TRABAJO FIN DE MÁSTER

Máster de Seguridad de las Tecnologías de la Información y las Comunicaciones (UOC, URV, UAB)

Estado de la criptografía post-cuántica y simulaciones de algoritmos post-cuánticos

Autor
Álvaro Rodrigo Reyes Rosado

Tutor académico
Josep Rifà Coma

Tutor industrial
Marcos Arjona Fernández

Directora del máster
Helena Rifà Pous

Madrid, diciembre de 2018

Resumen / Abstract

Desde que apareció la idea de utilizar computadores cuánticos para resolver problemas matemáticos de gran complejidad que no podrían ser resueltos en tiempo polinomial por computadores clásicos inició una carrera en la que compiten dos equipos: los que desarrollarán el primer computador cuántico de uso general y los que descubrirán como poder resistir al uso malicioso de estos nuevos computadores.

La criptografía post-cuántica trata de imponerse a esta situación investigando nuevos algoritmos que reemplacen a los algoritmos actuales de clave pública como RSA dando lugar a sistemas resistentes tanto a algoritmos clásicos como a cuánticos antes de que la amenaza de un computador cuántico pueda acabar, entre otros peligros y en especial, con la seguridad y la privacidad en las comunicaciones a través de Internet.

Se refleja en este proyecto el estado actual de la investigación en esta materia: algoritmos criptográficos basados en problemas que no pueden ser resueltos por computadores cuánticos, desarrollos y patentes, aplicaciones actuales y futuras, etc. dejando al lector una fuente de recursos óptima para su iniciación en la criptografía post-cuántica.

Since the idea of using quantum computers appeared to solve complex mathematical problems that could not be solved in polynomial time by classical computers, it started a race in which two teams are competing: those that will develop the first general-purpose quantum computer and those that will discover how to resist the malicious use of these new computers.

Post-quantum cryptography tries to impose itself on this situation by investigating new algorithms that replace current public-key algorithms such as RSA, giving rise to systems resistant to both classical and quantum algorithms before the threat of a quantum computer can end, between other dangers and in particular, with security and privacy in communications over the Internet.

The current state of research in this field is reflected in this project: cryptographic algorithms based on problems that can not be solved by quantum computers, developments and patents, current and future applications, etc. leaving the reader an optimal source of resources for his initiation into post-quantum cryptography.

Listado de acrónimos

- PQC: Post-Quantum Cryptography
- QKD: Quantum Key Distribution
- NIST: National Institute of Standarts and Technology
- NSA: National Security Agency (USA)
- SIDH: Supersingular Isogeny Diffie-Hellman
- FPGA: Field-programmable Gate Array
- ASIC: Application-Specific Integrated Circuit
- CPA: Chosen Plaintext Attack
- CCA: Chosen Ciphertext Attack
- CCA2: Adaptive Chosen Ciphertext Attack
- IND-CCA: Indistinguishable Chosen Ciphertext Attack
- EUF-CMA: Existential Unforgeability under Chosen Message Attack
- SUF-CMA/sEUF-CMA: Strong Existential Unforgeability under Chosen Message Attack
- LWE: Learning With Errors
- SIS: Short Integer Solution
- KEM: Key Encapsulation Mechanism
- PKE: Public Key Encryption
- ITU: International Telecommunication Union
- NTRU: N-th degree truncated polynomial ring
- ZKP: Zero-knowledge proof/protocol
- OTP: One-Time Password
- OTS: One-Time Signature
- MPKC: Multivariate Public Key Cryptography

Palabras clave

Computación, post-cuantica, cuántico, criptografía, algoritmo cuántico, criptosistema, NIST, rendimiento, retículo, multivariable, corrector, errores, hash, McEliece, Niederreiter, simulación, Hamming, Goppa.

Cryptography, computation, post-quantum, quantum-safe, quantum-resistant, quantum algorithm, cryptosystem, NIST, performance, lattice-based, code-based, multivariate, hash-based, McEliece, Niederreiter, simulation, Hamming, Goppa.

Índice

Índice de figuras

Introducción

Cuando se habla de computación cuántica se tiende a pensar que es un campo de investigación que todavía no ha alcanzado un nivel de madurez suficiente pero la verdad es que poco a poco van apareciendo los primeros frutos del trabajo de los últimos años. Dentro de este mundo, la criptografía cuántica apunta a ser el futuro de la seguridad de las comunicaciones de aquí a unos años, pero eso no quiere decir que nos debamos alejar del sistema de computación clásico que ya dominamos por la nueva posibilidad de un sistema más eficiente.

Al contrario de lo que se pueda pensar la criptografía cuántica no es un tema que haya aparecido hace poco ya que las primeras apariciones de este tipo de criptografía datan de los años 80, pero es ahora cuando más énfasis estamos poniendo en el desarrollo de nuevos algoritmos criptográficos que puedan protegernos de las altas capacidades de computación de estos nuevos sistemas, los cuales vemos progresar con paso firme. Estos desarrollos tendrán multiples ventajas a nivel global sobre todo por la gran cantidad de aplicaciones que dependen del protocolo TLS.

Este es el objetivo de la criptografía post-cuántica: luchar contra una amenaza creciente de la que sabemos su existencia desde hace mas de dos décadas.

Estructura

Orientado para un nivel intermedio de conocimientos en criptografía, algebra lineal y teoría de cuerpos, este trabajo de fin de máster se estructura en cinco capítulos que recogen el estado actual de la criptografía post-cuántica con el fín de ofrecer una visión lo más completa posible del panorama actual con el objetivo de que sirva de asistencia a futuros investigadores e ingenieros a entrar de lleno en un área muy compleja:

- En el primer capítulo veremos el estado del arte, la cronología de la computación cuántica enfocada a la criptografía y qué ha motivado la aparición de la criptografía post-cuántica.
- En el segundo capitulo nos introduciremos más a fondo en los tipos de algoritmos que se investigan como base para crear criptosistemas seguros contra ataques cuánticos.
- En el tercer capítulo revisaremos la situación actual en el ámbito del i+D+I, tanto por parte de las universidades como de las empresas que están desarrollando nuevas soluciones y patentes y el estado del concurso promovido por el NIST para elegir nuevos estándares.
- En el cuarto capítulo analizaremos las consecuencias de la criptografía post-cuántica en determinadas tecnologías como blockchain o IoT.
- En el quinto y último capítulo simularemos en una aplicación cómo funcionan dos de los criptosistemas clásicos basados en códigos con corrección de errores que son capaces de resistir ataques cuánticos: el criptosistema de McEliece y el de Niederreiter.

Junto con la entrega de esta memoria se entregará un ejecutable .jar que contiene la aplicación desarrollada para simular los algoritmos, además de los recursos utilizados para hacer las gráficas que serán listadas en el tablón de figuras.

1 – Estado del arte

1.1 - Computación cuántica y criptografía cuántica

La computación cuántica aparece como una nueva forma de computación que tiene su base en la física cuántica y su origen se atribuye a Paul Benioff y Yuri Manin en 1980. Brevemente la física cuántica posee ciertas características que no suceden en la física clásica (Montanaro, 2015):

- Superposición: un sistema que puede estar en un estado A o B, también puede estar en una mezcla de ambos.
- Colapso: tras medir el sistema y obtener de forma probabilística uno de los dos resultados posibles, cualquier otra medición indicará siempre el mismo resultado.
- Entrelazamiento: un cambio en un sistema A afectaría simultáneamente a otro sistema B sin importar la distancia que los separe.
- Incertidumbre: no se puede determinar simultáneamente la posición de una partícula y su movimiento.

Estas características tan contraintuitivas en el mundo en el que nos movemos podrían servirnos de mucha ayuda cuando lleguemos a los límites de la Ley de Moore, ya que llegaremos a un punto en el cual no podremos reducir físicamente mucho más el número de transistores que podemos incorporar en los procesadores actuales, y la idea de pasar a un sistema cuántico es prácticamente necesaria para seguir progresando en el campo de las tecnologías de la información.

En la teoría descrita por Paul Benioff se exponía la idea de trabajar a nivel cuántico en lugar de utilizar voltajes eléctricos, permitiendo que en lugar de utilizar bits clásicos con posiciones 0 y 1 existiese una tercera opción en la que se puede estar en ambos estados a la vez, permitiendo realizar varias operaciones de forma simultánea. Esta nueva unidad de representación de la información pasaría a llamarse qubit (quantum bit), y la simultaneidad que nos ofrece no debe entenderse como un reparto de tareas en paralelo como estamos acostumbrados a implementar, sino como un estado en el que se pueden plantear todas las situaciones posibles al mismo tiempo (Cummins, 2018). Cada qubit permite tomar un valor exponencial, por lo que sistemas con N qubits poseerían la capacidad de realizar 2^N operaciones simultáneas. Para dar un ejemplo de lo que esto significa un computador cuántico con 30 qubits equivaldría a un procesador clásico de 10 teraflops. Ronald de Wolf, investigador del CWI, presentó un artículo sobre el posible impacto de estos computadores (de Wolf, 2017).

Actualmente se han desarrollado computadores cuánticos que son capaces de resolver problemas de optimización muy concretos, pero para ver computadores cuánticos generales de forma comercial todavía es necesaria mucha más investigación. En abril de 2018 investigadores de la Universidad de Innsbruck anunciaron la obtención de un entrelazamiento cuántico estable de 20 qubits (Universidad de Innsbruck, 2018) (Nieto, 2018).

Como se ha dicho antes se estima que en un futuro próximo la Ley de Moore que tanto ha servido de referencia para evidenciar el crecimiento computacional a lo largo de los años dejará de tener efecto una vez lleguemos a un punto en el que no podamos crear transistores más pequeños, por lo que nuestro progreso tecnológico en esta rama esta sentenciado. Sin embargo, con la aparición de los computadores cuánticos ha surgido una nueva ley: la Ley de Rose, la cual no basa el progreso en la relación inversamente proporcional del número de transistores y el tamaño con el que se crean sino en el aumento del número de qubits, con la diferencia de que

esta última no está limitada físicamente (Noor-ul-Ain, Atta-ur-Rahman, Nadeem, & Ghafoor Abbasi, 2015).

La criptografía tiene una de las aplicaciones de la computación cuántica con mayor perspectiva de futuro, en tanto que será posible evitar en gran medida los ataques de eavesdropping gracias a los efectos cuánticos en las comunicaciones y detectar cuándo se está produciendo un acto de espionaje. Las primeras menciones de la criptografía cuántica se producen en la convención Crypto 82 con el artículo "Quantum cryptography, or Unforgeable subway tokens" (Bennett, Brassard, Breidbart, & Wiesner, 1983) escrito por Stephen Wiesner, Charles Bennett, Gilles Brassard y Seth Breidbart, en el que se hablaba de la posibilidad de la creación de una memoria que no puede ser leída, solo verificada, y la multiplexación de dos mensajes de forma que solo pueda recuperarse uno a costa de destruir el otro. Wiesner ya era conocido por su trabajo sobre el código conjugado, considerado como los primeros esbozos de la criptografía cuántica, pero el trabajo nunca fue publicado hasta 1983 donde apareció en la revista SIGACT News (Brassard, Arxiv, 2005).

En 1984 Charles Bennett y Gilles Brassard desarrollaron el primer protocolo de distribución de claves cuánticas (QKD) llamado BB84, basado en las leyes de la mecánica cuántica en la correlacion del estado de polarización de fotones y no en la complejidad computacional de los problemas matemáticos subyacentes en los criptosistemas clásicos, con el objetivo de poder intercambiar claves de forma secreta. Este protocolo hace uso del teorema de no clonado, que asegura que no existe ningún procedimiento para copiar un estado cuántico a otro sistema sin que este sea destruido o modificado. Este teorema es imprescindible para la criptografía cuántica ya que, si en un canal de comunicación cuántico existiese un tercero intentando interceptar la comunicación, cualquier tipo de medición que éste haga resultará en una detección del intruso por parte del sistema, creando una comunicación segura teóricamente infalible (Lomonaco, 1998) (Gisin, Ribordy, Tittel, & Zbinden, 2008).

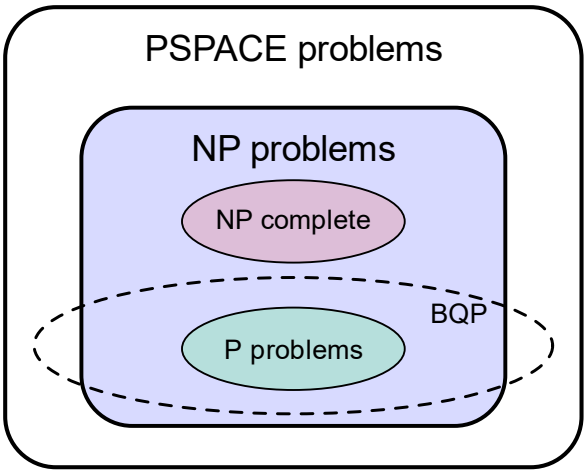
Además de BB84 existen otros protocolos de distribución de claves:

- Artur K. Ekert desarrolló de forma independiente en 1991 otro sistema QKD, el protocolo E91, basándose en fotones entrelazados.
- B92, una generalización del BB84 propuesto por Charles Bennett en 1992.
- Decoy-state QKD, propuesto por Won-Young Hwang en 2002, utiliza distintos niveles de intensidad con el objetivo de variar las estadísticas del número de fotones en el canal de comunicación, evitando los ataques de separación del número de fotones (PNS).
- SARG04, otra generalización del BB84 más robusta y también resistente a ataques PNS. Este protocolo fue propuesto en 2004 por Valerio Scarani, Sofyan Iblisdir, Nicolas Gisin, y Antonio Acín, de la Universidad de Ginebra.

Más tarde en 1993 Ekert organizaría el primer congreso internacional de criptografía cuántica, cuyo objetivo principal era recolectar todos los artículos posibles que tratasen el tema, hayan sido artículos publicados, esbozos, ideas, etc. A partir de este momento la criptografía cuántica gana notoriedad y se convierte en un campo de investigación con muchas expectaciones (Brassard & Crépeau, ResearchGate, 1996).

1.2 - Necesidad de la criptografía post-cuántica

La idea de la criptografía post-cuántica es desarrollar criptosistemas basados en problemas matemáticos que sean demostrablemente intratables para computadores cuánticos. Estos problemas pueden pertenecer a diferentes categorías según su complejidad:



- P: algoritmos resolubles en tiempo polinomial determinístico.
- NP: algoritmos resolubles en tiempo polinomial no determinístico.
- PSPACE: algoritmos resolubles en espacio polinomial (tiempo ilimitado, cantidad polinomial de memoria).
- BQP: algoritmos resolubles por un computador cuántico con una cantidad polinomial de compuertas cuánticas. Un problema de decisión pertenece a BQP si se resuelve la decisión con alta probabilidad y en tiempo polinomial con un algoritmo cuántico.

La idea de la criptografía post-cuántica es desarrollar criptosistemas basados en problemas matemáticos que sean demostrablemente intratables para computadoras cuánticas. Esto es, si suponemos que BQP es lo mismo que NP, buscar problemas que no estén en BQP.

En 1994 Peter Shor Williston, profesor de matemáticas aplicadas del MIT, ideó un algoritmo durante su trabajo en los Laboratorios Bell que permitía encontrar factores de un número de una manera eficiente (Shor, 1994), ampliando la información aún más en 1996 (Shor, 1996). Este algoritmo se cree que se halla en la clase de complejidad BQP antes descrita pero no en P. Dicho algoritmo, utilizado en un computador cuántico que por aquel se teorizaba con su existencia, sería capaz de resolver fácilmente aquellos problemas en los que se basaban los algoritmos de clave pública que utilizamos hoy en día: RSA (el algoritmo criptográfico creado por Ron Rivest, Adi Shamir y Leonard Adleman que forma la mayor parte de las comunicaciones seguras en Internet), DH (Diffie-Hellman, un algoritmo criptográfico de establecimiento de claves), ECC (Criptografía de curva elíptica) y DSA (Algoritmo de firma digital):

Algoritmo	Tipo de cifrado	Propósito	Impacto por computadores cuánticos a larga escala
AES	Simétrico	Cifrado	Se necesitarán claves más largas
SHA-2, SHA-3	-	Funciones hash	Se necesitarán salidas más largas
RSA	Asimétrico	Firma, establecimiento de clave	Dejará de ser seguro

ECDSA, ECDH (Criptografía de Curvas Elípticas)	Asimétrico	Firma, establecimiento de clave	Dejará de ser seguro
DSA (Criptografía de Cuerpos Finitos)	Asimétrico	Firma, establecimiento de clave	Dejará de ser seguro

(NIST, 2016)

Con el amplio despliegue de estos algoritmos a través de los protocolos SSL y TLS en Internet se hace urgente la investigación de nuevos algoritmos cuánticos que puedan reemplazar a estos antes de que aparezcan los primeros computadores cuánticos que sean capaces de ejecutar el algoritmo de Shor. Es en este punto donde aparece la criptografía post-cuántica.

El algoritmo de Shor no afecta a los esquemas de criptografía simétrica, aunque estos quedan alcanzados por otro algoritmo distinto: el algoritmo de Grover. Este es capaz de reducir la complejidad del problema de $O(N)$ a $O(\sqrt{N})$, lo cual podemos solventar de momento duplicando el tamaño de las claves, pero nos hace estar alertas de la aparición de nuevos algoritmos. En los siguientes puntos se mostrarán estos algoritmos en detalle.

Aunque para ejecutar el algoritmo de Shor para romper una clave pública de 2048 bits de RSA se calcula que harán falta entre 4000 y 10000 qubits (según el numero de puertas) en un computador cuántico de uso general, los expertos no creen que pueda tardar mucho más en llegar esta situación y hablan de intervalos de tiempo entre diez y cuarenta años.

Otro de los problemas que se añaden a la carrera por migrar a un estado en el que los criptosistemas sean resistentes a ataques cuánticos es la protección de los datos personales de los usuarios. En estos últimos años se ha puesto mucho énfasis a la privacidad del usuario a raíz de las grandes brechas de seguridad producidas en empresas multinacionales, y con la aparición del GDPR las empresas que no sean capaces de proteger sus datos confidenciales se arriesgan a multas considerables, poniendo en peligro su presencia en el mercado. Es un hecho que no todas las empresas tienen la capacidad técnica para avanzar en la resolución de este problema y por eso grandes tecnológicas como Google, Microsoft o IBM dedican tiempo y recursos para solventar esta situación.

Un detalle que no debe pasar desapercibido es la confidencialidad de los datos una vez aparezcan los primeros computadores cuánticos. Tanja Lange, profesora de criptología en la Technische Universiteit Eindhoven de los Países Bajos y experta en criptografía post-cuántica, explicaba en la conferencia de seguridad dotSecurity 2017 en su presentación sobre el estado de la criptografía (Lange, 2018) que hoy en día grandes agencias gubernamentales y de seguridad nacional de algunos países están almacenando información encriptada a la espera de poder ser desencriptada más adelante, por lo que es relevante realizar actualizaciones sobre aquellos datos que necesiten seguir siendo confidenciales una vez llegue el momento. Esto es especialmente crítico en datos personales como por ejemplo historias clínicas que deben permanecer secretas aún después del fallecimiento de una persona (Chang, 2017) (Holden, 2017).

1.2.1 - Algoritmo de Shor

Consideremos el problema de encontrar el periodo de una función exponencial modular. En aritmética modular el periodo de una función para un número aleatorio *a* módulo *N* es el menor positivo *r* tal que $a^r \equiv 1\ (mod\ N)$, o en otras palabras el número de veces que tiene que "rotar" la función para volver al mismo estado. Por ejemplo: para *N* = 15 y *a* = 7 y aplicando fuerza bruta encontramos que *r* = 4, es decir, $7^4 \equiv 1\ (mod\ 15)$, lo que implica que la función $7^{(x+4)}$ será congruente con $7^x\ (mod\ 15)$ para cualquier *x*.

Sabiendo el objetivo de este algoritmo, ¿por qué es interesante encontrar el periodo en este tipo de funciones? Porque el problema que resuelve el algoritmo de Shor consiste en generalizar tanto el problema de la factorización de números como el problema del logaritmo discreto en encontrar dicho periodo.

Pongamos a prueba el algoritmo de Shor para factorizar números enteros. El objetivo es encontrar los enteros α y β que resultan de la factorización de $N = \alpha\beta$. Para este caso escogeremos *N* = 1155 y *a* = 29.

Como *N* y *a* son coprimos, es decir, su máximo común divisor es 1, obtenemos la siguiente formula:

$$a \equiv 1\ (mod\ N) \Rightarrow a - 1 \equiv 0\ (mod\ N)$$

En esta formula podemos observar, aprendido el paso anterior, que $a \equiv 1\ (mod\ N)$ es igual que $a^r = 1\ (mod\ N)$ gracias al periodo de la función.

Una vez tenemos este punto, comprobamos que $a^r - 1 = 0\ (mod\ N)$ o, en otras palabras, *N* divide de forma exacta a $a^r - 1$. Para esta formula sabemos que existe una identidad notable y en consiguiente una factorización:

$$a^r - 1 = \left(a^{r}/_2 - 1\right)\left(a^{r}/_2 + 1\right)$$

En la siguiente tabla podemos ver cómo se encontraría uno de los factores propios de *N* utilizando el periodo de la función:

N	1155		
a	29		
Paso 1:			
Comprobamos que N y a son coprimos			
mcd(N,a)=	1		
Lo son, seguimos el algoritmo			
Paso 2:			
Buscamos un valor r tal que a^r = 1 (mod N)			
Realizamos la búsqueda por **fuerza bruta**			
r			
1	29		N
2	841		N
3	134		N
4	421		N
5	659		N
6	631		N
7	974		N
8	526		N
9	239		N
10	1		S
11	#¡NUM!		#¡NUM!
12	#¡NUM!		#¡NUM!

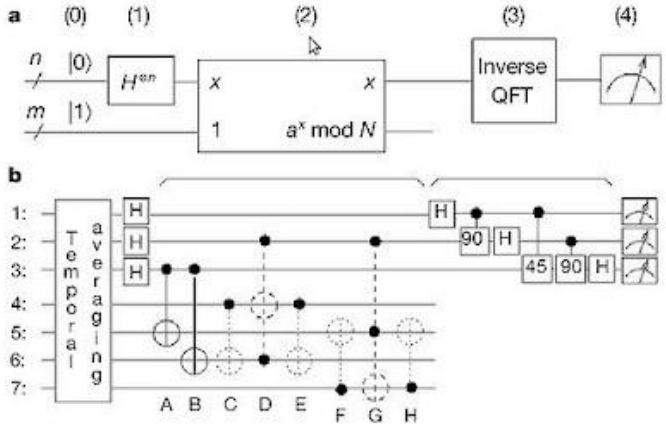
1 Primeros pasos del algoritmo de Shor. Observamos que en Excel la operación a^r cuando r>10 produce un número que desborda y provoca error

Paso 3:			
Elegimos el r que corresponde al primer candidato de la lista			
Poner aquí:	10		
mcd(a^(r/2)-1, N)=		7	
mcd(a^(r/2)+1, N)=		165	
Resultado:			
Los factores son 165 y 7			

2 Último paso del algoritmo con el que calcular los factores

Aplicando reiteradas veces el algoritmo de Shor sobre los factores que vayamos encontrando podremos encontrar fácilmente su descomposición en factores primos.

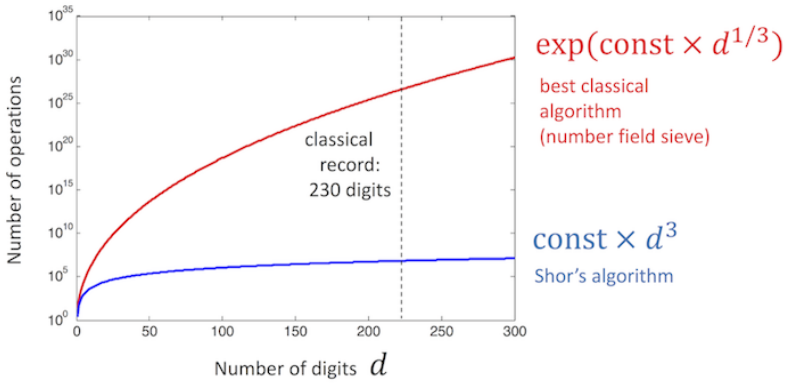
El problema de este algoritmo en un sistema clásico es que su complejidad es exponencial por lo que no importa lo potente que sea si no podemos usarlo para descomponer factores de números enormes como los usados para las claves RSA (Recordemos que RSA utiliza un módulo N resultado de la multiplicación entre los factores utilizados para la generación de claves). Es en este momento cuando aparece la aplicación de la transformada de Fourier cuántica para encontrar el periodo y tomar ventaja de la computación cuántica:



3 Algoritmo de Shor en un sistema cuántico. Fuente: Sambit Bikas Pal

La transformada de Fourier cuántica (QFT) utiliza paralelismo cuántico para computar de forma rápida las transformadas discretas de Fourier de las funciones en $\mathbb{Z}^n$. Gracias a la QFT podemos resolver el problema de encontrar el periodo de forma probabilística, que es una de las cualidades de los algoritmos cuánticos, y con una complejidad algorítmica $O(log^2(N))$

En la imagen (IBM, 2017) podemos apreciar la diferencia de coste computacional según la cantidad de dígitos del número:

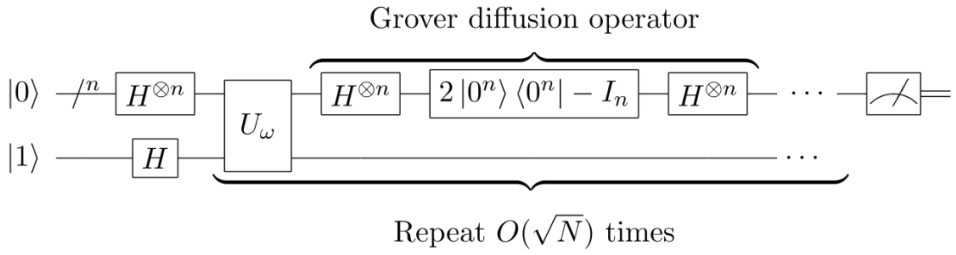


4 Comparación de complejidad

En comparación con uno de los mejores algoritmos de factorización para números con menos de 80 cifras, el algoritmo rho de Polland (con complejidad $O(2^{\frac{n}{4}}n^2)$ siendo *n* el número de cifras) o el algoritmo de la criba del cuerpo de números para cifras superiores, el algoritmo de Shor en un computador cuántico los supera a ambos con una complejidad de $O(log^4(N)log\ log(N))$ con $N = 2^n$, pasando de una complejidad exponencial a una polinómica.

1.2.2 - Algoritmo de Grover

El algoritmo que Lov K. Grover publicó en 1996 se resumía como un problema que se basaba en buscar un número de teléfono en un listado telefónico ordenado por nombre. En un computador clásico habría que realizar una búsqueda secuencial para encontrar el valor que se busca y por ende su complejidad sería lineal ($O(N)$). Sin embargo, en un computador cuántico en que se pudiera precargar el listado en memoria esta operación tendría una complejidad de $O(\sqrt{N})$.

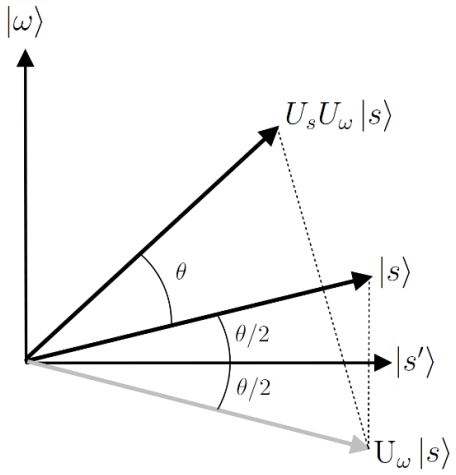


5 Esquema cuántico del algoritmo de Grover

Esta reducción de complejidad se debe a que no realiza una búsqueda lineal elemento por elemento hasta encontrar el que se está buscando, sino que confía en funciones oráculo para realizar un calculo de probabilidad de si el elemento es o no el correcto.

En la Figura 5 vemos el esquema cuántico de este algoritmo. Cada caja simboliza un operador o función cuántica. En el orden que aparecen tenemos los qubits $|0\rangle$ y $|1\rangle$, la puerta Hadamard H que asigna el estado base $|0\rangle$ a $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$ y el $|1\rangle$ a $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$, representando una rotación de π sobre los ejes X y Z, la puerta Hadamard $H^{\otimes n}$ que se describe como $\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle$, la función unitaria U_ω que realiza una operación de negación sobre el qubit si una función oráculo f devuelve un resultado positivo (1) para un valor y deja el qubit como está si la función devuelve un resultado negativo (0), la función U_s representada en la imagen como el operador de difusión de Grover y el símbolo de medición, que devuelve el resultado del algoritmo.

En la imagen lateral se puede ver el funcionamiento del algoritmo:



1) Se inicializa el sistema dando una probabilidad uniforme a todos los estados del qubit.

2) Se ejecuta U_ω, invirtiendo sobre la media aquella probabilidad del estado que se vea afectado por la función oráculo. Esto se representa con un ángulo θ/2.

3) Se ejecuta U_s, invirtiendo de nuevo la probabilidad, provocando que el valor afectado en el paso anterior incremente su probabilidad.

4) Se ejecutan los pasos 2 y 3 si es necesario hasta aproximarse todo lo posible al vector ω.

5) Se mide y se obtiene la probabilidad del estado que corresponde al valor buscado.

Evitamos explicar de forma técnica las formulas del algoritmo porque lo que queremos explicar es la idea que hay detrás: con cada iteración se puede observar como el vector que representa el qubit $|s\rangle$ se acerca más al vector que representa el qubit $|\omega\rangle$, o dicho de otra forma, cómo aumenta la probabilidad de que el elemento buscado cada vez que ejecutamos el bucle.

Destacamos que este algoritmo funciona de forma probabilística al igual que el algoritmo de Shor, y entre sus aplicaciones puede reducir el tiempo que se puede tardar en descifrar una clave simétrica con un algoritmo simétrico actual como AES. La estandarización de AES define los tamaños de clave de 128, 196 y 256 bits, lo cual ante un computador cuántico vería reducido su nivel de seguridad al que tendrían las claves de 64, 98 y 128 bits respectivamente.

Investigadores del National Institute of Technology, India (Kumar Rao, Mahto, Kumar Yadav, & Ali Khan, 2017) realizaron un artículo en el que comparaban el nivel de seguridad de los algoritmos AES, RSA y ECC frente a computadores clásicos y computadores cuánticos, llegando a la conclusión de que AES128 podría romperse, pero AES256 aún puede resistir. Los otros algoritmos, al ser de clave pública, pasarían a no tener seguridad alguna. Si fuera necesario los ordenadores clásicos tienen la suficiente potencia para poder duplicar el tamaño de clave de AES y seguir teniendo claves seguras, pero no podemos asegurar que no aparezcan nuevos algoritmos cuánticos más potentes que puedan romper estos algoritmos a base de fuerza bruta.

1.3 - Cronología

~1950 — Richard Feynman menciona la posibilidad de usar efectos cuánticos en computación

~1969 — Stephen Wiesner propone el primer acercamiento a la criptografía cuántica a finales de los años 60, en un artículo titulado "Conjugate Coding" que no fue publicado hasta una década más tarde, ya en los años 80.

1981 — Paul Benioff expone su teoría para aprovechar las leyes cuánticas en el entorno de la computación.

~1982 — Presentación en la conferencia CRYPTO'82: "Quantum cryptography, or Unforgeable subway tokens" por Stephen Wiesner, Charles H. Bennett, Gilles Brassard y Seth Breidbard, siendo esta la primera vez que se publicaba el término "criptografía cuántica".

1984 — Aparece BB84, el primer protocolo de criptografía cuántica para realizar distribución de claves cuánticas (Quantum Key Distribution), creado por Charles Bennett y Gilles Brassard. Más tarde aparecería el protocolo B92 aunque con mayores debilidades en su diseño.

1989 — Se publica el primer prototipo de criptografía cuántica en SIGACT News, que no es más que una prueba de viabilidad que permitia la transmisión segura sobre una distancia de 32 centímetros.

1991 — Artur Ekert desarrolla una aproximación diferente a la distribución de claves cuántica (QKD) basado en lo que conocemos como entrelazamiento cuántico.

1993 — Ekert organiza la primera conferencia internacional de criptografía cuántica en Broadway, Inglaterra. Se considera el primer evento con gran alcance dedicado a éste área.

1994 — Peter W. Shor descubre un algoritmo que permite a un teórico computador cuántico factorizar enteros muy grandes en tiempo polinómico, resolviendo tanto el problema de la factorización de enteros como el problema del logaritmo discreto.

1996 — Lov K. Grover expone el algoritmo de Grover para la búsqueda en una secuencia no ordenada de datos con N componentes en un tiempo de $O(\sqrt{N})$

1997 — Se demuestra la teleportación cuántica de forma experimental.

2000

David DiVincenzo publica “The Physical Implementation of Quantum Computation”, en el que enumera cinco requisitos para que un ordenador pueda funcionar basándose en efectos cuánticos.

2001

IBM demostró que era posible ejecutar el algoritmo de Shor factorizando el número 15 = 3 * 5 en un computador cuántico de 7 qubits

2004

Sale a la luz el primer criptosistema cuántico comercial (QKD), creado por la compañía suiza ID Quantique.

2007

D-Wave, una empresa canadiense, fabrica un computador cuántico con 16 qubits especializado en la optimización de problemas a través del llamado “quantum annealing” o algoritmo del temple cuántico, algo diferente a lo que se investigaba en las universidades.

2011

Se implementa la primera arquitectura Von Neumann cuántica con circuitos superconductores.

2015

La Agencia Nacional de Seguridad de Estados Unidos (NSA) realiza un anuncio con el que van a iniciar la actualización de la lista de algoritmos Suite B a nuevos algoritmos cuánticos “en un futuro no muy distante”.

2016

Se celebra el congreso PQCrypto en Fukuoka, Japón, donde el NIST anuncia que se ha abierto la convocatoria para elegir los algoritmos post-cuánticos que serán estándar en las comunicaciones.

2017

Para el concurso se identifican 69 aportes aceptados como “correctos y completos” y se cierra la convocatoria.

2018

Se realiza la primera conferencia donde comienza la presentación de algoritmos por parte de los equipos desarrolladores.

2018

Investigadores de la Universidad de Innsbruck (Austria) consiguen un entrelazamiento estable de 20 qubits, marcando el record actual.

2 –Algoritmos criptográficos

2.1 - Conceptos

Para desarrollar algoritmos resistentes a ataques cuánticos como los vistos en el apartado anterior es necesario tener claro el objetivo que se pretende conseguir. Los criptosistemas se diseñan en base a la utilización de unas piezas denominadas primitivas criptográficas como pueden ser el cifrado simétrico o la función hash que a su vez se diseñan respecto a un problema matemático, usualmente con difícil solución.

Por ejemplo, el criptosistema RSA utiliza el algoritmo de cifrado asimétrico del mismo nombre para realizar funciones de cifrado y firma, mientras que este algoritmo deja la carga de la complejidad en el problema de la factorización de enteros. Este criptosistema se alimenta de una sola clave para producir dos salidas distintas con las que se producirá una acción según el orden y el número de veces que se utilicen las claves pública y privada sobre el texto plano: autenticación, integridad, confidencialidad, no-repudio, etc.

Los algoritmos post-cuánticos no se diferencian en nada particular al algoritmo RSA y al resto de algoritmos de otros criptosistemas salvo en el problema matemático subyacente, el cual se busca que sea resistente a los ataques cuánticos. De hecho, se ha presentado como candidato a estándar una versión de RSA post-cuántica por lo que no es necesario descartar totalmente los algoritmos que usamos hoy en día. Tampoco es necesario que un solo criptosistema implemente todas las primitivas criptográficas que existan, puede haber algoritmos solamente dedicados a firmar y otros dedicados exclusivamente al intercambio de claves.

Recordamos que la necesidad de la criptografía post-cuántica surge por la necesidad de desarrollar algoritmos de cifrado asimétrico, que son los que apuntan a ser vulnerables próximamente. Es común encontrarnos algunos conceptos al investigar sobre estos algoritmos:

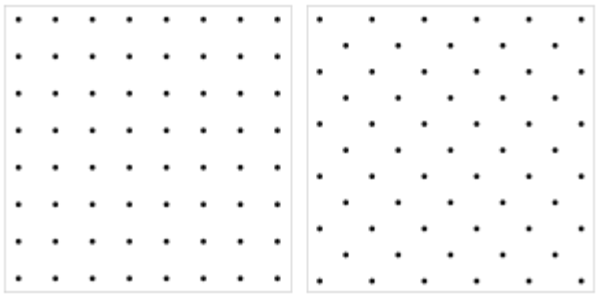
- Clave secreta/privada (sk, secret key): es la clave producida por el cifrado asimétrico que debe permanecer en secreto.
- Clave pública (pk, public key): es la clave producida por el cifrado asimétrico que puede ser compartida públicamente.
- Secreto compartido (ss, shared secret): es un dato que contiene información que necesitan saber ambas partes para iniciar una comunicación cifrada.
- Texto plano (msg, plaintext): es el texto que se pretende enviar.
- Texto cifrado (ct, ciphertext): es el texto resultado de la operación de cifrado.
- Firma (sm, signed message): es el resultado de la operación de firma.

La viabilidad de un cifrado asimétrico se puede ver reflejada en el tamaño de estos valores. Es conocido que los tamaños de las claves de RSA dependen de la versión que se esté utilizando (RSA-1024, RSA-2048, RSA-4096, a mayor número de bits mayor tamaño de claves y mayor seguridad contra ataques de fuerza bruta) pero tenemos otros cifrados asimétricos, como el criptosistema de McEliece que pese a que resiste ataques cuánticos si se construye con determinados parámetros produce unas claves públicas de casi un megabyte, convirtiéndolo en un cifrado que puede no ser óptimo en cuanto a memoria o velocidad en determinadas situaciones.

En los siguientes apartados vamos a resumir los diferentes tipos de algoritmos que se están investigando para la criptografía según la categoría de problemas que abordan.

2.2 - Criptografía basada en retículos

Los retículos se definen como un conjunto de puntos que se distribuyen de forma regular en un plano n-dimensional infinito. Debido a la infinitud de este plano y a los recursos limitados de un computador es necesario representar los retículos a través de bases. Las bases son un conjunto de vectores que se utilizan para representar cualquier punto del plano que forma el retículo, de forma que a partir de ellas se pueda expresar cualquier vector en un espacio vectorial como una combinación lineal de los elementos de la base.



6 Retículos en un plano bidimensional

Estas bases pueden ser cortas o largas dependiendo de la longitud de los vectores, es decir, la longitud de su representación en un plano. Para representar un punto cualquiera del plano puede existir más de una base pero cuanto más pequeños sean los valores de las bases más fácil será determinar la estructura del retículo, de ahí la importancia de buscar bases cortas para que los cálculos sean más eficientes.

Definidas las propiedades de los retículos se destacan principalmente tres problemas, partiendo de que nos dan de entrada unas bases largas para el retículo L:

- El problema del vector más corto (Alwen & Peikert, 2010), en el que se nos pide encontrar un punto lo más cercano al punto origen en L.
- El problema de las bases cortas, en el que se pide encontrar una base corta para L.
- El problema del vector más cercano, en el que se nos pide encontrar el punto del retículo L lo más cercano posible a un punto P aleatorio en el plano.

En criptografía se empezaron a trabajar con retículos a partir de las investigaciones de Miklós Ajtai sobre la dificultad de algunos problemas basados en estos objetos matemáticos. Se experimenta además de con estos problemas con otros como los basados en aprendizaje con errores (LWE, Ring-LWE) o los problemas del entero más corto (Short Integer Solution (SIS), Ring-SIS) para crear criptosistemas de clave pública, funciones hash, esquemas de firma o cifrado homomórfico (Peikert, 2013). Oded Regev explica en más detalle en qué consiste el problema LWE (Regev, 2009) mientras que Ajtai presentó el problema SIS en su investigación (Ajtai, 1996). Algunos ejemplos de criptosistemas basados en retículos los tenemos en el esquema de firma GGH (Goldreich, Shafi, & Halevi, 1997), el algoritmo NTRU (Hoffstein, Pipher, & Silverman, 1998) o el anillo de Peikert para intercambio de claves (Peikert, 2014).

2.3 - Criptografía basada en funciones polinomiales multivariables

La criptografía basada en polinomios multivariable, tambien llamada MPKC, es una de las categorías de criptografía post-cuántica con más desarrollo junto a la criptografía basada en retículos y en códigos correctores de errores. El termino multivariable hace referencia a que los polinomios que se utilizan para este tipo de criptografía tienen más de una variable, por ejemplo $f(x,y) = xy^2 + xy + y$. Comúnmente se utilizan polinomios cuadráticos sobre el cuerpo finito $GF(2)$, es decir, son sistemas no lineales (Mohamed & Petzoldt, 2016)(Yang, 2017)(Linde, 2018).

El tipo de problemas que se pueden adaptar para este tipo de criptografia derivan de los siguientes (Ding & Yang, 2009):

- Problema MQ: Dados m polinomios cuadráticos multivariables $p_1(x_1, \dots, x_n), p_2(x_1, \dots, x_n), \dots, p_m(x_1, \dots, x_n)$ en n variables $x_1, x_2, \dots, x_n$, encontrar un vector $xv = (x_1, \dots, x_n)$ tal que $p_1(xv) = p_2(xv) \dots = p_m(xv) = 0$. En 1979 se demostró que el problema MQ es NP-completo y para $m \approx n$ el problema es NP-duro incluso para $GF(2)$.
- Problema EIP: dada una clave publica P de un MPKC, encontrar unas funciones afines S y T y una funcion Q cuadratica tal que $P = T \circ Q \circ S$ donde el operador $\circ$ representa la composición de funciones.

En general, un esquema multivariable se construye a partir de una función fácil de resolver que es camuflada entre dos transformaciones lineales para obtener un nuevo sistema a partir de su composición. Este nuevo sistema es la clave publica, mientras que el sistema original y las transformaciones forman la clave privada.

Entre los esquemas más notables están (Goubin, Patarin, & Yang, 2011):

- C* crea un sistema cuadrático a partir de una transformación monomial sobre una extensión de un cuerpo finito.
- HFE (Hidden Field Equations), en lugar de utilizar una transformación monomial como C* utiliza una transformación binomial y construye el sistema cuadrático con una base explicita para la extensión del cuerpo finito.
- UOV (Unbalanced Oil and Vinegar) describe un sistema dado por m polinomios cuadráticos en $n = m + v$ variables $(w_1, \dots, w_m, w_{m+1}, \dots, w_{m+v})$ donde las m primeras se denominan *oil variables* mientras que el resto se llaman *vinegar variables*. Cada polinomio puede contener términos cuadráticos de la forma *oil*vinegar* o *vinegar*vinegar*, pero no *oil*oil*. Rainbow es uno de los criptosistemas presentados al NIST como candidato a estándar que se basa en esta idea.
- IFS (Intermediate Field System) construye un sistema Q a partir de un sistema Q' de k ecuaciones en k variables sobre una extensión de un cuerpo finito. Este sistema aún no se ha conseguido romper.

2.4 - Criptografía basada en códigos con corrección de errores

Durante una comunicacion se pueden producir errores en los bits de los mensajes causados por interferencias externas (ruido, problemas de red, etc.) provocando que al receptor le llegue un mensaje sin sentido. Para mejorar la calidad de la comunicacion se implementaron códigos correctores de errores, con los cuales se podría arreglar automaticamente un mensaje erroneo gracias a la redundancia de informacion. El ejemplo clasico es la repeticion triple del mensaje que se quiere enviar. Si se quiere enviar un 0 o un 1, se enviará 000 y 111 respectivamente. Al receptor le podrán llegar 8 posibles combinaciones de 0 y 1, pero el valor correcto será el que más ocurrencias tenga. Evidentemente este ejemplo no es muy efectivo para las comunicaciones reales, ya que tendriamos que enviar el mismo mensaje más de una vez, lo cual puede no ser optimo si el mensaje excede determinada longitud (para un tamaño de n bits hay que enviar 3n bits).

En general hablaríamos de códigos C(n,k), con los que se añaden n-k bits de paridad a los mensajes de tamaño k produciendo mensajes de tamaño n. Comparado con el ejemplo anterior, por cada k bits de informacion se enviarán n-k bits extra en lugar de 2n.

Pero estos errores que se pueden producir no son necesariamente provocados por el entorno. La criptografía basada en codigos correctores de errores se basa en la idea de añadir errores a proposito a los mensajes cifrados de forma que sea el receptor el unico con capacidad para arreglar y extraer la informacion. Entre los códigos más comunes en criptografía están los códigos Goppa y Golay binarios, Hamming, BCH, Reed-Solomon y Walsh-Hadamard.

El uso de los códigos correctores de errores en criptografía post-cuántica se basa principalmente en dos problemas de decisión NP-completos (Berkelamp, McEliece, & van Tilborg, 1978) y en derivaciones de estos:

- Problema del peso de las clases laterales:
    - Entrada: una matriz binaria H, un vector binario s y un entero no negativo w.
    - Propiedad: existe un vector e, cuyo peso de Hamming[1] es menor que w, tal que s = eH
- Problema de los pesos del subespacio:
    - Entrada: una matriz binaria H y un entero no negativo w.
    - Propiedad: existe un vector x, cuyo peso de Hamming es w, tal que xH = 0

En este trabajo se hablará en más detalle de los criptosistemas de McEliece y Niederreiter, que pueden implementar un sistema de cifrado asimétrico y de firma utilizando códigos correctores.

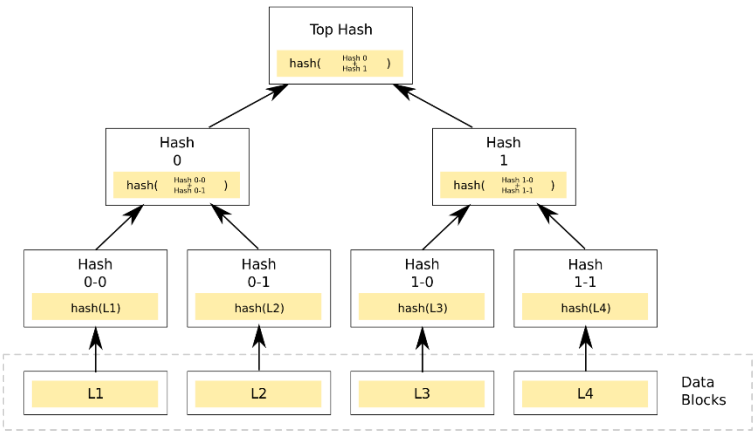
[1] Peso de Hamming: cantidad de 1 que resultan de aplicar xor entre una palabra de un código binario y la palabra todo ceros.

2.5 - Criptografía basada en funciones hash

En 1979 Leslie Lamport publica en su artículo *Constructing Digital Signatures from One Way Function* (Lamport, 1979) un esquema para implementar firmas a través de funciones hash. Se seleccionan dos claves secretas A y B; por cada bit del mensaje que se quiere firmar, se añade parte de la clave A si el bit es 0 y parte de la clave B si el bit es 1, formando al final una cadena formada por partes de A y B que representan la firma del mensaje. Este esquema, aunque simple, tiene la desventaja de que tanto las claves utilizadas como las firmas producidas pueden llegar a ser muy grandes además de funcionar como una firma de un solo uso ya que las claves no se pueden reutilizar.

Ralph Merkle desarrolló en su tesis (Merkle, 1979) una forma de utilizar el esquema de Lamport para implementar un esquema de firma multiuso basado en un árbol que pasaría posteriormente a llamarse árbol hash de Merkle.

Esta estructura permite que se puedan agrupar los hashes de distintos bloques de datos en un solo hash, con el coste de aumentar el tamaño de la salida.



7 Árbol hash de Merkle

Posteriormente Robert Winternitz propuso una mejora del esquema de Merkle en el que se utilizasen 256 claves distintas para firmar más de un bit. Dichas claves se irían produciendo de forma automática a partir de una clave semilla a la que se le aplicaría una función hash y produciendo finalmente una clave secreta equivalente a una clave pública.

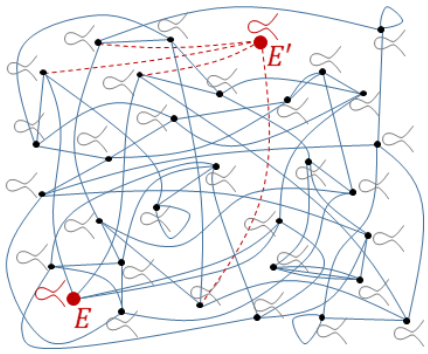
Estos fueron los inicios de la criptografía basada en hash, aunque el desarrollo de este tipo de funciones está limitado a los esquemas de firma. Para el concurso del NIST solo se han presentado dos algoritmos basados en hash:

- SPHINCS (Bernstein D. J., 2015) (E. & M., 2015), esquema de firma sin estado basados en hiper árboles.
- Gravity-SPHINCS (Aumasson & Guillaume, 2017), variante de SPHINCS que implementa algunas mejoras sobre el esquema anterior.

2.6 - Otros tipos

Random walk, polinomios de Chevychev, numeros hipercomplejos, derivaciones de RSA, pruebas de conocimiento cero, ecuaciones lineales… la criptografía post-cuántica no busca soluciones solamente en la clase de problemas que aparecen con mayor asiduidad, sino que cualquier tipo de problema matemático es viable para ser parte de un criptosistema. No todos son eficientes ni cumplen con los requisitos mínimos de seguridad, pero abren una puerta al descubrimiento de nuevas formas de proteger nuestra información. En este punto describimos dos categorías que consideramos dignas de mención.

2.6.1 - Isogenia de curvas elípticas supersingulares

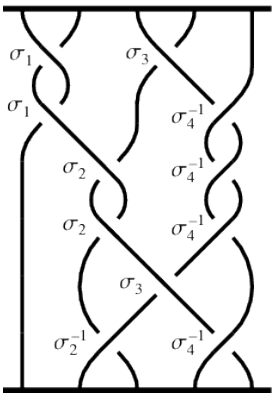


La isogenia de curvas elipticas supersingulares trata de encontrar una conexión entre dos curvas elípticas dentro de un conjunto de curvas definido como clase de isogenias supersingulares, que contiene todas aquellas curvas elipticas supersingulares sobre un cuerpo finito con p^2, siendo p un numero primo elegido de forma conveniente. Cada curva puede estar conectada con otras tres o cuatro curvas diferentes segun el tipo de mapeo que se realice gracias al numero primo elegido.

El funcionamiento básico del intercambio de claves es el siguiente: Alice y Bob eligen una curva E y E' respectivamente. Cada uno aplica su mapeo (bi-isogenias o tri-isogenias) hasta llegar a una curva EA y EB que serán compartidas publicamente. Ambos, una vez tengan las curvas del otro, seguirán aplicando su mapeo, de forma que llegaran a una curva E'' común, que será el secreto compartido para iniciar la comunicacion. A este proceso se le llama SIDH (Supersingular Isogeny Diffie-Hellman) (Castryck, 2017)

2.6.2 - Grupo de trenzas

Un grupo de trenzas se representa como un conjunto B_n de 2n puntos separados en dos partes superior e inferior y n cuerdas de forma que todos los puntos superiores se conectan con los inferiores a través de cuerdas, las cuerdas no intersectan entre si, y ninguna intersecta con cualquier linea horizontal mas de una vez (Garber, 2007).



Entre los problemas que se pueden abordar con los grupos de trenzas están el problema de decisión del conjugado (determinar si dos trenzas de un conjunto son conjugados o no), el problema de la búsqueda del conjugado (para dos trenzas $x, y \in B_n \times B_n$ que sean conjugados encontrar una trenza $a \in B_m, m < n$ tal que $y = axa^{-1}$) y el problema de la descomposición del conjugado (para dos trenzas $x, y \in B_n \times B_n$ y una trenza $b \in B_m, m < n$ que cumpla que $y = bxb^{-1}$, encontrar $a', a'' \in B_m$ tal que $y = a'xa''$) (Ko, 2000).

Estableciendo como base alguno de estos problemas es posible construir funciones de un solo sentido y esquemas de intercambio de claves.

3 – Investigaciones y proyectos en desarrollo

Detrás del desarrollo de los algoritmos post-cuánticos se encuentran los grupos de investigación que continúan con el avance en el descubrimiento de formas de aplicar nuevos problemas matemáticos para su uso en criptografía. Se exponen en la siguiente tabla los grupos con mayor presencia que se encuentran en activo investigando en este campo:

Grupos de investigación	Localización	Intereses y proyectos
Norwegian University of Science and Technology (NTNU)	Trondheim, Noruega	Investigacion en primitivas, ecosistemas e intercambio de claves post-cuánticos que puedan ser eficientes en entornos IoT (EURAXESS, 2018).
Universidad de Innsbruck	Innsbruck, Austria	Desarrollo de computadores cuánticos estables (Universidad de Innsbruck, 2018).
Institute for Quantum Computing (IQC), Universidad de Waterloo	Waterloo, Canadá	Open Quantum Safe (libOQS) (OpenQuantumSafe, 2018)
Universidad de Surrey	Guildford, Inglaterra	Aplicación de criptografía post-cuántica en sistemas basados en blockchain, como criptomonedas o contratos inteligentes (Universidad de Surrey, 2018).
Mathematical Institute, Universidad de Oxford	Oxford, Inglaterra	Diseño de protocolos criptográficos y evaluación de la seguridad de la criptografía clásica y post-cuántica ante los algoritmos clásicos y cuánticos (Mathematical Institute, s.f.).
Royal Holloway, Universidad de Londres	Londres, Inglaterra	Parámetros de seguridad de LWE para PQC y cifrado totalmente homomorfico, implementación hardware de criptosistemas post-cuánticos (Royal Holloway, s.f.).
Cybersecurity Research Lab, Ryerson University	Toronto, Canadá	Areas de investigación en seguridad IoT, tecnología blockchain y criptografía post-cuántica (IACR, 2018).
Applied Security and Information Assurance Group (APSIA) de la Universidad de Luxemburgo	Luxemburgo	Proyectos FutureTPM (en progreso) y PLAyBACk (Universidad de Luxemburgo, s.f.).
Universidad de Amsterdam / Universidad de Leiden /	(Leiden, Amsterdam), Países Bajos	Criptoanálisis de los algoritmos más prometedores presentados en la competición del NIST, con el objetivo de

Centrum Wiskunde & Informatica		encontrar vulnerabilidades, fallos en los esquemas o diferentes parámetros que puedan soportar ataques cuánticos (Centrum Wiskunde & Informatica, s.f.).
Majulab (UMI 3654), JFLI (UMI 3527), Equipo de criptografía de la Universidad de Rennes, equipo de Algoritmos y Complejidad del IRIF (CNRS UMR 8243)	Singapur, Francia, Japón	Equipos de cooperación que investigan sobre nuevas aplicaciones de algoritmos cuánticos para el criptoanálisis (Majulab, s.f.) (JFLI, s.f.) (IRIF, s.f.).
Universidad de Florida del Sur	Florida, Estados Unidos	Criptosistemas post-cuánticos basados en isogenias (Universidad de Florida del Sur, s.f.).
Universidad de Buenos Aires	Buenos Aires, Argentina	Sistemas de prueba de conocimiento cero.
Coding Theory and Cryptology group, Universidad Técnica de Eindhoven	Eindhoven, Paises Bajos	Desarrollo de criptosistemas que provean de seguridad post-cuántica, desarrollo de algoritmos cuánticos para criptoanálisis (Universidad Técnica de Eindhoven, s.f.).
Universidad Ruhr de Bochum	Bochum, Alemania	Criptografía aplicada y post-cuántica (Universidad Ruhr de Bochum, s.f.).
CDC, Universidad Técnica de Darmstadt	Darmstadt, Hesse, Alemania	Investigación en criptografía basada en retículos, en códigos, en hash y en multivariables (Universidad Técnica de Darmstadt, s.f.).
Universidad Técnica de Dinamarca	Kongens Lyngby, Dinamarca	Criptosistemas de firma basados en hash.
SECRET, Instituto Nacional de Investigación en Informática y Automática (INRIA)	París, Francia	Seguridad, implementación y diseño de primitivas criptográficas basadas criptografía de código con corrección de errores (INRIA, s.f.).
Academia Sinica	Taipei, Taiwan, China	Investigación en criptosistemas de clave pública multivariable (MPKC) y otros basados en retículos.
COSIC, Universidad Católica de Lovaina	Lovaina, Bélgica	Criptoanálisis contra los esquemas basados en isogenias supersingulares (Universidad Católica de Lovaina, s.f.).
Digital Security Group, Universidad de Radboud	Nimega, Países Bajos	PQC para Computación confiable (Trusted Computing), criptoanálisis (Universidad de Radboud, s.f.).

Universidad de Haifa	Haifa, Israel	Implementacion en dispositivos empotrados de algoritmos basados en retículos
Stichting Nederlandse Wetenschappelijk Onderzoek Instituten	Paises Bajos	Implementación de los algoritmos presentados al NIST para comprobar sus características y estudio de los ataques side channel sobre dichos algoritmos. Investigación sobre cómo afectan los nuevos algoritmos a los protocolos actuales y si es necesario realizar cambios en el funcionamiento de los protocolos (Stichting Nederlandse Wetenschappelijk Onderzoek Instituten, s.f.).
Nederlande Organisatie Voor Toegepast Natuurwetenschappelijk Onderzoek Tno	Paises Bajos	Investigación sobre el diseño y rendimiento de esquemas híbridos (Nederlande Organisatie Voor Toegepast Natuurwetenschappelijk Onderzoek Tno, s.f.).

Ademas de los proyectos de investigación también se encuentran colaborando empresas que poseen sus propios equipos de investigación. Entre estas empresas se encuentran Microsoft, IBM, Google, NXP Semiconductors, Thales Comm. Ltd., HW Comm. Ltd. EMC Inf. Systems International, Bundesdruckerei, SCYTL Secure Electronic Voting S.A., etc. colaborando tanto en los proyectos de investigación europeos como en el desarrollo de nuevo hardware para ejecutar criptografía cuántica.

Actualmente hay tres grandes proyectos financiados con fondos europeos para el desarrollo de PQC:

- El proyecto PQ-CRYPTO (CORDIS, 2015) terminó en febrero de 2018 y ha presentado 22 de los 69 algoritmos elegidos para ser estándar a través del concurso del NIST, entre los que aparecen criptosistemas de intercambio de claves como de firma basados en distintos tipos de algoritmos como los vistos en el apartado anterior. Entre los productos obtenidos con este proyecto se encuentra la librería libpqcrypto que implementa 19 de las 22 soluciones presentadas (Bernstein & Lange, 2018).
- SAFECrypto (CORDIS, 2015) tiene como objetivo desarrollar nuevas soluciones criptográficas basándose en problemas de retículos, enfocándose en parámetros como coste, consumo de energía, rendimiento y robustez según el tipo de aplicación ya sea restringida por los recursos o que opere en tiempo real. Este proyecto tiene su fecha de fin a finales de 2018.
- PROMETHEUS (CORDIS, 2018) al igual que el anterior enfocará su investigación en problemas basados en retículos para desarrollar un nuevo conjunto de primitivas criptográficas que permitan el desarrollo de protocolos para mantener la privacidad de los usuarios ante ataques cuánticos. El proyecto promete mostrar la factibilidad de la privacidad post-cuántica en distintos casos de uso como los pagos electrónicos, el voto electrónico y los sistemas de inteligencia contra ciberamenazas. Este proyecto tiene su fecha de fin a finales de 2021.

3.1 - Candidatos a estándar NIST

El Instituto Nacional de Estándares y Tecnología (NIST) es una agencia de la Administración de Tecnología del Departamento de Comercio de los Estados Unidos y su objetivo es promover la innovación y competencia en Estados Unidos mediante avances en metrología, normas y tecnología. Además, desde 1970 ha creado una serie de competiciones abiertas para encontrar los algoritmos criptográficos que se tomarían como estándar. En estas competiciones aparecen los algoritmos de cifrado DES y Rijndael (AES), el algoritmo de hash Keccak (SHA-3), y en 2016 el NIST convocó la cuarta competición abierta para encontrar un algoritmo criptográfico post-cuántico con el objetivo proteger las comunicaciones en el caso de que se llegue a construir un computador cuántico general y la seguridad de la criptografía asimétrica se vea comprometida.

Este concurso surge a raíz de un anuncio que realizó la Agencia de Seguridad Nacional de Estados Unidos en la que avisaban que iniciarán una transición hacia algoritmos resistentes a potenciales computadores cuánticos en un tiempo no muy distante (IAD, 2015). A pesar de la sensación que se propaga por las redes de que la necesidad de la criptografía post-cuántica se ha convertido en una carrera a contrarreloj en lugar de un avance científico los expertos en criptografía aconsejan no perder la calma y dejar que la estandarización asiente las bases para la protección cuántica (Schwartz, 2017).

A continuación, se exponen los candidatos que han presentado algoritmos que han sido validados como correctos y completos hasta finales del 2017:

Algoritmo	Tipo	Clase
Compact LWE	PKE/KEM	Criptografía basada en enrejado/retículo
CRYSTALS-KYBER		
Ding Key Exchange		
EMBLEM / R.EMBLEM		
FrodoKEM		
HILA5		
KCL (pka OKCN/AKCN/CNKE)		
KINDI		
LAC		
LIMA		
Lizard		
LOTUS		
NewHope		
NTRUEncrypt		
NTRU-HRSS-KEM		
NTRU Prime		
Odd Manhattan		
Round2		
SABER		
Three Bears		
Titanium		
CRYSTALS-DILITHIUM	Firma	
DRS		
FALCON		

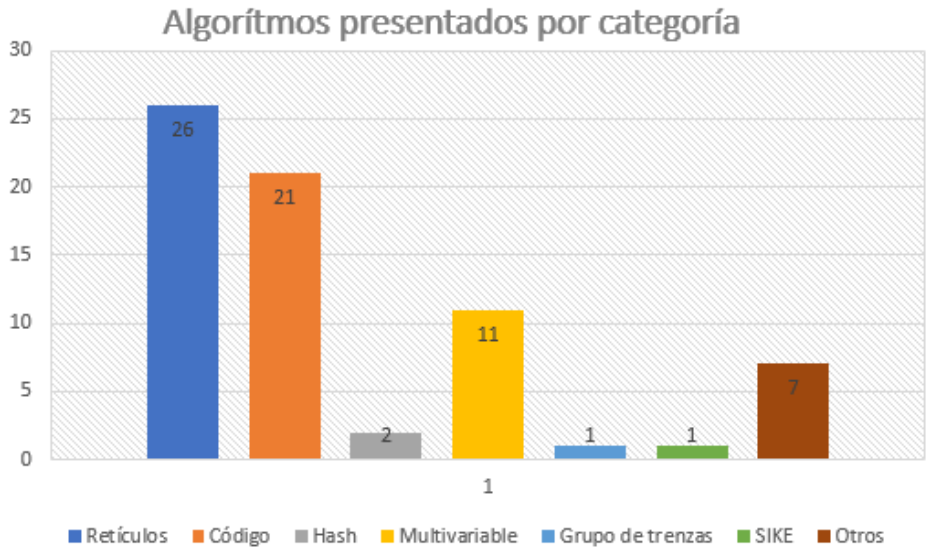
pqNTRUSign		
qTESLA		
BIG QUAKE	PKE/KEM	Criptografía basada en códigos con corrección de errores
BIKE		
Classic McEliece		
DAGS		
Edon-K		
HQC		
LAKE		
LEDAkem		
LEDApkc		
Lepton		
LOCKER		
McNie		
NTS-KEM		
Ouroboros-R		
QC-MDPC KEM		
Ramstake		
RLCE-KEM		
RQC		
pqsigRM	Firma	
RaCoSS		
RankSign		
Gravity-SPHINCS	Firma	Criptografía basada en funciones hash
SPHINCS+		
CFPKM	PKE/KEM	Criptografía basada en funciones polinómicas multivariables
Giophantus		
DualModeMS	Firma	
GeMSS		
Gui		
HiMQ-3		
LUOV		
MQDSS		
Rainbow		
SRTPI	PKE/KEM y firma	
DME		
WalnutDSA	Firma	Grupo de trenzas
SIKE	PKE/KEM	Isogenia de curvas elípticas supersingulares
pqRSA-Encryption	PKE/KEM	Satírico
pqRSA-Signature	Firma	Satírico
Guess Again	PKE/KEM	Otros
HK17		
Mersenne-756839		
RVB		
Picnic	Firma	

(NIST, 2018) Histórico (NIST, 2017) Tabla

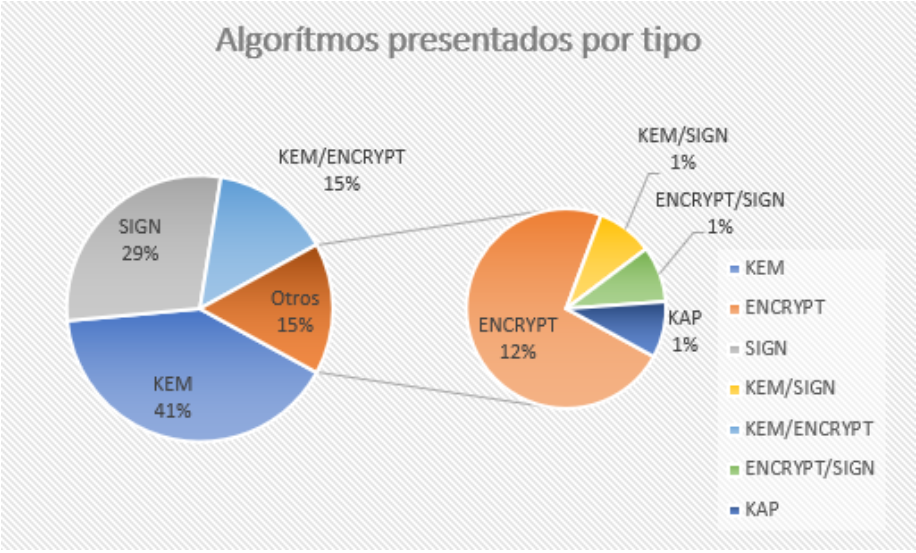
3.2 – Análisis global de las propuestas

Los algoritmos presentados incluyen, como medida de pre-validación, una serie de ficheros denominados KAT (Known Answer Test) que según define el NIST se utilizan para comprobar que la implementación es correcta, de forma que todos los participantes estandaricen la salida de su algoritmo para producir el mismo resultado y facilitar el proceso de validación.

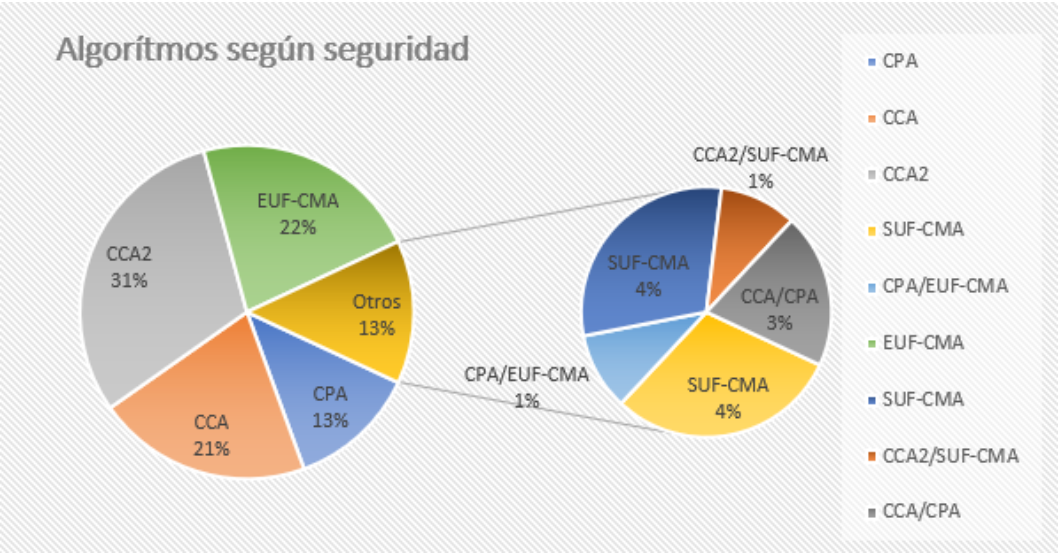
El análisis de datos se ha realizado a partir de los ficheros .rsp que corresponden a los KATs que ha presentado cada proyecto, en concreto a la última iteración de cada ejecución, y la fusión con los datos proporcionados por el proyecto SAFECrypto para aquellos proyectos que no han adjuntado KAT o lo han hecho siguiendo otro formato distinto al establecido (SAFECrypto, 2018).



8 Algoritmos presentados por categoría

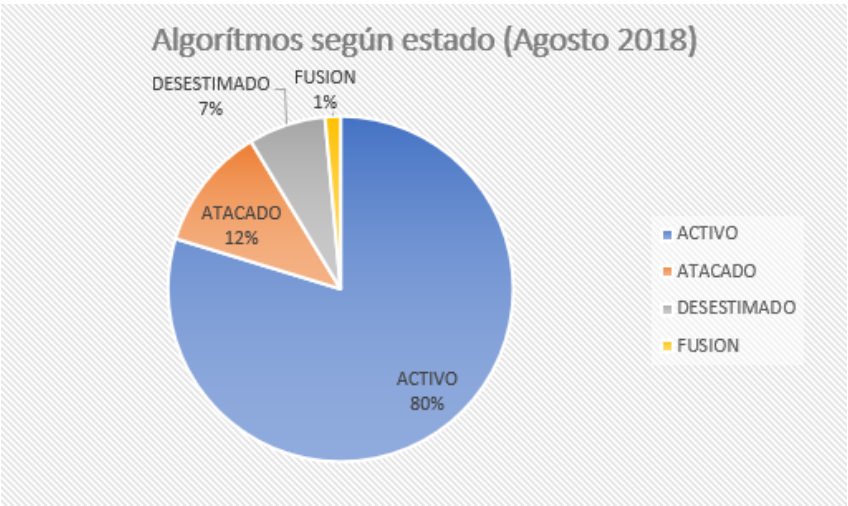


9 Algoritmos presentados por tipo

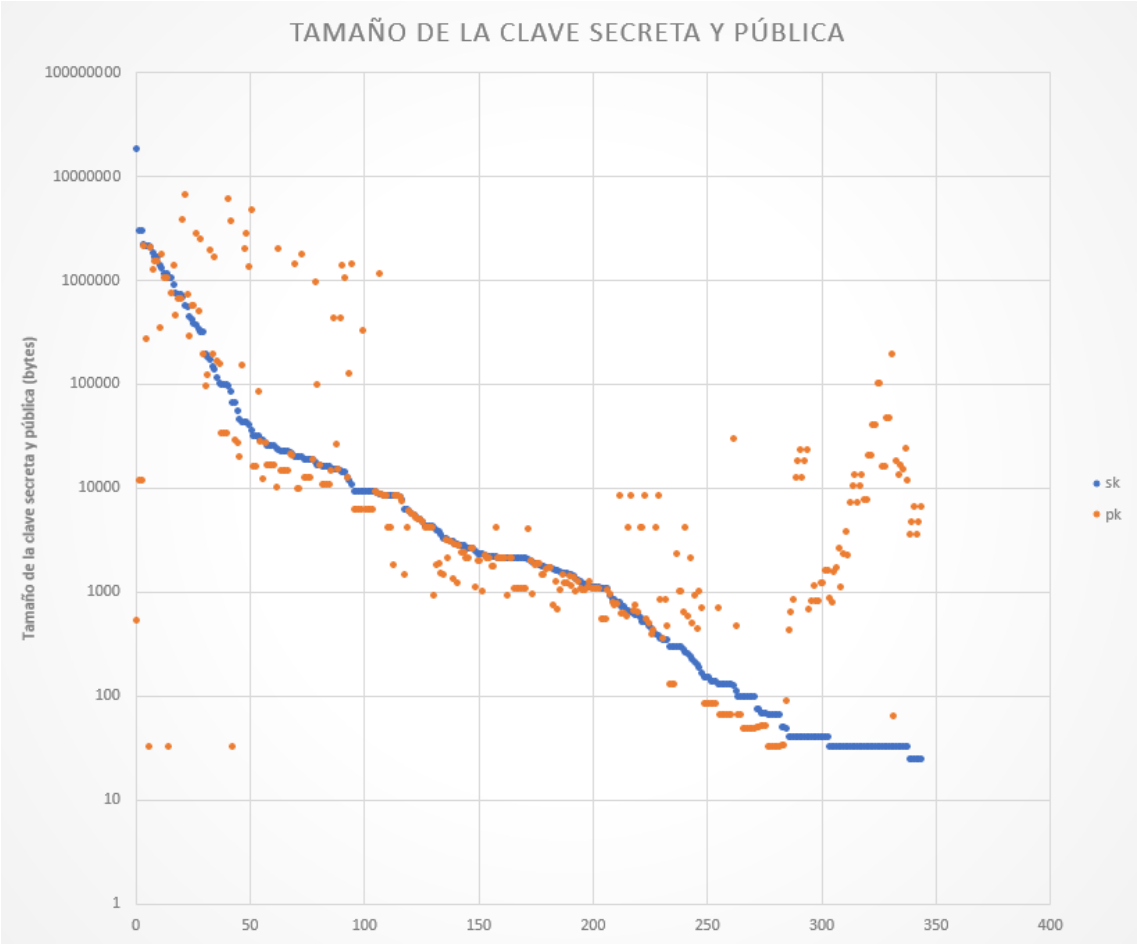


10 Algoritmos según seguridad

La mitad de los algoritmos presentados son teóricamente seguros contra ataques de texto cifrado elegido (CCA). Esto significa que los atacantes no pueden obtener información a partir de los textos descifrados de palabras seleccionadas.

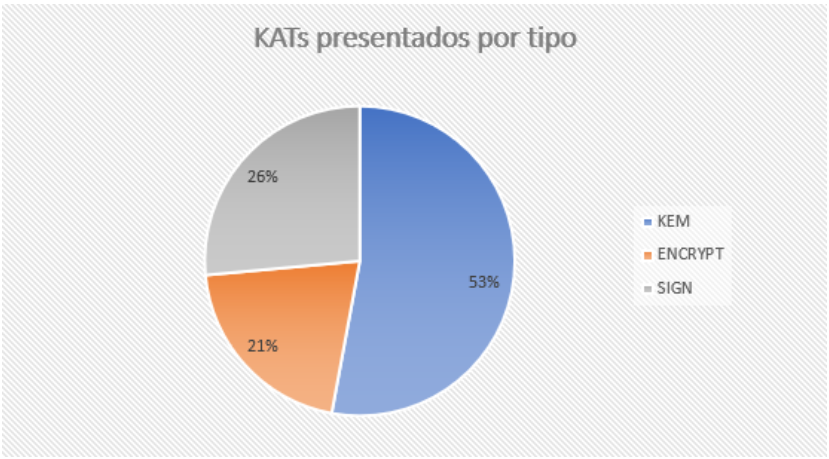


11 Algoritmos según estado (Agosto 2018)



12 Tamaño de claves secreta y pública ordenado descendientemente y a escala logarítmica

Observamos en la Figura 13 que se dan muchos casos en los que tanto la clave secreta como la clave pública son muy grandes. Se esperaría ver que la grafica siguiese una tendencia lineal a medida que decrece el tamaño de la clave secreta, pero como se puede ver existen muchos algoritmos que destacan por tener una clave pública muy grande y una clave secreta muy pequeña.



13 KATs presentados por tipo

Se han eliminado del análisis los siguientes algoritmos por no cumplir los estándares de salida en los KATs proporcionados por cada desarrollador o porque no se hallaron datos:

- pqRSA-Encryption
    - Los ficheros están vacíos.
        - encrypt/pqrsa20/kat_encrypt.rsp
        - encrypt/pqrsa25/kat_encrypt.rsp
        - encrypt/pqrsa30/kat_encrypt.rsp
        - kem/pqrsa20/kat_kem.rsp
        - kem/pqrsa25/kat_kem.rsp
        - kem/pqrsa30/kat_kem.rsp
- pqRSA-Signature
    - Los ficheros están vacíos.
        - sign/pqrsa20/kat_sign.rsp
        - sign/pqrsa25/kat_sign.rsp
        - sign/pqrsa30/kat_sign.rsp
- RaCoSS
    - No funciona el enlace de descarga del fichero rsp.
- HK17
    - Los ficheros rsp no siguen el patrón de los demás algoritmos.
        - PQCkemKATWithIntermediateValues_8.rsp
        - PQCkemKATWithIntermediateValues_16.rsp
        - PQCkemKATWithIntermediateValues_32.rsp
- LAC_AKE
    - Los ficheros rsp no siguen el patrón de los demás algoritmos.
        - LAC128/PQCakeKAT_1056.rsp
        - LAC192/PQCakeKAT_2080.rsp
        - LAC256/PQCakeKAT_2080.rsp

Los datos utilizados para generar las gráficas se adjuntarán con este trabajo.

4 – Criptografía post-cuántica aplicada

Hasta ahora hemos revisado la historia y los hechos más importantes durante estas últimas décadas, los tipos de algoritmos que se están investigando y aquellas implementaciones concretas que se han propuesto para estándar y que están validándose en estos momentos, pero nos queda por hablar de las aplicaciones de la criptografía post-cuántica y los efectos que tendrá una vez esté totalmente adaptada en nuestras comunicaciones.

4.1 - Dispositivos empotrados y chips

Infineon, empresa considerada el segundo mayor fabricante europeo de chips, consigue implementar el algoritmo NewHope en un chip sin contacto de una tarjeta inteligente, convirtiéndose en un fabricante pionero en la implementación de criptografía post-cuántica comercial (Infineon, 2017). Afirman que el pequeño tamaño del chip, la poca capacidad de almacenamiento y la velocidad de transmisión fueron retos que tuvieron que resolver al margen del diseño del propio esquema del algoritmo. Dicho algoritmo también está siendo utilizado por Google de forma experimental en su navegador Google Canary, cuya descarga está abierta al público. Google seleccionó el algoritmo por lo prometedor que aparentaba cuando se hicieron las primeras investigaciones.

Otro de los éxitos en implementar PQC lo tienen Brian Koziel, Reza Azarderakhsh y Mehran Mozaffari-Kermani quienes implementaron el primer intercambio de claves SIDH con tiempo constante en una FPGA, persiguiendo el objetivo de ejecutar SIDH tan rápido como fuera posible, posicionando este algoritmo como un competidor fuerte que ya tiene su implementación en lenguaje Go en TLS 1.3.

También se han realizado avances con librerías más ligeras como mbed TLS, anteriormente conocida como PolarSSL. Más abajo se define esta librería (Academia Sinica, 2014).

4.2 - Suites criptográficas híbridas

Las suites hibridas usan tanto algoritmos pre-cuánticos como post-cuánticos, protegiendo la clave de sesión si cualquiera de los dos problemas es difícil de resolver. En estos casos el algoritmo post-cuántico protegería frente a los futuros computadores cuánticos mientras que los pre-cuánticos o actuales seguirían protegiendo la comunicación si avanza el criptoanálisis de los algoritmos post-cuánticos. Por ejemplo, hablamos antes del algoritmo NewHope que se ha implementado en la versión Canary de Chrome (CECPQ1), pero realmente se ha implementado como una combinación con el protocolo ECDSA y la curva elíptica X25519/Curve25519 (Bos, 2016).

Los mayores progresos en el desarrollo de algoritmos post-cuánticos utilizables se basan en forks de OpenSSL, la librería de seguridad con más alcance que implementa los protocolos SSL y TLS. Precisamente por el amplio alcance de esta librería, porque es de código abierto y porque está programada en C, la convierten en un buen punto de partida para integrar los nuevos algoritmos de forma que la actualización en un futuro sea lo más suave posible. En concreto destacamos tres forks (Soto Velázquez, 2017):

- wolfSSL: pensada como una implementación de TLS para dispositivos empotrados, domótica, juegos para móviles y otros. Esta librería implementa la suite TLS_QSH con el criptosistema NTRU. Si ambos componentes de una comunicación soportan NTRU aseguran conseguir una mejora de velocidad de transmisión entre 20 y 200 por ciento.
- BoringSSL: es el fork que utiliza Google para implementar su suite CECPQ1.
- mbedTLS: para dispositivos de bajo coste y chips, este fork implementa algoritmos de intercambio de claves basados en retículos y algoritmos multivariables para firmas.

Uno de los proyectos de código abierto más destacados es la librería liboqs (OpenQuantumSafe, 2018), creada como parte del proyecto OpenQuantumSafe dirigido por Michael Mosca y Douglas Stebila, investigadores expertos en criptografía cuántica en el Institute for Quantum Computing (IQC) de la Universidad de Waterloo. Este proyecto utiliza un fork de OpenSSL en el que han implementado los algoritmos Frodo, BCNS, NewHope, MSrln, Kyber, NTRU (basados en retículos), McBits (basado en código), IQC-REF y MSR SIDH (basado en curvas elípticas supersingulares). En el SCIS 2018 (Symposium on Cryptography and Information Security) investigadores del KAIST presentaron una publicación sobre el rendimiento de esta librería, con resultados positivos para los algoritmos basados en retículos (An, Choi, Lee, & Kim, 2018).

En negrita se marca el objetivo al que se está apuntando según Stebila para 2030, para ir poco a poco progresando hasta tener algoritmos post-cuánticos de forma común como tenemos ahora, por ejemplo, RSA y ECDHE:

Intercambio de claves	Firma digital
Híbrida (clásica tradicional + post-cuántica)	**Clásica tradicional**
Híbrida (clásica tradicional + post-cuántica)	Híbrida (clásica tradicional + post-cuántica)
Clásica post-cuántica	Clásica tradicional
Clásica post-cuántica	Clásica post-cuántica

(Stebila, 2017)

Mark Pecen, miembro del IQC y director de operaciones de ISARA Corporation, una compañía dedicada a ofrecer productos resistentes a la computación cuántica, explica que están desarrollando certificados digitales post-cuánticos e híbridos con el objetivo de que la migración sea lo más fluida posible. En concreto están desarrollando nuevas características para el certificado estándarizado ITU X.509 de forma que operen de ambas formas, permitiendo la elección de algoritmos post-cuánticos si los clientes lo soportan y algoritmos clásicos en otros casos, e igualmente para los servidores (ISARA, 2017) (ISARA Corporation, 2017).

4.3 - Blockchain

En los últimos años se ha oído mucho hablar de la tecnología de cadena de bloques (blockchain), una estructura de datos donde la información se almacena en grupos llamados bloques con información de una operación e información extra de otros bloques de la cadena, formando gracias a técnicas criptográficas un histórico de operaciones que son validadas de forma descentralizada.

La primera aparición de la tecnología de cadena de bloques fue como la base de la criptomoneda Bitcoin en 2009, en la que la información que se almacena y valida en la cadena de bloques son operaciones financieras, con la particularidad de que no hay intermediarios entre las dos partes implicadas en la transacción; las operaciones en Bitcoin se validan por los propios usuarios del sistema quienes obtienen un beneficio al validar un bloque de la cadena, convirtiendo la "minería de Bitcoin" en un negocio lucrativo.

La cadena de bloques deja caer la responsabilidad de las validaciones de bloques en las funciones hash. Por ejemplo, Bitcoin utiliza un doble hash SHA-256 sobre la cabecera del bloque para crear el hash que lo identifica y una combinación de una clave pública generada con ECDSA (secp256k1) a la que se le aplica posteriormente RIPEMD-160 y SHA-256 para crear direcciones de Bitcoin.

Como destacábamos al principio de este trabajo, los algoritmos clásicos se pueden ver superados gracias a la computación cuántica, lo que supondría que aquellos negocios basados en cadena de bloques que no utilicen funciones hash resistentes podrían desaparecer al romperse la fiabilidad de la cadena. No solo eso, sino que a causa del algoritmo de Grover se puede prever una mejora sustancial para el minado de criptomonedas, dando ventaja a los usuarios de computadores cuánticos. Sin embargo, esta mejora aún no sería capaz de alcanzar las marcas que consiguen los circuitos ASIC diseñados con el único propósito de minar criptomonedas (Australian Cybersecurity Magazine, 2018). Frente a esta situación aparecen los criptosistemas post-cuánticos, que deben evitar estas situaciones si se descubren algoritmos cuánticos que puedan invalidar el esquema de la cadena de bloques.

Entre los últimos progresos en esta línea podemos hablar de investigaciones en nuevos esquemas, como por ejemplo una publicación reciente de un esquema basado en algoritmos basados en retículos para implementar una cadena de bloques post-cuántica (Gao, 2018), pero además de esquemas también existen plataformas disponibles como Blockchain Board of Derivatives (BBOD), utilizando el algoritmo SIDH para securizar su sistema (Junghee, 2017) o Kelvin Blockchain que utiliza diversos algoritmos (NewHope, NTRU, Frodo, SIDH, Picnic…) (Kelvin, 2018).

4.4 - Internet de las Cosas

Al revisar la situación de la criptografía en el Internet de las Cosas es fácil adivinar que el principal problema es tanto la falta de memoria como la escasa capacidad de computo que pueda tener el dispositivo. Simona Samardjiska expuso en el congreso RIOT 2017 una presentación sobre PQC en IoT (Samardjiska, 2017) que los dispositivos se ejecutan en entornos muy restringidos, por ejemplo con memoria de almacenamiento a niveles de kilobyte, la consumición de energía que gastan respecto a lo que pueden almacenar o conseguir de otras fuentes (como en el caso de los RFID) o el tipo de chip que utilizan (FPGA, ASIC). A medida que hemos ido describiendo diferentes algoritmos durante este trabajo hemos comprobado que sus tamaños crecen respecto a los algoritmos clásicos, además del rendimiento que ofrecen cada uno en su cómputo particular, lo que añade una capa más de problemas junto al entorno en el que se deben ejecutar.

Pero a pesar de esta perspectiva tan negativa ya se están realizando investigaciones sobre la eficiencia de los criptosistemas disponibles como por ejemplo el trabajo realizado por el Technical Research Centre of Finland sobre placas RaspberryPi 2 y 3, presentando los resultados del análisis energético y el coste de radiotransmisión en el intercambio de claves al utilizar los algoritmos Frodo y NewHope implementados en la librería liboqs, y una implementación de un esquema de código secreto sobre redes inalámbricas (Suomalainen, Kotelba, Kreku, & Lehtonen, 2018).

También se hace hincapié en la llamada criptografía ligera, que intenta conseguir un balance entre seguridad y rendimiento y de la cual se espera que sea uno de los pilares para este tipo de dispositivos. Como era de esperar el NIST también organiza convenciones para analizar el estado de este tipo de criptografía y en un futuro estandarizar nuevos protocolos. En cuanto a la investigación se están encontrando posibles opciones con futuro entre los algoritmos basados en retículos (Saarinen, 2016) (Rui, Chi, Yue, & Tao, 2018) (PQCRYPTO, 2015).

El lector puede encontrar más datos y ejemplos sobre PQC en IoT en los trabajos de Rickard Johansson y Thomas Strahl de la Universidad de Lund (Johansson & Strahl, 2016) y en el artículo de Oscar Garcia-Morchon et al. *Considerations for a lightweight, usable, and quantum-secure IoT* en el que detalla una serie de consideraciones para el diseño e implementación en IoT (Garcia-Morchon, 2016).

5 - Caso práctico: criptosistemas de McEliece y Niederreiter

Para completar este proyecto vamos a desarrollar una aplicación que implemente los criptosistemas de McEliece clásico y Niederreiter, de forma que podamos ver cómo funciona un algoritmo asimétrico basado en códigos con corrección de errores paso por paso.

El criptosistema de McEliece, publicado en 1978 por Robert McEliece (McEliece, 1978), tiene como base el problema de la decodificación vía síndrome de los códigos lineales. Se considera este problema NP-completo cuando el número de errores no está acotado. No obstante, hay ciertos tipos de códigos lineales que tienen un algoritmo de decodificación muy rápido por lo que, mientras que un atacante se vería forzado a decodificar vía síndrome para poder descifrar el mensaje, el usuario verdadero al haber establecido las condiciones del sistema puede usar el algoritmo de decodificación y obtener el mensaje de forma rápida. Podemos observar un ejemplo del funcionamiento de este criptosistema en (Denver University, 2012).

Los pasos para simular una comunicación con este algoritmo son los siguientes:

- Bob, para permitir que se puedan comunicar con él, debe:
  - Elegir un código lineal $C(n,k)$ que tenga un algoritmo de decodificación rápido y que pueda corregir *t* errores como mínimo.
  - Generar:
    - La matriz generadora G del código lineal $C(n,k)$ elegido.
    - La matriz no singular e invertible $S_{k,k}$.
    - La matriz de permutación $P_{n,n}$ (solo un 1 por fila y columna).
  - Publicar la matriz $G' = SGP$ que se convertirá en la clave pública junto con el parámetro *t*.
- Alice, para comunicarse con Bob, debe:
  - Dividir su mensaje en bloques *m* de tamaño *k.*
  - Construir un vector de errores aleatorio y de peso *t* o menor.
  - Generar y enviar $y = mG' + e$.
- Bob, para encontrar el mensaje, debe:
  - Calcular $y' = yP^{-1}$.
  - Decodificar *y'* con el algoritmo de decodificación del código lineal para generar *y''*.
  - Se tiene que $y'' = mS \Rightarrow m = mSS^{-1}$ y así Bob obtiene el mensaje m.

Como ya adelantábamos antes, el criptosistema de McEliece tiene la desventaja de que produce unas claves públicas muy grandes para ofrecer seguridad contra ataques cuánticos. En concreto se establecen unos parámetros determinados: el algoritmo debe funcionar con códigos Goppa binarios de longitud n = 6960, dimensión k = 5413 y con la capacidad de corregir t=119 errores produciendo de esta forma claves públicas de 8.373.911 bits (PQCRYPTO, 2015). Desde 2008 los parámetros de seguridad n=1024, k = 524 y t=50 no son seguros, tal y como expuso Daniel Bernstein et at. en un artículo en el que se detalla cómo atacar a este algoritmo con dichos parámetros, el tiempo y recursos que llevó y sugerencias sobre nuevos parámetros (Bernstein D. J., 2008). Como se apuntaba en el apartado 2.4 este algoritmo se basa en el problema NP-completo de encontrar los valores necesarios para crear una tabla de decodificación de síndromes, ya que el crecimiento es exponencial según el número de bits.

En esta tabla presentamos algunos de los códigos lineales que se han sugerido para la implementación del criptosistema de McEliece:

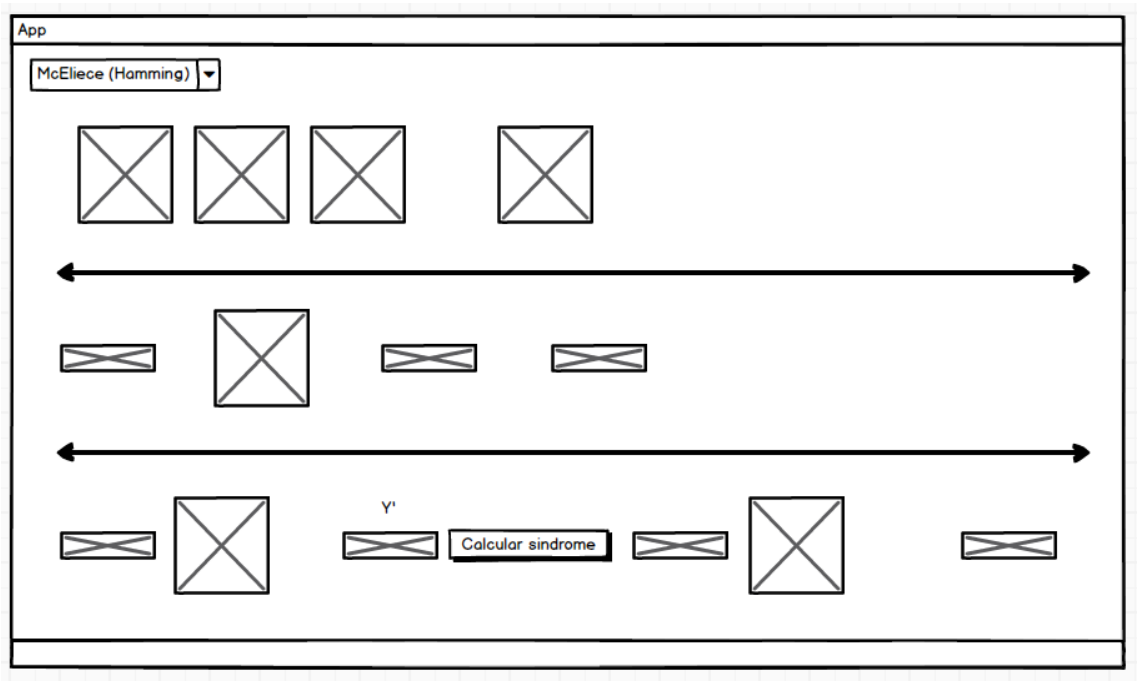
Código subyacente	Estado
Códigos Goppa binarios irreducibles	Seguro
Códigos Goppa sobre cuerpos finitos n-arios (n ≥ 31)	Seguro
Codigos de comprobación de paridad de densidad moderada (MDPC)	Seguro
Códigos convolucionales	No seguro (Landais & Tillich, 2013)
Códigos Reed-Solomon generalizados	No seguro (Couvreur et al., 2013; Gauthier, Otmani & Tillich, 2012)
Códigos Goppa sobre cuerpos finitos n-arios (2 < n < 31)	No seguro (Peters, 2010)
Códigos de distancia de rango máxima (MRD)	No seguro (Gaborit, Ruatta & Schrek, 2013)
Códigos Reed-Muller	No seguro (Chizhov & Borodin, 2013)
Códigos casi cíclicos alternantes, códigos Goppa casi diádicos, BCH, códigos de comprobación de paridad de densidad baja	No seguro (Faugère et al., 2010)
Códigos algebraicos geométricos en el caso de curvas hiperelípticas con genus bajo	No seguro (Faure & Minder, 2008)
Códigos Srivastava generalizados	No seguro (Sidelnikov, Shestakov, 1992)

Fuente: (Berkelamp, McEliece, & van Tilborg, 1978)

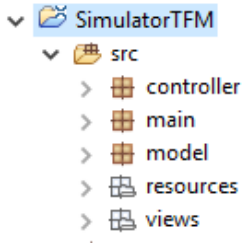
El criptosistema de Niederreiter, creado por Harald Niederreiter (Niederreiter, 1986), es una variante del de McEliece con la particularidad de que también permite firmar mensajes y cifrar hasta diez veces más rápido (Hudde, 2013). El funcionamiento se describe de forma similar:

- Bob, para permitir que se puedan comunicar con él, debe:
  - Elegir un código lineal $C(n,k)$ que tenga un algoritmo de decodificación rápido y que pueda corregir *t* errores como mínimo.
  - Generar:
    - La matriz de control H del código lineal $C(n,k)$ elegido.
    - La matriz no singular e invertible $S_{(n-k),(n-k)}$.
    - La matriz de permutación $P_{n,n}$ (solo un 1 por fila y columna).
  - Publicar la matriz $H' = SHP$ que se convertirá en la clave pública junto con el parámetro *t*.
- Alice, para comunicarse con Bob, debe:
  - Dividir su mensaje en bloques *m* de tamaño *n* y de peso *t* o menor.
  - Generar y enviar $c = H'm^T$.
- Bob, para descifrar el mensaje, debe:
  - Calcular $c' = S^{-1}c = S^{-1}H'(m)^T = HP(m)^T = H(m')^T$ donde $m' = mP^T$ siendo c' el síndrome de m'.
  - Aplicar el algoritmo de decodificación al síndrome c' para encontrar mP^T.
  - Se tiene que $m = mP^T(P^{-1})^T$ y así Bob obtiene el mensaje m.

El desarrollo de la aplicación comenzó con una fase de diseño de la interfaz a través del programa Balsamiq y un primer acercamiento a los algoritmos a través de MATLAB, que se adjuntarán con esta memoria.

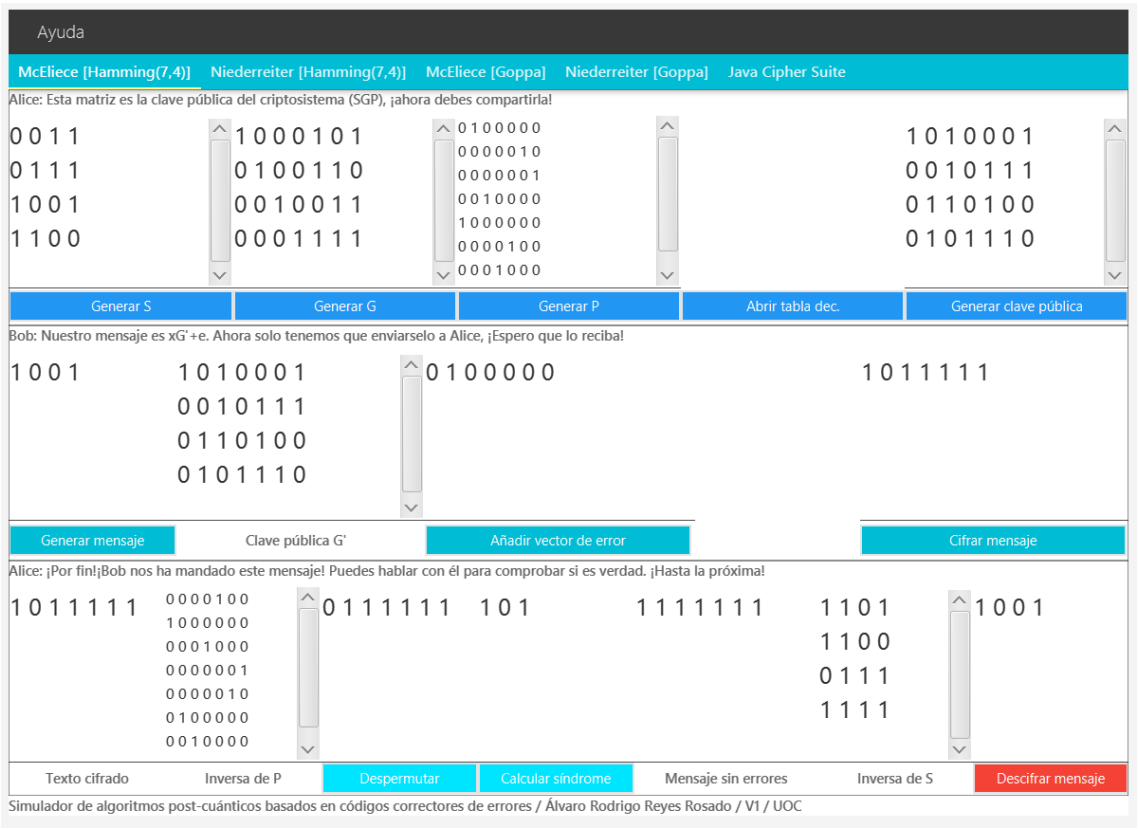


La aplicación sigue un patrón MVC. Para crear la interfaz se ha utilizado JavaFX junto con el software de diseño de interfaces Gluon Scene Builder, de forma que las interfaces se crean como ficheros .fxml y separan completamente las vistas del resto de la aplicación.

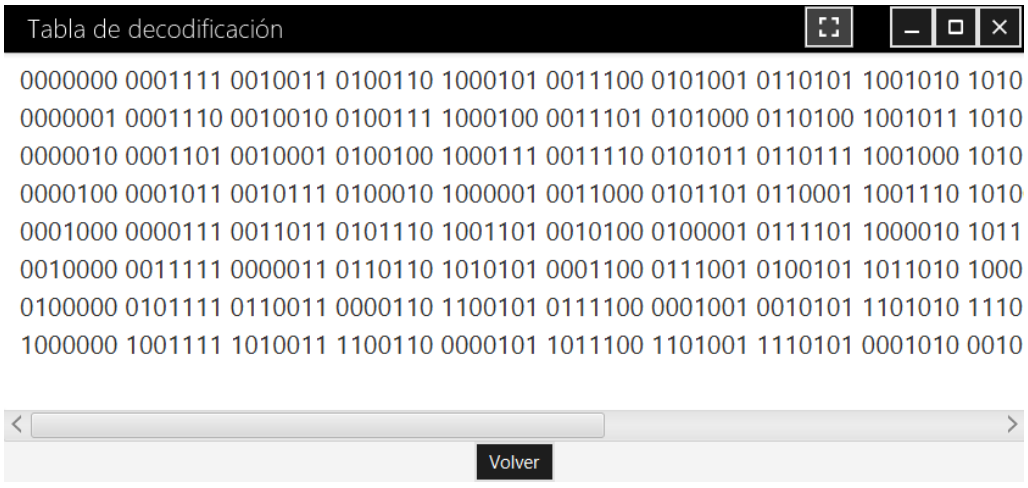


Todos los recursos que necesita la aplicación para funcionar se exportan en un solo ejecutable, excepto la configuración de seguridad de Java que debe realizar cada usuario de forma personal.

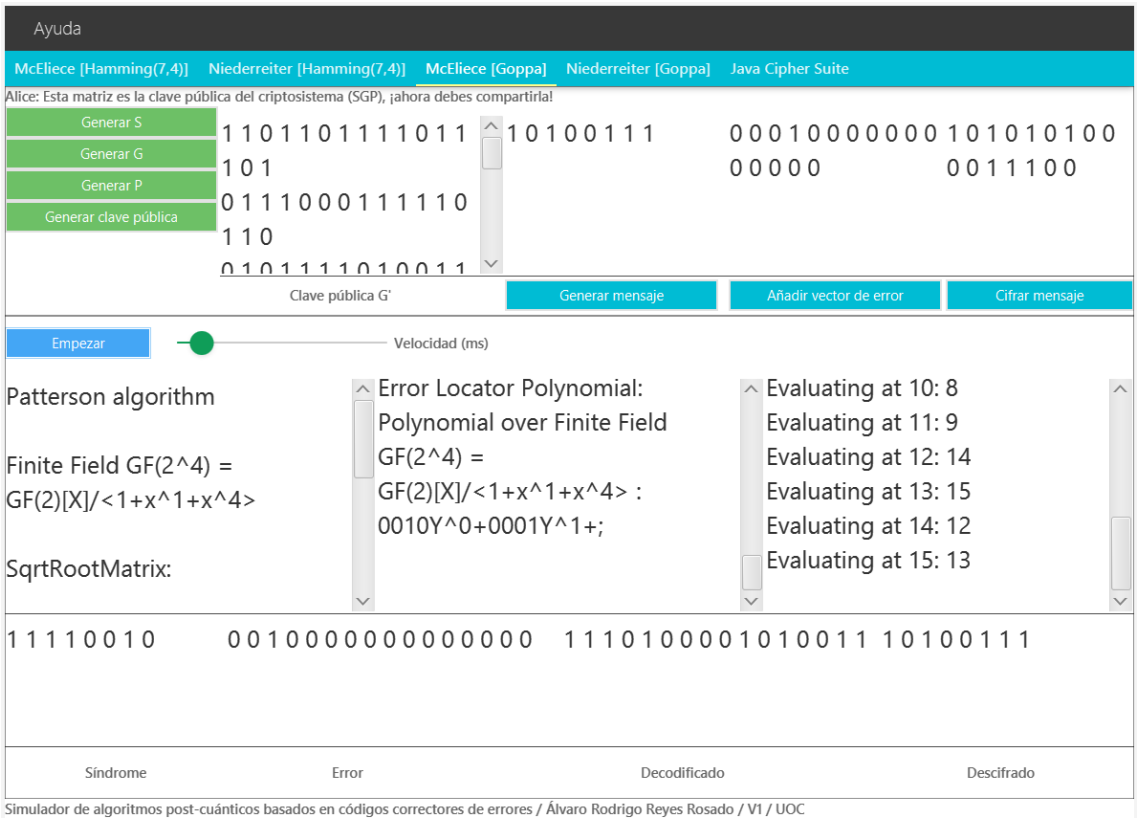
La aplicación muestra el funcionamiento de ambos algoritmos utilizando códigos Hamming(7,4) y códigos Goppa binarios irreducibles. Los primeros (Hamming) se utilizan para mostrar cómo funcionan los algoritmos en cada paso mientras que los segundos (Goppa) sirven para ilustrar la diferencia respecto al tipo de código lineal usado.



En la sección de McEliece con códigos Goppa vemos que la interfaz es distinta puesto que lo que se intenta mostrar aquí es la diferencia entre utilizar un código u otro. En el caso de los códigos Hamming realizamos la decodificación del síndrome basándonos en una propiedad característica del código Hamming(7,4) que permite saber la posición del error de forma exacta según el síndrome del error, sin tener que crear una tabla de decodificación. Sin embargo, a efectos de poder ver en detalle cómo sería una decodificación "ineficiente" se realiza la decodificación utilizando la tabla.

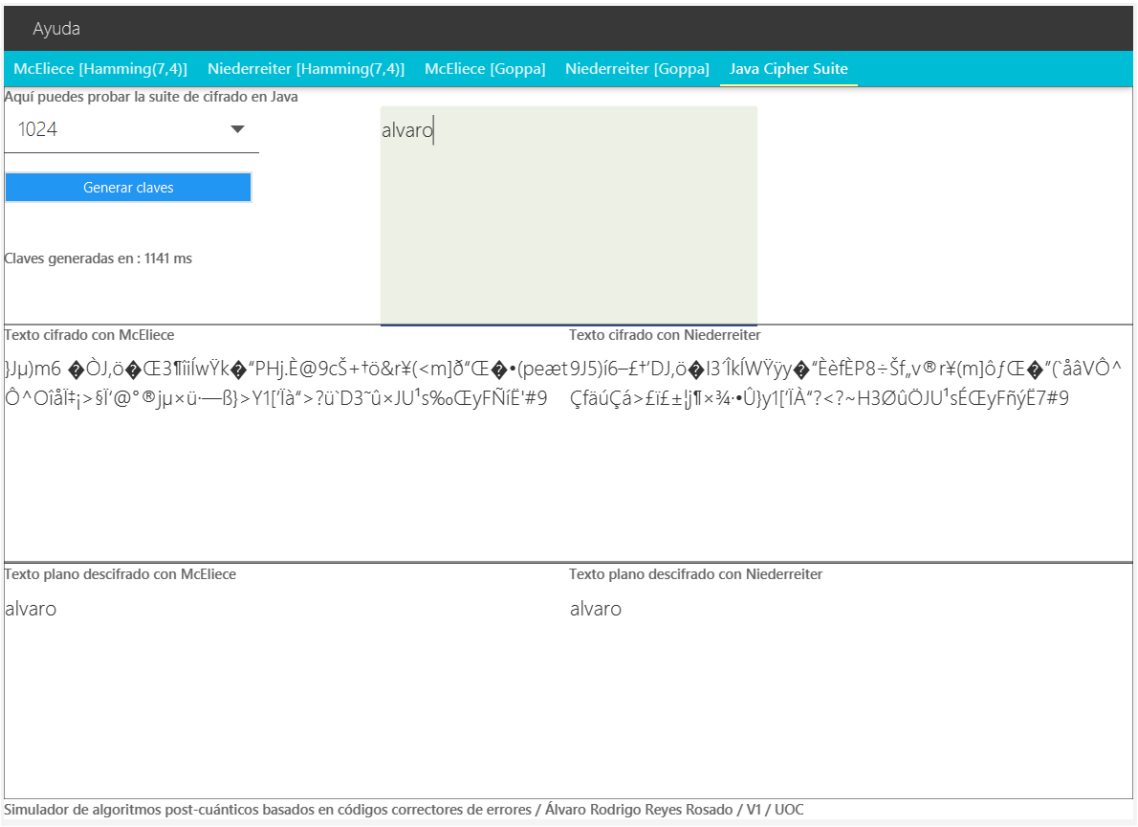


Por otro lado para los códigos Goppa se implementa el algoritmo de Patterson, que permite realizar la decodificación del síndrome sin generar dicha tabla de decodificación, algo que es especialmente interesante cuando la longitud de la clave es muy elevada. La implementación de este algoritmo en Java ha sido desarrollada por Elena Klintsevich, de la Technische Universitat Darmstadt, para el software FlexiProvider. Dicha implementación se ha segmentado en diferentes partes para ofrecer la visualización paso a paso del algoritmo de Patterson, no obstante es necesario mencionar la autoría del código original.



Los parámetros utilizados para generar el código Goppa para esta aplicación son n=16, k=4 y t=2. Es posible ajustar el código para que se puedan cambiar los parámetros k y t, pero se ha decidido establecer unos parámetros fijos porque lo importante es el seguimiento del algoritmo y hacer notar que parte del interés del desarrollo de esta aplicación recae en la visualización del código fuente y en poner a disposición un punto de entrada para entender estos algoritmos con la única ayuda de seguir el flujo del programa.

Por último se muestra la diferencia en el tamaño de claves que se pueden utilizar en la implementación desarrollada por FlexiProvider, con el que podemos ver la diferencia en el tiempo de generación de claves y cómo cambian los valores que se obtienen al cifrar con ambos algoritmos.



Los requisitos para hacer funcionar la aplicación de forma independiente son los siguientes:

- El programa debe ejecutarse con una versión igual o superior a Java 8 (Para la compilación se ha utilizado jdk1.8.0_111).
- Extraer las políticas sin restricciones en ${JRE_HOME}[2]/lib/security. Esto se debe a las restricciones de los proveedores de algoritmos criptográficos en ciertos países, controlados a través del Java Cryptographic Extension (JCE).
- Extraer el proveedor FlexiProvider (FlexiProvider-1.7p7.signed.jar y CoDec-build21-jdk13.jar) en ${JRE_HOME}/lib/ext.
- Añadír al fichero java.security el nuevo proveedor: *security.provider.**N**=de.flexiprovider.pqc.FlexiPQCProvider* (donde **N** es un número que depende de la configuración de seguridad de Java personal)

Para este trabajo se adjunta el programa junto con una versión de Java (JRE) preparada para su ejecución en Windows bajo una resolución de pantalla recomendable de 1920x1080 o superior.

[2] JRE_HOME es la ruta en la que se encuentra el entorno JRE de Java

Para terminar este apartado aclaramos de nuevo que el programa realizado se basa en las versiones clásicas de los algoritmos y que, a medida que han ido pasando los años, han ido apareciendo variaciones que intentan mejorar las deficiencias de estos. Sin ir más lejos el criptosistema de Niederreiter representa una mejora respecto al de McEliece ya que permite la firma digital. En un artículo Bolkema et al. documentan dos variaciones del criptosistema de McEliece (Bolkema, 2017):

- McEliece con máscara de peso 2
    - Este sistema propone la sustitución de las matrices de permutación clásicas por matrices invertibles cuyas filas contengan exactamente un par de unos. Explican que este cambio también se puede aplicar al criptosistema de Niederreiter.
- McEliece clásico basado en códigos de comprobación de paridad con densidad moderada espacialmente acoplados (*spatially coupled moderate density parity-check code*/SD-MDPC)

En el artículo se comenta además que un atacante puede intentar romper estos criptosistemas utilizando algoritmos de decodificación o buscando formas de explotar la estructura del código para crear un decodificador eficiente.

Conclusiones

Tras el recorrido realizado sobre el estado de la criptografía post-cuántica podemos observar varios hechos. El primero y más evidente es que, como se lleva diciendo desde hace años, hace falta desarrollar cuanto antes nuevos algoritmos de cifrado. Se está poniendo mucho énfasis en ciertos algoritmos como NewHope, Frodo, NTRU o SIDH ya que cuentan con recursos muy potentes tanto en el lado técnico como en el financiero. El resultado de la estandarización dentro de unos años verificará si alguno de estos algoritmos se convierte en el RSA del siglo XXI.

El segundo hecho es que al margen de la estandarización muchas empresas ya están implementando algoritmos post-cuánticos sin saber si resultarán vulnerables tras la revisión del NIST. El tipo de implementación suele estar ligado al lugar: las universidades se centran mayoritariamente en encontrar y adaptar problemas matemáticos para el diseño de algoritmos post-cuánticos mientras que las empresas se centran más en la implementación de los algoritmos y el desarrollo de hardware para su ejecución, al mismo tiempo que siguen investigando sobre el computador cuántico. No solo se están implementando los algoritmos, sino que también se están desarrollando ideas de negocio basados en ellos como por ejemplo nuevas criptomonedas que prometen ser más resistentes que las actuales.

El tercer hecho es que la criptografía post-cuántica no es algo nuevo y que antes de que apareciesen las primeras menciones a ésta ya existían algoritmos resistentes a los computadores cuánticos. Prueba de ello es el criptosistema de McEliece que se publicó en 1978 el cual con parámetros adecuados puede resistir ataques cuánticos a pesar de sus desventajas de tamaño.

Como punto final tenemos que la criptografía post-cuántica viene acompañada, como en cualquier otro tipo de criptografía, con su parte complementaria de criptoanálisis basada en el análisis de un escenario en el que el atacante tiene acceso a todas las capacidades que pueda ofrecer un computador cuántico.

Bibliografía

Academia Sinica. (2014). Recuperado el 2018, de http://www.iis.sinica.edu.tw/papers/byyang/18988-F.pdf

Ajtai, M. (1996). Generating Hard Instances of Lattice Problems. Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing.

Alwen, J., & Peikert, C. (13 de Julio de 2010). Obtenido de https://link.springer.com/article/10.1007%2Fs00224-010-9278-3

An, H., Choi, R., Lee, J., & Kim, K. (23-26 de Junio de 2018). Obtenido de https://pdfs.semanticscholar.org/7ef2/2ef120fd8dcc64a7e1865b99b02e37ad3a48.pdf

Aumasson, J.-P., & Guillaume, E. (29 de Noviembre de 2017). Obtenido de https://github.com/gravity-postquantum/gravity-sphincs/blob/master/Supporting_Documentation/submission.pdf

Australian Cybersecurity Magazine. (11 de Junio de 2018). Obtenido de https://australiancybersecuritymagazine.com.au/quantum-safe-cryptology-and-cryptocurrencies/

Bennett, C., Brassard, G., Breidbart, S., & Wiesner, S. (1983). Obtenido de https://link.springer.com/chapter/10.1007/978-1-4757-0602-4_26

Berkelamp, E., McEliece, R., & van Tilborg, H. (1978). En *On the inherent intractability of certain coding problems. IEEE Transactions on Information Theory, 24(3)* (págs. 384-386).

Bernstein, D. J. (7 de Agosto de 2008). Obtenido de https://cr.yp.to/codes/mceliece-20080807.pdf

Bernstein, D. J. (2 de Febrero de 2015). Obtenido de https://sphincs.cr.yp.to/sphincs-20150202.pdf

Bernstein, D. J., & Lange, T. (18 de Julio de 2018). *PQCrypto*. Obtenido de https://pqcrypto.org/

Bolkema, J. (1 de Mayo de 2017). Obtenido de https://arxiv.org/abs/1612.05085

Bos, J. W. (Septiembre de 2016). *joppebos.com*. Obtenido de http://joppebos.com/presentations/ULB2016.pdf

Brassard, G. (17 de Octubre de 2005). *Arxiv*. Obtenido de https://arxiv.org/pdf/quant-ph/0604072.pdf

Brassard, G., & Crépeau, C. (31 de Julio de 1996). *ResearchGate*. Obtenido de https://www.researchgate.net/publication/220556114_25_years_of_quantum_cryptography

Castryck, W. (31 de Mayo de 2017). Obtenido de https://www.esat.kuleuven.be/cosic/elliptic-curves-are-quantum-dead-long-live-elliptic-curves/

Centrum Wiskunde & Informatica. (s.f.). Obtenido de https://www.cwi.nl/research/groups/cryptology

Chang, L. (14 de Octubre de 2017). *Betanews*. Obtenido de https://betanews.com/2017/10/13/current-encryption-vs-quantum-computers/

CORDIS. (2015). Obtenido de https://cordis.europa.eu/project/rcn/194240_en.html

CORDIS. (2015). Obtenido de https://cordis.europa.eu/project/rcn/194347_en.html

CORDIS. (2018). Obtenido de https://cordis.europa.eu/project/rcn/213162_en.html

Cummins, H. (29 de Junio de 2018). *InfoQ*. Obtenido de https://www.infoq.com/articles/quantum-computing-intro-one

de Wolf, R. (14 de Diciembre de 2017). Obtenido de https://arxiv.org/pdf/1712.05380.pdf

Denver University. (2012). Obtenido de http://www-math.ucdenver.edu/~wcherowi/courses/m5410/ctcmcel.html

Ding, J., & Yang, B.-Y. (2009). Obtenido de http://www.moscito.org/by-publ/recent/pqc.pdf

E., O., & M., F. (2015). Advances in Cryptology - EUROCRYPT 2015. Lecture Notes in Computer Science, vol 9056. Berlin, Heidelberg: Springer.

EURAXESS. (2018). Obtenido de https://cdn3.euraxess.org/jobs/303272

Gao, Y. &. (Abril de 2018). Obtenido de https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8340794

Garber, D. (Junio de 2007). Obtenido de http://www.ims.nus.edu.sg/Programs/braids/files/david_tut1.pdf

Garcia-Morchon, O. (19 de Octubre de 2016). Obtenido de https://www.nist.gov/sites/default/files/documents/2016/10/19/garcia-morchon-paper-lwc2016.pdf

Gisin, N., Ribordy, G., Tittel, W., & Zbinden, H. (1 de Febrero de 2008). Obtenido de https://arxiv.org/pdf/quant-ph/0101098.pdf

Goldreich, O., Shafi, G., & Halevi, S. (1997). Advances in Cryptology - CRYPTO '97. Lecture Notes in Computer Science, vol. 1294. Springer-Verlag.

Goubin, L., Patarin, J., & Yang, B.-Y. (2011). Multivariate Cryptosystems, Encyclopedia of Cryptography and Security. Springer. Obtenido de http://precision.moscito.org/by-publ/recent/00421-multivariate.pdf

Hoffstein, J., Pipher, J., & Silverman, J. H. (21 de Junio de 1998). Obtenido de https://link.springer.com/chapter/10.1007/BFb0054868

Hoffstein, J., Pipher, J., & Silverman, J. H. (21 de Junio de 1998). Obtenido de https://link.springer.com/chapter/10.1007/BFb0054868

Holden, J. (27 de Diciembre de 2017). *Nautilus*. Obtenido de http://nautil.us/blog/-how-classical-cryptography-will-survive-quantum-computers

Hudde, H. C. (Marzo de 2013). Obtenido de https://www.emsec.rub.de/media/attachments/files/2013/03/mastersthesis-hudde-code-based-cryptography-library.pdf

IACR. (2018). Obtenido de https://www.iacr.org/jobs/

IAD. (19 de Agosto de 2015). Obtenido de https://www.iad.gov/iad/programs/iad-initiatives/cnsa-suite.cfm

IBM. (2017). *QuantumExperience*. Obtenido de https://quantumexperience.ng.bluemix.net/proxy/tutorial/full-user-guide/004-Quantum_Algorithms/110-Shor%27s_algorithm.html

Infineon. (30 de Mayo de 2017). Obtenido de https://www.infineon.com/cms/en/about-infineon/press/press-releases/2017/INFCCS201705-056.html

INRIA. (s.f.). Obtenido de https://www.inria.fr/en/teams/secret

IRIF. (s.f.). Obtenido de https://www.irif.fr/en/index

ISARA. (8 de Noviembre de 2017). Obtenido de https://www.isara.com/cryptographic-certificates-quantum-safe/

ISARA Corporation. (7 de Septiembre de 2017). *prnewswire*. Obtenido de https://www.prnewswire.com/news-releases/isara-sets-new-international-standard-for-quantum-safe-security-300515858.html?tc=eml_cleartime

JFLI. (s.f.). Obtenido de http://jfli.cnrs.fr/

Johansson, R., & Strahl, T. (8 de Junio de 2016). *http://lup.lub.lu.se*. Obtenido de http://lup.lub.lu.se/luur/download?func=downloadFile&recordOId=8878692&fileOId=8878700

Junghee, R. (26 de Octubre de 2017). *Medium*. Obtenido de https://medium.com/@bbod/why-post-quantum-encryption-in-blockchain-97a6b8f906a8

Kelvin. (14 de Abril de 2018). Obtenido de https://klvn.io/assets/wp.pdf

Ko, K. (11 de Agosto de 2000). Obtenido de https://www.iacr.org/archive/crypto2000/18800166/18800166.pdf

Lamport, L. (18 de Octubre de 1979). Obtenido de https://www.microsoft.com/en-us/research/uploads/prod/2016/12/Constructing-Digital-Signatures-from-a-One-Way-Function.pdf

Lange, T. (19 de Marzo de 2018). Obtenido de https://www.youtube.com/watch?v=ePhHFtAOi6A

Linde, J. (2018). Obtenido de https://www.icmat.es/Thesis/2018/Tesis_Jorge_Linde.pdf

Lomonaco, S. (8 de Noviembre de 1998). Obtenido de https://arxiv.org/pdf/quant-ph/9811056.pdf

Majulab. (s.f.). Obtenido de http://majulab.cnrs.fr

Mathematical Institute. (s.f.). Obtenido de https://www.maths.ox.ac.uk/groups/cryptography

McEliece, R. J. (Febrero de 1978). Obtenido de https://ipnpr.jpl.nasa.gov/progress_report2/42-44/44N.PDF

Merkle, R. (Junio de 1979). Obtenido de http://www.merkle.com/papers/Thesis1979.pdf

Mohamed, M. S., & Petzoldt, A. (2016). Obtenido de https://eprint.iacr.org/2016/911.pdf

Montanaro, A. (25 de Noviembre de 2015). Obtenido de https://people.maths.bris.ac.uk/~csxam/teaching/history.pdf

Nederlande Organisatie Voor Toegepast Natuurwetenschappelijk Onderzoek Tno. (s.f.). Obtenido de https://www.nwo.nl/en/research-and-results/research-projects/i/62/31862.html

Niederreiter, H. (1986). Knapsack-type cryptosystems and algebraic coding theory. Problems of Control and Information Theory. Obtenido de http://real-j.mtak.hu/7997/1/MTA_ProblemsOfControl_15.pdf

Nieto, M. (18 de Abril de 2018). *BlogThinkBig*. Obtenido de https://blogthinkbig.com/computacion-cuantica-record

NIST. (Abril de 2016). Obtenido de http://dx.doi.org/10.6028/NIST.IR.8105

NIST. (3 de Enero de 2017). Recuperado el Agosto de 2018, de https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-1-Submissions

NIST. (2018). Obtenido de https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/history-pqc-round-1-updates.pdf

Noor-ul-Ain, Atta-ur-Rahman, Nadeem, M., & Ghafoor Abbasi, A. (Noviembre de 2015). Obtenido de https://www.researchgate.net/publication/282972925_Quantum_Cryptography_Trends_A_Milestone_in_Information_Security

OpenQuantumSafe. (4 de Enero de 2018). Obtenido de https://github.com/open-quantum-safe/liboqs

Peikert, C. (2013). Obtenido de https://web.eecs.umich.edu/~cpeikert/pubs/slides-abit2.pdf

Peikert, C. (16 de Julio de 2014). Obtenido de https://eprint.iacr.org/2014/070.pdf

PQCRYPTO. (1 de Abril de 2015). Obtenido de https://pqcrypto.eu.org/WP1.html

PQCRYPTO. (7 de Septiembre de 2015). Obtenido de https://pqcrypto.eu.org/docs/initial-recommendations.pdf

Quantum Communications Hub. (s.f.). Obtenido de https://www.quantumcommshub.net/news/phd-studentships-quantum-communications-post-post-quantum-cryptography-projects/

Regev, O. (2 de Mayo de 2009). Obtenido de https://cims.nyu.edu/~regev/papers/qcrypto.pdf

Royal Holloway. (s.f.). Obtenido de https://intranet.royalholloway.ac.uk/isg/research/research-themes/quantum-safe-cryptography.aspx

Rui, X., Chi, C., Yue, Q., & Tao, J. (13 de Mayo de 2018). Obtenido de https://arxiv.org/pdf/1805.04880.pdf

Saarinen, M.-J. O. (10 de Noviembre de 2016). Obtenido de https://eprint.iacr.org/2016/1058

SAFECrypto. (2018). Obtenido de https://www.safecrypto.eu/pqclounge/

Samardjiska, S. (2017). Obtenido de http://riot-os.org/files/RIOT-Summit-2017-slides/3-4-Security-session-Simona.pdf

Schwartz, M. J. (22 de Febrero de 2017). *Bank Info Security*. Obtenido de https://www.bankinfosecurity.com/quantum-crypto-dont-do-anything-a-9737

Shor, P. (1994). Obtenido de https://www.computer.org/csdl/proceedings/focs/1994/6580/00/0365700.pdf

Shor, P. (25 de Junio de 1996). Obtenido de https://arxiv.org/pdf/quant-ph/9508027.pdf

Soto Velázquez, J. A. (18 de Diciembre de 2017). Obtenido de https://courses.cs.ut.ee/MTAT.07.022/2017_fall/uploads/Main/antonio-report-f17.pdf

Stebila, D. (30 de Abril de 2017). Obtenido de https://s3.amazonaws.com/files.douglas.stebila.ca/files/research/presentations/20170430-TLSDIV.pdf

Stichting Nederlandse Wetenschappelijk Onderzoek Instituten. (s.f.). Obtenido de https://www.ncsc.nl/english/current-topics/factsheets/factsheet-post-quantum-cryptography.html

Suomalainen, J., Kotelba, A., Kreku, J., & Lehtonen, S. (7 de Febrero de 2018). Obtenido de www.mdpi.com/2410-387X/2/1/5/pdf

Universidad Católica de Lovaina. (s.f.). Obtenido de https://www.esat.kuleuven.be/cosic/

Universidad de Florida del Sur. (s.f.). Obtenido de http://www.math.usf.edu/

Universidad de Innsbruck. (2018). Obtenido de https://arxiv.org/pdf/1711.11092.pdf

Universidad de Luxemburgo. (s.f.). Obtenido de https://wwwfr.uni.lu/snt/research/apsia/projects

Universidad de Radboud. (s.f.). Obtenido de https://www.ru.nl/ds/

Universidad de Surrey. (Marzo de 2018). Obtenido de https://www.surrey.ac.uk/fees-and-funding/studentships/exploring-post-quantum-cryptography-block-chain-technology

Universidad Ruhr de Bochum. (s.f.). Obtenido de https://www.seceng.rub.de/research/projects/pqc/

Universidad Técnica de Darmstadt. (s.f.). Obtenido de https://www.informatik.tu-darmstadt.de/cdc/home_cdc/index.en.jsp

Universidad Técnica de Eindhoven. (s.f.). Obtenido de https://www.tue.nl/en/research/research-groups/center-for-quantum-materials-and-technology-eindhoven/impacts/post-quantum-cryptography/

Yang, B.-Y. (23 de Junio de 2017). Obtenido de https://2017.pqcrypto.org/exec/slides/mpkc-A.pdf