The state of the isogeny

Lorenz Panny

Technische Universität München

Berlin Crypto Meetup, 23 September 2024

Big picture

- Isogenies are a type of maps between elliptic curves.

Big picture

- Isogenies are a type of maps between elliptic curves.
- Sampling an isogeny *from* some curve is easy, recovering an isogeny *between* given curves seems very hard.

Big picture

- Isogenies are a type of maps between elliptic curves.
- Sampling an isogeny *from* some curve is easy, recovering an isogeny *between* given curves seems very hard.

$\rightsquigarrow$ ***Cryptography!***

Plan for this talk

- Some high-level intuition.
- Elliptic curves & isogenies.
- The CSIDH non-interactive key exchange.
- The SIKE attacks.
- The SQIsign signature scheme.

Diffie–Hellman key exchange 1976

Public parameters:

- a finite group G (traditionally $\mathbb{F}_p^*$, today elliptic curves)
- an element $g \in G$ of prime order q

Diffie–Hellman key exchange 1976

Public parameters:

- a finite group G (traditionally $\mathbb{F}_p^*$, today elliptic curves)
- an element $g \in G$ of prime order q

Alice	public	Bob
$a \xleftarrow{\text{random}} \{0 \dots q-1\}$		$b \xleftarrow{\text{random}} \{0 \dots q-1\}$
g^a		g^b
$s := (g^b)^a$		$s := (g^a)^b$

Diffie–Hellman key exchange 1976

Public parameters:

- a finite group G (traditionally $\mathbb{F}_p^*$, today elliptic curves)
- an element $g \in G$ of prime order q

Alice	public	Bob
$a \xleftarrow{\text{random}} \{0 \ldots q-1\}$		$b \xleftarrow{\text{random}} \{0 \ldots q-1\}$
g^a		g^b
$s := (g^b)^a$		$s := (g^a)^b$

Fundamental reason this works: $\cdot^a$ and $\cdot^b$ are commutative!

Diffie–Hellman: Bob vs. Eve

<u>Bob</u>

1. Set $t \leftarrow g$.
2. Set $t \leftarrow t \cdot g$.
3. Set $t \leftarrow t \cdot g$.
4. Set $t \leftarrow t \cdot g$.

...

$b-2$. Set $t \leftarrow t \cdot g$.

$b-1$. Set $t \leftarrow t \cdot g$.

b. Publish $B \leftarrow t \cdot g$.

Diffie–Hellman: Bob vs. Eve

Bob

1. Set $t \leftarrow g$.
2. Set $t \leftarrow t \cdot g$.
3. Set $t \leftarrow t \cdot g$.
4. Set $t \leftarrow t \cdot g$.

...

$b-2$. Set $t \leftarrow t \cdot g$.

$b-1$. Set $t \leftarrow t \cdot g$.

b. Publish $B \leftarrow t \cdot g$.

Is this a good idea?

Diffie–Hellman: Bob vs. Eve

Bob

1. Set $t \leftarrow g$.
2. Set $t \leftarrow t \cdot g$.
3. Set $t \leftarrow t \cdot g$.
4. Set $t \leftarrow t \cdot g$.

...

$b-2$. Set $t \leftarrow t \cdot g$.

$b-1$. Set $t \leftarrow t \cdot g$.

b. Publish $B \leftarrow t \cdot g$.

Attacker Eve

1. Set $t \leftarrow g$. If $t = B$ return 1.
2. Set $t \leftarrow t \cdot g$. If $t = B$ return 2.
3. Set $t \leftarrow t \cdot g$. If $t = B$ return 3.
4. Set $t \leftarrow t \cdot g$. If $t = B$ return 3.

...

$b-2$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b-2$.

$b-1$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b-1$.

$\boxed{b.\ \text{Set } t \leftarrow t \cdot g.\ \text{If } t = B \text{ return } b.}$

$b+1$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b+1$.

$b+2$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b+2$.

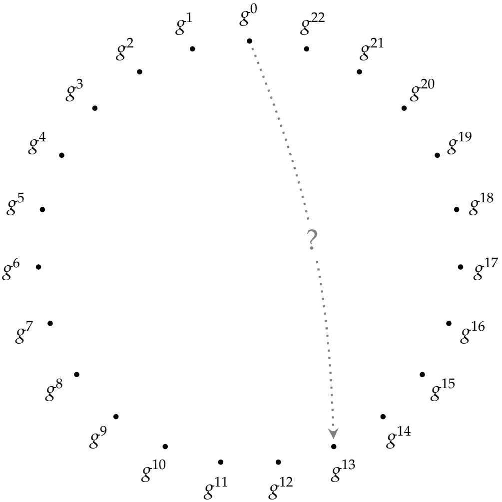
...

Diffie–Hellman: Bob vs. Eve

<u>Bob</u>	<u>**Attacker** Eve</u>
1. Set $t \leftarrow g$.	1. Set $t \leftarrow g$. If $t = B$ return 1.
2. Set $t \leftarrow t \cdot g$.	2. Set $t \leftarrow t \cdot g$. If $t = B$ return 2.
3. Set $t \leftarrow t \cdot g$.	3. Set $t \leftarrow t \cdot g$. If $t = B$ return 3.
4. Set $t \leftarrow t \cdot g$.	4. Set $t \leftarrow t \cdot g$. If $t = B$ return 3.
...	...
$b-2$. Set $t \leftarrow t \cdot g$.	$b-2$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b-2$.
$b-1$. Set $t \leftarrow t \cdot g$.	$b-1$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b-1$.
b. Publish $B \leftarrow t \cdot g$.	b. Set $t \leftarrow t \cdot g$. If $t = B$ return b.
	$b+1$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b+1$.
	$b+2$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b+2$.
	...

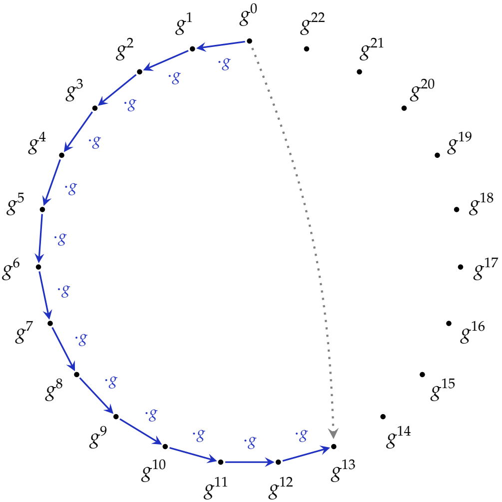
Effort for both: $O(\#G)$. Bob needs to be smarter.

(This attacker is also kind of dumb, but that doesn't matter for my point here.)



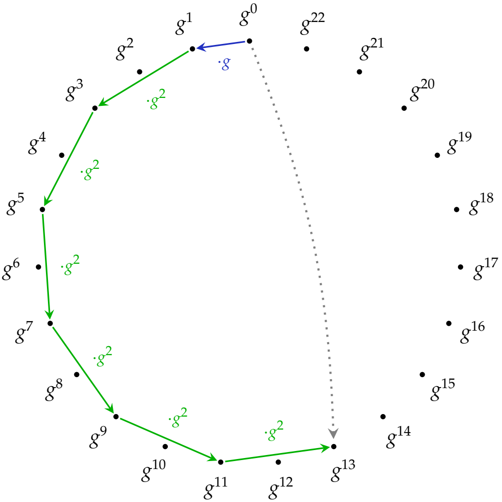
Bob computes his public key g^{13} from g.

multiply



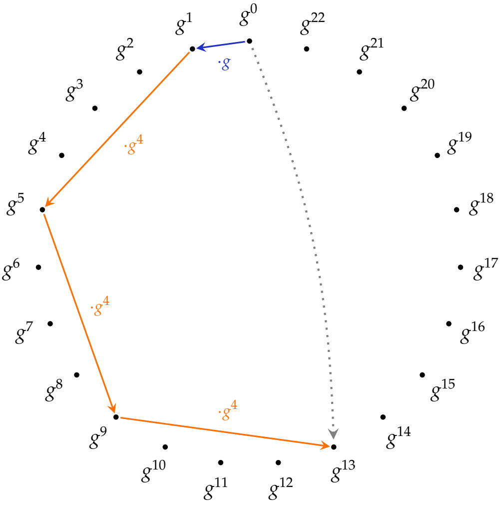
Bob computes his public key g^{13} from g.

Square-and-multiply



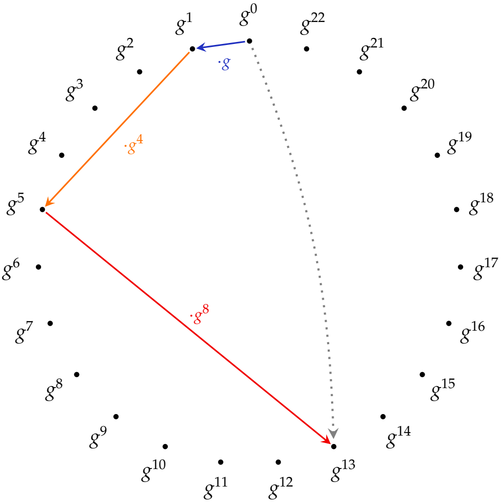
Bob computes his public key g^{13} from g.

Square-and-multiply-and-square-and-multiply



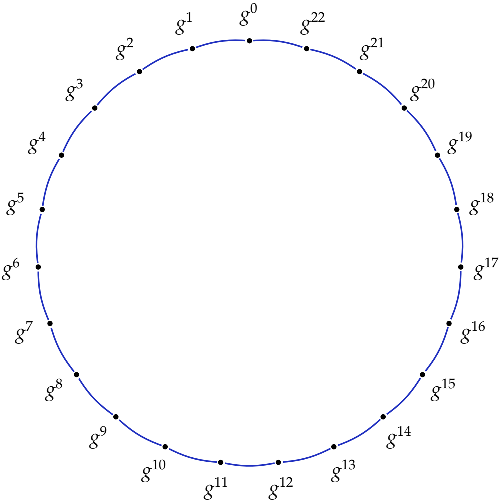
Bob computes his public key g^{13} from g.

Square-and-multiply-and-square-and-multiply-and-squ

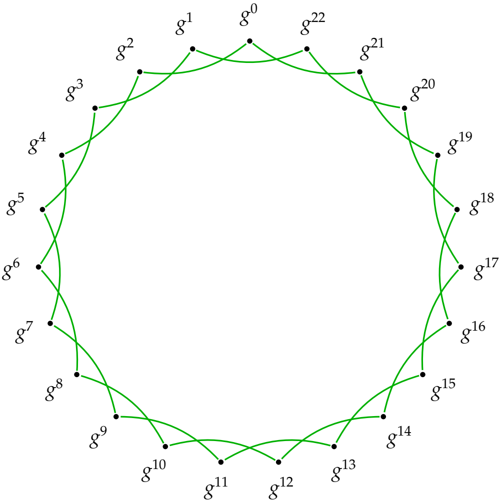


Bob computes his public key g^{13} from g.

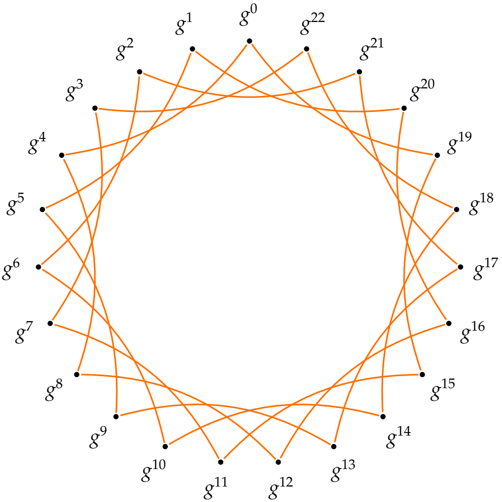
Square-and-multiply as graphs



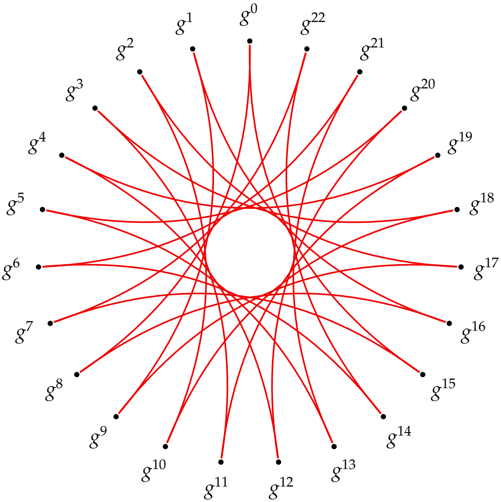
Square-and-multiply as graphs



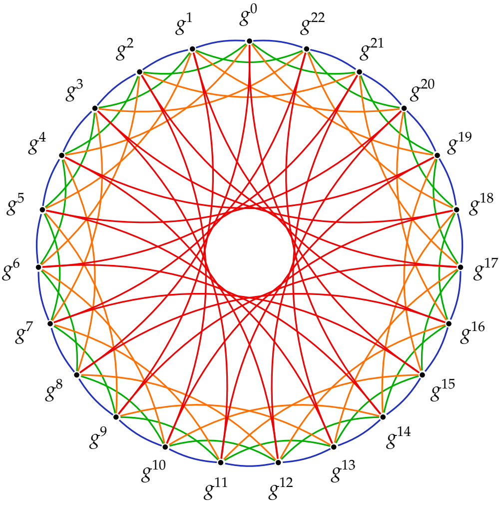
Square-and-multiply as graphs



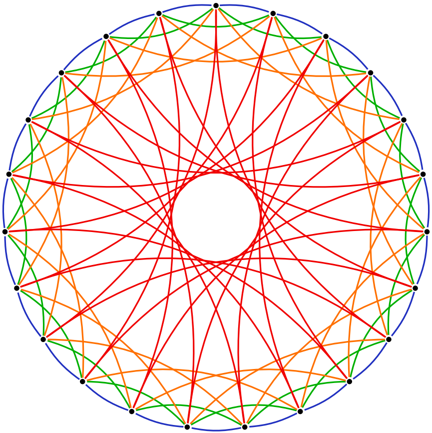
Square-and-multiply as graphs



Square-and-multiply as a graph



Square-and-multiply as a graph



Fast mixing: paths of length $\log(\#\,\text{nodes})$ to everywhere.

Shor's algorithm vs. DLP

Shor's quantum algorithm computes α from g^α in any group in polynomial time.

Shor's algorithm vs. DLP

Shor's quantum algorithm computes α from g^α in any group in polynomial time.

Shor computes α from $h = g^\alpha$ by finding the kernel of the map

$$f\colon\ \mathbb{Z}^2 \to G,\ (x, y) \mapsto g^x \cdot h^y\,.$$

Shor's algorithm vs. DLP

Shor's quantum algorithm computes α from g^α in any group in polynomial time.

Shor computes α from $h = g^\alpha$ by finding the kernel of the map

$$f\colon\ \mathbb{Z}^2 \to G,\ (x, y) \mapsto g^x \cdot h^y\,.$$

$\rightsquigarrow$ New plan: Get rid of the group, keep the graph.

Plan for this talk

- Some high-level intuition. ✓
- Elliptic curves & isogenies.
- The CSIDH non-interactive key exchange.
- The SIKE attacks.
- The SQIsign signature scheme.

Stand back!

We’re going to do math.

Elliptic curves

An elliptic curve over a field F of characteristic $\notin \{2, 3\}$ is* an equation of the form

$$E\colon\; y^2 = x^3 + ax + b$$

with $a, b \in F$ such that $4a^3 + 27b^2 \neq 0$.

Elliptic curves

An elliptic curve over a field F of characteristic $\notin \{2, 3\}$ is* an equation of the form

$$E\colon\ y^2 = x^3 + ax + b$$

with $a, b \in F$ such that $4a^3 + 27b^2 \neq 0$.

A point on E is a solution (x, y), <u>or</u> the "fake" point ∞.

Elliptic curves

An elliptic curve over a field F of characteristic $\notin \{2, 3\}$ is* an equation of the form

$$E\colon\ y^2 = x^3 + ax + b$$

with $a, b \in F$ such that $4a^3 + 27b^2 \neq 0$.

A point on E is a solution (x, y), or the "fake" point ∞.

E is an abelian group: we can "add" points.

Elliptic curves

An elliptic curve over a field F of characteristic $\notin \{2, 3\}$ is* an equation of the form

$$E\colon\ y^2 = x^3 + ax + b$$

with $a, b \in F$ such that $4a^3 + 27b^2 \neq 0$.

A point on E is a solution (x, y), or the "fake" point ∞.

E is an abelian group: we can "add" points.

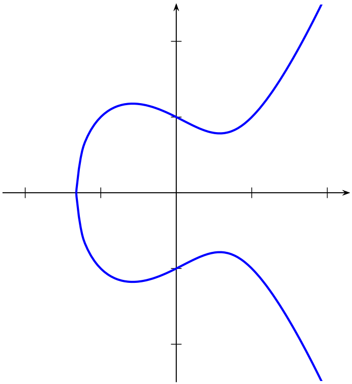
- The neutral element is ∞.
- The inverse of (x, y) is $(x, -y)$.
- The sum of (x_1, y_1) and (x_2, y_2) is

$$\left(\lambda^2 - x_1 - x_2,\ \lambda(2x_1 + x_2 - \lambda^2) - y_1\right)$$

where $\lambda = \frac{y_2 - y_1}{x_2 - x_1}$ if $x_1 \neq x_2$ and $\lambda = \frac{3x_1^2 + a}{2y_1}$ otherwise.

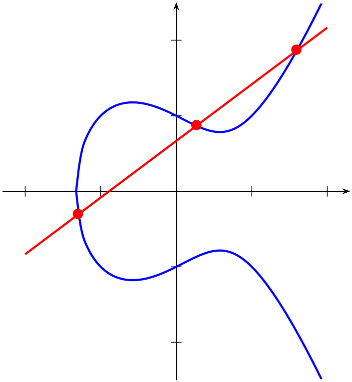
do **not** remember these formulas!

Elliptic curves (picture over $\mathbb{R}$)



The elliptic curve $y^2 = x^3 - x + 1$ over $\mathbb{R}$.

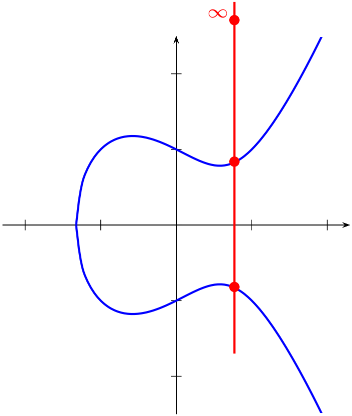
Elliptic curves (picture over $\mathbb{R}$)



Addition law:

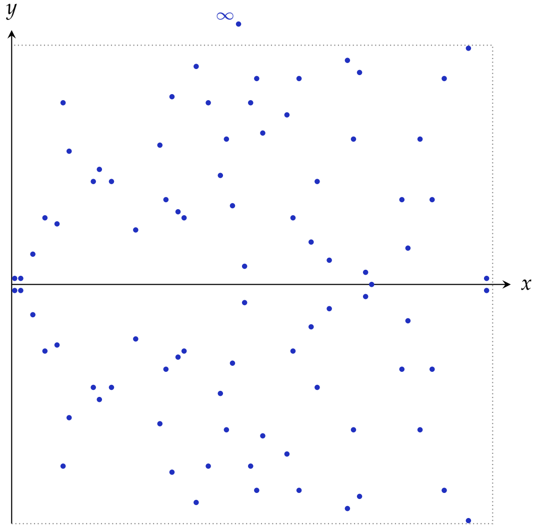
$$P + Q + R = \infty \quad \Longleftrightarrow \quad \{P, Q, R\} \text{ on a straight line.}$$

Elliptic curves (picture over $\mathbb{R}$)



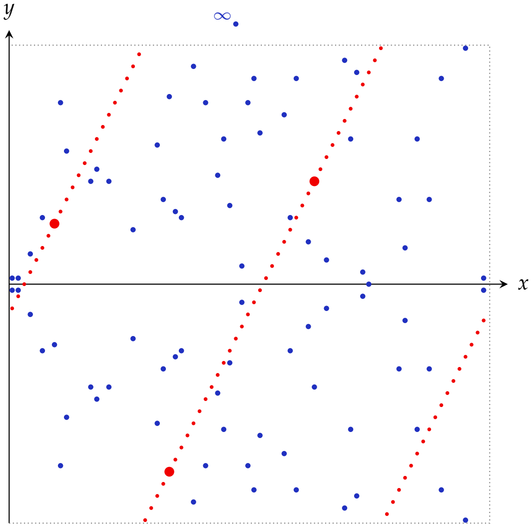
The *point at infinity* ∞ lies on every vertical line.

Elliptic curves (picture over $\mathbb{F}_p$)



The same curve $y^2 = x^3 - x + 1$ over the finite field $\mathbb{F}_{79}$.

Elliptic curves (picture over $\mathbb{F}_p$)



The addition law of $y^2 = x^3 - x + 1$ over the finite field $\mathbb{F}_{79}$.

Isogenies

Isogenies

...are just fancily-named

nice maps

between elliptic curves.

Isogenies

An isogeny of elliptic curves is a non-zero map $E \to E'$ that is:

Isogenies

An isogeny of elliptic curves is a non-zero map $E \to E'$ that is:

- given by rational functions.

Isogenies

An isogeny of elliptic curves is a non-zero map $E \to E'$ that is:

- given by rational functions.
- a group homomorphism.

Isogenies

An isogeny of elliptic curves is a non-zero map $E \to E'$ that is:

- given by rational functions.
- a group homomorphism.

<u>Reminder</u>:

A rational function is $f(x,y)/g(x,y)$ where f, g are polynomials.

A group homomorphism φ satisfies $\varphi(P+Q) = \varphi(P) + \varphi(Q)$.

Isogenies

An isogeny of elliptic curves is a non-zero map $E \to E'$ that is:

- given by rational functions.
- a group homomorphism.

Reminder:

A rational function is $f(x,y)/g(x,y)$ where f, g are polynomials.

A group homomorphism φ satisfies $\varphi(P+Q) = \varphi(P) + \varphi(Q)$.

The kernel of an isogeny $\varphi\colon E \to E'$ is $\{P \in E : \varphi(P) = \infty\}$.

The degree of a separable* isogeny is the size of its kernel.

Isogenies (examples)

An isogeny of elliptic curves is a non-zero map $E \to E'$ that is:

- given by rational functions.
- a group homomorphism.

Isogenies (examples)

An isogeny of elliptic curves is a non-zero map $E \to E'$ that is:

- given by rational functions.
- a group homomorphism.

Example #1: For each $m \neq 0$, the multiplication-by-m map

$$[m] \colon E \to E$$

is a degree-m^2 isogeny. If $m \neq 0$ in the base field, its kernel is

$$E[m] \;\cong\; \mathbb{Z}/m \times \mathbb{Z}/m.$$

Isogenies (examples)

An isogeny of elliptic curves is a non-zero map $E \to E'$ that is:

- given by rational functions.
- a group homomorphism.

Example #2: $(x, y) \mapsto \left(\frac{x^3-4x^2+30x-12}{(x-2)^2}, \frac{x^3-6x^2-14x+35}{(x-2)^3} \cdot y \right)$
defines a degree-3 isogeny of the elliptic curves

$$\{y^2 = x^3 + x\} \longrightarrow \{y^2 = x^3 - 3x + 3\}$$

over $\mathbb{F}_{71}$. Its kernel is $\{(2, 9), (2, -9), \infty\}$.

Isogenies and kernels

For any finite subgroup G of E, there exists a unique* separable* isogeny $\varphi_G \colon E \to E'$ with kernel G.

Isogenies and kernels

For any finite subgroup G of E, there exists a unique* separable* isogeny $\varphi_G \colon E \to E'$ with kernel G.

The curve E' is denoted by E/G. (cf. quotient groups)

Isogenies and kernels

For any finite subgroup G of E, there exists a unique* separable* isogeny $\varphi_G \colon E \to E'$ with kernel G.

The curve E' is denoted by E/G. (cf. quotient groups)

$\rightsquigarrow$ To choose an isogeny, simply choose a finite subgroup.

Isogenies and kernels

For any finite subgroup G of E, there exists a unique* separable* isogeny $\varphi_G \colon E \to E'$ with kernel G.

The curve E' is denoted by E/G. (cf. quotient groups)

$\rightsquigarrow$ To choose an isogeny, simply choose a finite subgroup.

- We have formulas to compute and evaluate isogenies.
  (...but they are only efficient for "small" degrees!)

Isogenies and kernels

For any finite subgroup G of E, there exists a unique* separable* isogeny $\varphi_G \colon E \to E'$ with kernel G.

The curve E' is denoted by E/G. (cf. quotient groups)

$\rightsquigarrow$ To choose an isogeny, simply choose a finite subgroup.

- We have formulas to compute and evaluate isogenies.
  (...but they are only efficient for "small" degrees!)

$\rightsquigarrow$ Decompose large-degree isogenies into prime steps.
That is, walk in an isogeny graph.

One-wayness from isogenies

One-wayness from isogenies

Keep in mind: Constructing isogenies $E \to \_$ is (usually) easy,
constructing an isogeny $E \to E'$ given (E, E') is (usually) hard.

Plan for this talk

- Some high-level intuition. ✓
- Elliptic curves & isogenies. ✓
- The CSIDH non-interactive key exchange.
- The SIKE attacks.
- The SQIsign signature scheme.

CSIDH [ˈsiːˌsaɪd]

[Castryck–Lange–Martindale–Panny–Renes 2018]

Isogeny-based key exchange: High-level view

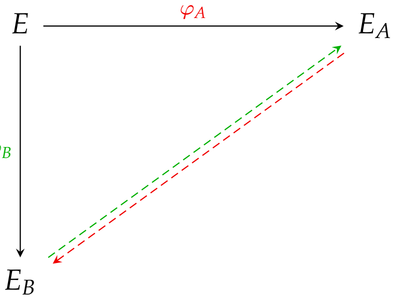
E

Isogeny-based key exchange: High-level view

$$\begin{tikzcd} E \arrow[r, "\varphi_A"] \arrow[d, "\varphi_B"'] & E_A \\ E_B & \end{tikzcd}$$

- Alice & Bob pick secret $\varphi_A : E \to E_A$ and $\varphi_B : E \to E_B$.
  (These isogenies correspond to walking on the isogeny graph.)

Isogeny-based key exchange: High-level view



- Alice & Bob pick secret $\varphi_A : E \to E_A$ and $\varphi_B : E \to E_B$.
  (These isogenies correspond to walking on the isogeny graph.)
- Alice and Bob transmit the end curves E_A and E_B.

Isogeny-based key exchange: High-level view

$$\begin{tikzcd}
E \arrow[rr, "\varphi_A"] \arrow[dd, "\varphi_B"'] & & E_A \arrow[d, "\varphi'_B"] \arrow[ddll, dashed, bend left=5] \\
& & E_{AB} \\
E_B \arrow[r, "\varphi'_A"'] \arrow[uurr, dashed, bend left=5] & E_{BA} &
\end{tikzcd}$$

- Alice & Bob pick secret $\varphi_A : E \to E_A$ and $\varphi_B : E \to E_B$.
  (These isogenies correspond to walking on the isogeny graph.)
- Alice and Bob transmit the end curves E_A and E_B.
- Alice <u>somehow</u> finds a "parallel" $\varphi_{A'} : E_B \to E_{BA}$, and Bob <u>somehow</u> finds $\varphi_{B'} : E_A \to E_{AB}$,

Isogeny-based key exchange: High-level view

$$
\begin{tikzcd}
E \arrow[rr, "\varphi_A"] \arrow[dd, "\varphi_B"'] & & E_A \arrow[d, "\varphi'_B"] \arrow[ddll, dashed, bend left=5] \\
& & E_{AB} \\
E_B \arrow[r, "\varphi'_A"'] \arrow[uurr, dashed, bend left=5] & E_{BA} \arrow[ur, equal] &
\end{tikzcd}
$$

- Alice & Bob pick secret $\varphi_A : E \to E_A$ and $\varphi_B : E \to E_B$.
  (These isogenies correspond to walking on the isogeny graph.)
- Alice and Bob transmit the end curves E_A and E_B.
- Alice <u>somehow</u> finds a "parallel" $\varphi_{A'} : E_B \to E_{BA}$, and Bob <u>somehow</u> finds $\varphi_{B'} : E_A \to E_{AB}$, such that $E_{AB} \cong E_{BA}$.

How to find “parallel” isogenies?

$$\begin{tikzcd}
E \arrow[rr, "\varphi_A"] \arrow[dd, "\varphi_B"'] & & E_A \arrow[d, "\varphi'_B"] \\
& & E_{AB} \\
E_B \arrow[r, "\varphi'_A"'] & E_{BA} \arrow[ur, equal] &
\end{tikzcd}$$

How to find “parallel” isogenies?

$$
\begin{tikzcd}
E \arrow[rr, "\varphi_A"] \arrow[dd, "\varphi_B"'] & & E_A \arrow[d, "\varphi'_B"] \\
& & E_{AB} \\
E_B \arrow[r, "\varphi'_A"'] & E_{BA} \arrow[ur, equal] &
\end{tikzcd}
$$

<u>**CSIDH**’s solution</u>:

Use special isogenies φ_A which can be transported to the curve E_B totally independently of the secret isogeny φ_B.

(Similarly with reversed roles, of course.)

“Special” isogenies

We fix an elliptic curve $E/\mathbb{F}_p$ such that $E(\mathbb{F}_p) \cong \mathbb{Z}/(p+1)$.

"Special" isogenies

We fix an elliptic curve $E/\mathbb{F}_p$ such that $E(\mathbb{F}_p) \cong \mathbb{Z}/(p+1)$.

$\Rightarrow$ For every $\ell \mid (p+1)$ exists a unique order-ℓ subgroup H_ℓ.

“Special” isogenies

We fix an elliptic curve $E/\mathbb{F}_p$ such that $E(\mathbb{F}_p) \cong \mathbb{Z}/(p+1)$.

$\Rightarrow$ For every $\ell \mid (p+1)$ exists a unique order-ℓ subgroup H_ℓ.

$\rightsquigarrow$ For all such E can canonically find an isogeny $\varphi_\ell \colon E \to E'$.

“Special” isogenies

We fix an elliptic curve $E/\mathbb{F}_p$ such that $E(\mathbb{F}_p) \cong \mathbb{Z}/(p+1)$.

$\Rightarrow$ For every $\ell \mid (p+1)$ exists a unique order-ℓ subgroup H_ℓ.

$\rightsquigarrow$ For all such E can canonically find an isogeny $\varphi_\ell \colon E \to E'$.

We consider prime ℓ and refer to φ_ℓ as a “special” isogeny.

Cycles from “special” isogenies

What happens when we iterate such a “special” isogeny?

Cycles from “special” isogenies

What happens when we iterate such a “special” isogeny?

$$
\begin{tikzcd}
 & & & & E_{\ell^{r-1}} \arrow[dl] & \cdots \arrow[l] & E_{\ell^6} \arrow[l] \\
 & & & E_{\ell^r} \arrow[d] & & & E_{\ell^5} \arrow[u] \\
E \arrow[r] & E_\ell \arrow[r] & E_{\ell^2} \arrow[r] & E_{\ell^3} \arrow[rr] & & E_{\ell^4} \arrow[ur] &
\end{tikzcd}
$$

Cycles from "special" isogenies

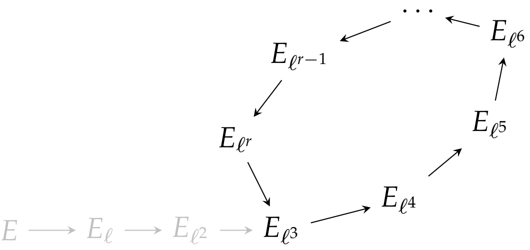
What happens when we iterate such a "special" isogeny?

$$\begin{tikzcd}
 & & & & \cdots \arrow[dl] & E_{\ell^6} \arrow[l] \\
 & & & E_{\ell^{r-1}} \arrow[d] & & \\
 & & & E_{\ell^r} \arrow[dd] & & E_{\ell^5} \arrow[uu] \\
 & & & & E_{\ell^4} \arrow[ur] & \\
E \arrow[r] & E_\ell \arrow[r] & E_{\ell^2} \arrow[r] & E_{\ell^3} \arrow[ur] & &
\end{tikzcd}$$

- Fact: Each curve has only one other rational ℓ-isogeny.

Cycles from "special" isogenies

What happens when we iterate such a "special" isogeny?

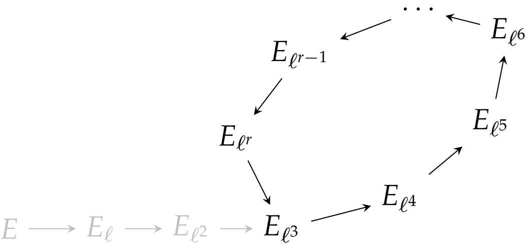


- Fact: Each curve has only one other rational ℓ-isogeny.

!! Reverse arrows are unique; the "tail" $E \to E_{\ell^3}$ cannot exist.

Cycles from "special" isogenies

What happens when we iterate such a "special" isogeny?



- Fact: Each curve has only one other rational ℓ-isogeny.

!! Reverse arrows are unique; the "tail" $E \to E_{\ell^3}$ cannot exist.

$\implies$ The "special" isogenies φ_ℓ form isogeny cycles!

Compatible cycles from "special" isogenies

What happens when we compose those "special" isogenies?

Compatible cycles from “special” isogenies

What happens when we compose those “special” isogenies?

$$
\begin{tikzcd}
 & E_{m\ell} & E_{\ell m} \\
E_m \arrow[ur, "\varphi'_\ell"] & E_\ell \arrow[ur, "\varphi_m"] & \\
E \arrow[u, "\varphi'_m"] \arrow[ur, "\varphi_\ell"] & &
\end{tikzcd}
$$

Compatible cycles from "special" isogenies

What happens when we compose those "special" isogenies?

$$
\begin{tikzcd}
 & & E_{m\ell} & E_{\ell m} \\
E_m \arrow[urr, "\varphi'_\ell"] & & & \\
 & & E_\ell \arrow[uur, "\varphi_m"] & \\
E \arrow[uu, "\varphi'_m"] \arrow[urr, "\varphi_\ell"] & & &
\end{tikzcd}
$$

- Fact: $\ker(\varphi'_\ell \circ \varphi'_m) = \ker(\varphi_m \circ \varphi_\ell) = \langle \ker \varphi_\ell, \ker \varphi'_m \rangle$.

Compatible cycles from “special” isogenies

What happens when we compose those “special” isogenies?

$$\begin{tikzcd} E_m \arrow[r, "\varphi'_\ell"] & E_{\ell m} \\ E \arrow[u, "\varphi'_m"] \arrow[r, "\varphi_\ell"] & E_\ell \arrow[u, "\varphi_m"'] \end{tikzcd}$$

- Fact: $\ker(\varphi'_\ell \circ \varphi'_m) = \ker(\varphi_m \circ \varphi_\ell) = \langle \ker \varphi_\ell, \ker \varphi'_m \rangle$.

!! The order cannot matter $\implies$ cycles must be compatible.

CSIDH in one slide

CSIDH in one slide

- Choose some small odd primes $\ell_1, ..., \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.

CSIDH in one slide

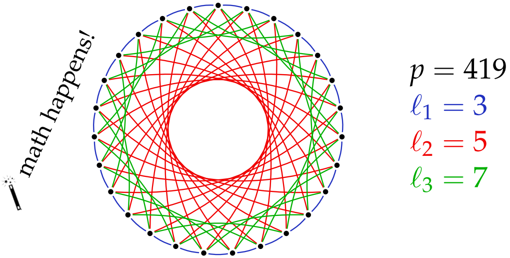
- Choose some small odd primes $\ell_1, ..., \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.
- Let $X = \{y^2 = x^3 + Ax^2 + x \text{ supersingular with } A \in \mathbb{F}_p\}$.

CSIDH in one slide

- Choose some small odd primes $\ell_1, ..., \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.
- Let $X = \{y^2 = x^3 + Ax^2 + x$ supersingular with $A \in \mathbb{F}_p\}$.
- Look at the "special" ℓ_i-isogenies within X.

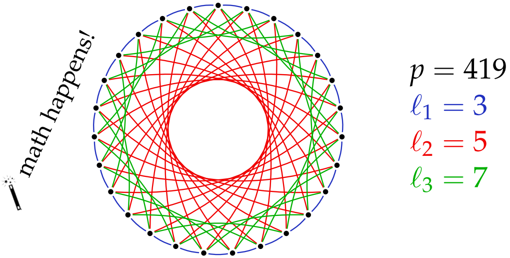
CSIDH in one slide

- Choose some small odd primes $\ell_1, \ldots, \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.
- Let $X = \{y^2 = x^3 + Ax^2 + x$ supersingular with $A \in \mathbb{F}_p\}$.
- Look at the "special" ℓ_i-isogenies within X.



CSIDH in one slide

- Choose some small odd primes $\ell_1, \ldots, \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.
- Let $X = \{y^2 = x^3 + Ax^2 + x \text{ supersingular with } A \in \mathbb{F}_p\}$.
- Look at the "special" ℓ_i-isogenies within X.



- Walking "left" and "right" on any ℓ_i-subgraph is efficient.

Walking in the CSIDH graph (in SageMath)

```
sage: E = EllipticCurve(GF(419^2), [1,0])
sage: E
Elliptic Curve defined by y^2 = x^3 + x
         over Finite Field in z2 of size 419^2
```

Walking in the CSIDH graph (in SageMath)

```
sage: E = EllipticCurve(GF(419^2), [1,0])
sage: E
Elliptic Curve defined by y^2 = x^3 + x
          over Finite Field in z2 of size 419^2
sage: while True:
....:     x = GF(419).random_element()
....:     try:
....:         P = E.lift_x(x)
....:     except ValueError: continue
....:     if P[1] in GF(419):   # "right" step: invert
....:         break
....:
sage: P
(218 : 403 : 1)
```

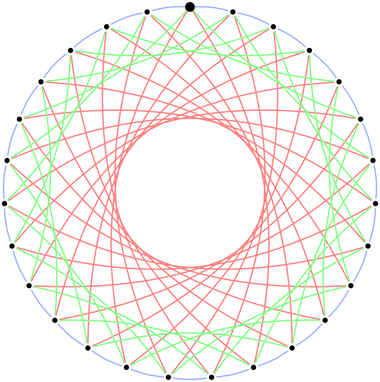
Walking in the CSIDH graph (in SageMath)

```
sage: E = EllipticCurve(GF(419^2), [1,0])
sage: E
Elliptic Curve defined by y^2 = x^3 + x
        over Finite Field in z2 of size 419^2
sage: while True:
....:     x = GF(419).random_element()
....:     try:
....:         P = E.lift_x(x)
....:     except ValueError: continue
....:     if P[1] in GF(419):    # "right" step: invert
....:         break
....:
sage: P
(218 : 403 : 1)
sage: P.order().factor()
2 * 3 * 7
sage: EE = E.isogeny_codomain(2*3*P)  # "left" 7-step
sage: EE
Elliptic Curve defined by y^2 = x^3 + 285*x + 87
        over Finite Field in z2 of size 419^2
```

CSIDH key exchange

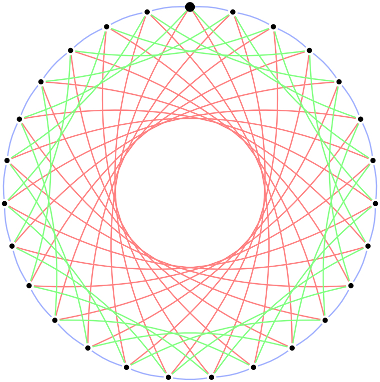
Alice

[+, +, −, −]



Bob

[−, +, −, −]



CSIDH key exchange

Alice

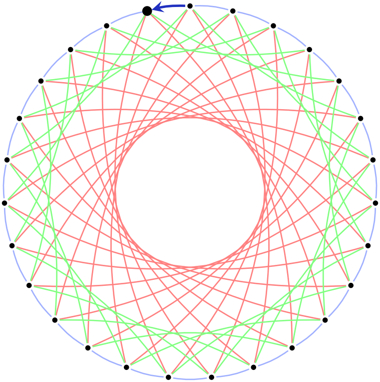
[+, +, −, −]

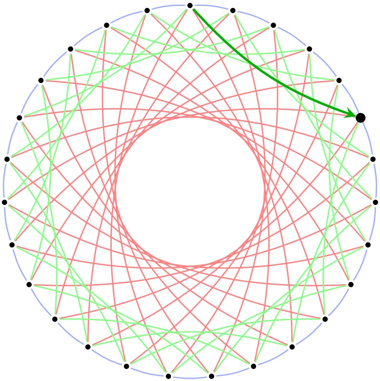
↑

Bob

[−, +, −, −]

↑

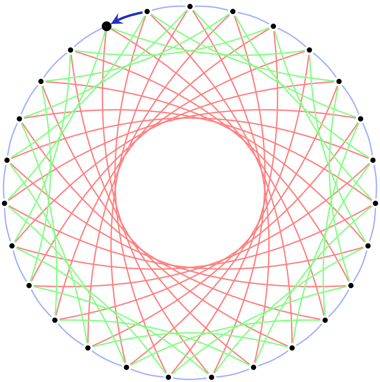




CSIDH key exchange

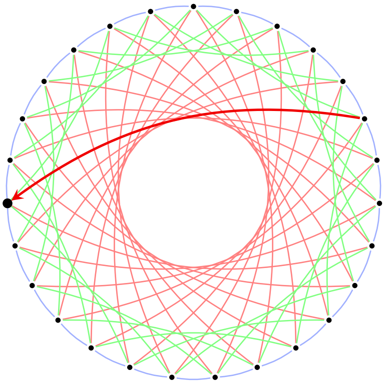
Alice

$[+, +, -, -]$



Bob

$[-, +, -, -]$



CSIDH key exchange

Alice

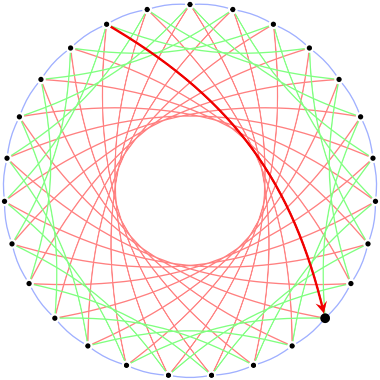
$[\textcolor{blue}{+}, \textcolor{blue}{+}, \textcolor{red}{-}, \textcolor{green}{-}]$

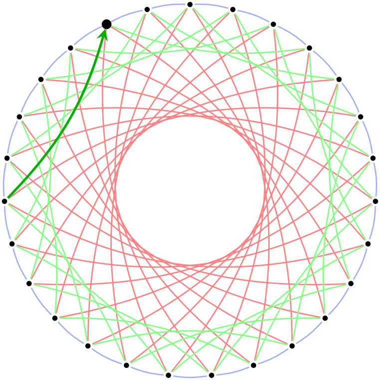
$\uparrow$

Bob

$[\textcolor{green}{-}, \textcolor{red}{+}, \textcolor{green}{-}, \textcolor{blue}{-}]$

$\uparrow$





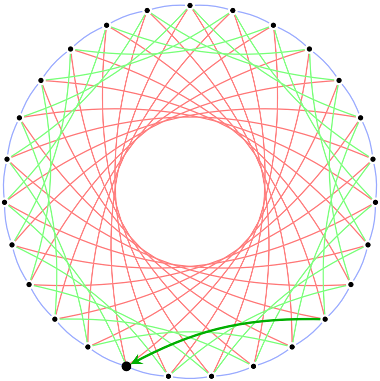
CSIDH key exchange

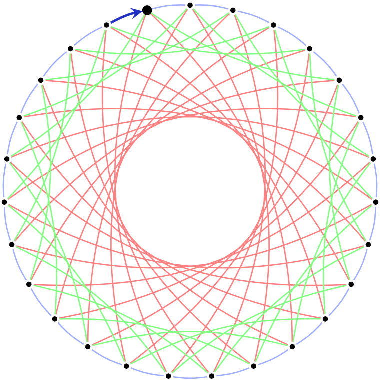
Alice

$[+, +, -, -]$
$\uparrow$

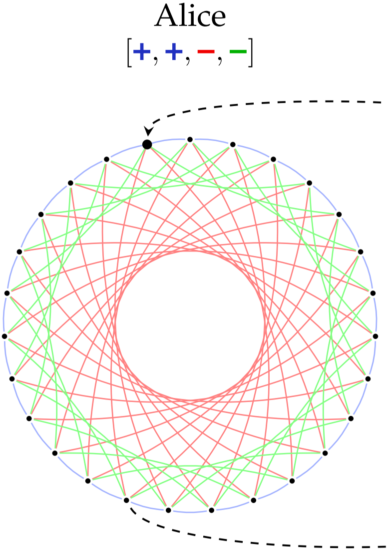
Bob

$[-, +, -, -]$
$\uparrow$





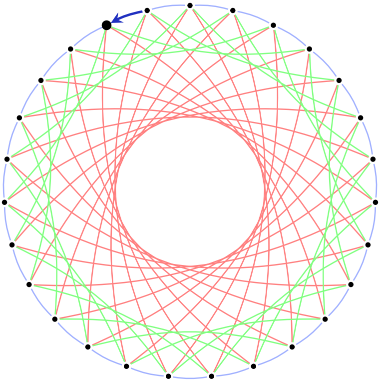
CSIDH key exchange



CSIDH key exchange

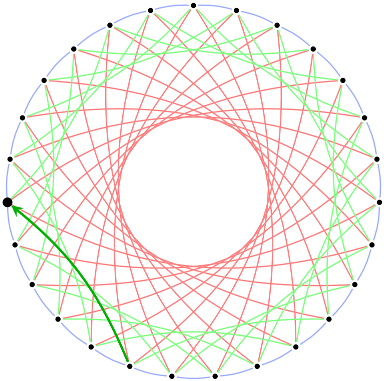
Alice

[+, +, −, −]



Bob

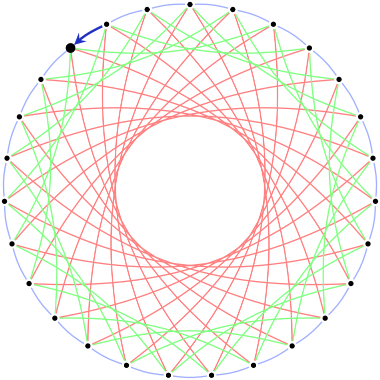
[−, +, −, −]



CSIDH key exchange

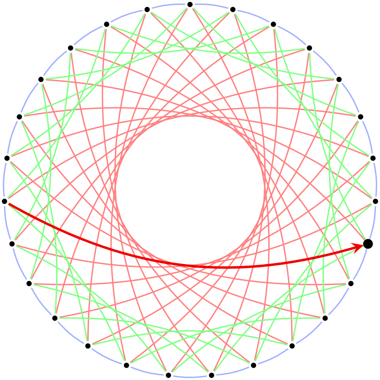
Alice

$[+, +, -, -]$



Bob

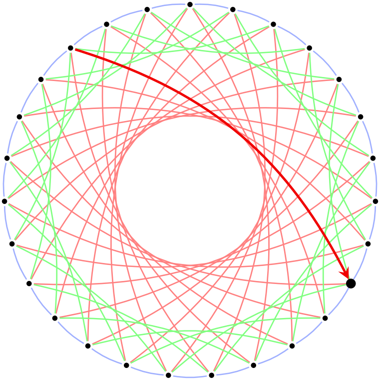
$[-, +, -, -]$



CSIDH key exchange

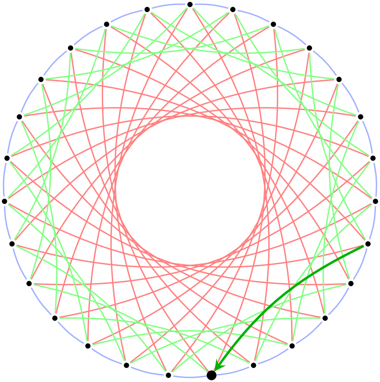
Alice

$[+, +, -, -]$
$\uparrow$



Bob

$[-, +, -, -]$
$\uparrow$

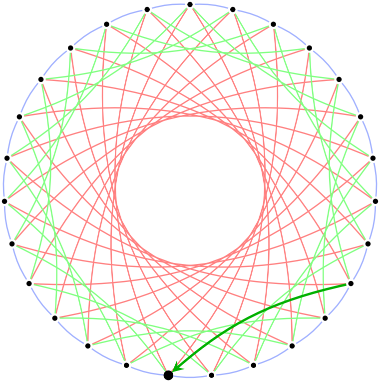


CSIDH key exchange

Alice

$[+, +, -, -]$

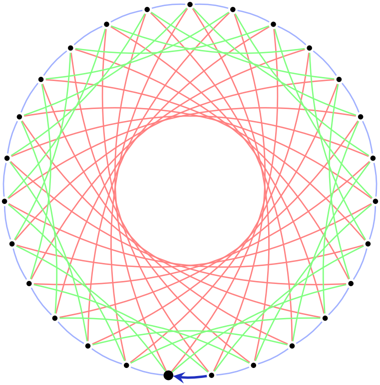
$\uparrow$



Bob

$[-, +, -, -]$

$\uparrow$



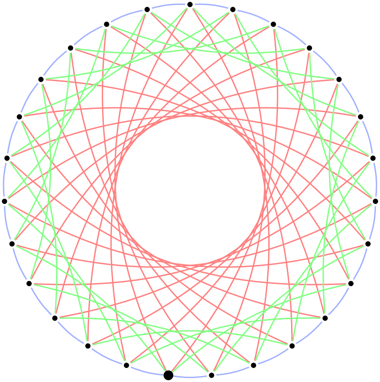
CSIDH key exchange

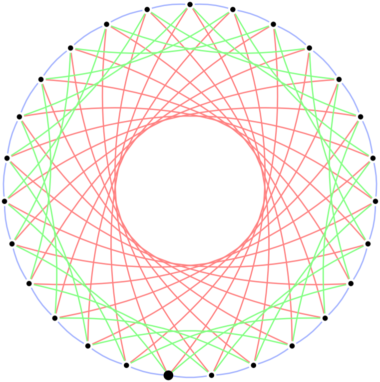
Alice

[+, +, −, −]

Bob

[−, +, −, −]





Action!

Cycles are compatible: [right then left] = [left then right]

Action!

Cycles are compatible: [right then left] = [left then right]
$\rightsquigarrow$ only need to keep track of total step counts for each ℓ_i.

Example: [**+**, **+**, **−**, **−**, **−**, **+**, **−**, **−**] just becomes $(+1, \quad 0, -3) \in \mathbb{Z}^3$.

Action!

Cycles are compatible: [right then left] = [left then right]
$\rightsquigarrow$ only need to keep track of total step counts for each ℓ_i.

Example: [**+**, **+**, **−**, **−**, **−**, **+**, **−**, **−**] just becomes $(+1, \quad 0, -3) \in \mathbb{Z}^3$.

There is a group action of $(\mathbb{Z}^n, +)$ on our set of curves X!

(An action of a group $(G, \cdot)$ on a set X is a map $*\colon G \times X \to X$
such that $id * x = x$ and $g * (h * x) = (g \cdot h) * x$ for all $g, h \in G$ and $x \in X$.)

Action!

Cycles are compatible: [right then left] = [left then right]
$\rightsquigarrow$ only need to keep track of total step counts for each ℓ_i.
Example: $[+, +, -, -, -, +, -, -]$ just becomes $(+1, \quad 0, -3) \in \mathbb{Z}^3$.

There is a group action of $(\mathbb{Z}^n, +)$ on our set of curves X!

(An action of a group $(G, \cdot)$ on a set X is a map $*\colon G \times X \to X$
such that $id * x = x$ and $g * (h * x) = (g \cdot h) * x$ for all $g, h \in G$ and $x \in X$.)

!! We understand the structure: By complex-multiplication theory, the quotient $\mathbb{Z}^n/\ker$ is the ideal-class group $\mathrm{cl}(\mathbb{Z}[\sqrt{-p}])$.

Action! 🎬

Cycles are compatible: [right then left] = [left then right]
$\rightsquigarrow$ only need to keep track of total step counts for each ℓ_i.

Example: $[\mathbf{+}, \mathbf{+}, \mathbf{-}, \mathbf{-}, \mathbf{-}, \mathbf{+}, \mathbf{-}, \mathbf{-}]$ just becomes $(+1, \quad 0, -3) \in \mathbb{Z}^3$.

There is a group action of $(\mathbb{Z}^n, +)$ on our set of curves X!

(An action of a group $(G, \cdot)$ on a set X is a map $*\colon G \times X \to X$
such that $id * x = x$ and $g * (h * x) = (g \cdot h) * x$ for all $g, h \in G$ and $x \in X$.)

!! We understand the structure: By complex-multiplication theory, the quotient $\mathbb{Z}^n/\ker$ is the ideal-class group $\mathrm{cl}(\mathbb{Z}[\sqrt{-p}])$.

!! This group characterizes *when two paths lead to the same curve*.

CSIDH: Where things stand

- Key sizes: Public keys are 4λ bits, where λ is the *classical* security level. (For λ-bit *quantum* security, need $\Theta(\lambda^2)$ bits.)

CSIDH: Where things stand

- Key sizes: Public keys are 4λ bits, where λ is the *classical* security level. (For λ-bit *quantum* security, need $\Theta(\lambda^2)$ bits.)
- Quantum security: Asymptotically $\exp\big((\log p)^{1/2+o(1)}\big)$ due to Kuperberg's quantum algorithm.

⚠ Concrete security estimates vary wildly.

CSIDH: Where things stand

- Key sizes: Public keys are 4λ bits, where λ is the *classical* security level. (For λ-bit *quantum* security, need $\Theta(\lambda^2)$ bits.)
- Quantum security: Asymptotically $\exp\big((\log p)^{1/2+o(1)}\big)$ due to Kuperberg's quantum algorithm.

  ⚠ Concrete security estimates vary wildly.
- Performance: Some tens of milliseconds per group-action evaluation at the 128-bit *classical* security level.

CSIDH: Where things stand

- Key sizes: Public keys are 4λ bits, where λ is the *classical* security level. (For λ-bit *quantum* security, need $\Theta(\lambda^2)$ bits.)
- Quantum security: Asymptotically $\exp\big((\log p)^{1/2+o(1)}\big)$ due to Kuperberg's quantum algorithm.

  ⚠ Concrete security estimates vary wildly.
- Performance: Some tens of milliseconds per group-action evaluation at the 128-bit *classical* security level.
- New: "Clapoti" — a polynomial-time algorithm for arbitrary combinations of operations in the group and evaluations of the action.

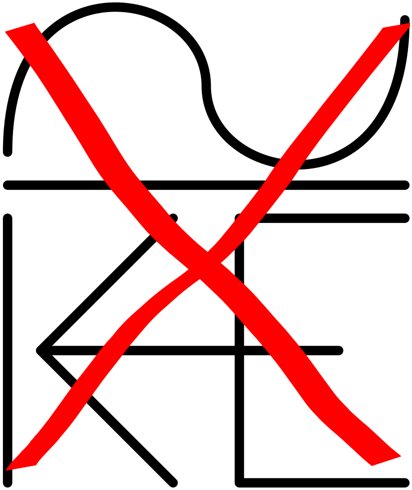
  (Previously, only restricted sequences of operations were efficient.)

Plan for this talk

- Some high-level intuition. ✓
- Elliptic curves & isogenies. ✓
- The CSIDH non-interactive key exchange. ✓
- The SIKE attacks.
- The SQIsign signature scheme.

SIDH/SIKE

SIDH/SIKE



SIDH/SIKE

...*was* another well-known isogeny-based key exchange scheme:

- The "isogeny poster child" from ≈ 2011 to ≈ 2022.
- Part of NISTPQC, which found no security flaws.

SIDH/SIKE

…*was* another well-known isogeny-based key exchange scheme:

- The "isogeny poster child" from ≈ 2011 to ≈ 2022.
- Part of NISTPQC, which found no security flaws.

It was catastrophically broken in 2022.

The SIDH/SIKE attacks

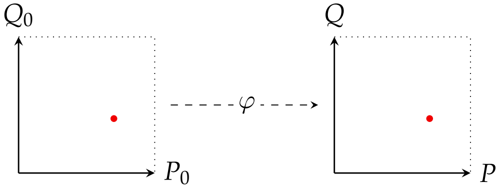
- **Not** a case of everyone overlooking something stupid.

The SIDH/SIKE attacks

- **Not** a case of everyone overlooking something stupid.
- The attack uses an unexpected profound new technique.

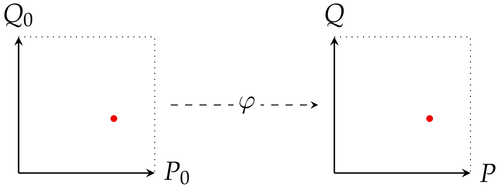
The SIDH/SIKE attacks

- **Not** a case of everyone overlooking something stupid.
- The attack uses an unexpected profound new technique.
- SIKE revealed how a secret isogeny acts on lots of points.



The SIDH/SIKE attacks

- **<u>Not</u>** a case of everyone overlooking something stupid.
- The attack uses an unexpected profound new technique.
- SIKE revealed how a secret isogeny acts on lots of points.

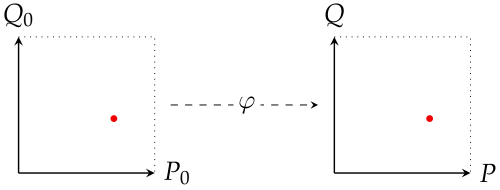


This **isogeny interpolation** problem turns out to be **easy!**

(at least in some cases — it's complicated, etc., etc.)

The SIDH/SIKE attacks

- **Not** a case of everyone overlooking something stupid.
- The attack uses an unexpected profound new technique.
- SIKE revealed how a secret isogeny acts on lots of points.

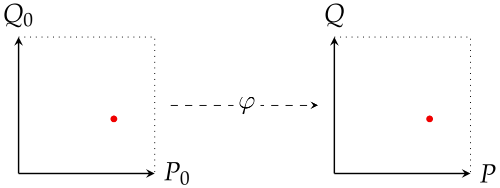


This **isogeny interpolation** problem turns out to be **easy!**
(at least in some cases — it's complicated, etc., etc.)

- It has since found groundbreaking constructive uses.
- The general **isogeny problem** is entirely unaffected!

The SIDH/SIKE attacks

- **Not** a case of everyone overlooking something stupid.
- The attack uses an unexpected profound new technique.
- SIKE revealed how a secret isogeny acts on lots of points.



This **isogeny interpolation** problem turns out to be **easy!**

(at least in some cases — it's complicated, etc., etc.)

- It has since found groundbreaking constructive uses.
- The general **isogeny problem** is entirely unaffected!

$\rightsquigarrow$ The best thing to ever happen to isogenies! 🎉

SoK: Isogeny problems

Is SIKE broken yet?

Home About

Schemes

Name	Type	Classical Security	Quantum Security	References	Additional Information
▹ SIDH	Key Exchange	$\tilde{O}(n^3)$	$\tilde{O}(n^3)$	JDF11 DJP14 CLN16	▹ Comment
SIKE	KEM	$\tilde{O}(n^3)$	$\tilde{O}(n^3)$	SIKE	▹ Comment
B-SIDH	Key Exchange	$\tilde{O}(n^3)$	$\tilde{O}(n^3)$	Cos19	▹ Comment
CRS	Key Exchange, Non Interactive Key Exchange	$\exp(n)^{1/2}$	L(1/2)	Cou06 RS06 DKS18	▹ Comment
CSIDH	Key Exchange, Non Interactive Key Exchange	$\exp(n)^{1/2}$	L(1/2)	CL+18 CD19	▹ Comment

https://issikebrokenyet.github.io

Plan for this talk

- Some high-level intuition. ✓
- Elliptic curves & isogenies. ✓
- The CSIDH non-interactive key exchange. ✓
- The SIKE attacks. ✓
- The SQIsign signature scheme.

SQIsign: What?

SQIsign

https://sqisign.org

SQIsign: What?

https://sqisign.org

- A new and very hot post-quantum signature scheme.
- Based on a super cool part of number theory/geometry. $\ddot\smile$

More "special" isogenies

- Earlier: "Special" isogenies φ_ℓ with rational kernel points.

More "special" isogenies

- Earlier: "Special" isogenies φ_ℓ with rational kernel points.
- In other words: $\ker \varphi_\ell = \ker[\ell] \cap \ker(\pi - 1)$.
  (Here π is the Frobenius endomorphism $\pi \colon (x, y) \mapsto (x^p, y^p)$.)

More "special" isogenies

- Earlier: "Special" isogenies φ_ℓ with rational kernel points.
- In other words: $\ker\varphi_\ell = \ker[\ell] \cap \ker(\pi - 1)$.
  (Here π is the Frobenius endomorphism $\pi\colon (x,y) \mapsto (x^p, y^p)$.)

!! Over $\mathbb{F}_{p^2}$, we can have more endomorphisms.
Example: $y^2 = x^3 + x$ has $\iota\colon (x,y) \mapsto (-x, \sqrt{-1}\cdot y)$.

More "special" isogenies

- Earlier: "Special" isogenies φ_ℓ with rational kernel points.
- In other words: $\ker \varphi_\ell = \ker[\ell] \cap \ker(\pi - 1)$.
  (Here π is the Frobenius endomorphism $\pi\colon (x, y) \mapsto (x^p, y^p)$.)

!! Over $\mathbb{F}_{p^2}$, we can have more endomorphisms.
Example: $y^2 = x^3 + x$ has $\iota\colon (x, y) \mapsto (-x, \sqrt{-1} \cdot y)$.

- Extremely non-obvious fact in this setting:

Every isogeny $\varphi\colon E \to E'$ comes from a subset $I_\varphi \subseteq \mathrm{End}(E)$.

More "special" isogenies

- Earlier: "Special" isogenies φ_ℓ with rational kernel points.
- In other words: $\ker \varphi_\ell = \ker[\ell] \cap \ker(\pi - 1)$.
  (Here π is the Frobenius endomorphism $\pi\colon (x, y) \mapsto (x^p, y^p)$.)

!! Over $\mathbb{F}_{p^2}$, we can have more endomorphisms.
Example: $y^2 = x^3 + x$ has $\iota\colon (x, y) \mapsto (-x, \sqrt{-1} \cdot y)$.

- Extremely non-obvious fact in this setting:

Every isogeny $\varphi\colon E \to E'$ comes from a subset $I_\varphi \subseteq \mathrm{End}(E)$.

:) We understand the structure of $\mathrm{End}(E)$.

More “special” isogenies

- Earlier: “Special” isogenies φ_ℓ with rational kernel points.
- In other words: $\ker \varphi_\ell = \ker[\ell] \cap \ker(\pi - 1)$.
  (Here π is the Frobenius endomorphism $\pi\colon (x, y) \mapsto (x^p, y^p)$.)

!! Over $\mathbb{F}_{p^2}$, we can have more endomorphisms.
Example: $y^2 = x^3 + x$ has $\iota\colon (x, y) \mapsto (-x, \sqrt{-1} \cdot y)$.

- Extremely non-obvious fact in this setting:

Every isogeny $\varphi\colon E \to E'$ comes from a subset $I_\varphi \subseteq \mathrm{End}(E)$.

:-) We understand the structure of $\mathrm{End}(E)$.

:-) We understand how I_φ, I_ψ relate for isogenies $\varphi, \psi\colon E \to E'$.
(NB: Same E'.)

The Deuring correspondence

...is the formal version of what I just said.

The Deuring correspondence

...is the formal version of what I just said.

...is a strong connection between two *a priori* very different worlds:

The Deuring correspondence

...is the formal version of what I just said.

...is a strong connection between two *a priori* very different worlds:

- Supersingular elliptic curves defined over $\mathbb{F}_{p^2}$.

The Deuring correspondence

...is the formal version of what I just said.

...is a strong connection between two $\overset{\textit{a priori}}{\curlyvee}$ very different worlds:

- Supersingular elliptic curves defined over $\mathbb{F}_{p^2}$.
- Quaternions: Maximal orders in a certain algebra $B_{p,\infty}$.

The Deuring correspondence

...is the formal version of what I just said.

...is a strong connection between two $\overset{\textit{a priori}}{\curlyvee}$ very different worlds:

- Supersingular elliptic curves defined over $\mathbb{F}_{p^2}$.
- Quaternions: Maximal orders in a certain algebra $B_{p,\infty}$.

Isogenies become "connecting ideals" in quaternion land.

The Deuring correspondence

...is the formal version of what I just said.

...is a strong connection between two *a priori* very different worlds:

- Supersingular elliptic curves defined over $\mathbb{F}_{p^2}$.
- Quaternions: Maximal orders in a certain algebra $B_{p,\infty}$.

Isogenies become “connecting ideals” in quaternion land.

☺ One direction is easy, the other seems hard! $\rightsquigarrow$ *Cryptography!*

The Deuring correspondence (examples)

Let $p = 7799999$ and let $\mathbf{i}, \mathbf{j}$ satisfy $\mathbf{i}^2 = -1,\ \mathbf{j}^2 = -p,\ \mathbf{ji} = -\mathbf{ij}$.

The ring $\mathcal{O}_0 = \mathbb{Z} \oplus \mathbb{Z}\,\mathbf{i} \oplus \mathbb{Z}\,\frac{\mathbf{i}+\mathbf{j}}{2} \oplus \mathbb{Z}\,\frac{1+\mathbf{ij}}{2}$
corresponds to the curve $E_0\colon\ y^2 = x^3 + x$.

The ring $\mathcal{O}_1 = \mathbb{Z} \oplus \mathbb{Z}\,4947\mathbf{i} \oplus \mathbb{Z}\,\frac{4947\mathbf{i}+\mathbf{j}}{2} \oplus \mathbb{Z}\,\frac{4947+32631010\mathbf{i}+\mathbf{ij}}{9894}$
corresponds to the curve $E_1\colon\ y^2 = x^3 + 1$.

The ideal $I = \mathbb{Z}\,4947 \oplus \mathbb{Z}\,4947\mathbf{i} \oplus \mathbb{Z}\,\frac{598+4947\mathbf{i}+\mathbf{j}}{2} \oplus \mathbb{Z}\,\frac{4947+598\mathbf{i}+\mathbf{ij}}{2}$
defines an isogeny $E_0 \to E_1$ of degree $4947 = 3 \cdot 17 \cdot 97$.

Signing with isogenies

- Fiat–Shamir: signature scheme from identification scheme.

$$E_0 \xrightarrow{\textit{secret}} E_A$$

Signing with isogenies

- Fiat–Shamir: signature scheme from identification scheme.

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "secret"] \arrow[d, dashed, "commitment"'] & E_A \\
E_1 &
\end{tikzcd}$$

Signing with isogenies

- <u>Fiat–Shamir</u>: signature scheme from identification scheme.

$$
\begin{tikzcd}
E_0 \arrow[rrr, dashed, "\textit{secret}"] \arrow[d, dashed, "\textit{commitment}"'] & & & E_A \\
E_1 \arrow[rrr, "\textit{challenge}"'] & & & E_2
\end{tikzcd}
$$

Signing with isogenies

- Fiat–Shamir: signature scheme from identification scheme.

$$\begin{tikzcd}[column sep=huge, row sep=huge]
E_0 \arrow[r, dashed, "\textit{secret}"] \arrow[d, dashed, "\textit{commitment}"'] & E_A \arrow[d, very thick, "\textit{signature}"] \\
E_1 \arrow[r, "\textit{challenge}"'] & E_2
\end{tikzcd}$$

Signing with isogenies

- Fiat–Shamir: signature scheme from identification scheme.

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "secret"] \arrow[d, dashed, "commitment"'] & E_A \arrow[d, "signature"] \\
E_1 \arrow[r, "challenge"'] & E_2
\end{tikzcd}$$

- Easy signature: $E_A \to E_0 \to E_1 \to E_2$. *Obviously broken.*

Signing with isogenies

- Fiat–Shamir: signature scheme from identification scheme.

$$
\begin{tikzcd}
E_0 \arrow[r, dashed, "\textit{secret}"] \arrow[d, dashed, "\textit{commitment}"'] & E_A \arrow[d, "\textit{signature}"] \\
E_1 \arrow[r, "\textit{challenge}"'] & E_2
\end{tikzcd}
$$

- Easy signature: $E_A \to E_0 \to E_1 \to E_2$. *Obviously broken.*
- **SQIsign**'s solution: Construct new path $E_A \to E_2$ (using *secret*).

Signing with isogenies

- Fiat–Shamir: signature scheme from identification scheme.

$$
\begin{tikzcd}
E_0 \arrow[r, dashed, "\textit{secret}"] \arrow[d, dashed, "\textit{commitment}"'] & E_A \arrow[d, thick, "\textit{signature}"] \\
E_1 \arrow[r, "\textit{challenge}"'] & E_2
\end{tikzcd}
$$

- Easy signature: $E_A \to E_0 \to E_1 \to E_2$. *Obviously broken.*
- **SQIsign**'s solution: Construct new path $E_A \to E_2$ (using *secret*).
- It relies on an explicit form of the Deuring correspondence.

SQIsign: Why?

- + It's extremely small compared to the competition.
- – It's relatively slow compared to the competition.
- + ...but performance is getting better by the $\approx$ week!

SQIsign: Why?

- \+ It's extremely <u>small</u> compared to the competition.
- – It's relatively <u>slow</u> compared to the competition.
- \+ ...but performance is getting better by the $\approx$ week!

SQIsign (original version): Numbers

sizes

parameter set	public keys	signatures
NIST-**I**	**64** bytes	**177** bytes
NIST-**III**	**96** bytes	**263** bytes
NIST-**V**	**128** bytes	**335** bytes

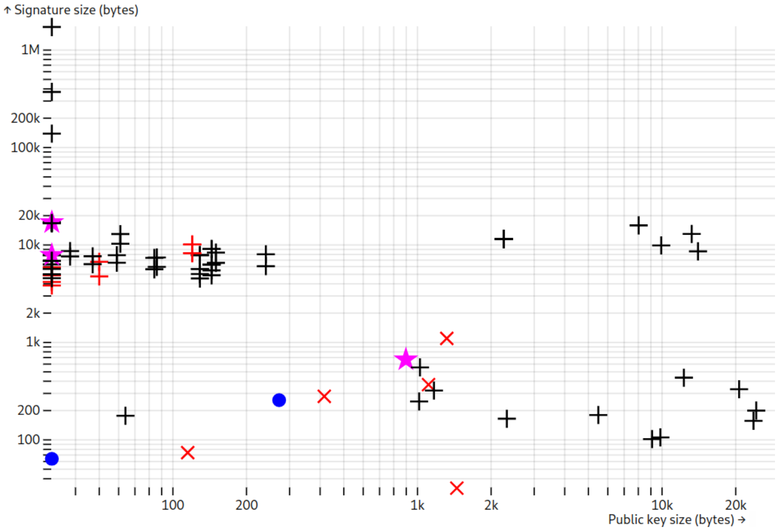
performance

Cycle counts for a *generic C implementation* running on an Intel *Ice Lake* CPU.
Optimizations are certainly possible and work in progress.

parameter set	keygen	signing	verifying
NIST-**I**	**3728** megacycles	**5779** megacycles	**108** megacycles
NIST-**III**	**23734** megacycles	**43760** megacycles	**654** megacycles
NIST-**V**	**91049** megacycles	**158544** megacycles	**2177** megacycles

Source: `https://sqisign.org`

SQIsign (original version): Comparison



Source: `https://pqshield.github.io/nist-sigs-zoo`

SQIsign**2D-West**: New and dramatically improved!

Table 1. Parameter sizes and performance of SQIsign2D-West. Average running times computed using an Intel Xeon Gold 6338 (Ice Lake, 2GHz) using finite field arithmetic optimised for the x64 architecture, turbo boost disabled. See Section 7 for details.

	Sizes (bytes)		Timings (ms)		
	Public key	Signature	Keygen	Sign	Verify
NIST I	66	148	30	80	4.5
NIST III	98	222	85	230	14.5
NIST V	130	294	180	470	31.0

SQIsign**2D-West**: New and dramatically improved!

Table 1. Parameter sizes and performance of SQIsign2D-West. Average running times computed using an Intel Xeon Gold 6338 (Ice Lake, 2GHz) using finite field arithmetic optimised for the x64 architecture, turbo boost disabled. See Section 7 for details.

	Sizes (bytes)		Timings (ms)		
	Public key	Signature	Keygen	Sign	Verify
NIST I	66	148	30	80	4.5
NIST III	98	222	85	230	14.5
NIST V	130	294	180	470	31.0

- The $\approx 10\times$ speedup over the original version of SQIsign comes from the new tools underlying the SIKE attacks.

Plan for this talk

- Some high-level intuition. ✓
- Elliptic curves & isogenies. ✓
- The CSIDH non-interactive key exchange. ✓
- The SIKE attacks. ✓
- The SQIsign signature scheme. ✓

Questions?

(Also feel free to email me: `lorenz@yx7.cc`)