadores es crítica para la seguridad global del algoritmo.

3. La descripción del algoritmo es oscuro y "variable". Los algoritmos "serios" describen una serie de pasos a realizar de manera impermutable. La descripción de este algoritmo deja abierta la selección de multitud de cuestiones al emisor y receptor; esto puede introducir problemas.

4. Cualquier algoritmo criptográfico que se precie debe resistir un ataque con texto en claro conocido; es algo básico y no lo vemos reflejado en el documento. Sin estas pruebas elementales, ningún algoritmo puede darse por bueno.

5. El algoritmo tiene pinta de mezcla de procedimientos básicos de criptografía clásica. En ocasiones, recuerda a cifradores por homófonos, en otros casos a cifradores basados en matrices y transposiciones de sus elementos. En cualquier caso, no queda claro que los procedimientos establezcan procesos no lineales en los que basar una seguridad real.

   Hablando de sistemas de cifra lineales con espacios de claves muy grandes (que parece ser el fuerte de este sistema), más grandes incluso que los actuales cifradores modernos que son no lineales, hay que ser muy cautos; e.g. si en el algoritmo de matrices de Hill se usa una simple matriz de 8x8 y un cuerpo de cifra de todos los caracteres ASCII imprimibles, se obtiene un espacio de claves válidas exorbitante ...

pero el sistema se rompe con una simple matriz de Gauss-Jordan y tres o cuatro palabras del texto en claro y su correspondiente texto cifrado.

6. Si se desea probar la seguridad del sistema es bien sencillo, hay dos opciones no excluyentes. La primera es publicar sus resultados en cualquiera de los congresos o conferencias científicas de renombre en criptografía, tanto en España como en otros países, y someter sus resultados a investigaciones independientes. Una segunda, y aconsejable, liberar una herramienta concreta que implemente su algoritmo y ser sometido al escrutinio de la comunidad científica e internautas. Si se incentiva a la comunidad de alguna manera para atacar su sistema, tanto mejor.

Ante todo lo anterior no sólo podemos afirmar que este algoritmo ni mucho menos va a desplazar a los algoritmos actuales, sino que no le auguramos ningún futuro.

Sería conveniente, especialmente para los profesionales del periodismo que difunden este tipo de noticias en grandes medios, recordar que en España disponemos de excelentes profesionales, tanto expertos en seguridad informática como criptólogos; a la mayoría de ellos se les podría consultar antes de publicar noticias sensacionalista como las que nos ocupa.

Atentamente,

Dr. Jorge Ramió
Profesor Titular de la Universidad Politécnica de Madrid. Desde el año 1994 imparte diversas asignaturas relacionadas con la seguridad y criptografía. Entre sus facetas destaca su interés por la difusión de la seguridad informática en Iberoamérica. Autor del libro electrónico de Seguridad Informática y Criptografía en 2006, de libre distribución en Internet y con más de 120.000 descargas, creador en 1999 de la Red Temática de Criptografía y Seguridad de la Información Criptored, organización de los congresos iberoamericanos CIBSI y TIBETS, creador y director de la cátedra UPM Applus+ de seguridad, creación y dirección de la Enciclopedia de la Seguridad de la Información Intypedia y del primer MOOC en español sobre seguridad de la información Crypt4you, participación como profesor invitado en cursos de posgrado en España y países de Latinoamérica.

Dr. Alfonso Muñoz
Doctor de Telecomunicaciones por la Universidad Politécnica de Madrid e investigador postdoctoral en Computer Security & Advanced Switching Network en la Universidad Carlos III de Madrid. Especialista en protección de datos digitales y diseño de sistemas seguros (criptografía, esteganografía, dpi, etc.). Ha publicado más de 20 artículos en revistas y congresos científicos de alto impacto (revisión ciega por pares) en el campo de la seguridad informática, y ha trabajado en proyectos con organismos europeos, ministerios y multinacionales. Su trabajo de investigador lo combina con su faceta de divulgación. Algún ejemplo destacable es la creación y dirección técnica de Intypedia, creación del primer MOOC en español sobre seguridad de la información Crypt4you, artículos de divulgación en revistas o blogs del sector o la participación en conferencias de seguridad informática y hacking.