MAA PROGRAMS Virtual Programming

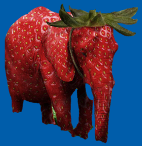
MATHEMATICAL ASSOCIATION OF AMERICA

RSA is dead, long live PQC!

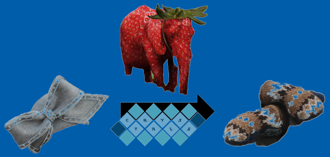
Teaching cryptography in the quantum era

Joshua Holden (he/him/his)

Rose-Hulman Institute of Technology



27 Oct
2022



MAA Recording Policy

- The MAA is recording this event.
- We retain the right to show it again and to distribute it.
- By participating, you are agreeing that your contributions (e.g., text in the chat, your video if enabled, etc.) become part of the recording.
  - *Note:* Breakout rooms are NOT recorded.
- The recording is available by request to registered participants. Please email programs@MAA.org to request.

Letters of Participation

- If you would like a letter of participation, you must request that by emailing programs@MAA.org within 48 hours of attendance.
- Letters will not be sent to those who only view the recording, or those who are in attendance for less than 30 minutes.
- The MAA staff will verify attendance and then send a PDF of the letter to each requester.

Land Acknowledgement

This talk is being broadcast from land that is part of the traditional territories of the Očeti Šakówiŋ (Sioux), Kiikaapoi (Kickapoo), and Myaamia (Miami) nations. These peoples, and many others, are still fighting for the rights promised them by treaties with the United States government.

BIPOC lives and heritages matter.

What is Post-Quantum Cryptography?

Not ...

- ..."The thing that comes after Quantum Cryptography"
- ...Quantum Key Distribution
- ...Quantum Computation

What is Post-Quantum Cryptography?

Not . . .

- . . . "The thing that comes after Quantum Cryptography"
- . . . Quantum Key Distribution
- . . . Quantum Computation

Post-Quantum Cryptography (PQC) **is**

- cryptography that we can run on today's computers
- which will be resistant to cryptanalysis by quantum computers.

In 2016, NIST estimated that a cryptographically relevant quantum computer could be built by 2031.

- RSA: dead

In 2016, NIST estimated that a cryptographically relevant quantum computer could be built by 2031.

- RSA: dead
- Diffie-Hellman Key Agreement: dead

In 2016, NIST estimated that a cryptographically relevant quantum computer could be built by 2031.

- RSA: dead
- Diffie-Hellman Key Agreement: dead
- Digital Signature Algorithm: dead

In 2016, NIST estimated that a cryptographically relevant quantum computer could be built by 2031.

- RSA: dead
- Diffie-Hellman Key Agreement: dead
- Digital Signature Algorithm: dead
- Elliptic Curve Cryptography: dead

In 2016, NIST estimated that a cryptographically relevant quantum computer could be built by 2031.

- RSA: dead
- Diffie-Hellman Key Agreement: dead
- Digital Signature Algorithm: dead
- Elliptic Curve Cryptography: dead
- AES: still alive

In 2016, NIST estimated that a cryptographically relevant quantum computer could be built by 2031.

- RSA: dead
- Diffie-Hellman Key Agreement: dead
- Digital Signature Algorithm: dead
- Elliptic Curve Cryptography: dead
- AES: still alive
- SHA-3: still alive (maybe even SHA-2)

In 2016, NIST estimated that a cryptographically relevant quantum computer could be built by 2031.

- RSA: dead
- Diffie-Hellman Key Agreement: dead
- Digital Signature Algorithm: dead
- Elliptic Curve Cryptography: dead
- AES: still alive
- SHA-3: still alive (maybe even SHA-2)

Several types of cryptography will just need longer keys, but public-key cryptography will need a complete revamp.

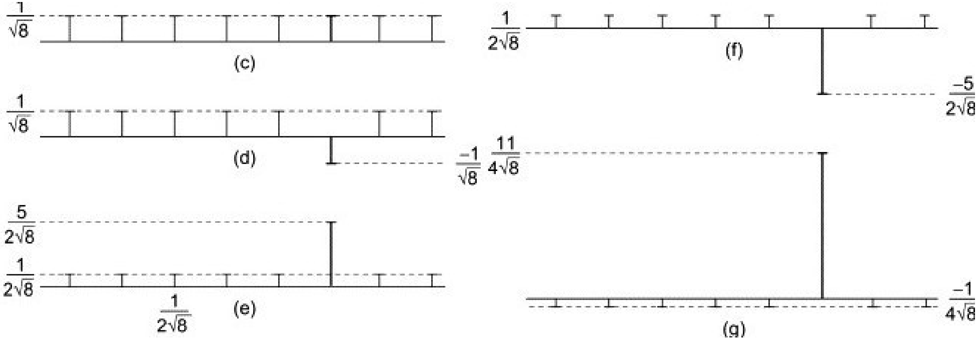
NIST started a process to choose new public-key systems for standardization.

- Submissions received by NIST: 82
- Submissions meeting minimum specified requirements: 69
- Submissions still in contention as of the First PQC Standardization Conference: 64
- Submitters involved: 278, from "25 Countries, 16 States, 6 Continents"

These are the systems referred to as "post-quantum cryptography", although quantum-resistant cryptography might be more accurate.

There are two main quantum algorithms relevant to cryptanalysis.

Grover's algorithm speeds up searching (e.g. for keys), but only by a quadratic amount.

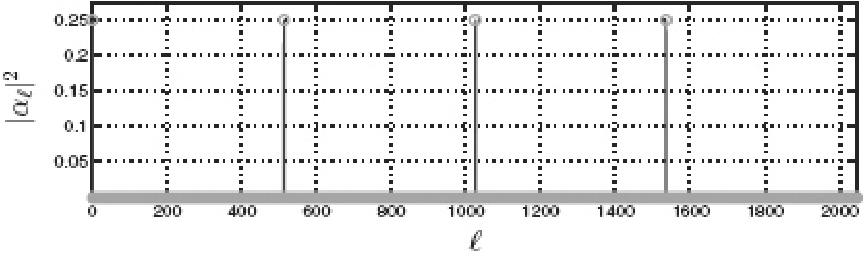


[Marinescu and Marinescu, 2012]

Solution: Double the key size.

There are two main quantum algorithms relevant to cryptanalysis.

Shor's algorithm speeds up finding periodic patterns, by an exponential amount.

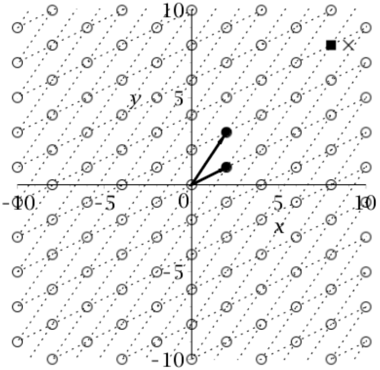


[Neilson and Chuang, 2010]

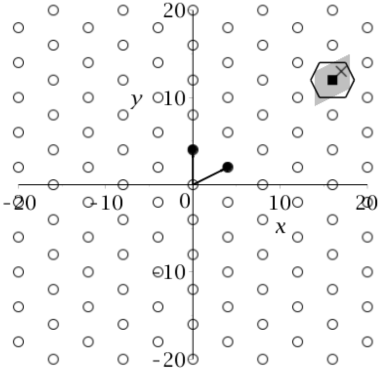
This completely breaks anything based on factoring or any variant of the discrete logarithm problem.

There are five major types of hard problems under consideration for post-quantum cryptography.

1. Lattice problems



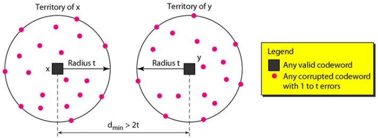
Aretha can encrypt a point using these generators and a small error.



Bernie can decrypt because he knows a better set of generators.

There are five major types of hard problems under consideration for post-quantum cryptography.

2. Code problems



[www.myreadingroom.co.in]

Datawords	Codewords	Datawords	Codewords
0000	0000000	1000	1000110
0001	0001101	1001	1001011
0010	0010111	1010	1010001
0011	0011010	1011	1011100
0100	0100011	1100	1100101
0101	0101110	1101	1101000
0110	0110100	1110	1110010
0111	0111001	1111	1111111

[www.myreadingroom.co.in]

Aretha can encrypt a bitstring using a set of generators and a small error.

Bernie can decrypt because he knows a better set of generators.

There are five major types of hard problems under consideration for post-quantum cryptography.

3. Multivariable polynomial problems

$$\begin{aligned} y_1 &= f_1(x_1, \ldots, x_n) \\ y_2 &= f_2(x_1, \ldots, x_n) \\ &\vdots \\ y_m &= f_m(x_1, \ldots, x_n) \end{aligned}$$

$$\begin{aligned} y_1 &= f_1(x_1, \ldots, x_n) \\ y_2 &= f_2(x_1, \ldots, x_n) \\ &\vdots \\ y_m &= f_m(x_1, \ldots, x_n) \end{aligned}$$

Aretha can solve these equations using a trap door $f = S \circ f^* \circ T$.

Given the message $y_1, \ldots, y_m$, Bernie can verify that the signature $x_1, \ldots, x_n$ is correct.

There are five major types of hard problems under consideration for post-quantum cryptography.

4. Hash function problems

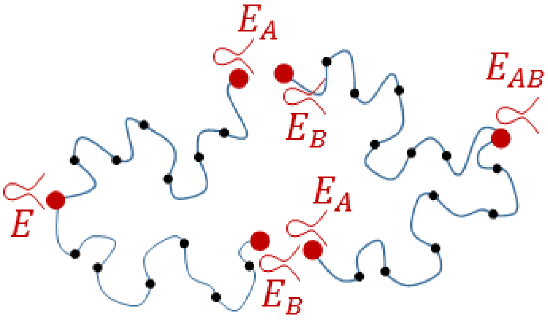
					1	0	0	$\cdots$
$H(X_{0,1})$	$H(X_{0,2})$	$H(X_{0,3})$	$\cdots$		$H(X_{0,1})$	$X_{0,2}$	$X_{0,3}$	$\cdots$
$H(X_{1,1})$	$H(X_{1,2})$	$H(X_{1,3})$	$\cdots$		$X_{1,1}$	$H(X_{1,2})$	$H(X_{1,3})$	$\cdots$

Aretha's secret key is two sets of random hashes.

Aretha can sign by revealing some of the random numbers.

There are five major types of hard problems under consideration for post-quantum cryptography.

5. Elliptic Curve Isogenies (some systems have recently been broken!)



[Castryck, 2017]

Aretha can reach the secret curve by going around the top route.

Bernie can reach the secret curve by going around the bottom route.

NIST has so far selected four submissions for standardization.

- CRYSTALS-Kyber (key-establishment for most use cases)
- CRYSTALS-Dilithium (digital signatures for most use cases)
- FALCON (digital signatures for use cases requiring smaller signatures)
- SPHINCS+ (digital signatures not relying on the security of lattices)

The first three are lattice systems; the fourth is a hash function system.

CRYSTALS

Cryptographic Suite for Algebraic Lattices

Joppe Bos Leo Ducas

Eike Kiltz Tancrede Lepoint

Vadim Lyubashevsky John Schanck

Peter Schwabe Gregor Seiler Damien Stehle

CRYSTALS-Kyber is based on a version of the Learning With Errors (LWE) problem.

Given a vector $\mathbf{t}$ of the form

$$\mathbf{t} \equiv A\mathbf{s} + \mathbf{e} \pmod{q}, \tag{1}$$

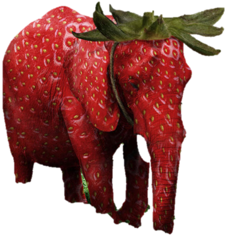
where

- A is a public matrix with at least as many rows as columns
- $\mathbf{e}$ is a "small error vector" drawn from some probability distribution

find the secret vector $\mathbf{s}$.

This seminar will introduce three toy cryptosystems based on variations of LWE:

- Smeagol (a simplified version of Frodo), based on LWE
- Phantom (a simplified version of NewHope), based on Ring LWE
- Alkaline (a simplified version of Kyber), based on Module LWE



ElectroNic ExeRcises for CiphEricaL Learning

Smeagol key generation:

Bernie's private key is a $k \times \ell$ matrix S and his public key is (A, T), where

- A is a $k \times k$ public matrix,
- E is a $k \times \ell$ error matrix, and

$$T \equiv AS + E \pmod{q} \tag{2}$$

(The columns of S, T, and E are thought of as collections of vectors **s**, **t**, and **e**.)

[New Line Cinema]

Smeagol example ($k = 4$, $\ell = 4$, $q = 23$)

Bernie first generates the random matrices

$$A = \begin{pmatrix} 7 & 21 & 8 & 18 \\ 15 & 6 & 14 & 16 \\ 7 & 7 & 5 & 13 \\ 20 & 3 & 4 & 13 \end{pmatrix}, \qquad S = \begin{pmatrix} 3 & 0 & 2 & -3 \\ 0 & 3 & -2 & -1 \\ -1 & 0 & 1 & -1 \\ 1 & 1 & -2 & 2 \end{pmatrix},$$

$$E = \begin{pmatrix} -1 & 1 & 2 & 0 \\ 0 & 0 & -1 & -1 \\ 1 & -2 & -1 & 0 \\ 1 & -2 & 2 & 0 \end{pmatrix}.$$

He then uses these to compute the public matrix

$$T = (AS + E) \operatorname{MOD} 23 = \begin{pmatrix} 8 & 14 & 15 & 9 \\ 7 & 16 & 21 & 12 \\ 21 & 0 & 16 & 1 \\ 2 & 9 & 5 & 11 \end{pmatrix}$$

and publishes (A, T) as his public key.

Smeagol encryption:

$\mathbf{p}$ is a plaintext message represented as a vector of bits.

Aretha chooses

- a random k-bit nonce vector $\mathbf{r}$;
- small random errors $\mathbf{e}_1$ and $\mathbf{e}_2$.

She computes the ciphertext $(\mathbf{u}, \mathbf{v})$, where

$$\mathbf{u} = \left(A^T\mathbf{r} + \mathbf{e}_1\right) \text{ MOD } q, \qquad \mathbf{v} = \left(T^T\mathbf{r} + \mathbf{e}_2 + \lfloor q/2 \rceil \, \mathbf{p}\right) \text{ MOD } q.$$

hint

blind

plaintext

($\lfloor x \rceil$ means round x to the nearest integer. MOD is the "coder's mod".)

Smeagol example (continued)

Aretha wants to send the message "hi" to Bernie. For the first letter, Aretha encodes "h" as the number 8, or 1000 in binary, corresponding to

$$\mathbf{p} = (1\ 0\ 0\ 0)^T.$$

She computes random vectors

$$\mathbf{r} = (0\ 1\ \ -1\ \ -1)^T, \qquad \mathbf{e_1} = (1\ 0\ 0\ 0)^T, \qquad \mathbf{e_2} = (0\ 0\ 0\ \ -1)^T.$$

She then computes

$$\begin{aligned}
\mathbf{u} &= (A^T\mathbf{r} + \mathbf{e}_1)\,\mathrm{MOD}\, q = (13\ 4\ 5\ 22)^T \\
\mathbf{v} &= (T^T\mathbf{r} + \mathbf{e}_2 + \lfloor q/2 \rceil\, \mathbf{p})\,\mathrm{MOD}\, q \\
&= \left[[(7\ 7\ 0\ 0)^T + (0\ 0\ 0\ 1)^T + (12\ 0\ 0\ 0)^T \right]\mathrm{MOD}\, 23 \\
&= (19\ 7\ 0\ 22)^T \qquad \text{and sends } (\mathbf{u}, \mathbf{v}) \text{ to Bernie.}
\end{aligned}$$

Smeagol decryption:

Bernie tests each coordinate of $\mathbf{v} - S^T\mathbf{u}$ to see if it is closer to 0 or to $q/2$ modulo q:

$$\mathbf{p}' = \left\lfloor \frac{(\mathbf{v} - S^T\mathbf{u}) \,\mathrm{MOD}\, q}{\lfloor q/2 \rceil} \right\rceil \mathrm{MOD}\, 2.$$

Since

$$\begin{aligned}
\mathbf{v} - S^T\mathbf{u} &\equiv T^T\mathbf{r} + \lfloor q/2 \rceil \mathbf{p} + \mathbf{e_2} - S^T A^T \mathbf{r} - S^T\mathbf{e_1} \pmod q \\
&\equiv (AS)^T\mathbf{r} + E^T\mathbf{r} + \lfloor q/2 \rceil \mathbf{p} + \mathbf{e_2} - S^T A^T \mathbf{r} - S^T\mathbf{e_1} \pmod q \\
&\equiv E^T\mathbf{r} + \mathbf{e_2} - S^T\mathbf{e_1} + \lfloor q/2 \rceil \mathbf{p} \pmod q
\end{aligned}$$

as long as the coordinates of $E^T\mathbf{r} + \mathbf{e_2} - S^T\mathbf{e_1}$ have magnitude less than $q/4$, $\mathbf{p}'$ will be equal to $\mathbf{p}$.

Smeagol example (concluded)

You are Bernie . You receive the message:

$$\mathbf{u} = (13\ 4\ 5\ 22)^T$$
$$\mathbf{v} = (19\ 7\ 0\ 22)^T$$

from Aretha . You compute

$$\mathbf{p}' = \left\lfloor \frac{(\mathbf{v} - S^T\mathbf{u}) \text{ MOD } q}{\lfloor q/2 \rceil} \right\rceil \text{MOD } 2 = ???$$

Do you get the message that Aretha sent?

Some things you should note:

- There is a lot of ciphertext expansion!
- The encryption is probabilistic.
- There is a chance that the decryption will fail.

What kinds of PQC can you teach to undergraduates?

- Lattice-based: NTRU, Smeagol, Phantom, Alkaline (`https://github.com/joshuarbholden/alkaline`)
- Code-based: McEliece
- Hash-based: Merkle signature scheme
- Multivariable: Oil and Vinegar
- Elliptic curve isogenies: Charles-Goren-Lauter hash function

My *Resource Guide for Teaching Post-Quantum Cryptography* is at `https://arxiv.org/abs/2207.00558` (to appear in *Cryptologia*).

Also watch for the second edition of *The Mathematics of Secrets*, from Princeton University Press.

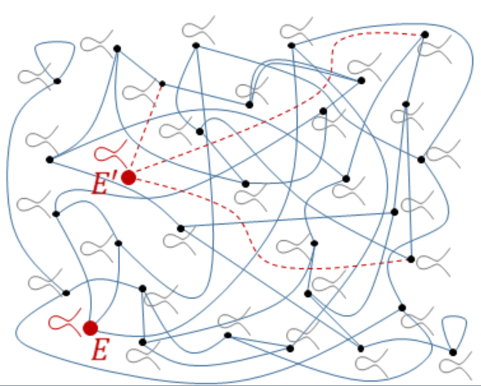
What undergraduate classes would this be appropriate for?

- Linear Algebra: Smeagol, Oil and Vinegar, McEliece
- Abstract Algebra: Phantom, Alkaline, McEliece
- Second course in Algebra or in Number Theory: CGL
- Data Structures: Merkle signature scheme
- Cryptography and/or student research project: any of these!

Next session:

- Finish Phantom / Alkaline
- Comparing Alkaline with Kyber
- Breaking lattice-based systems

Thanks for joining us for this MAA virtual event!



[Wouter Castryck, 2017]

MAA MATHEMATICAL ASSOCIATION OF AMERICA

PROGRAMS Virtual Programming