Triển khai mật mã hậu lượng tử trên các thiết bị tích hợp NPU

Người trình bày

Tạ Duy Hoàng

Trường Công nghệ Thông tin và Truyền thông
Đại học Bách khoa Hà Nội

Hà Nội · 2026

NỘI DUNG

Buổi nói này gồm những gì

01

Mật mã hậu lượng tử

- Vì sao mật mã khóa công khai cổ điển sụp đổ
- Thu thập hôm nay, giải mã ngày mai
- Lộ trình và mốc thời hạn toàn cầu

02

Các họ mật mã hậu lượng tử

- Năm giả thiết khó
- Danh mục NIST hiện nay
- Mật mã dựa trên mã

03

HQC

- Cơ chế hoạt động
- Bộ giải mã ghép
- Vai trò, hiện trạng, vấn đề mở

Phần cuối dẫn tới bài báo: giải mã HQC là nút thắt hiệu năng trên nền tảng di động và nhúng.

PHẦN I

Mật mã hậu lượng tử

Vì sao cần nó, và thế giới đã quyết định những gì

- Mối đe dọa lượng tử với mật mã khóa công khai
- Thu thập hôm nay, giải mã ngày mai
- Lộ trình di trú và mốc thời hạn của các quốc gia

Thuật toán lượng tử phá vỡ tầng khóa công khai

Thuật toán Shor (1994)

Phân tích thừa số và logarit rời rạc trong thời gian đa thức trên máy tính lượng tử đủ lớn có sửa lỗi. Mọi nguyên thủy khóa công khai đang dùng rộng rãi — RSA, Diffie–Hellman, ECDH, ECDSA — đều đứng trên đúng hai bài toán đó.

Thuật toán Grover (1996)

Chỉ tăng tốc bậc hai cho tìm kiếm không cấu trúc. Hệ đối xứng sống sót bằng cách tăng gấp đôi biên an toàn — AES-256 và SHA-384 vẫn vững. Tầng bất đối xứng không có lối thoát tương tự.

Mật mã đối xứng được vá. Mật mã bất đối xứng buộc phải thay thế.

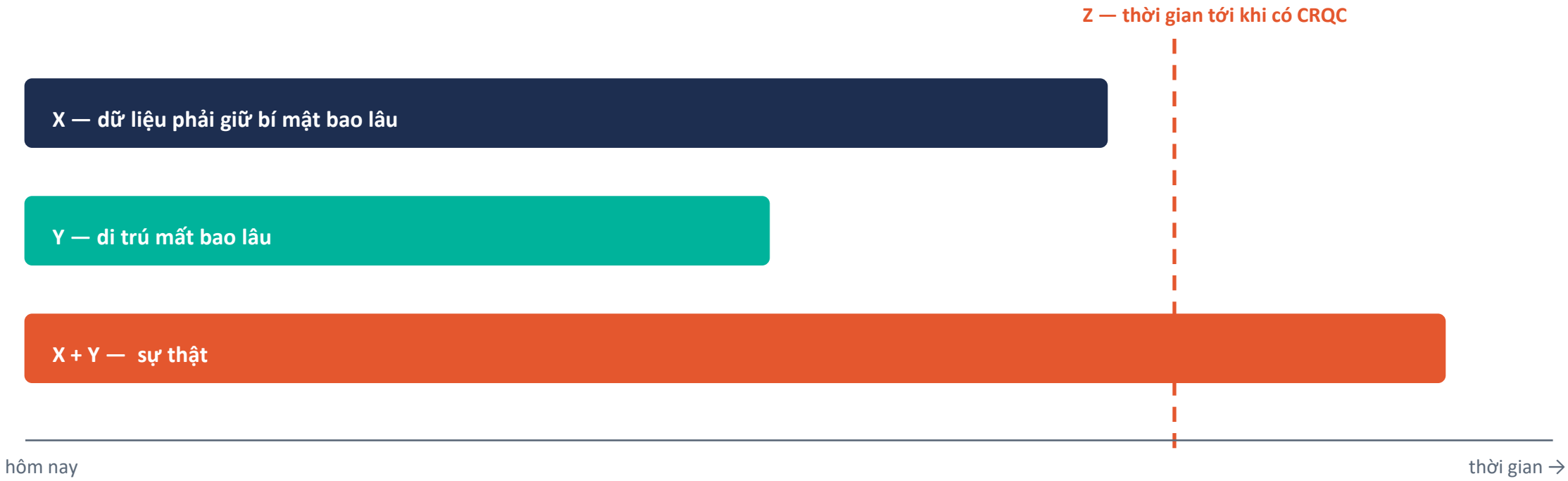
RSA-2048 / RSA-4096	BỊ PHÁ
ECDH · ECDSA · EdDSA	BỊ PHÁ
DH trường hữu hạn · DSA	BỊ PHÁ
AES-128	SUY YẾU
SHA-256 (hàm băm)	AN TOÀN

NIST IR 8547: RSA và mật mã đường cong elliptic sẽ bị loại bỏ dần từ 2030 và bị cấm sau 2035.

1.2 TÍNH CẤP BÁCH

Thu thập hôm nay, giải mã ngày mai

Kẻ tấn công không cần máy tính lượng tử ngay hôm nay. Lưu lượng đã mã hóa và bản mã lưu trữ có thể được ghi lại từ bây giờ, rồi giải mã vào ngày xuất hiện máy tính lượng tử đủ mạnh (CRQC). Mọi bí mật có vòng đời dài vì thế đã bị phơi bày.



Bất đẳng thức Mosca: nếu X + Y > Z, thì dữ liệu bạn bảo vệ hôm nay đã mất an toàn.

Con số điển hình: X = 10–25 năm với hồ sơ y tế, tài chính, quốc phòng, định danh; Y = 5–10 năm cho một tổ chức lớn.

1.3 CHUẨN HÓA

Một thập kỷ tuyển chọn, giờ là một thập kỷ di trú

2016

NIST kêu gọi đề xuất
69 ứng viên hợp lệ

2017–2022

Vòng 1–3
CRYSTALS-Kyber được chọn

8/2024

FIPS 203 / 204 / 205
ML-KEM, ML-DSA, SLH-DSA
chốt

3/2025

HQC được chọn (IR 8545)
ứng viên duy nhất vòng 4

2026

Dự thảo FIPS cho HQC
FIPS 206 (FN-DSA) đang soạn

2027

Dự kiến chốt
chuẩn HQC

HQC là phương án dự phòng, không phải phương án thay thế. NIST nói rõ: các tổ chức nên di trú sang ML-KEM ngay, và giữ HQC như lựa chọn thứ hai độc lập về mặt toán học.

Mọi khu vực pháp lý lớn đều đã có lộ trình kèm mốc thời gian

Khu vực	Văn bản	Các mốc chính
Hoa Kỳ — hệ thống ANQG	NSA CNSA 2.0	Mua sắm mới phải kháng lượng tử từ 1/2027; chỉ dùng PQC từ 2033
Hoa Kỳ — dân sự	NSM-10, OMB M-23-02, NIST IR 8547	RSA / ECC loại bỏ dần 2030, cấm hẳn 2035
Liên minh châu Âu	Lộ trình NIS Cooperation Group (2025)	Kế hoạch quốc gia cuối 2026; hệ thống rủi ro cao 2030; hoàn tất 2035
Vương quốc Anh	Lộ trình di trú- chuyển đổi NCSC (2025)	Kiểm kê và lập kế hoạch tới 2028; hệ thống ưu tiên 2031; hoàn tất 2035
Đức	Hướng dẫn kỹ thuật BSI	Hạ tầng trọng yếu 2030; bắt buộc triển khai hybrid
Úc	Hướng dẫn ASD	Cấm thuật toán khóa công khai cổ điển sau 2030
Canada	Lộ trình CCCS (2025)	Hệ thống chính phủ không mật hoàn tất trước 2035

Hai mốc lặp lại gần như ở mọi nơi: 2030 cho hệ thống rủi ro cao, 2035 cho phần còn lại.

1.5 DI TRÚ

Quá trình chuyển đổi diễn ra như thế nào

1

Kiểm kê

Lập danh mục mật mã (CBOM): mọi khóa, chứng chỉ và giao thức đang nằm ở đâu.

2

Đánh giá

Xếp hạng theo vòng đời dữ liệu và mức phơi nhiễm; bí mật sống lâu đi trước.

3

Lập kế hoạch

Thiết kế để linh hoạt thuật toán: thuật toán là cấu hình, không phải mã cứng.

4

Chuyển đổi

Triển khai chế độ hybrid (cổ điển + PQC) để một thành phần hỏng vẫn còn thành phần kia.

5

Loại bỏ

Gỡ thuật toán khóa công khai cổ điển, kiểm định và theo dõi hướng dẫn mới.

Linh hoạt thuật toán mới là đích thực sự

Tham số và thuật toán sẽ còn thay đổi: tham số ML-KEM tăng lên, HQC tới năm 2027, rồi kết quả vòng chữ ký bổ sung. Hệ thống thay được nguyên thủy mà không phải thiết kế lại sẽ hấp thụ được tất cả.

Hiệu năng là rào cản thực tế

Khóa và bản mã PQC lớn hơn, phép toán nặng hơn — đặc biệt trên thiết bị di động, IoT và nhúng. Đây chính là chỗ mà nghiên cứu cài đặt như bài báo này có ý nghĩa.

PHẦN II

Các họ mật mã hậu lượng tử

Năm giả thiết khó — và vì sao hệ dựa trên mã lại quan trọng

- Lattice, mã, hàm băm, đa biến, isogeny
- Danh mục NIST tại thời điểm hiện tại
- Mật mã dựa trên mã và bài toán giải mã hội chứng

2.1 TOÀN CẢNH

Năm giả thiết khó độc lập nhau

Dựa trên lattice

Bài toán khó: Learning With Errors, NTRU, SIS

Hệ mật: ML-KEM, ML-DSA, FN-DSA

Đã chuẩn hóa — lựa chọn chính

Dựa trên mã

Bài toán khó: Giải mã mã tuyến tính ngẫu nhiên

Hệ mật: HQC, BIKE, Classic McEliece

HQC được chọn tháng 3/2025

Dựa trên hàm băm

Bài toán khó: Kháng va chạm / kháng tiền ảnh

Hệ mật: SLH-DSA, XMSS, LMS

Đã chuẩn hóa — chỉ cho chữ ký

Đa biến

Bài toán khó: Giải hệ phương trình bậc hai nhiều biến

Hệ mật: MAYO, UOV, SNOVA

Đang trong vòng chữ ký bổ sung

Dựa trên isogeny

Bài toán khó: Vành tự đồng cấu của đường cong elliptic

Hệ mật: SQIsign

Vòng bổ sung; SIDH/SIKE bị phá 2022

Đa dạng hóa mới là mấu chốt

SIKE bị phá trong một cuối tuần trên máy tính cá nhân năm 2022, sau chín năm được cộng đồng soi kỹ. Một danh mục dựa trên một giả thiết duy nhất là một điểm chết duy nhất — đó chính là lập luận để chuẩn hóa một KEM dựa trên mã bên cạnh KEM dựa trên lattice.

Mật mã đối xứng (AES, SHA-2/3) không nằm trong danh sách này: nó vốn đã kháng lượng tử khi tăng gấp đôi tham số.

Danh mục hậu lượng tử của NIST

Chuẩn	Thuật toán	Họ	Chức năng	Trạng thái
FIPS 203	ML-KEM (Kyber)	Lattice	Đóng gói khóa	**Đã chốt — 8/2024**
FIPS 204	ML-DSA (Dilithium)	Lattice	Chữ ký số	**Đã chốt — 8/2024**
FIPS 205	SLH-DSA (SPHINCS+)	Hàm băm	Chữ ký số	**Đã chốt — 8/2024**
FIPS 206	FN-DSA (Falcon)	Lattice	Chữ ký số	**Đang soạn dự thảo**
FIPS 2xx	HQC	Mã	Đóng gói khóa	**Chọn 3/2025 · dự thảo 2026 · chốt 2027**

Chỉ một KEM có phương án dự phòng

Chữ ký đã có ba chuẩn trên hai giả thiết. Trao đổi khóa thì chỉ dựa vào lattice — HQC lấp đúng khoảng trống đó.

Vòng bốn, một ứng viên sống sót

Vòng 4 xét BIKE, HQC và Classic McEliece. NIST chuẩn hóa HQC; hai ứng viên còn lại không được chọn.

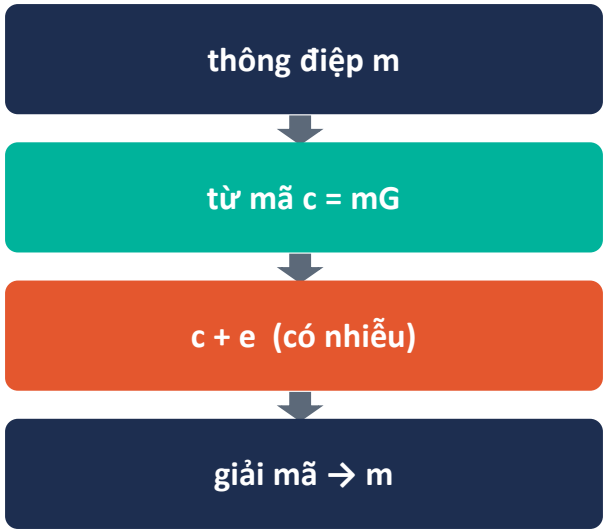
An toàn đến từ độ khó của bài toán giải mã

Bài toán giải mã hội chứng

cho H và lấy ngẫu nhiên, s, t
tìm e sao cho $H \cdot e^T = s$ và $wt(e) \leq t$

- NP-khó trong trường hợp tổng quát (Berlekamp, McEliece, van Tilborg, 1978).
- Tấn công tốt nhất là information-set decoding — độ phức tạp hàm mũ, Grover chỉ cải thiện được bậc hai.
- Gần 50 năm phân tích kể từ hệ McEliece 1978, chưa có đột phá cấu trúc nào với giả thiết mã ngẫu nhiên.

Cấu trúc tổng quát



mã hóa công khai

cố ý thêm lỗi

cửa sập bí mật:
một bộ giải mã hiệu quả

Đánh đổi: hệ dựa trên mã có được một giả thiết rất được hiểu rõ và số học chủ yếu là XOR nên rất nhanh — đổi lại khóa công khai hoặc bản mã lớn, và một bộ giải mã bắt buộc phải chạy trong thời gian hằng.

2.4 VÒNG 4

Ba KEM dựa trên mã — một được chọn

Hệ mật	Họ mã	Khóa công khai	Bản mã	Kết quả
Classic McEliece	Mã Goppa nhị phân	261 KB	128 B	Không được chọn — khóa quá lớn cho hầu hết giao thức
BIKE	Mã QC-MDPC	1,5 KB	1,6 KB	Không được chọn — phân tích xác suất giải mã sai chưa đủ chín
HQC	Mã QC ngẫu nhiên + bộ giải mã ghép công khai	2,2 KB	4,3 KB	**ĐƯỢC CHỌN, tháng 3/2025**

Vì sao HQC được chọn

Quy dẫn

An toàn quy về giải mã mã tựa tuần hoàn ngẫu nhiên — không có cấu trúc mã ẩn để tấn công.

Phân tích được

Xác suất giải mã sai được chứng minh nhỏ hơn $2^{-\lambda}$, điều BIKE không đạt được chặt bằng.

Thực dụng

Kích thước khóa và bản mã cân bằng, số học nhanh và đã được hiểu rõ.

PHẦN III

Hamming Quasi-Cyclic (HQC)

HQC là gì, dùng để làm gì, và đang đứng ở đâu

- Cấu trúc: một mã ghép công khai
- Giải mã Reed-Solomon và Reed-Muller nhân bản
- Vai trò, hiện trạng và khoảng trống cài đặt

3.1 ĐỊNH NGHĨA

HQC là gì

1 Một KEM dựa trên mã

Đề xuất bởi Aguilar-Melchor, Aragon, Bettaieb, Bidoux, Blazy, Deneuville, Gaborit và cộng sự (2016); một hệ mã hóa khóa công khai IND-CPA được biến thành KEM IND-CCA2 nhờ biến đổi HHK.

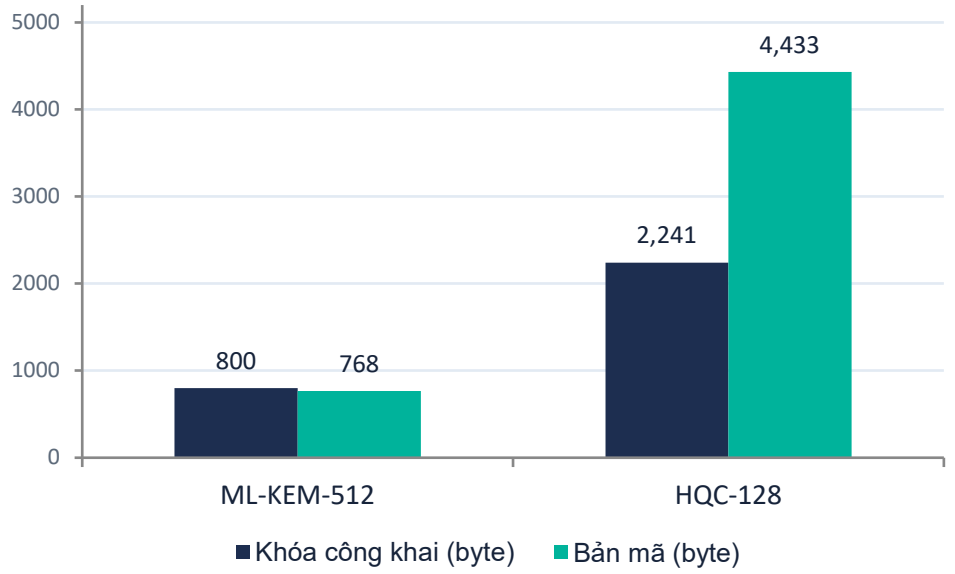
2 Không giấu mã

Khác với McEliece, mã sửa sai ở đây là công khai và cố định. An toàn chỉ đến từ bài toán giải mã hội chứng tựa tuần hoàn — không có gì được giấu trong cấu trúc mã.

3 Xác suất lỗi chứng minh được

Tham số được chọn sao cho xác suất giải mã sai chắc chắn nhỏ hơn $2^{-\lambda}$, đúng như chứng minh IND-CCA2 đòi hỏi.

Kích thước một lần trao đổi khóa (byte)



HQC-128 truyền khoảng 6,7 KB mỗi lần trao đổi so với 1,6 KB của ML-KEM-512 — cái giá của việc đa dạng hóa giả thiết.

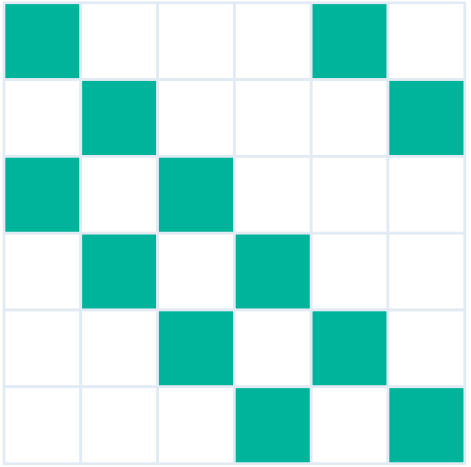
Kích thước theo đặc tả HQC ngày 22/8/2025 (khóa đóng gói, bản mã).

Mọi thứ diễn ra bên trong một vành tuần hoàn

$$R = F_2[X] / (X^n - 1)$$

- Vector nhị phân độ dài n được đọc như một đa thức; tích u · v là tích chập vòng, tính bằng Karatsuba và Toom-Cook.
- Một vector h duy nhất mô tả trọn ma trận vòng rot(h) cỡ n × n. Nhờ vậy, công bố một mã trông ngẫu nhiên chỉ tốn n bit thay vì n^2 — đây chính là lý do khóa nhỏ.
- n được chọn là số nguyên tố nguyên thủy, nên $X^n - 1$ chỉ có hai nhân tử bất khả quy. Điều này chặn các tấn công cấu trúc khai thác tính tựa tuần hoàn.

rot(h) với h = (1, 0, 1, 0, 0, 0)



mỗi hàng là hàng phía trên dịch đi một vị trí

Giả thiết khó

Giải mã hội chứng tựa tuần hoàn (QCSD): cho h và s = x + h · y, với x, y là các vector bí mật trọng số thấp ω, hãy tìm lại (x, y). Chứng minh IND-CPA của HQC quy đúng về các biến thể quyết định 2-QCSD và 3-QCSD của bài toán này — không có gì được giấu trong bản thân mã.

Ba thuật toán, và vị trí của bộ giải mã

Sinh khóa

$h \leftarrow R$ ngẫu nhiên đều
$x, y \leftarrow R$ trọng số ω
$s = x + h \cdot y$

$ek = (h, s)$
$dk = y$

Mã hóa

$r_1, r_2 \leftarrow$ trọng số ω_r
$e \leftarrow$ trọng số ω_e
$u = r_1 + h \cdot r_2$
$v = \text{Encode}(m) + s \cdot r_2 + e$

$ct = (u, v)$

Giải mã

$v - u \cdot y$
$= \text{Encode}(m) + e'$

$m = \text{Decode}(v - u \cdot y)$

e' nhỏ $\Rightarrow$ khôi phục m

Toàn bộ lập luận an toàn dồn về một bước

Decapsulation trước hết khôi phục từ mã có nhiễu $v - u \cdot y$ qua một phép nhân đa thức thưa, rồi giao nó cho bộ giải mã ghép. Bộ giải mã phải khử nhiễu đúng với xác suất ít nhất $1 - 2^{-\lambda}$, trong thời gian hằng, trên chính thiết bị đang chạy handshake. Bước giải mã đó là chủ đề của bài báo này.

Vì sao bộ giải mã chỉ nhìn thấy một lỗi nhỏ

$$
\begin{aligned}
v - u \cdot y &= (\mathrm{Encode}(m) + s \cdot r_2 + e) - (r_1 + h \cdot r_2) \cdot y \\
&= \mathrm{Encode}(m) + (x + h \cdot y) \cdot r_2 + e - r_1 \cdot y - h \cdot r_2 \cdot y \\
&= \mathrm{Encode}(m) + x \cdot r_2 - r_1 \cdot y + e
\end{aligned}
$$

Vành giao hoán nên hai hạng tử $h \cdot r_2 \cdot y$ triệt tiêu — và cùng với chúng là toàn bộ khóa công khai.

Phần còn lại là vector lỗi $e' = x \cdot r_2 - r_1 \cdot y + e$. Mọi hạng tử đều là tích của hai vector thưa, nên e' vẫn cách xa phân bố đều. Giải mã thành công đúng khi $\omega(e') \le \Delta$, tức khả năng sửa lỗi của mã ghép — và phép kiểm tra đó là thứ duy nhất đứng giữa bản mã và thông điệp.

≈ 34 %

Tỉ lệ lật bit hiệu dụng $p\star$ mà mỗi tọa độ của e' phải chịu ở HQC-128. Mã ghép phải giải được một kênh nhiễu tới mức đó.

DFR < 2^{-128}

ω, ω_r, n_1 và n_2 được chọn sao cho xác suất giải mã sai nằm dưới mức an toàn — một chặn có chứng minh, không phải số đo.

Giải mã thất bại không chỉ là mất một thông điệp. Dưới tấn công bản mã chọn lọc, việc quan sát các lần thất bại làm rò rỉ thông tin về bí mật (x, y) — đó là lý do HQC ấn định xác suất lỗi ngay từ thiết kế, thay vì coi nó là một chi tiết kỹ thuật.

Từ mã hóa tới đóng gói khóa

Đóng gói khóa — bên gửi

1. Sinh thông điệp ngẫu nhiên m 128 bit và salt 16 byte.
2. Dẫn xuất $(K, \theta) = G(H(ek) \| m \| salt)$ bằng SHA3-512.
3. Mã hóa m, dùng θ làm mầm sinh r_1, r_2 và e.
4. Gửi (u, v, salt). Khóa chung chính là K.

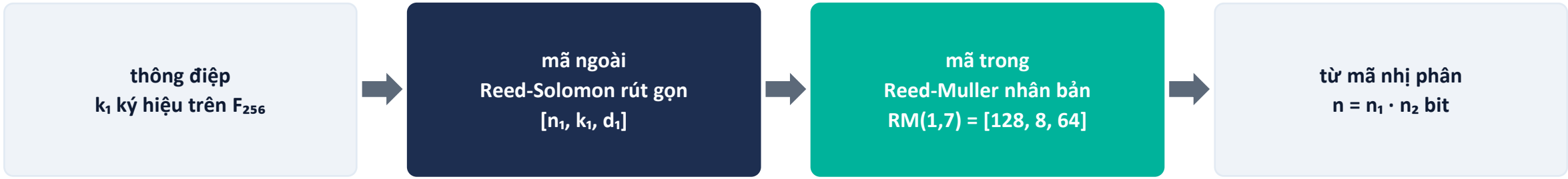
Mở gói khóa — bên nhận

1. Giải mã: $m' = Decode(v - u \cdot y)$ — bộ giải mã chạy ở đây.
2. Tính lại $(K', \theta') = G(H(ek) \| m' \| salt)$.
3. Mã hóa lại m′ với θ′ rồi so sánh với bản mã nhận được.
4. Nếu lệch, trả về khóa giả ngẫu nhiên dẫn xuất từ bí mật σ.

Mã hóa lại chính là thứ mua được an toàn IND-CCA2

Đây là biến đổi Fujisaki-Okamoto có salt với từ chối ngầm: một bản mã hỏng sẽ cho ra một khóa mà kẻ tấn công không phân biệt được với khóa thật, thay vì một thông báo lỗi. Nó cũng giải thích yêu cầu thời gian hằng — mọi chênh lệch thời gian giữa chấp nhận và từ chối đều là một oracle bản mã chọn lọc, đúng như tấn công rejection-sampling năm 2022 lên HQC và BIKE đã cho thấy.

Cấu trúc ghép Reed-Solomon / Reed-Muller



Mỗi ký hiệu Reed-Solomon trên F_{256} được ánh xạ thành một từ mã Reed-Muller, sau đó nhân bản (bội 3 hoặc 5) để tăng khả năng chịu nhiễu.

Tham số	HQC-128	HQC-192	HQC-256
Độ dài Reed-Solomon n_1	46	56	90
Bội nhân bản Reed-Muller	3	5	5
Khối RM nhân bản n_2	384	640	640
Số lỗi RS sửa được δ	15	16	29
Số vector HVX cho Chien search	1	1	2

Vì sao cấu trúc này quan trọng

Giải mã Reed-Muller là một biến đổi Hadamard nhanh rồi chọn đỉnh; giải mã Reed-Solomon là syndrome, Berlekamp-Massey và tìm nghiệm. Cả hai đều thao tác trên vector số nguyên và F_2^8 đều đặn, độ dài cố định — đúng cấu trúc mà một engine SIMD rộng được sinh ra để xử lý.

Chuỗi xử lý giải mã của HQC

Mã trong — Reed-Muller nhân bản

1. Chia từ mã có nhiễu thành các khối RM
2. Gộp các bản lặp thành vector độ tin cậy
3. Biến đổi Hadamard nhanh (7 tầng bướm)
4. Chọn đỉnh với quy tắc phá hòa cố định

Mã ngoài — Reed-Solomon rút gọn

5. Tính 2δ syndrome trên F_2^8
6. Berlekamp-Massey → đa thức định vị lỗi
7. Tìm nghiệm trên support rút gọn
8. Tính độ lớn lỗi và sửa từ mã

Các kernel này chiếm phần lớn chi phí decapsulation, và tất cả đều là tính toán đều đặn, song song theo lane — đó là quan sát mà bài báo dựa vào.

3.8 THAM SỐ

HQC-128 qua các con số

Tham số	HQC-128
Độ dài Reed-Solomon n_1	46
Độ dài Reed-Muller nhân bản n_2	384
Từ mã ghép $n_1 n_2$	17.664 bit
Độ dài vành n (nguyên tố nguyên thủy)	17.669
Độ dài thông điệp k	128 bit
Trọng số ω của bí mật x, y	66
Trọng số $\omega_r = \omega_e$ của r_1, r_2, e	75
Khả năng sửa lỗi Reed-Solomon δ	15
Xác suất giải mã sai	$< 2^{-128}$

5 bit dư ($n - n_1 n_2$) bị cắt bỏ; n là số nguyên tố nguyên thủy để chặn các tấn công cấu trúc.

Kích thước

Khóa đóng gói 2.241 B · bản mã 4.433 B · khóa chung 32 B. Khóa mở gói có thể lưu chỉ bằng một seed 32 byte.

Chi phí ba phép toán

Bản C tham chiếu, đơn vị kilocycle: KeyGen 4.557 · Encaps 9.116 · Decaps 13.918. Với AVX2: 76 · 150 · 353.

Decapsulation là phép đắt nhất

Nó tốn khoảng 1,5–2 lần một lần encapsulation, và bộ giải mã ghép nằm ngay bên trong — đúng kernel mà bài báo đưa xuống backend vector.

3.9 HIỆN TẠI

Vai trò của HQC và vị trí hiện nay

Vai trò: bảo hiểm thuật toán

HQC là phương án dự phòng được chỉ định cho ML-KEM trong trao đổi khóa. Nó không nhằm thay thế ML-KEM, và NIST khuyến nghị các tổ chức triển khai ML-KEM ngay. Giá trị của nó nằm ở chỗ một cú phá vỡ giả thiết lattice sẽ không chạm tới nó.

Đã được triển khai ở đâu

Đã có trong liboqs và Bouncy Castle, và xuất hiện trong các thử nghiệm hybrid X25519 + HQC. Mức độ áp dụng còn hạn chế khi chuẩn vẫn ở dạng dự thảo.

Hiện trạng và các vấn đề kỹ thuật còn mở

Chuẩn hóa

Được chọn tháng 3/2025 (NIST IR 8545). Dự thảo FIPS dự kiến 2026, chốt năm 2027.

Chi phí

Bản mã lớn hơn ML-KEM khoảng 5–6 lần và decapsulation nặng hơn rõ rệt.

Kênh phụ

Tấn công định thời đã nhắm vào mã tham chiếu (Guo và cộng sự, TCHES 2022). Lấy mẫu và giải mã thời gian hằng là bắt buộc.

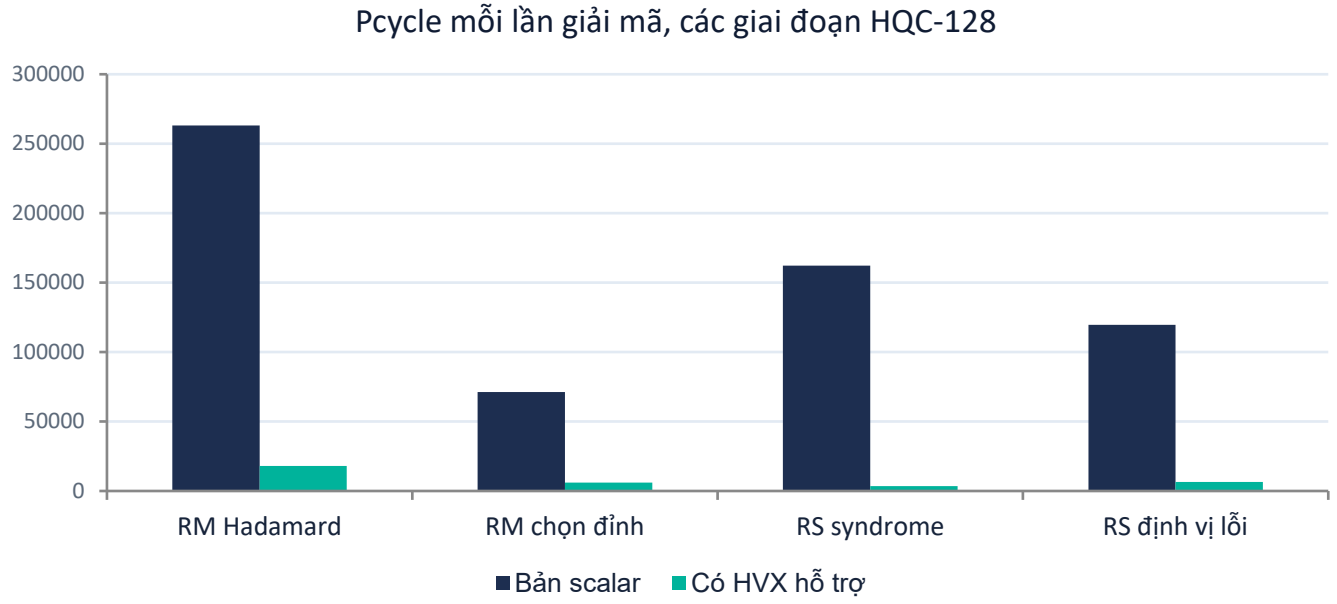
Nền tảng

Đã có cài đặt tối ưu cho Cortex-M4, FPGA và RISC-V SoC — nhưng chưa có cho bộ tăng tốc vector trong điện thoại hiện đại.

3.10 BÀI BÁO NÀY

Giải mã HQC trên thiết bị tích hợp NPU

Điện thoại hiện đại đã sẵn có một engine SIMD rộng — Hexagon Vector eXtensions (HVX) của Qualcomm nằm trong khối NPU. Bài báo thiết kế lại các kernel giải mã HQC tốn kém nhất theo mô hình thực thi theo lane của nó, đồng thời giữ kết quả bằng bit với bộ giải mã scalar tham chiếu.



23,00× tăng tốc trên mô phỏng cho một lần giải mã HQC-128 đầy đủ
953.763 → 41.471 Pcycle

2,07× cải thiện độ trễ thực đo trên kit phát triển Snapdragon 8 Gen 2

18,13× hiệu quả năng lượng mỗi lần giải mã, hơn 99% việc của CPU được chuyển đi

Lưu ý: các con số là cho giai đoạn giải mã dưới lời gọi FastRPC theo lô, không phải decapsulation đầu-cuối, và backend chưa đạt thời gian hằng.

Vu Minh Chau, Nguyen Ngoc Kiet, Pham Quang Minh, Mai Xuan Ngoc, Nguyen Duc Anh, Hoang Ta — arXiv:2606.01968.

Tổng kết

1 **Nguy cơ**

Thu thập hôm nay - giải mã ngày mai khiến việc chuyển đổi thành bài toán của hiện tại; 2030 và 2035 là hai mốc quyết định.

2 **Đa dạng hóa là có chủ đích**

HQC tồn tại để việc trao đổi khóa không phụ thuộc riêng vào lattice.

3 **Khoảng trống nằm ở cài đặt**

HQC đã chuẩn hóa trên giấy; làm cho nó rẻ trên thiết bị người ta thực sự mang theo vẫn là việc còn mở — và bộ tăng tốc vector là câu trả lời chưa được khai thác.

Xin cảm ơn!

Tạ Duy Hoàng · SoICT, HUST