Encriptación post-cuántica

Hecho por:
Mario Ventura
Joan Teruel

Índice

Introducción

Los ordenadores cuánticos, aunque se encuentren en su “infancia”, ponen en riesgo la arquitectura de los sistemas de cifrado más modernos como:

- ECC (Eliptic Curves)
- RSA

Los algoritmos clásicos se basan en problemas computacionalmente tediosos.

Ej.: el algoritmo de Shor (Factorización rápida de números)

Definición de PQC e importancia

Debido a la amenaza de las máquinas cuánticas se han ideado algoritmos y mecanismos que permiten que aún se puedan cifrar comunicaciones de manera segura.

Estos son seguros debido a que no se han encontrado algoritmos cuánticos que los rompan. **No necesitan ejecutarse en máquinas cuánticas.**

También hay protocolos mediante hardware cuántico que serían usados para distintos algoritmos, pero no nos centraremos en ellos.

Definición de PQC e importancia

La PQC desarrolla algoritmos criptográficos resistentes a ataques de computadoras cuánticas con el fin de proteger

- Confidencialidad
- Integridad
- Autenticidad

de la información.

Resulta importante porque se basa en problemas matemáticos difíciles para computadoras clásicas y cuánticas.

Algoritmos y fundamentos matemáticos

Los algoritmos PQC pueden clasificarse según su base matemática de la siguiente forma:

- **Basados en rejillas:** CRYSTALS-Dilithium, CRYSTALS-KYBER, NTRU
- **Basados en códigos:** HQC, McEliece
- **Basados en hash:** SPHINCS+, XMSS
- **Multivariados:** Rainbow
- **Basados en isogenias**: SIKE (roto en 2022, no recomendado)

Relevancia del NIST

La organización NIST, la misma que estandarizó AES o SHA, tiene un rol muy importante en la estandarización de algoritmos PQC (además de otras SDO) (IETF, X9, ISO, ITU-T…).

El NIST es responsable de:

- Estandarizar algunos algoritmos de PQC tales como CRYSTALS-KYBER, CRYSTALS-Dilithium, SPHINCS+, y HQC.
- Lanzar competiciones abiertas al público en su "Post-Quantum Cryptography Standardization Project", cualquiera puede participar.

Proceso de estandarización

- El proyecto del NIST se inició en 2016 con el fin de encontrar algoritmos públicos que sean resistentes a los *Quantum Computers*.
- En 2022 se seleccionan CRYSTALS-KYBER, Dilithium, SPHINCS+, y en 2025 se añade HQC como apoyo por su variedad matemática.
- Esto facilita la generación de interoperabilidad y la adopción internacional de estos algoritmos.

Algoritmos PQC basados en ‘lattices’

Algoritmos basados en rejillas (lattices):

- Basados en estructuras matemáticas de vectores dónde estos forman bases. El problema más conocido es SVP (Shortest Vector Problem) donde se debe aproximar la longitud Euclidiana mínima.
- No todos problemas basados en lattices son criptográficamente seguros.

Ejemplos:

- **CRYSTALS-Dilithium**
- **CRYSTALS-Kyber**

Algoritmos Post-Cuánticos no basados en ‘lattices’

- Muchos de los algoritmos post-cuánticos que se intentan solventar usan nuevos métodos como **Firma digital multivariada-cuadrática**
- Otros se basan en la fiabilidad y conocimiento de problemas existentes, como la **Firma mediante Hash o basada en código de corrección**

Ejemplos:

- **SPHINCS+**
- **HQC**

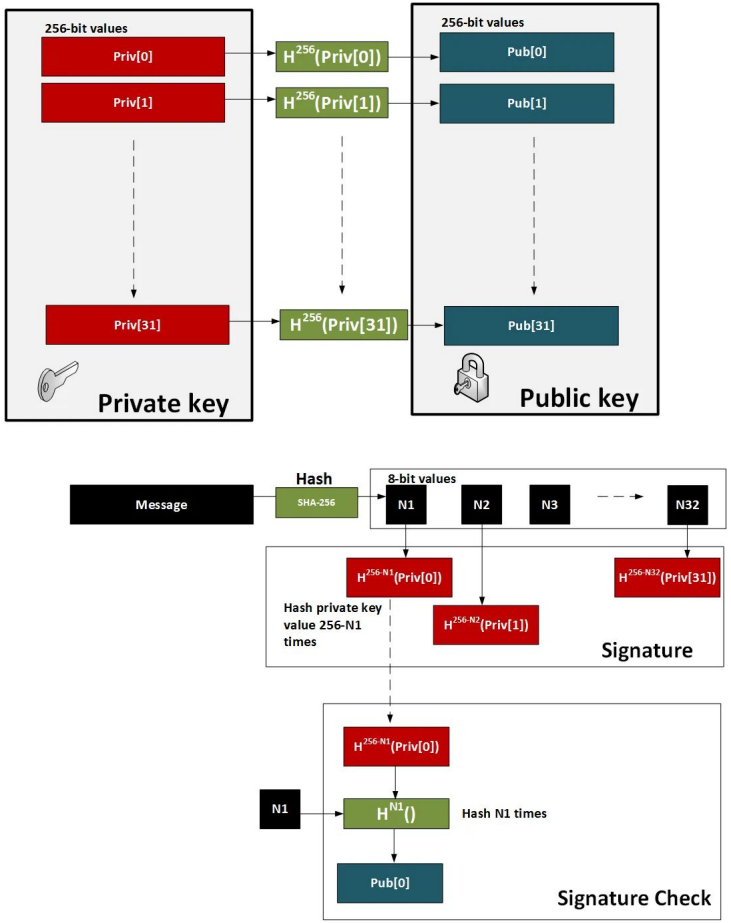
SPHINCS+

Presentado en 2015. Uno de los cuatro algoritmos recomendados por NIST desde 2022 y adscrito al “Post Quantum Cryptography Project”

- **Propósito:** Firma digital sin estado basada en hash.
- **Base:** Funciones hash.
- **Uso:** Típicamente, autenticación.
- **Ventajas:** Claves pública/privada pequeñas
- **Desventajas:** Firmas grandes (41 KB), tiempo alto de firma/verificación.

Aún por estandarizar

SPHINCS+

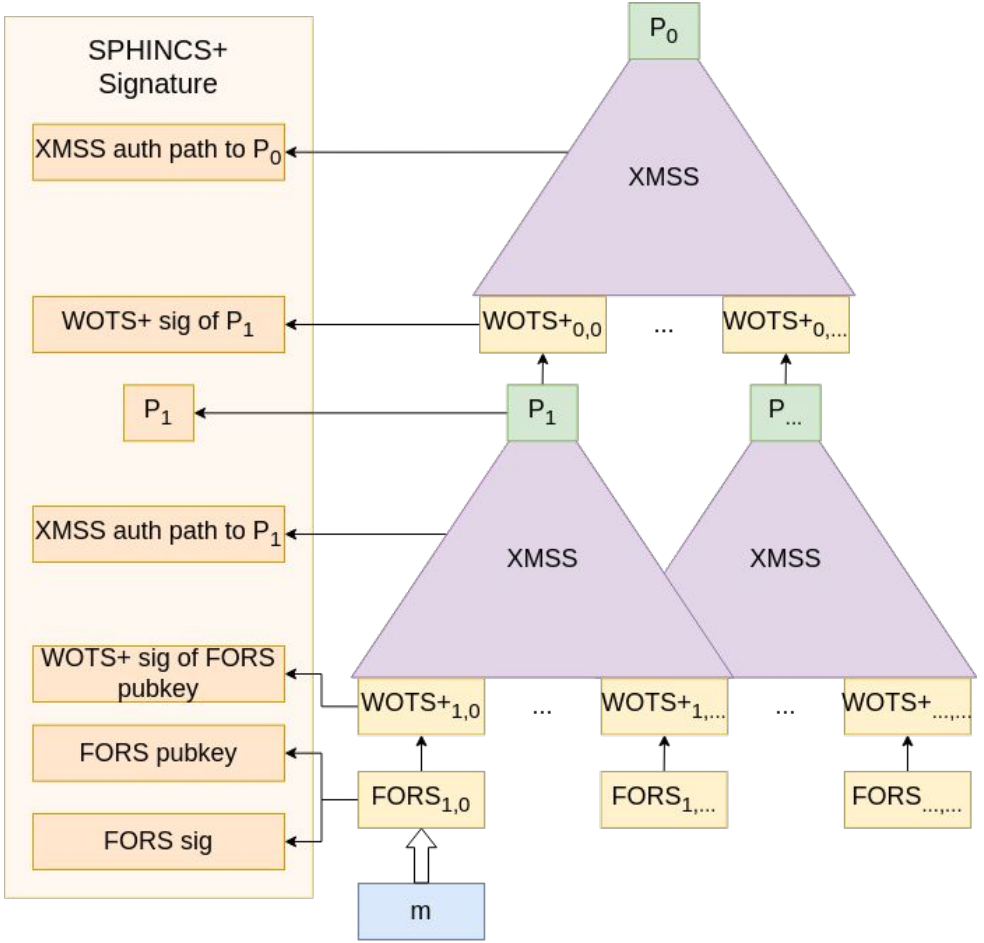

256-bit values
Priv[0]
Priv[1]
Priv[31]
$H^{256}(Priv[0])$
$H^{256}(Priv[1])$
$H^{256}(Priv[31])$
Private key
256-bit values
Pub[0]
Pub[1]
Pub[31]
Public key
Message
Hash
SHA-256
8-bit values
N1
N2
N3
N32
$H^{256\text{-}N1}(Priv[0])$
$H^{256\text{-}N2}(Priv[1])$
$H^{256\text{-}N32}(Priv[31])$
Hash private key value 256-N1 times
Signature
$H^{256\text{-}N1}(Priv[0])$
N1
$H^{N1}()$
Hash N1 times
Pub[0]
Signature Check

SPHINCS+

SPHINCS+ usa otro algoritmo llamado XMSS que es stateful, pero al estar contenido dentro de los árboles SPHINCS mantiene la propiedad de stateless.

- One-Time Signing implica que el camino usado para generar las claves ya no puede ser usado.
- Se garantiza con semillas provenientes de el mensaje y la clave pública con algoritmos pseudo-aleatorios.

SPHINCS+


SPHINCS+ Signature
XMSS auth path to P_0
WOTS+ sig of P_1
P_1
XMSS auth path to P_1
WOTS+ sig of FORS pubkey
FORS pubkey
FORS sig
P_0
XMSS
$WOTS+_{0,0}$
...
$WOTS+_{0,...}$
P_1
$P_{...}$
XMSS
XMSS
$WOTS+_{1,0}$
...
$WOTS+_{1,...}$
...
$WOTS+_{...,...}$
$FORS_{1,0}$
$FORS_{1,...}$
$FORS_{...,...}$
m

HQC

Hamming Quasi-Cyclic. Algoritmo basado en códigos de corrección de errores de códigos cuasi-cíclicos de la métrica de Hamming.

- **Propósito:** KEM basado en códigos de corrección de errores.
- **Base:** Decodificación de síndromes en códigos cuasi-cíclicos.
- **Uso:** Respaldo para KYBER, VPNs, TLS, almacenamiento de datos sensibles, etc.
- **Ventajas:** Robusto contra ataques cuánticos, base matemática diversa.
- **Desventajas:** Claves/cifrados grandes (ej., 4.5 KB para HQC-128), menos eficiente que KYBER.

Aún por estandarizar

HQC

Escogido en 2025 por el NIST como **apoyo para CRYSTALS-KYBER** por ofrecer una alternativa matemática. Basado en 'Syndrome-Decoding', complicado para computadores convencionales y cuánticos.

3 pasos:

1. **Generación de claves:** Genera una clave pública y una privada.
2. **Encapsulación:** Usar clave pública del receptor para crear cifrado que contiene clave secreta.
3. **Desencapsulación:** Receptor recibe cifrado y usa clave privada para recuperar clave original.

HQC

Recuperación de claves pública y privada

Algorithm 1 KeyGen

Input: parameters

$\mathbf{h} \xleftarrow{\$} \mathcal{R}$

$(\mathbf{x}, \mathbf{y}) \xleftarrow{\$} \mathcal{R}_\omega^2$

$s_k \leftarrow (\mathbf{x}, \mathbf{y})$

$\mathbf{s} \leftarrow \mathbf{x} + \mathbf{hy}$

$p_k \leftarrow (\mathbf{h}, \mathbf{s})$

return (s_k, p_k)

Otros algoritmos

Existen otros esquemas criptográficos que han sido evaluados o están en desarrollo dentro del ámbito de la criptografía post-cuántica (PQC).

Destacan:

- **XMSS**
- **McEliece**
- **Rainbow** (Descartado desde 2022 por vulneraciones)

Otros algoritmos: XMSS

XMSS (eXtended Merkle Signature Scheme) es un algoritmo de firma digital basado en **funciones hash con estado**.

- **Tipo:** Firma digital con estado basada en hash.
- **Uso:** Sistemas embebidos, blockchains.
- **Ventajas:** Seguridad probada, estandarizado por NIST (SP 800-208).
- **Desventajas:** Requiere gestión de estado, firmas grandes.

Otros algoritmos: McEliece

Esquema de encriptación basado en códigos lineales algebraicos como los códigos Goppa binarios.

- **Tipo:** Encriptación basada en códigos lineales (Goppa).
- **Uso:** Potencial para aplicaciones específicas.
- **Ventajas:** Resistente a ataques cuánticos, eficiente en cifrado/descifrado.
- **Desventajas:** Claves públicas grandes (ej., 261,120 bytes), no estandarizado por NIST, poco práctico para aplicaciones generales

Otros algoritmos: Rainbow

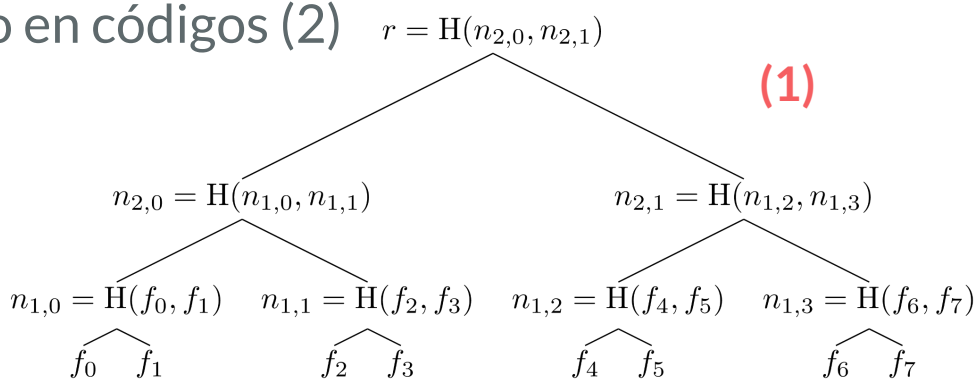
Algoritmo de firma digital basado en criptografía multivariada, específicamente en el esquema Oil-Vinegar desbalanceado.

- **Tipo:** Firma digital multivariada (Oil-Vinegar).
- **Uso:** Estudio académico tras vulnerabilidad descubierta en 2022.
- **Ventajas:** Firmas pequeñas, eficiente antes de la ruptura.
- **Desventajas:** No recomendado debido a ataque que recupera claves.

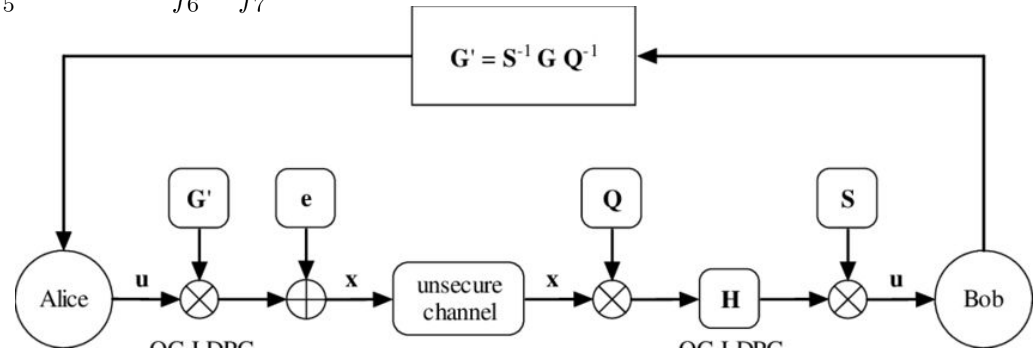
Otros algoritmos

Firma digital con estados basado en hash (1) y McEliece cryptosystem basado en códigos (2)

(1)



(2)



Casos de uso

La PQC tiene aplicaciones en varios sectores en los que la seguridad de los datos y comunicaciones es de vital importancia. Algunos ejemplos son:

1. Protección de Comunicaciones en Internet
2. Autenticación y Firmas Digitales
3. Seguridad en Infraestructuras Críticas como la defensa
4. Almacenamiento de Datos Sensibles

Ventajas y desventajas

PQC no basados en lattices presenta numerosas ventajas pero también desventajas respecto a los clásicos.

Ventajas

- Mayor Resistencia a algoritmos cuánticos (Shor, Grover).
- Diversidad matemática reduce riesgos de vulnerabilidades únicas.
- Estandarización (NIST) asegura fiabilidad e interoperabilidad.
- Implementaciones optimizadas para múltiples aplicaciones.

Desventajas

- Mayor costo computacional que métodos clásicos (ej., SPHINCS+, HQC).
- Claves y firmas grandes requieren (más almacenamiento y ancho de banda).
- Incertidumbre matemática deriva en posibles vulnerabilidades futuras en algoritmos nuevos.

Debate:

1. ¿Cual es el riesgo y coste de no adoptar los algoritmos PQC lo antes posible; aumentará la brecha digital al requerir hardware más avanzado, dejando atrás a regiones o sectores con menos recursos?
2. ¿Deberían las empresas y gobiernos adoptar de inmediato esquemas híbridos que combinen PQC con algoritmos pre-cuánticos, o esperar a que los algoritmos PQC estén más maduros?
3. ¿Deberían los algoritmos PQC ser obligatorios en sectores críticos, como la banca o la defensa, incluso si esto implica mayores costos y complejidad?

Debate:

1. ¿Es realista esperar una coordinación global para la transición a PQC, considerando las diferencias en infraestructura tecnológica y prioridades entre países?
2. ¿Qué implicaciones éticas surgen si un país o entidad desarrolla un ordenador cuántico capaz de romper algoritmos tradicionales antes de que PQC esté ampliamente adoptada?
3. ¿Cómo podríamos justificar el alto coste computacional y los grandes tamaños de clave de algoritmos como SPHINCS+ en aplicaciones prácticas, como dispositivos IoT con recursos limitados?

GRACIAS

Requisitos para el “Post-Quantum Cryptography Project”

1. Algoritmos totalmente Open-sourced o hechos públicos para su posterior revisión.
2. No usarán métodos poco seguros respecto a ordenadores cuánticos (ej. factorización o logaritmos discretos).
3. Los algoritmos como mínimo deberán implementar una de estas funcionalidades:
    a. Encriptación con clave pública: Generación de claves para encriptar/desencriptar.
    b. Mecanismos de intercambio de claves: Generación de claves, encapsulación y desencapsulación.
    c. Firmas digitales: Generación de claves, firma y su verificación.
4. Deberán proveerse toda la configuración y parámetros usados para llegar a la seguridad que se indica.