SEGURIDAD EN CERTIFICADOS DIGITALES BASADOS EN RSA, PRESENTE Y FUTURO

Nury A. Palomino Martínez
Universidad Piloto de Colombia
Especialización Seguridad informática
Bogotá D.C., Colombia

fanalp25@gmail.com

Abstract. Since august of 1977, when three students of the Massachusetts Institute of Technology presented a new cryptographic scheme named RSA, private and public organizations, as well as scientific communities and individuals have been dedicating their efforts to break the mathematical theories used to sustain its discover with no other achievement than shake and knock down some of its implementations. However, upcoming technologies and scientific discoveries, brings a ray of light for the materialization of theories otherwise utopic due to their complexity and algorithmic cost.

Resumen. Desde agosto de 1977, cuando tres estudiantes del instituto de tecnología de Massachusetts presento un nuevo esquema criptográfico denominado RSA, organismos públicos y privados, al igual que comunidades científicas e individuos han dedicado sus esfuerzos para romper las teorías matemáticas usadas para sustentar este descubrimiento, sin ningún logro significativo más que tumbar y hacer temblar algunas de sus implementaciones, sin embargo las tecnologías emergentes y los descubrimientos científicos, brindan un rayo de esperanza para la materialización de teorías que de otra manera serian utópicas debido a la complejidad y costo del algorítmico.

Palabras claves. Algoritmo, Factorización, criptosistema.

1. INTRODUCCIÓN

Un año antes de que Ronald Rivest, Adi Shamir y Leonard Adleman publicaran su esquema criptográfico, Whitfield Diffie y Martin Hellman mostraban al público su protocolo criptográfico para el establecimiento de claves entre dos o más partes que no han tenido previo contacto [1], sentando las bases para la creación de protocolos muy seguros, incluyendo RSA.

En 1977 se publicaron documentos en los que se demostraba que en 1973, los criptográficos del grupo de seguridad de comunicaciones Electrónicas (CESG) del gobierno británico, ya disponían de conocimientos sobre este tipo de criptografía. [2]

En cuanto a los algoritmos basados en Diffie-Hellman, la complejidad básica radica en la gran dificultad que supone calcular logaritmos discretos en un cuerpo finito, sin embargo cabe anotar que esta dificultad, matemáticamente hablando es una simple conjetura, pues hasta la fecha no ha sido demostrada.

Como se mencionaba anteriormente, RSA es uno de los varios esquemas criptográficos que se basan en el protocolo Diffie-Hellman para el intercambio de claves, agregando complejidad mediante el establecimiento de dichas claves por intermedio de números primos de gran tamaño, de la forma ilustrada en la Figura. 1. La factorización de números de gran tamaño supone una gran dificultad y ha sido objeto de estudios que se han

fortalecido desde la aparición de este criptosistema, sin que a la fecha, computacionalmente hablando y tomando como base la tecnología existente, se haya logrado resolver en un tiempo aceptable. Sin embargo, la computación cuántica, tema tecnológico y científico de actualidad brinda luces para la implementación del algoritmo de Shor, que permite descomponer un número en factores primos en tiempo polinómico y que fuera demostrado en 2001 por un grupo de científicos de IBM quienes lograron descomponer 15 en sus factores primos 3 y 5 usando una computadora de 7 qbits [3].

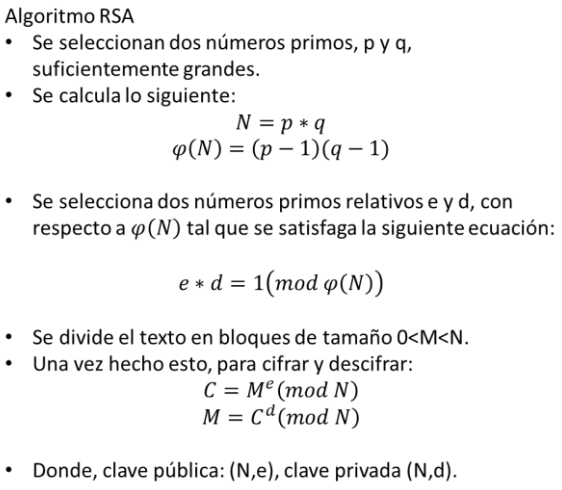

Figura. 1 - Algoritmo RSA. Fuente: Articulo- Improvement over Public Key Cryptographic Algorithm

Para lograr entender el impacto que puede tener la llegada de la computación cuántica frente a los esquemas de seguridad basados en RSA y en el criptoanálisis en general, es necesario dar una mirada con mayor profundidad a las bases del criptosistema RSA y a algunos de los tipos de ataque que ha recibido durante estas décadas.

2. TIPOS DE ATAQUES AL ALGORITMO RSA.

A continuación se explican algunos de los mas importantes tipos de ataques a RSA, según la clasificación hecha por Dan Boneh en su artículo "Twenty years of attacks on the RSA cryptosystem" [4].

2.1 ATAQUES ELEMENTALES QUE APROVECHA EL EVIDENTE MAL USO DEL CRIPTOSISTEMA.

Durante los 35 años que tiene de existencia el algoritmo RSA, la cantidad de ataques que ha recibido es incontable, sin embargo, de acuerdo con el doctor Dan Boneh [4], estos no ha sido catastróficos para el criptosistema, pues hasta el momento solo se ha demostrado que se implementa de manera incorrecta.

Como se menciona al principio de este artículo, el algoritmo RSA, basado en Deffie-Hellman, supone un alto nivel de seguridad dados los problemas que representan el calculo de logaritmos discretos y la factorización de enteros de gran tamaño, pero desafortunadamente se tiende a pensar que la aplicación plana del algoritmo es suficiente para dar seguridad a una comunicación. Sin embargo, el uso del algoritmo RSA simple no provee algo conocido como "Seguridad Semántica" que consiste en la certeza de que ninguna información del mensaje original a través del mensaje encriptado, utilizando, por ejemplo, criptoanálisis por frecuencia. Para evitar este tipo de ataques, normalmente se utilizan métodos de cifrado aleatorios y otras técnicas que han sido descritas por variados autores expertos en el tema [5]. Además de esto, se han emitido una serie de estándares conocidos por sus siglas PKCS (Public Key Cryptography Standards), cuya implementación reduce el riesgo de ocurrencia de los problemas antes descritos.

Otra de los ataques que hacen uso de malas implementaciones del algoritmo podría catalogarse como un ataque de fuerza bruta, pero a pesar de ello es uno de los tipos de ataque mas estudiado durante las últimas décadas: La factorización de números enteros muy grandes. Los métodos más rápidos conocidos en la actualidad son "General number Field Sieve" que demuestra se el mas eficiente para números de más de 100 dígitos y "Quadratic Sieve" [6] que se considera el más eficiente para números de menos de 100 dígitos y que se basan en la búsqueda de congruencias para encontrar los factores. En el caso descrito

anteriormente, uno de los factores que puede facilitar un ataque es la elección de dos números primos muy cercanos para generar N. Por ejemplo [7]:

Si se escogen dos números p y q tal que $p \approx q$, y suponiendo que $p > q$, entonces:

- $\frac{(p-q)}{2}$ es un entero muy pequeño.
- $\frac{(p+q)}{2}$, será un entero ligeramente superior a $\sqrt{N}$
- Además, se cumplirá que:

$$N = \frac{(p+q)^2}{4} - \frac{(p-q)^2}{4}$$

- Nótese aquí la similitud con la ecuación básica de la diferencia de cuadrados $N = x^2 - y^2$, donde:

$$x = \frac{p+q}{2}, \qquad y = \frac{p-q}{2}$$

- En este momento se escogen valores de x tal que $x < \sqrt{N}$, hasta que encontremos un valor que convierta a $x^2 - N^2$ en cuadrado perfecto.
- Una vez hecho esto, obtendremos los valores de p y q donde:

$$p = (x+y), q = (x-y)$$

En la introducción de este articulo se anota que los avances tecnológicos han traído luz a nuevas implementaciones de algoritmos que reducen exponencialmente los tiempos para encontrar los factores de enteros relativamente grandes y se menciona el algoritmo de Shor, que a la fecha ha sido probado e implementado en pequeña escala.

Continuando con los ejemplos de malas implementaciones del criptosistema, vale la pena mencionar aquellas en donde se comparte el módulo N, calculado a través de los números p y q, utilizándolo para muchos usuarios, cambiando simplemente los valores de las claves públicas y privadas (e y d). Este tipo de prácticas resultan peligrosas un atacante podría recuperar el texto en claro basado tan solo en las llaves públicas, basados en el teorema de los primos relativos, como se muestra en la Figura. 2.

2.2 ATAQUES POR EXPONENTE PRIVADO MUY PEQUEÑO.

Cuando se eligen exponentes privados d inferiores a aproximadamente un cuarto del tamaño de N, se posibilita realizar un ataque para recuperar dichas claves o exponentes privados, basados en el fundamento matemático de que un número racional puede ser representado en fracciones continuas finitas.

El uso de exponentes privados pequeños es una práctica común en las comunicaciones que implementan el criptosistema RSA cuando se quiere disminuir el costo del descifrado o de la firma de los mensajes. Expertos como los doctores D. Boneh y G. Durfee [8], demuestran que el exponente privado debe tener una longitud equivalente a un cuarto del tamaño de N, es decir, si N tiene un tamaño de 1024 bits, d debe tener un tamaño al menos de 256 bits para poder evitar este tipo de ataques.

Sea M el texto en claro, e1 y e2 las claves públicas y N el cuerpo o módulo. Se tiene que para cada exponente los textos cifrados están dados por:

$$C_1 = M^{e_1} mod\ N$$
$$C_2 = M^{e_2} mod\ N$$

El atacante puede acceder sin problema a C_1, C_1, C_1, C_1 y a N, y basado en el teorema de los primos relativos fácilmente puede deducir el mensaje en texto claro despejando la siguiente ecuación:

$$(C_1{}^{-1})^{-r} * C_2{}^{s} = M\ mod\ N$$

Donde r y s son dos enteros tales que se satisfaga

$$re_1 + se_2 = 1$$

Figura. 2 - Pasos para obtener el texto en claro con 2 claves públicas diferentes. Fuente: Articulo- Improvement over Public Key Cryptographic Algorithm

2.3 ATAQUES POR EXPONENTE PÚBLICO MUY PEQUEÑO.

De igual manera que con la llave privada, el uso de llaves públicas pequeñas supone un ahorro en costos de procesamiento en el cifrado o verificación de la firma de un mensaje. Sin embargo este tipo de prácticas también ponen en riesgo las implementaciones del criptosistema y lo hacen fácil presa de ciertos tipos de ataques enumerados a continuación: Ataque por Teorema de Coppersmith [9] que se basa en el algoritmo LLL (Lenstra, Lenstra y Lovász) para reducción de bases, el ataque por difusión de Hastad (Hastad's Broadcast Attack [10]) donde básicamente el atacante captura una cantidad k de mensajes cifrados y mediante una serie de manipulaciones de esta información basados en el teorema del resto chino, el atacante puede llegar a obtener el mensaje. Tambien encontramos el ataque Franklin-Reiter [11] por mensajes relacionados donde el atacante puede aprovecharse de una comunicación establecida entre dos individuos donde se envían varios mensajes.

Existe también el ataque Coppersmith's Short Padding, donde el atacante aprovecha las deficiencias en el padding (agregar bits aleatorios al final de el mensaje para homogeneizar el tamaño de los bloques de mensaje transmitidos), en este ataque, el atacante, suponiendo que el emisor envía un mensaje con un padding aleatorio al emisor, lo intercepta, evitando que el mensaje llegue al receptor. El emisor, al notar que el receptor no responde, renvía el mensaje cifrado, con un nuevo padding aleatorio, el cual, de nuevo es capturado por el atacante, quien basado en estos dos mensajes puede obtener el texto claro.

Por ultimo tenemos el ataque por exposición parcial de la llave o Partial Key Exposure Attack [12], en el que se demuestra que con una pequeña fracción de bits del exponente privado que se pueda obtener, se podría deducir la llave completa y de allí la importancia de ser cuidadosos al momento de implementar el criptosistema.

2.4 ATAQUES A LAS IMPLEMENTACIONES DEL CRIPTOSISTEMA.

Estos tipos de ataque se refieren principalmente a aquellos que se realizan fuera del dominio del algoritmo en si, como ataques de hardware, o ataques basados en el comportamiento de los canales sobre los cuales se implementan la comunicaciones cifradas. Estos ataques también son conocidos como Side Channel Attacks (Ataques de canal alterno) y suelen clasificarse en dos tipos: Differential Side-Channel Attacks DSCA y Simple Side-Channel Attacks SSCA. La diferencia entre estos dos tipos de ataques radica en que SSCA explota la información de una sola medición para revelar datos secretos de la comunicación y DSCA realiza varias mediciones y deduce información a partir de lsa diferencias entre las mediciones tomadas.

Para dar mayor claridad a este tipo de ataques, es importante resaltar que a nivel computacional, el cálculo de una operación hace que el consumo de potencia, tiempo de respuesta, emisiones electromagnéticas, temperatura del procesador y otras características físicas varíen frente al procesamiento de otras operaciones y analizando estos cambios en las características físicas puede recopilarse información sobre dichas operaciones.

Primero tenemos los Timing Attacks, que basan su fortaleza en el hecho de que el tamaño del exponente privado d influye en los tiempos de cifrado de un mensaje y mediante mediciones precisas de los intervalos de respuesta de la implementación del criptosistema para realizar el cifrado o la generación de la firma se podría descubrir el exponente d. Esto fue demostrado por Kocher [13] en 1996.

Dentro de estos tipos de ataques alas implementaciones del criptosistema también encontramos un tipo denominado Random Faults o Fallos Aleatorios. Estos ataques son potencialmente peligrosos cuando se abusa del teorema del resto chino TRC para aumentar el rendimiento del algoritmo [14]. El ataque se basa en fallos aleatorios imperceptibles inherentes al hardware sobre

el cual se realiza el proceso de cifrado o firma, o que pueden ser inducidos por campos electromagnéticos, picos en la tensión eléctrica, etc., y que pueden afectar el resultado de una operación, invalidando así la firma generada. Con la firma errada, un atacante podría factorizar fácilmente el cuerpo o módulo utilizado por el emisor.

3. AVANCES TECNOLÓGICOS: COMPUTACIÓN CUÁNTICA Y EL ALGORITMO DE SHOR.

El Algoritmo de Shor, presentado en 1994 por el matemático Peter Shor Williston y que reduce a complejidad polinómica el problema de factorización de un número en sus factores primos. Como se mencionaba anteriormente, este algoritmo fue implementado exitosamente en el 2001 en un pequeño procesador cuántico y con el cual lograron resolver la factorización del número 15 en sus 3 y 5. Aunque los fundamentos matemáticos de este algoritmo son muy densos, a continuación se presenta una explicación del funcionamiento de este algoritmo de una manera didáctica, extraída de

http://www14.brinkster.com/aleatoriedad/compcuant1.htm :

Fase 1:

La primera fase del algoritmo es poner un registro de memoria en una superposición coherente de todos sus estados posibles. La letra 'Q' será usada para denotar un qubit que está en el estado coherente.

Cuando un qubit está en el estado coherente, se puede pensar en cómo existir en dos universos diferentes. En un universo existe como un '1' y en el otro existe como un '0'. Extendiendo esta idea a el registro de 3 bits podemos imaginar que el registro existe en 8 universos diferentes como se representa en la Figura. 3, uno por cada uno de los estados clásicos que podría representar (i.e.000, 001, 010, 011, 100, 101, 110, 111). Para tener el número 15, un cuarto bit es requerido (capaz de representar los números 0 a 15 simultáneamente en el estado coherente). Un cálculo ejecutado en el registro se puede pensar como un grupo entero de cálculos ejecutados en paralelo, uno en cada universo. En efecto un cálculo ejecutado en el registro es un cálculo ejecutado en cada valor posible que un registro puede representar.

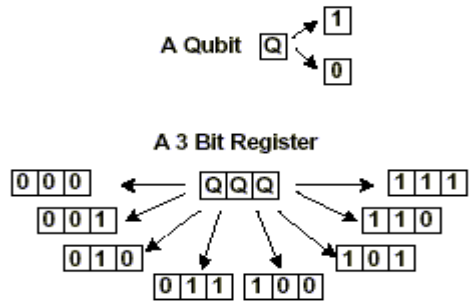

Figura 3. Un registro tres-qubit puede representar 8 estados clásicos simultáneamente. Fuente: http://beforeitsnews.com/alternative/2012/03/nsa-building-a-2-billion-quantum-computer-spy-center-1904783.html?currentSplittedPage=0

Fase 2

La segunda fase del algoritmo ejecuta un cálculo usando el registro.

- El número N es el número que deseamos factorizar, N = 15.
- Un número al azar X se escoge, donde 1<X<N-1.
- X es elevado a la potencia contenida en el registro (registro A) y entonces dividido por N.
- El resto de esta operación se pone en un segundo registro de 4 bits (registro B).

Después de ejecutar esta operación, el registro B contiene la superposición de cada uno de los universos resultantes. Esto se ilustra mejor con un ejemplo, si escogemos X igual a 2, entonces el contenido del registro B, por cada valor posible en registro A como se muestra en la Figura. 4.

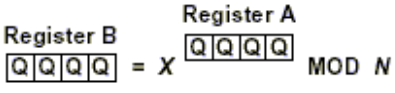

Figura 4. Funcionamiento ejecutado en Fase . Fuente: http://beforeitsnews.com/alternative/2012/03/nsa-building-a-2-billion-quantum-computer-spy-center-1904783.html?currentSplittedPage=0

Registro A	Registro B
0	1
1	2
2	4
3	8
4	1
5	2
6	4
7	8
8	1
9	2
10	4
11	8
12	1
13	2
14	4
15	8

Observe que los contenidos del registro B siguen una secuencia (1,2,4,8,1,2,4,8 …), la frecuencia en la cual se repite esta secuencia puede ser llamada f. En este caso, la secuencia (1,2,4,8) tiene cuatro valores, luego f=4.

Tabla 1. Contenido del registro B, cuando N=15 y X=2. Fuente: http://beforeitsnews.com/alternative/2012/03/nsa-building-a-2-billion-quantum-computer-spy-center-1904783.html?currentSplittedPage=0

Fase 3

La fase final es quizás la más difícil seguir. La frecuencia de repetición, f (ver Tabla 1.), puede ser encontrada usando una computadora cuántica. El valor resultante para f es entonces usado en la ecuación representada en la Figura. 5. para calcular un posible factor.

$$P = X^{\frac{f}{2}} - 1$$

Figura 5. Ecuación usada para calcular el factor. Fuente: http://beforeitsnews.com/alternative/2012/03/nsa-building-a-2-billion-quantum-computer-spy-center-1904783.html?currentSplittedPage=0

En nuestro ejemplo el valor f=4 da una respuesta correcta de 3.

Si bien es cierto que este algoritmo ha demostrado teóricamente ser eficiente para la factorización de los números en factores primos, su implementación a gran escala está directamente ligada a los avances en computación cuántica y más específicamente en el desarrollo de computadores cuánticos.

Desafortunadamente este campo ha tenido pocos avances en los últimos 10 años, principalmente por las técnicas utilizadas para la construcción de los procesadores cuánticos, que hacían supremamente difícil mantener el estado estable de los mismos y por ello generaban problemas de escalabilidad. A pesar de ello, la computación cuántica se vislumbra como una realidad tangible a un plazo de 20 años.

Estos son algunos de los aportes principales de la computación cuántica:

2008: La fundación nacional de ciencias (NSF) de los EEUU, consiguió almacenar por primera vez un Qubit, en el interior del núcleo de un átomo de fosforo, en donde se logró que la información permaneciera intacta durante 1.75 segundos. [15]

2009: Grupo de investigadores estadounidenses dirigidos por el profesor Robert Schoelkopf, de la universidad de Yale, el cual en el 2007 había desarrollado el bus cuántico, crea dos años después de esto el primer procesador cuántico de estado sólido, el cual funciona de forma similar al microprocesador convencional, con a diferencia que puede solo realizar pocas tareas muy simples como operaciones aritméticas o búsquedas de datos. [15]

2011: la empresa D-Wave System fundada en 1999, fabrica la primera computadora cuántica, construido dentro de un diamante por Robert Perkins. [15]

2012: Científicos de la universidad de california, han dado el primer paso en la construcción de una computadora cuántica al interior de un diamante, teneindo en cuenta el tema de la interferencia que se presentan en las funciones de estas computadoras, por lo cual se generó una cubierta de diamante, considerándolo el mental más resistente de la naturaleza; este computador es capaz de guardar 2 Qbits, es decir un bit en computación cuántica, pudiendo representar un uno o un cero al mismo tiempo. [15]

2015: Científicos de IBM lograron un importante avance hacia el camino de la construcción de un ordenador cuántico con uso práctico; en este avance se pudo detectar

y medir dos tipos de error cuánticos de forma simultanea bit-flip (consiste en cambiar el 0 al sitio del 1 y viceversa) y phase.flip (consiste en errores de cambio de fase que modifican la relación entre el 0 y 1 en una superposicion), hasta ese momento solo era posible tratar un error u otro, pero no los dos al mismo tiempo; lo cual lleva a un gran avance en lo que refiere a corregir errores cuánticos, considerando esto un punto muy importante en la construcción de un ordenador cuántico. [16]

4. CONCLUSIONES

Después del pequeño recorrido por la historia de RSA y por los intentos generados a lo largo de los años por vulnerar este criptosistema, podemos ver claramente las fortalezas que ofrece y las pocas debilidades que demuestra cuando se realizan implementaciones correctas de dicho criptosistema. También es claro que la factorización de los números primos no es a única manera de romper el criptosistema y que aún con la mejora en el rendimiento de los algoritmos que intentan atacar a RSA con este método, los sistemas computacionales actuales demuestran serias limitantes físicas para su implementación. Ante el asombro de muchos, el aprovechamiento de la tecnología y su profundo análisis han permitido llegar a conocer detalles de los datos manejados por un sistema a partir del estudio de su comportamiento al manipular dichos elementos. También es cierto que el sobrevalorado temor por la llegada de la computación cuántica hace que las bases de la criptografía actual se tambaleen, es seguro también que paralelo a este desarrollo, se crearán nuevas técnicas de criptografía que contrarrestaran los nefastos efectos de estos avances sobre la seguridad a nivel de las comunicaciones electrónicas.

REFERENCIAS

[1] W. Diffie and M.E. Hellman, "New directions in cryptography", IEEE Transactions on Information Theory 22 (1976), pp. 644-654.

[2] Daniel Lerch-Hostalot, «Ataque de Factorización a RSA», nov-2006. [En línea]. Disponible en: https://www.researchgate.net/publication/259475095_Ataque_de_Factorizacin_a_RSA.

[3] M. Ross, “IBM's Test-Tube Quantum Computer Makes History. First Demonstration of Shor's Historic Factoring Algorithm”. Internet: http://www-03.ibm.com/press/us/en/pressrelease/965.wss , Dic. 19, 2009 [Ago. 24, 2012]

[4] D. Boneh, “Twenty years of attacks on the RSA cryptosystem”, Notices of the American Mathematical Society (AMS), Vol. 46, No. 2, pp. 203-213, 1999.

[5] M. Bellare and P. Rogaway. Optimal asymmetric encryption. In EUROCRYPT '94, volume 950 of Lecture Notes in Computer Science, pages 92{111. Springer-Verlag, 1994.

[6] C. Pomerance,. "A Tale of Two Sieves". Notices of the AMS 43 (12) (1996), pp. 1473–1485

[7] J.Ramió Aguirre, “Vulnerabilidades y Ataques al Sistema de Cifra RSA” Curso de Seguridad Informática y Criptografía, Tenerife – España, 2005.

[8] D. Boneh and G. Durfee, “Cryptanalysis of RSA with private key d less than n 0,292”, IEEE Trans. Inform. Theory 36, 4 (2000), 1339-1349.

[9] D. Coppersmith, “Small Solutions to Polynomial Equations and Low Exponent RSA Vulnerabilities”, Journal of Cryptology, Vol. 10, 1997, pp. 233-260.

[10] J. Hastad, “Solving Simultaneous Modular Equations of Low Degree”, SIAM Journal of Computing, Vol. 17 No. 2. 1988, pp. 336-341.

[11] D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter, "Low Exponent RSA

with Related Messages", Eurocrypt '96, Vol. 1070, 1996, pp. 1-9

[12] D. Boneh, G. Durfee, and Y. Frankel. "An attack on RSA given a fraction of the private key bits". In AsiaCrypt '98, volume 1514 of Lecture Notes in Computer Science, pp. 25-34. Springer-Verlag, 1998.

[13] P. Kocher, "Timing Attacks on Implementations of Diffie- Hellman, RSA, DSS, and Other Systems", Crypto '96, Vol. 1109, 1996, pp. 104-113.

[14] D. Boneh, R. DeMillo, and R. Lipton. "On the importance of checking cryptographic protocols for faults". In EUROCRYPT '97, volume 1233 of Lecture Notes in Computer Science, pp. 37-51. Springer-Verlag, 1997.

[15] Jose Manuel Patiño Gutierrez, «Arquitectura de las computadoras y la computación cuántica», jul-2013. [En línea]. Disponible en: https://www.researchgate.net/publication/262871304_Arquitectura_de_las_computadoras_y_la_computacin_cuntica.

[16] IBM, «IBM Scientists Achieve Critical Steps to Building First Practical Quantum Computer», 29-abr-2015. [En línea]. Disponible en: http://www-03.ibm.com/press/es/es/pressrelease/46734.wss.