National Cyber Security Centre

Introduction to Quantum Safe Cryptography

ENISA

September 2018

Introduction

- This talk will introduce the mathematical background of the most popular PQC primitives
  - Code-based
  - Lattice-based
  - Multivariate
  - Isogenies
  - (Hash-based signatures)

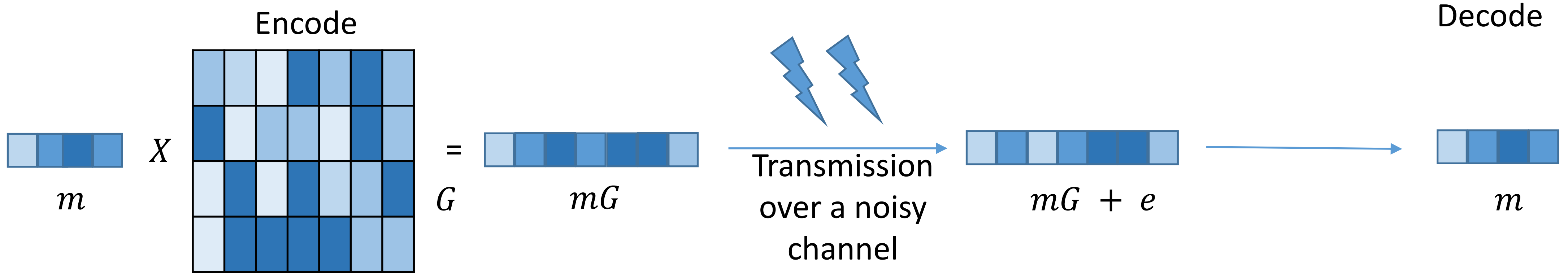
- Other talks will cover how to turn these primitives into real-world protocols, implementation considerations, etc.

- Basic ideas is to move away from “number theoretic” PKC constructions (factoring / discrete log) into other areas of algebraic or discrete mathematics to evade Shor’s algorithm

National Cyber Security Centre

Code-based protocols

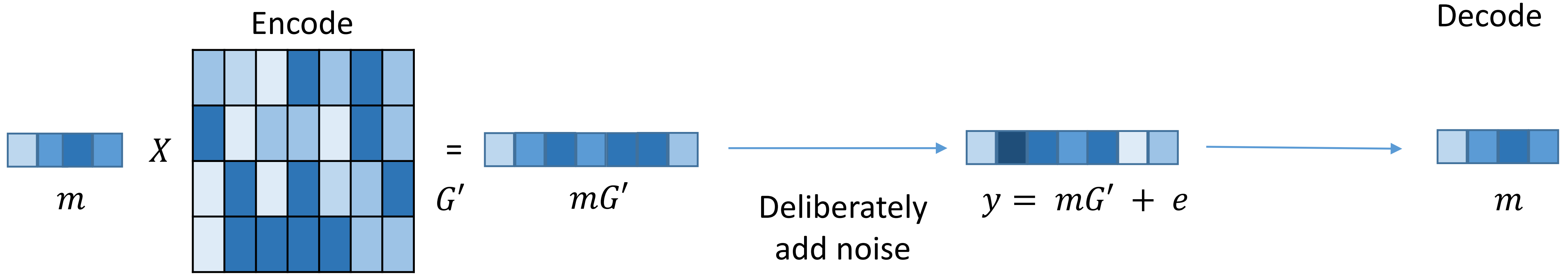
Linear codes in communications



- Mathematical model (Shannon 1948):
  - **Raw data** m is a k-long vector in F_q^k
  - Encode m into an n-long vector by multiplying by a $k \times n$ **code generator matrix** G over F_q^k
  - Transmit **codeword** mG. The channel **may** introduce noise, so assume we receive $y = mG + e$
  - **Decode** the received vector y to recover m
- Want a **code** with good transmission rate k/n and efficient decoder (up to some reasonable noise limit)

National Cyber Security Centre

Linear codes in cryptography



- Mathematical model (McEliece 1978):
  - **Raw data** m is a k-long vector in F_q^k
  - Encode m into an n-long vector by multiplying by a $k \times n$ **public key** matrix G' over F_q^k
  - Deliberately **add** noise, so we transmit and receive **ciphertext** $y = mG' + e$
  - **Decrypt** the received ciphertext y using a **private key** to recover m
- Want a **code** with good transmission rate, an efficient decoder <u>and</u> that is hard to distinguish from random

National Cyber Security Centre

Niederreiter scheme (1986)

A widely-used dual variant of (McEliece 1978) with equivalent security but better efficiency

Public key is $H' = SHP$

Private key is the decomposition of H' into
- A **code parity check matrix** H for a code $C = \{v \in F_q^n : Hv^T = 0\}$
- An invertible linear **scrambling matrix** S and a **permutation matrix** P, used to **hide** H

Together with an efficient **decoder** for code C

Initiator

To encrypt message to a chosen user with **public** key H':

- Encode **message** (error) as a vector m of weight $\leq w$
- **Ciphertext** (syndrome) $c = H'm^T$

$$\xrightarrow{s}$$

Receiver

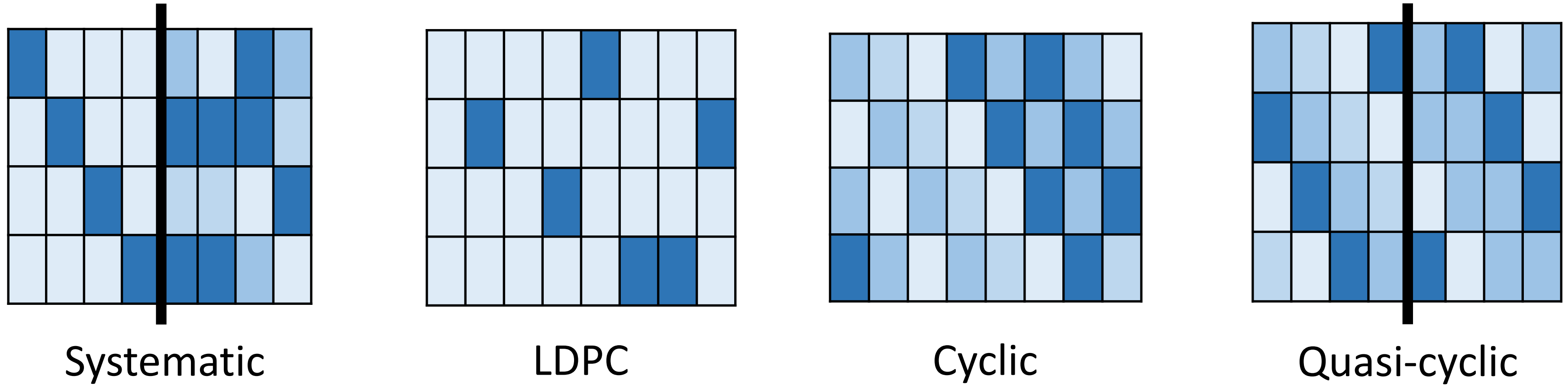
Use **private** S, H, P to recover m from s:

- Unscramble by multiplying c by S^{-1}
- Use the **decoder** to recover Pm^T
- Unpermute to recover **message** m

National Cyber Security Centre

Public key size

- Basic coding schemes have very large public keys and ciphertexts
- Many proposals to reduce the key size
  - Use different codes. Good codes for communications are often bad choices for crypto
  - Alternative algebraic properties to make attacks harder e.g. Rank-metric codes
  - Systematic forms or structured matrices



- It is important for us to understand the security properties of new proposals
  - Do they resist the standard attacks?
  - Can they be distinguished from random?
  - Do the security proofs and assumptions stand?

National Cyber Security Centre

Classical Attacks

- Security foundation: The general decoding problem (random H, G) is NP-hard (Berlekamp, McEliece, van Tilborg 1978) but efficient decoders exist for the classes of codes used in practical applications

- Private key recovery – deduce S, H, P from H'
    - Distinguishers – properties that differ from random
    - Equivalent codes – support splitting attack

- Message recovery and signature forgery attacks
    - Information Set Decoding (ISD)
    - Finding low-weight codewords

- Attacks on structured matrices
    - Groebner/linearization (see later) on several McEliece variants with compact keys
    - ISD attacks on LDPC proposals with too sparse public keys

National Cyber Security Centre

Information set decoding

- ISD algorithms find low weight vectors formed from rows of H’

  1. Permute columns $H'P = (A|B)$
  2. Check whether A is invertible
  3. If so compute $M = A^{-1}H' = (I_{n-k}|Q)$
  4. Look for low weight $e = vM$ (linear combinations of rows of M)
  5. If successful
     return $e' = vMP^{-1}$
  6. Else return to 1.

(Description from Perlner, 2014)

- Original ISD algorithm (Prange, 1962) chooses random permutations. Many subsequent refinements due to Lee-Brickell, Leon, Stern-Dumer, Bernstein-Lange-Peters, May-Ozerov and many others

- All ISD variants have exponential complexity = $O\left(2^{c(R,w)n}\right)$. For (Prange, 1962) $c = 0.1207$ and (May-Ozerov, 2015) $c = 0.0966$.

National Cyber Security Centre

Quantum Attacks

- (Bernstein 2009) showed how Grover's quantum search algorithm would significantly speed up the ISD search for k error-free coordinates. Reduced complexity exponent to $c = 0.06035$

- (Kachigar-Tillich, 2017 and Kirshanova, 2018) have recently reduced c to 0.058-0.059 by additionally incorporating a quantum random walk and nearest neighbor decoding

- Since all ISD variants have exponential complexity = $O\left(2^{c(R,w)n}\right)$ ISD may be defeated by increasing parameter sizes

- Grover speeds-ups are (approximately) quadratic and so the rule of thumb is that to retain the same level of security against a quantum computer the dimension n of the code needs to be doubled. For unstructured (binary Goppa) codes this corresponds to doubling the length of the ciphertext and quadrupling the size of the public key

National Cyber Security Centre

Summary and open questions

- McEliece’s original proposal to use binary Goppa codes is still considered very secure against both classical and quantum attacks

- But these have very large public keys (Megabytes) which can be difficult to integrate into communications protocols.

- Much work has gone into investigating more compact variants. Many compact schemes have been broken. QC-MDPC variants currently in favour. Rank metric proposals deserve more study

- Signatures have been problematic. Many code-based signatures have been proposed and broken. **All** submissions to NIST are broken
  - Some new research ideas and constructions but nothing solid yet

National Cyber Security Centre

Lattice-based protocols

Ring-LWE schemes

R-LWE uses correspondences between cyclic matrices ↔ vectors ↔ polynomials over F_q to gain efficiency in both storage and arithmetic

$$\begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{bmatrix} \leftrightarrow [1,2,3] \leftrightarrow 1 + 2x + 3x^2$$

R-LWE protocols are based on simple equations like $B = sA + e$ over polynomial rings $R = \frac{\mathbb{Z}[x]}{x^n \pm 1}$ where A and B are public values, s is the private key and e is a private error vector/polynomial

Generating polynomials is a more complicated process than in McEliece. A has uniform independent coefficients mod q but the private polynomials s and e have small coefficients sampled from more complicated probability distributions (e.g. a discrete Gaussian)

Alternatives schemes based on uniform distributions, rounding schemes and module-LWE variants have been submitted to NIST

National Cyber Security Centre

R-LWE key exchange

Idea: “Noisy Diffie-Hellman.” Public keys contain small noise terms. Values V_i , V_r will differ by a small amount V_r - V_i = $s_r e_i$ - $s_i e_r$ - e'_r and a reconciliation process is required to extract matching private keys

Initiator (i)

Generate small private key s_i

Generate small private noise e_i

Compute public key $B_i = s_i A + e_i$

$$\xrightarrow{B_i}$$

Responder (r)

Generate small private key s_r

Generate small private noise e_r

Compute public key $B_r = s_r A + e_r$

Generate small private noise e'_r

Compute value $V_r = s_r B_i + e'_r$

Compute check field $C = Check(V_r)$

$$\xleftarrow{B_r,\, C}$$

Initiator (i)

Compute value $V_i = s_i B_r$

Extract private key $K = Rec(V_i, C)$

Responder (r)

Extract private key $K = Extract(V_r)$

National Cyber Security Centre

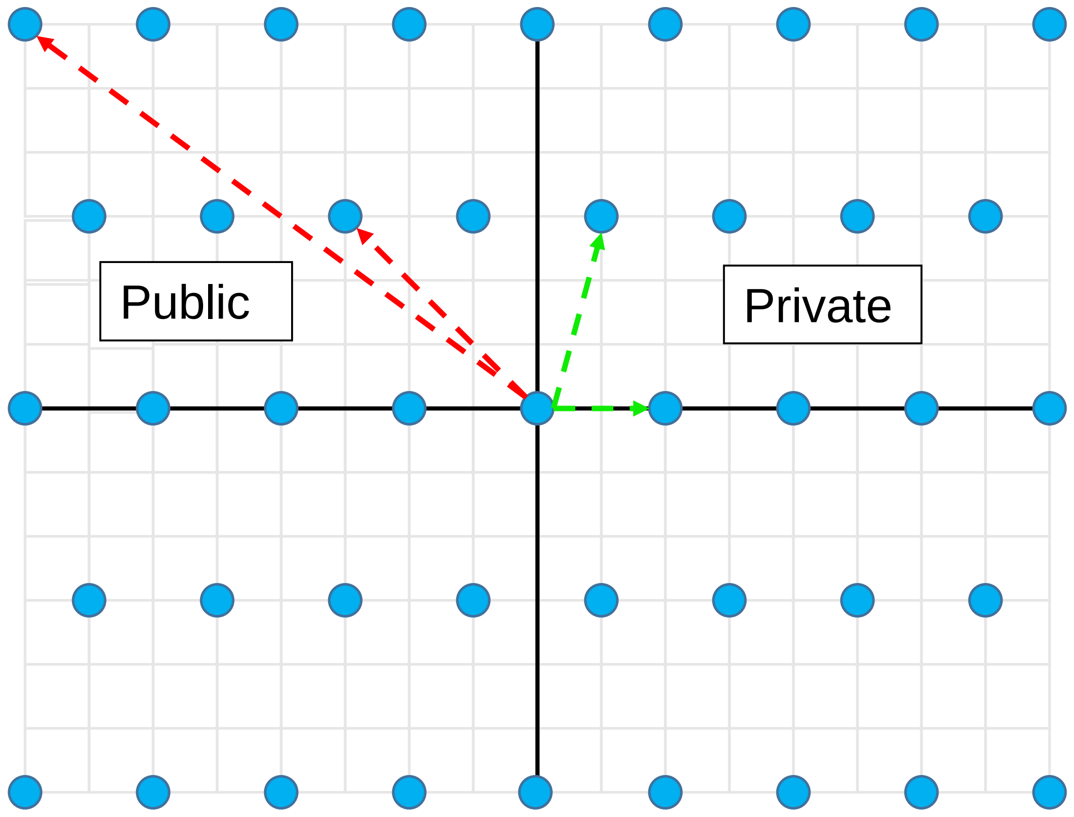
Generic Attacks: Lattices

Can often represent crypt problems in terms of a lattice

Lattice basis L = $\begin{bmatrix} 2 & 0 \\ 1 & 3 \end{bmatrix} \approx \begin{bmatrix} -3 & 3 \\ -8 & 6 \end{bmatrix}$

For LWE and NTRU use L = $\begin{bmatrix} qI & 0 \\ M & I \end{bmatrix}$ where M is formed from the public key and system parameters

Short vector found in (row span) of the reduced basis. Close vectors found e.g. via rounding



National Cyber Security Centre

Classical Attacks

- Security foundation: One of the cited advantages of lattice-based cryptography is the existence of worst-case to average-case reductions. It is known that certain cryptographic problems including R-LWE are as hard on average as well known NP-hard lattice problems (SVP) are in the worst case

- Attacks can often be presented as a short or close vector problem in a lattice
  - Lattice Basis reduction
  - Sieving
  - Enumeration

- Leaky signatures
  - Each signature is a vector close to the lattice
  - Collecting lots of signatures may reveal information about the shape of the lattice

National Cyber Security Centre

Lattice sieve

(Ajtai–Kumar–Sivakumar 2001, Nguyen–Vidick 2008)

- Form an initial list of many long lattice vectors e.g. by sampling from a discrete Gaussian distribution over the lattice
- Take all pairwise combinations of vectors in the input list. Form a new list from any of the sums or differences which are shorter than the original vectors.
- Iterate to produce new (smaller) lists of ever shorter lattice vectors until no improvement (exact SVP) or short enough (approximate SVP)

Many improvements and variants due to Micciancio–Voulgaris, Laarhoven, Herold-Kirshanova, Gama-Nguyen-Regev, Ducas and others. (Laarhoven-Mariano 2018) claims complexity $O\left(2^{0.42n}\right)$ for exact SVP

Quantum Attacks

- (Laarhoven, Mosca, van de Pol, 2014) describes several algorithms which combine lattice sieving with Grover's algorithm to solve SVP with claimed optimal complexity $O\left(2^{0.18n}\right)$ and heuristic complexity $O\left(2^{0.29n}\right)$

- (SOLILOQUY, 2014) outlined a new quantum attack idea on a lattice-based cryptosystem. However this addressed a very special case and did not threaten most general systems

National Cyber Security Centre

Summary and open questions

- Lattices offer flexibility – supports both practical key exchanges and signatures, plus high function cryptography like IBE, ABE and homomorphic encryption
- Reasonable parameter sizes – LWE tens of Kilobytes, R-LWE and M-LWE a few Kilobytes
- Some implementation details (Gaussian sampling) can be complex to implement. Alternatives based on uniform distributions, rounding schemes and module-LWE variants are under investigation
- Still some uncertainty around costing lattice attacks and hence choosing key sizes
- How does algebraic structure affect security?

National Cyber Security Centre

Multivariate quadratics

National Cyber Security Centre

MQ Signatures

MQ signature schemes are much more popular than key exchange schemes. A generic scheme:

Public Key is a vector of m equations in n unknowns

$$P(x_1, \dots, x_n) = \left(p_1(x_1, \dots, x_n), \dots, p_m(x_1, \dots, x_n)\right)$$

Private Key is the decomposition of P into a composition of a structured invertible quadratic map F with invertible linear maps to hide the structure

$$P(x_1, \dots, x_n) = L_2 \circ F \circ L_1(x_1, \dots, x_n)$$

To sign message M

Compute $Hash(M) = (h_1, \dots, h_m)$

Compute $P^{-1}(h_1, \dots, h_m) = (y_1, \dots, y_m)$. Only the valid user can do this

Send M and $(y_1, \dots, y_m)$

To verify

Anyone can verify whether $(h_1, \dots, h_m) = P(y_1, \dots, y_m)$

Structured inner polynomials

- The inner polynomials in NIST candidates **Ge*M*SS** and **Gui** look like

$$F(X, v_1, \dots, v_v) = \sum_{\substack{0 \le i < j \le n-1 \\ 2^{i+j} \le D}} A_{i,j} X^{2^{i+j}} + \sum_{\substack{0 \le i \le n-1 \\ 2^i \le D}} \beta_i(v_1, \dots, v_v) X^{2^i} + \gamma(v_1, \dots, v_v)$$

where $A_{i,j} \in \mathbb{F}_{2^n}$, $\beta_i : \mathbb{F}_2{}^v \to \mathbb{F}_{2^n}$ is a linear map and $\gamma: \mathbb{F}_2{}^v \to \mathbb{F}_{2^n}$ is a quadratic map

- For any fixed assignment of the v_i, F becomes a single-variable polynomial over $\mathbb{F}_{2^n}$

$$F'(X) = \sum_{\substack{0 \le i \le j \le n-1 \\ 2^{i+j} \le D}} A_{i,j} X^{2^{i+j}} + \sum_{\substack{0 \le i \le n-1 \\ 2^i \le D}} B_i X^{2^i} + C \in \mathbb{F}_{2^n}[X],$$

where $A_{i,j}, B_i, C \in \mathbb{F}_{2^n}$. The roots of $F'(X)$ can then be found via standard algorithms

Classical Attacks

- Security foundation: Solving random systems of (quadratic) multivariate polynomial equations with $m \sim n$ is NP-hard
  - MQ signature schemes generally use an underdetermined system, with $m < n < m^2$

- Try to attack structure
  - Early HFE systems had very low rank inner polynomials like $\sum c_i x^{q^{s(i)}+q^{t(i)}}$
  - Associated matrices defined a solvable system of n^2 quadratic equations in rn variables
  - MINRANK: Given a collection of matrices can find a linear combination of minimal rank (Kipnis-Shamir, 1999) and (Courtois, 2000)

- Head-on attacks used in cryptography are mainly based on linearization or Groebner basis techniques which are exponential in complexity

National Cyber Security Centre

Linearization

- Introduce variables for each monomial in the quadratic system. Construct a Macaulay matrix and solve the associated linear algebra problem. E.g. for $p_1 = x_1^2 - x_2$, $p_2 = x_1 - 2$, lin deg $D = 2$

$$\begin{array}{c|cccccc} & 1 & x_1 & x_2 & x_1^2 & x_1x_2 & x_2^2 \\ \hline p_1 & 0 & 0 & -1 & 1 & 0 & 0 \\ p_2 & -2 & 1 & 0 & 0 & 0 & 0 \\ x_1p_2 & 0 & -2 & 0 & 1 & 0 & 0 \\ x_2p_2 & 0 & 0 & -2 & 0 & 1 & 0 \end{array}$$

- Current state of the art for linearization (mqchallenge.org)
  - Hybrid XL solved 74 variables / 148 equations challenge in 18 hours on a CPU cluster
  - Parallel Crossbred solved 74 / 148 challenge in 32 hours on GPUs

Quantum Attacks

- The security of UOV and Rainbow signatures is determined by the cost of finding a collision in the hash function or attacking the system using linearization / Gröbner basis techniques, neither of which appears to be improved by quantum algorithms

- (Chen and Gao, 2017) observe that the HHL quantum algorithm can solve certain (well conditioned) systems of linear equations. They give a quantum linearization algorithm with claimed exponential speedup for solving sparse boolean quadratic systems - if the associated Macaulay matrix has small condition number

National Cyber Security Centre

Summary and open questions

- MQ systems are efficient, using only simple operations (matrices and vectors) and arithmetic over small fields
- Can achieve small signatures
- Main practical disadvantage is large keys (Megabytes)
  - **G*e*MSS tries** to minimize the sizes of the public-key and signature whilst **Gui** aims to maximize the efficiency of the signature generation process
  - Interesting new idea from (Beullens, Preneel and Szepieniec, 2018) allows trade offs between public key and signature sizes (not just MQ)
- Key establishment is problematic: **Most** NIST entries already broken
- Structured private maps are essential but can sometimes be exploited in attacks

National Cyber Security Centre

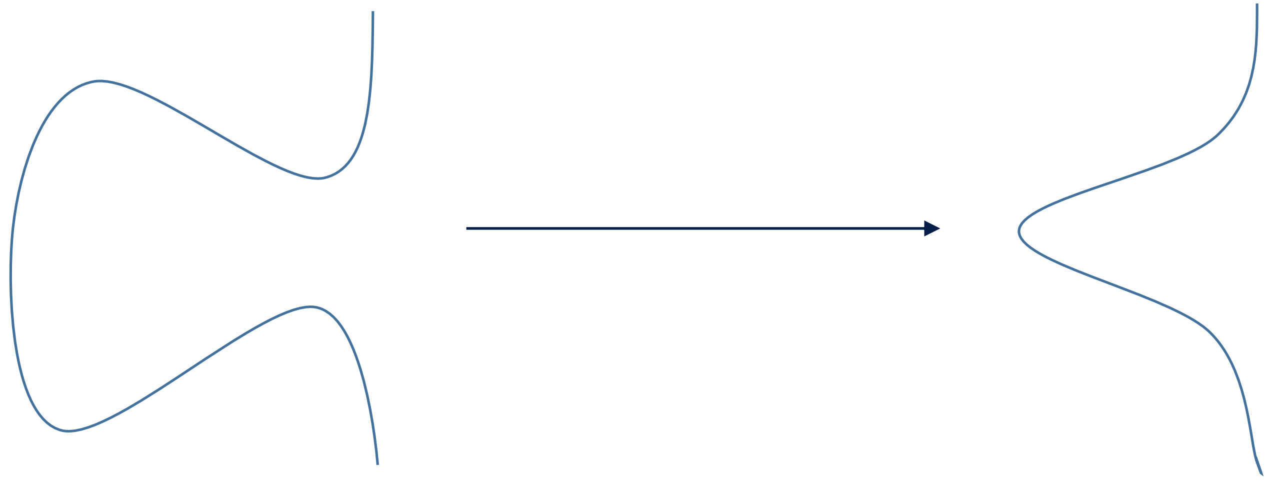
Isogenies

National Cyber Security Centre

Isogenies (many details omitted!)

- An elliptic curve can be described by an equation like $y^2 = x^3 + ax + b$ over F_q
- Points on the curve can be given a group (algebraic) structure. (So we can do things like add two points to find a third point on the curve, used in classical elliptic curve cryptography.)
- An Isogeny is a map between elliptic curves that preserves this group structure

E_1: $y^2 = x^3 + a_1x + b_1$



E_2: $y^2 = x^3 + a_2x + b_2$

- Hard problem: Given isogenous elliptic curves E_1, E_2 over F_q compute an isogeny $\phi : E_1 \rightarrow E_2$

National Cyber Security Centre

SIDH key exchange (details omitted!)

Uses a public supersingular curve $E: y^2 = x^3 + x$ over F_{p^2}

Given a subgroup G of E we can compute a high degree isogeny $\varphi_G\colon E \to E_G := E/\langle G\rangle$ as a composition of isogenies of small degree

$$\begin{tikzcd}
 & E_A \arrow[dr, "\psi_B"] \arrow[dd, dotted, bend left] & \\
E \arrow[ur, "\phi_A"] \arrow[dr, "\phi_B"'] & & E_{AB} \\
 & E_B \arrow[ur, "\psi_A"'] \arrow[uu, dotted, bend left] &
\end{tikzcd}$$

Party A generates a private random point R_A and isogeny $\varphi_A\colon E \to E_A$ and party B generates a private random point R_B and isogeny $\varphi_B\colon E \to E_B$

The images curves E_A and E_B are exchanged publicly (with other data), while the points R_A and R_B used to create the isogenies are kept private

A creates a function ψ_A on E_B from their private key R_A and finds its image $\psi_A(E_B)$ and B creates a function ψ_B on E_A from their private key R_B, and finds its image $\psi_B(E_A)$

The shared private key is computable from both $\psi_A(E_B)$ and $\psi_B(E_A)$, but cannot be computed without knowing R_A or R_B

National Cyber Security Centre

Attacking SIDH isogenies

- Security foundation: The difficulty of recovering an unknown isogeny between a pair of supersingular elliptic curves that are known to be isogenous
  - Not proved to be difficult but is well studied

- (Delfs-Galbraith, 2013) gave classical algorithms for computing general supersingular isogenies with complexity $O\left(p^{1/2}\right)$

- (Biasse, Jao and Sankar, 2014) detail the quantum speed-up to $O\left(p^{1/4}\right)$ via Grover's algorithm

- However the problems underlying SIDH are not general in that the degree of the isogeny φ_G, which is smooth and in $O\left(p^{1/2}\right)$, is known and public. (De Feo, Jao and Plut, 2011) note that this specialized problem can be viewed as an instance of the claw problem, and (Tani, 2007) gives an optimal quantum random walk algorithm for this with complexity $O\left(p^{1/6}\right)$

Features and open questions

- SIDH schemes have relatively small keys of around several hundred bytes
- Complexity of arithmetic makes SIDH lower than the other schemes mentioned
- No matching signature schemes
- More research required to reach a consensus of opinion on SIDH security

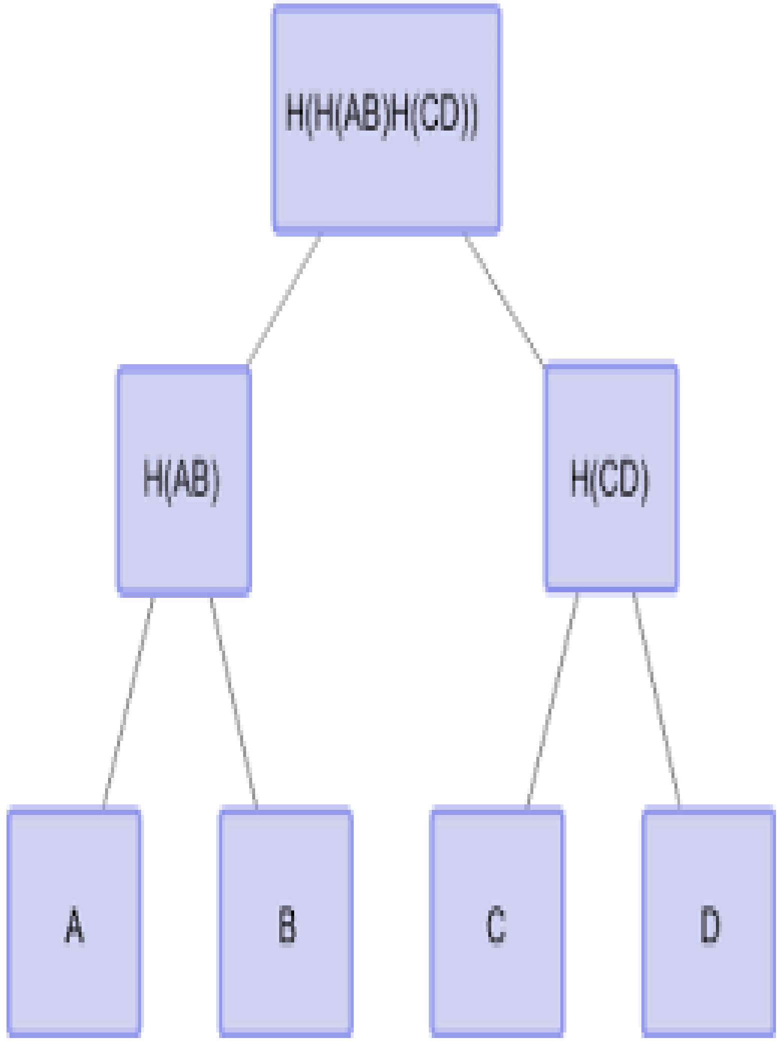
National Cyber Security Centre

Hash-based signatures

National Cyber Security Centre

Hash-based signatures

- Merkle signatures are well understood and considered secure
- The security of the signature depends on the security of the underlying hash function. Most schemes have a security reduction to one of the standard properties for cryptographic hash functions
- Some issues around practicality due to the need to maintain state between signatures. Recent work has been focused on efficiency improvements and the issue of statefulness
- Quantum attacks are limited to using Grover's algorithm to speed up the search for pre-images. Takes with $O(2^{n/2})$ iterations on a single quantum processor. This does not parallelize efficiently as finding the same preimage with 2^k quantum processor would require $O(2^{(n-k)/2})$ iterations


H(H(AB)H(CD))
H(AB)
H(CD)
A
B
C
D