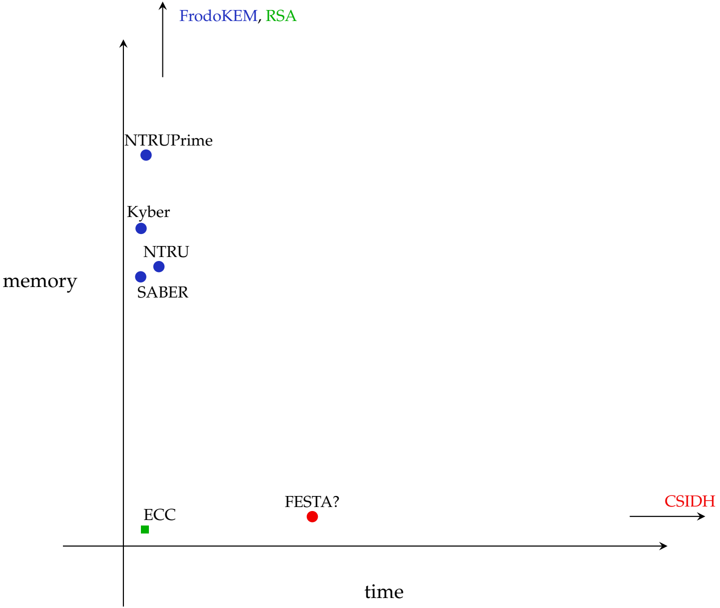
Isogeny-based cryptography: why, how, and what next?

Chloe Martindale

University of Bristol

ASCrypto, Ecuador

Zoo of lattice- and isogeny-based KEMs



Applications (non-exhaustive list)

	Lattices	Isogenies
KEM	✓	✓
Signatures	✓	✓
NIKE	(✓)	✓
FHE	✓	×
IBE	(✓)	×
Threshold	✓	✓
OPRF	✓	✓
VDF	(×)	(✓)
VRF	(✓)	(✓)

Big picture

- Isogenies are a source of exponentially-sized graphs.
- We can walk efficiently on these graphs.
- Fast mixing: short paths to (almost) all nodes.
- No known efficient algorithms to recover paths from endpoints.
- Enough structure to navigate the graph meaningfully.
  That is: some *well-behaved* 'directions' to describe paths. More later.

It is easy to construct graphs that satisfy *almost* all of these — not enough for crypto!

[ˈsiːˌsaɪd]

Recall: Diffie–Hellman key exchange '76

Public parameters:

- a finite group G (typically $\mathbb{F}_q^*$ or $E(\mathbb{F}_q)$)
- an element $g \in G$ of (large) prime order p

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

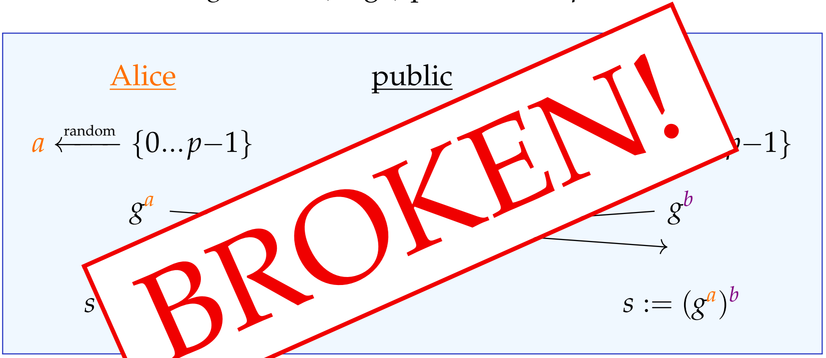
The Discrete Logarithm Problem, finding a given g and g^a, should be hard[1] in $\langle g \rangle$.

[1] Complexity (at least) subexponential in $\log(p)$.

Recall: Diffie–Hellman key exchange '76

Public parameters:

- a finite group G (typically $\mathbb{F}_q^*$ or $E(\mathbb{F}_q)$)
- an element $g \in G$ of (large) prime order p



The Discrete Logarithm Problem, finding a given g and g^a, should be hard[1] in $\langle g \rangle$.

[1] Complexity (at least) subexponential in $\log(p)$.

Quantumifying Exponentiation

- Idea to replace DLP: replace exponentiation

$$\begin{array}{ccc} \mathbb{Z} \times G & \to & G \\ (x, g) & \mapsto & g^x \end{array}$$

by a group action on a set.

Quantumifying Exponentiation

- Idea to replace DLP: replace exponentiation

$$\begin{array}{rcl} \mathbb{Z} \times G & \to & G \\ (x, g) & \mapsto & g^x \end{array}$$

by a group action on a set.

- Replace G by the set S of supersingular elliptic curves $E_A : y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_{419}$.
- Replace $\mathbb{Z}$ by a commutative group H that acts via isogenies.
- The action of $h \in H$ on S moves the elliptic curves one step around one of the cycles.

Couveignes-Rostovstev-Stolbunov key exchange

Public parameters:

- the finite set S (of some special $E/\mathbb{F}_q$),
- an element $E \in S$,
- the group H; acts on S via $*$.

Alice	public	Bob
$\alpha \xleftarrow{\text{random}} H$		$\beta \xleftarrow{\text{random}} H$
$\alpha * E$	$\longrightarrow$ / $\longleftarrow$	$\beta * E$
$s := \alpha * (\beta * E)$		$s := \beta * (\alpha * E)$

Finding α given E and $\alpha * E$, should be hard.[2]

[2] Complexity (at least) subexponential in $\log(\#S)$.

From CRS to CSIDH

1997 Couveignes proposes the now-CRS scheme.

- Uses ordinary elliptic curves/$\mathbb{F}_p$ with same end ring.
- Paper is rejected and forgotten.

2004 Rostovstev, Stolbunov rediscover now-CRS scheme.

- Best known quantum and classical attacks are exponential.

2005 Kuperberg: quantum subexponential attack for the dihedral hidden subgroup problem.

2010 Childs, Jao, Soukharev apply Kuperberg to CRS.

- Secure parameters $\rightsquigarrow$ key exchange of 20 minutes.

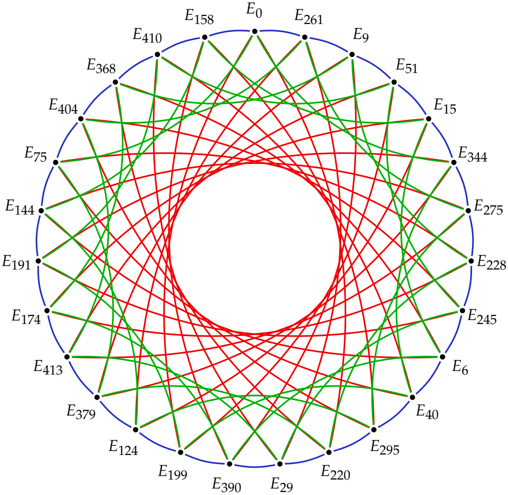
2011 Jao, De Feo propose SIDH [more to come!].

2017 De Feo, Kieffer, Smith use modular curves to do a CRS key exchange in 8 minutes.

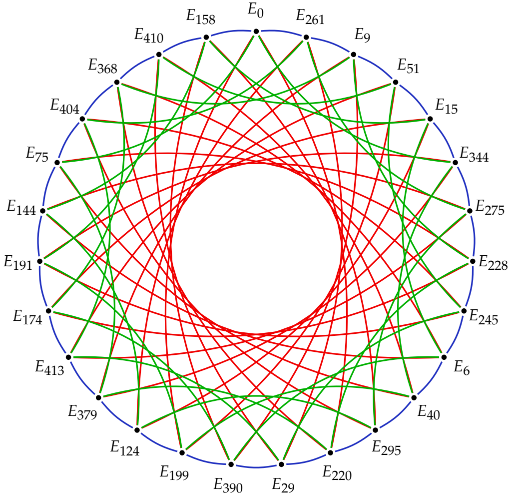
2018 Castryck, Lange, M., Panny, Renes propose CSIDH.

- CRS but with supersingular elliptic curves $/\mathbb{F}_p$.
- p constructed to make scheme efficient.
- Key exchange runs in 60ms.

Isogeny graphs at the CSIDH

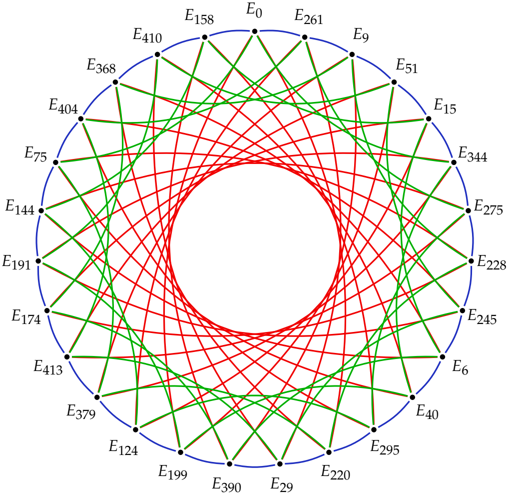


Isogeny graphs at the CSIDH



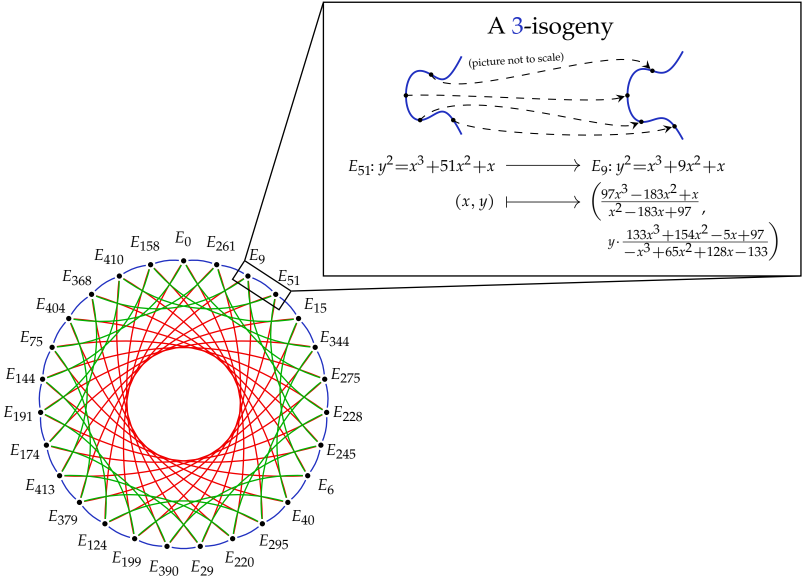
Nodes: Supersingular curves $E_A : y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_{419}$.

Isogeny graphs at the CSIDH



Nodes: Supersingular curves $E_A : y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_{419}$.
Edges: 3-, 5-, and 7-isogenies.

Graphs of elliptic curves



Compute neighbours in the graph

To compute a neighbour of E, we have to compute an ℓ-isogeny from E. To do this:

- Find a point P of order ℓ on E.

- Compute the isogeny with kernel $\{P, 2P, \ldots, \ell P\}$ using Vélu's formulas* (implemented in Sage).

Compute neighbours in the graph

To compute a neighbour of E, we have to compute an ℓ-isogeny from E. To do this:

- Find a point P of order ℓ on E.
  - Let $E/\mathbb{F}_p$ be supersingular and $p \geq 5$.
- Compute the isogeny with kernel $\{P, 2P, \ldots, \ell P\}$ using Vélu's formulas* (implemented in Sage).

Compute neighbours in the graph

To compute a neighbour of E, we have to compute an ℓ-isogeny from E. To do this:

- Find a point P of order ℓ on E.
  - Let $E/\mathbb{F}_p$ be supersingular and $p \geq 5$. Then $E(\mathbb{F}_p) \cong C_{p+1}$ or $C_2 \times C_{(p+1)/2}$.
- Compute the isogeny with kernel $\{P, 2P, \ldots, \ell P\}$ using Vélu's formulas* (implemented in Sage).

Compute neighbours in the graph

To compute a neighbour of E, we have to compute an ℓ-isogeny from E. To do this:

- Find a point P of order ℓ on E.
  - Let $E/\mathbb{F}_p$ be supersingular and $p \geq 5$. Then $E(\mathbb{F}_p) \cong C_{p+1}$ or $C_2 \times C_{(p+1)/2}$.
  - Suppose we have found $P = E(\mathbb{F}_p)$ of order $p+1$ or $(p+1)/2$.
- Compute the isogeny with kernel $\{P, 2P, \ldots, \ell P\}$ using Vélu's formulas* (implemented in Sage).

Compute neighbours in the graph

To compute a neighbour of E, we have to compute an ℓ-isogeny from E. To do this:

- Find a point P of order ℓ on E.
  - Let $E/\mathbb{F}_p$ be supersingular and $p \geq 5$. Then $E(\mathbb{F}_p) \cong C_{p+1}$ or $C_2 \times C_{(p+1)/2}$.
  - Suppose we have found $P = E(\mathbb{F}_p)$ of order $p+1$ or $(p+1)/2$.
  - For every odd prime $\ell | (p+1)$, the point $\frac{p+1}{\ell} P$ is a point of order ℓ.
- Compute the isogeny with kernel $\{P, 2P, \ldots, \ell P\}$ using Vélu's formulas* (implemented in Sage).

Compute neighbours in the graph

To compute a neighbour of E, we have to compute an ℓ-isogeny from E. To do this:

- Find a point P of order ℓ on E.
  - Let $E/\mathbb{F}_p$ be supersingular and $p \geq 5$. Then $E(\mathbb{F}_p) \cong C_{p+1}$ or $C_2 \times C_{(p+1)/2}$.
  - Suppose we have found $P = E(\mathbb{F}_p)$ of order $p+1$ or $(p+1)/2$.
  - For every odd prime $\ell | (p+1)$, the point $\frac{p+1}{\ell} P$ is a point of order ℓ.
- Compute the isogeny with kernel $\{P, 2P, \ldots, \ell P\}$ using Vélu's formulas* (implemented in Sage).
  - Given a $\mathbb{F}_p$-rational point of order ℓ, the isogeny computations can be done over $\mathbb{F}_p$.

Representing nodes of the graph

- Every node of G_{ℓ_i} is

$$E_A : y^2 = x^3 + Ax^2 + x.$$

Representing nodes of the graph

- Every node of G_{ℓ_i} is

$$E_A : y^2 = x^3 + Ax^2 + x.$$

$\Rightarrow$ Can compress every node to a single value $A \in \mathbb{F}_p$.

Representing nodes of the graph

- Every node of G_{ℓ_i} is

$$E_A : y^2 = x^3 + Ax^2 + x.$$

⇒ Can compress every node to a single value $A \in \mathbb{F}_p$.

⇒ Tiny keys!

Does any A work?

[3] This algorithm has a small chance of false positives, but we actually use a variant that *proves* that E_A has $p + 1$ points.

Does any A work?

No.

[3] This algorithm has a small chance of false positives, but we actually use a variant that *proves* that E_A has $p + 1$ points.

Does any A work?

No.

- About $\sqrt{p}$ of all $A \in \mathbb{F}_p$ are valid keys.

[3] This algorithm has a small chance of false positives, but we actually use a variant that *proves* that E_A has $p + 1$ points.

Does any A work?

No.

- About $\sqrt{p}$ of all $A \in \mathbb{F}_p$ are valid keys.
- Public-key validation: Check that E_A has $p + 1$ points.
  Easy Monte-Carlo algorithm: Pick random P on E_A and check $[p + 1]P = \infty$.[3]

[3] This algorithm has a small chance of false positives, but we actually use a variant that *proves* that E_A has $p + 1$ points.

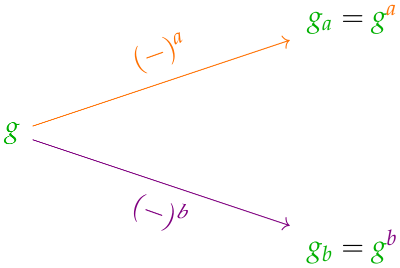
Venturing beyond the CSIDH

A selection of advances since original publication (2018):

- CSURF [CD19]: exploiting 2-isogenies.
- sqrtVelu [BDLS20]: square-root speed-up on computation of large-degree isogenies.
- Radical isogenies [CDV20]: significant speed-up on isogenies of small-ish degree.
- Some work on different curve forms (e.g. Edwards, Huff).
- Knowledge of $\mathrm{End}(E_0)$ and $\mathrm{End}(E_A)$ breaks CSIDH in classical polynomial time [Wes21].
- The SQALE of CSIDH [CCJR22]: carefully constructed CSIDH parameters less susceptible to Kuperberg's quantum algorithm.
- CTIDH [$B^2C^2LMS^2$21]: Efficient constant-time CSIDH-style construction.

Evolution of key exchange

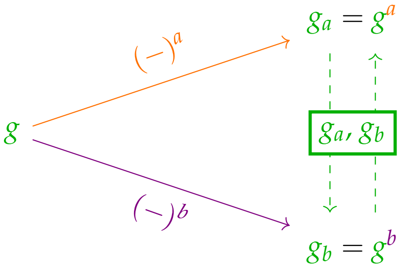
Diffie-Hellman



Colour code: Public, Alice's secret, Bob's secret

Evolution of key exchange

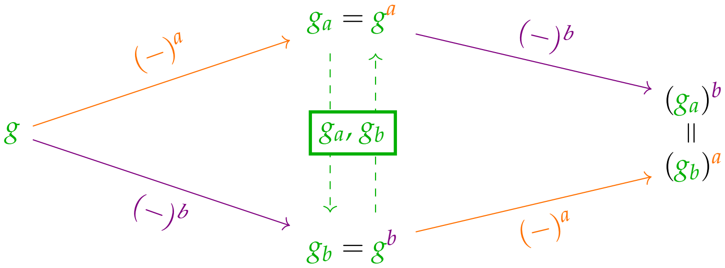
Diffie-Hellman



Colour code: Public, Alice's secret, Bob's secret

Evolution of key exchange

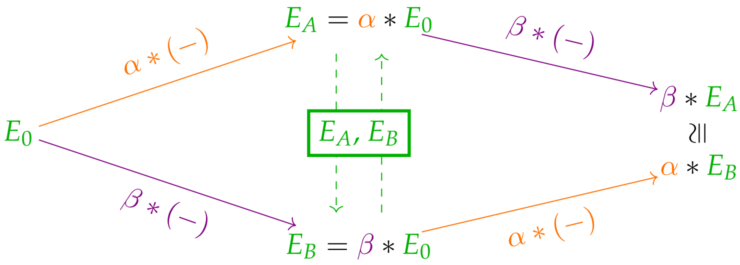
Diffie-Hellman



Colour code: Public, Alice's secret, Bob's secret

Evolution of key exchange

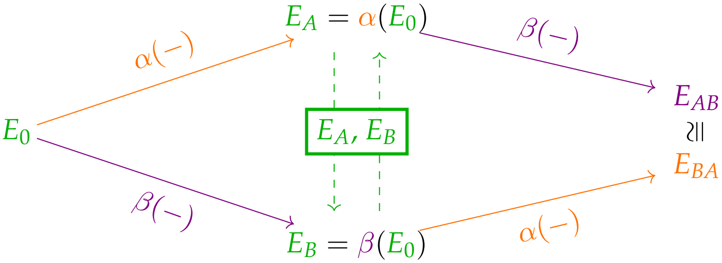
CRS or CSIDH



Colour code: Public, Alice's secret, Bob's secret

Evolution of key exchange

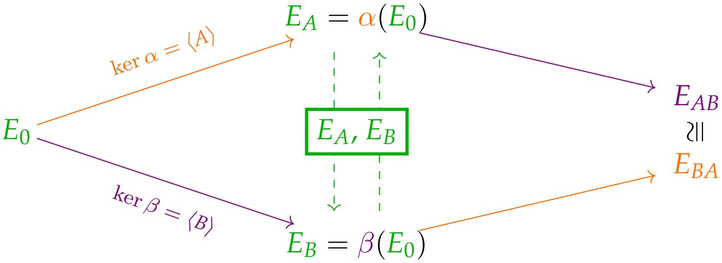
From CRS to SIDH



Colour code: Public, Alice's secret, Bob's secret

Evolution of key exchange

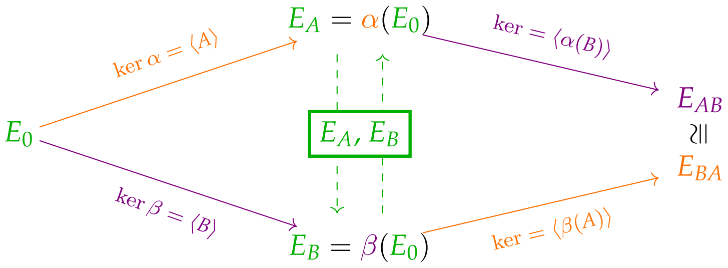
From CRS to SIDH



Colour code: Public, Alice's secret, Bob's secret

Evolution of key exchange

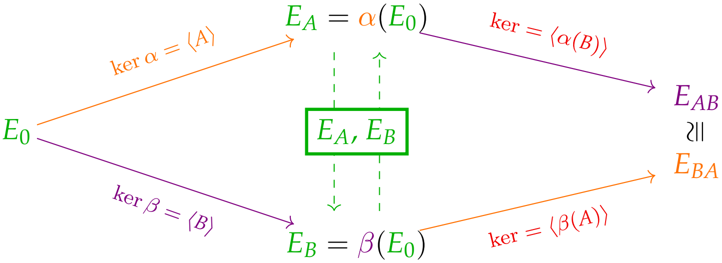
From CRS to SIDH



Colour code: Public, Alice's secret, Bob's secret

Evolution of key exchange

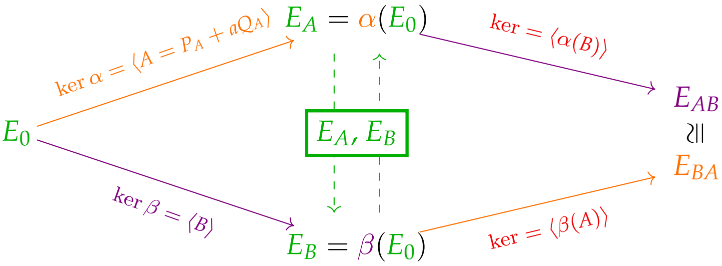
From CRS to SIDH



Colour code: Public, Alice's secret, Bob's secret, ?!

Evolution of key exchange

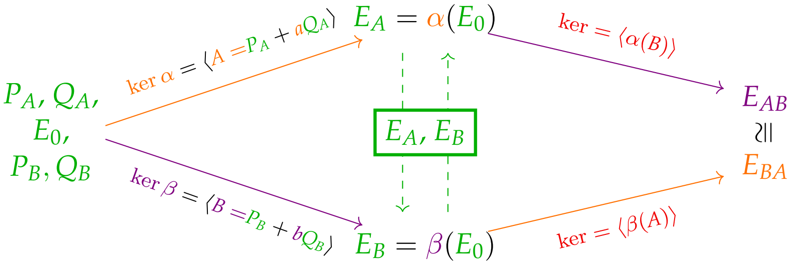
From CRS to SIDH



Colour code: Public, Alice's secret, Bob's secret, ?!

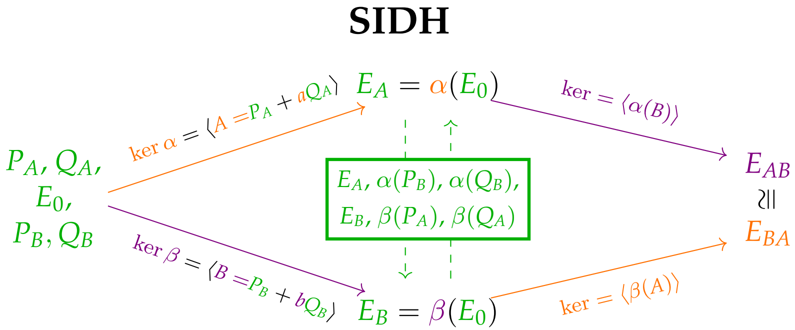
Evolution of key exchange

From CRS to SIDH



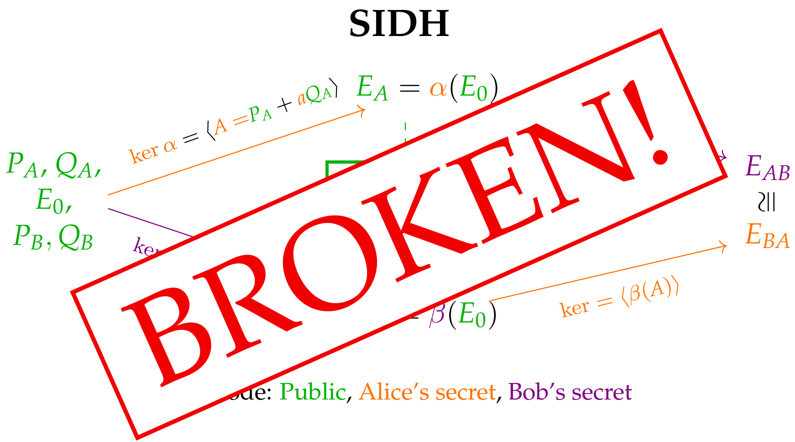
Colour code: Public, Alice's secret, Bob's secret, ?!

Evolution of key exchange



Colour code: Public, Alice's secret, Bob's secret

Evolution of key exchange



Code: Public, Alice's secret, Bob's secret

Summary of hard problems

- Diffie-Hellman – The Discrete Logarithm Problem, finding a given g and g^a.

Summary of hard problems

- Diffie-Hellman – The Discrete Logarithm Problem, finding a given g and g^a.
- CRS / CSIDH – Finding α given E and $\alpha * E$.

Summary of hard problems

- Diffie-Hellman – The Discrete Logarithm Problem, finding a given g and g^a.
- CRS / CSIDH – Finding α given E and $\alpha * E$.
- All isogeny-based schemes – Given elliptic curves E_0 and E_A, compute an isogeny $\alpha : E_0 \to E_A$ if it exists.

Summary of hard problems

- Diffie-Hellman – The Discrete Logarithm Problem, finding a given g and g^a.
- CRS / CSIDH – Finding α given E and $\alpha * E$.
- All isogeny-based schemes – Given elliptic curves E_0 and E_A, compute an isogeny $\alpha : E_0 \to E_A$ if it exists.
- SIDH –

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α. (modulo technical restrictions)*

*Details for the elliptic curve lovers:

p a large prime; $E_0/\mathbb{F}_{p^2}$ and $E_A/\mathbb{F}_{p^2}$ supersingular; $\deg(\alpha)$, N public large smooth coprime integers; points P_B, Q_B chosen such that $\langle P_B, Q_B \rangle = E_0[N]$.

History of the SIDH problem

2011 Problem introduced by De Feo, Jao, and Plut

2016 Galbraith, Petit, Shani, Ti give active attack

2017 Petit gives passive attack on some parameter sets

2020 de Quehen, Kutas, Leonardi, M., Panny, Petit, Stange give passive attack on more parameter sets

2022 Castryck-Decru and Maino-M.(-Panny-Pope-Wesolowski) give passive attack on SIKE parameter sets; Robert extends to all parameter sets

- CD and MMPPW attack is subexponential in most cases
- CD attack polynomial-time when $\mathrm{End}(E_0)$ known
- Robert attack polynomial-time in all cases

Technical interlude

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α . (modulo technical restrictions)*

Technical interlude

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α or $\ker(\alpha)$. (modulo technical restrictions)*

Technical interlude

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α or $\ker(\alpha)$. (modulo technical restrictions)*

- The set of points on an elliptic curve forms a group.

Technical interlude

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α or $\ker(\alpha)$. (modulo technical restrictions)*

- The set of points on an elliptic curve forms a group.
- $E_A[N]$ = set of points of order dividing N.

Technical interlude

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α or $\ker(\alpha)$. (modulo technical restrictions)*

- The set of points on an elliptic curve forms a group.
- $E_A[N]$ = set of points of order dividing N.
- $* \rightsquigarrow E_A[N] = \langle \alpha(P_B), \alpha(Q_B) \rangle$.

Technical interlude

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α or $\ker(\alpha)$. (modulo technical restrictions)*

- The set of points on an elliptic curve forms a group.
- $E_A[N]$ = set of points of order dividing N.
- $^* \rightsquigarrow E_A[N] = \langle \alpha(P_B), \alpha(Q_B) \rangle$.
- If $\deg(\theta : E_A \to E_A) = N$, then $\ker(\theta) \subseteq E_A[N]$.

Technical interlude

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B), \alpha(Q_B)$, compute α or $\ker(\alpha)$. (modulo technical restrictions)*

- The set of points on an elliptic curve forms a group.
- $E_A[N]$ = set of points of order dividing N.
- $^* \rightsquigarrow E_A[N] = \langle \alpha(P_B), \alpha(Q_B) \rangle$.
- If $\deg(\theta : E_A \to E_A) = N$, then $\ker(\theta) \subseteq E_A[N]$.
- Every isogeny (e.g. $\alpha : E_0 \to E_A$) has a dual isogeny (e.g. $\widehat{\alpha} : E_A \to E_0$)

Technical interlude

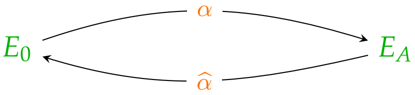
There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B), \alpha(Q_B)$, compute α or $\ker(\alpha)$. (modulo technical restrictions)*

- The set of points on an elliptic curve forms a group.
- $E_A[N]$ = set of points of order dividing N.
- $^* \rightsquigarrow E_A[N] = \langle \alpha(P_B), \alpha(Q_B) \rangle$.
- If $\deg(\theta : E_A \to E_A) = N$, then $\ker(\theta) \subseteq E_A[N]$.
- Every isogeny (e.g. $\alpha : E_0 \to E_A$) has a dual isogeny (e.g. $\widehat{\alpha} : E_A \to E_0$)

$\rightsquigarrow$ Petit's idea: Construct $\theta : E_A \to E_A$ such that $\ker(\widehat{\alpha}) \subseteq \ker(\theta)$.

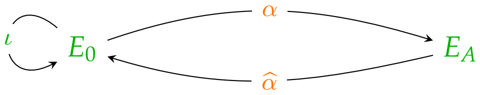
Petit's trick: torsion points to isogenies

Finding the secret isogeny α of known degree.



Petit's trick: torsion points to isogenies

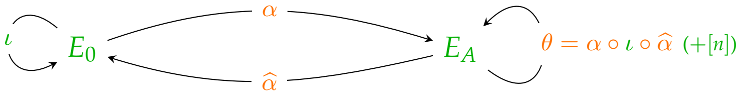
Finding the secret isogeny α of known degree.



- Restriction # 1: Assume we can choose $\iota : E_0 \to E_0$.

Petit's trick: torsion points to isogenies

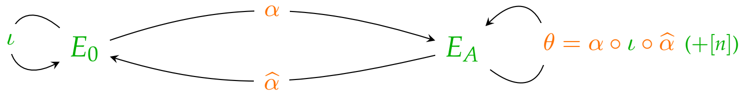
Finding the secret isogeny α of known degree.



- Restriction # 1: Assume we can choose $\iota : E_0 \to E_0$.

Petit's trick: torsion points to isogenies

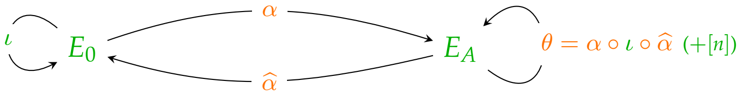
Finding the secret isogeny α of known degree.



- Restriction # 1: Assume we can choose $\iota : E_0 \to E_0$.
- Know $\alpha(E_0[N])$ (and $\widehat{\alpha(E_A[N])}$) from public torsion points.

Petit's trick: torsion points to isogenies

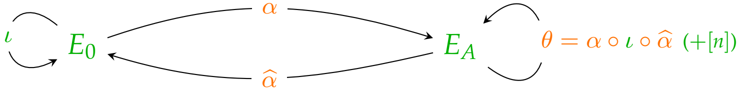
Finding the secret isogeny α of known degree.



- Restriction # 1: Assume we can choose $\iota : E_0 \to E_0$.
- Know $\alpha(E_0[N])$ (and $\widehat{\alpha(E_A[N]})$) from public torsion points.
- Know $\deg(\theta) = \deg(\alpha)^2 \deg(\iota) + n^2$.

Petit's trick: torsion points to isogenies

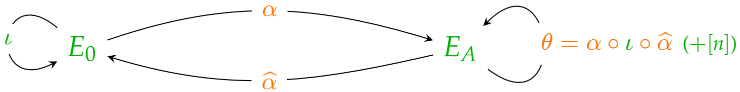
Finding the secret isogeny α of known degree.



- Restriction # 1: Assume we can choose $\iota : E_0 \to E_0$.
- Know $\alpha(E_0[N])$ (and $\widehat{\alpha(E_A[N])}$) from public torsion points.
- Know $\deg(\theta) = \deg(\alpha)^2 \deg(\iota) + n^2$.
- Restriction # 2: If there exist ι, n such that $\deg(\theta) = N$, then can completely determine θ, and α, in polynomial-time.

Petit's trick: torsion points to isogenies

Finding the secret isogeny α of known degree.



- Restriction # 1: Assume we can choose $\iota : E_0 \to E_0$.
- Know $\alpha(E_0[N])$ (and $\widehat{\alpha(E_A[N]})$) from public torsion points.
- Know $\deg(\theta) = \deg(\alpha)^2 \deg(\iota) + n^2$.
- Restriction # 2: If there exist ι, n such that $\deg(\theta) = N$, then can completely determine θ, and α, in polynomial-time.
- Restriction # 2 rules out SIKE parameters, where $N \approx \deg(\alpha)$ (and $p \approx N \cdot \deg \alpha$).

Enter Kani

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \rightarrow E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α. (modulo technical restrictions)*

Enter Kani

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α. (modulo technical restrictions)*

Problem:
Not enough choices $\theta : E_A \to E_A$.
'No θ of degree N.'

Enter Kani

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B), \alpha(Q_B)$, compute α. (modulo technical restrictions)*

Problem:
Not enough choices $\theta : E_A \to E_A$.
'No θ of degree N.'

Solution? $\theta : E_0 \times E_A \to E_0 \times E_A$?
$\rightsquigarrow$ still not enough.

Enter Kani

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α. (modulo technical restrictions)*

Problem:
Not enough choices $\theta : E_A \to E_A$.
'No θ of degree N.'

Solution? $\theta : E_0 \times E_A \to E_0 \times E_A$?
$\rightsquigarrow$ still not enough. But!

Enter Kani

There are public elliptic curves E_0 and E_A, and a secret isogeny $\alpha : E_0 \to E_A$. Given the points P_B, Q_B on E_0 and $\alpha(P_B)$, $\alpha(Q_B)$, compute α. (modulo technical restrictions)*

Problem:
Not enough choices $\theta : E_A \to E_A$.
'No θ of degree N.'

Solution? $\theta : E_0 \times E_A \to E_0 \times E_A$?
$\rightsquigarrow$ still not enough. But! Kani's theorem:

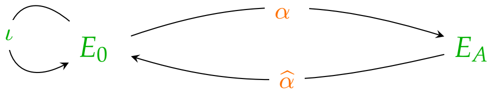
- Constructs E_1, E_2 such that there exists a (structure-preserving) isogeny

$$E_1 \times E_A \to E_0 \times E_2$$

  of the right degree, N^2.
- Petit's trick then applies.

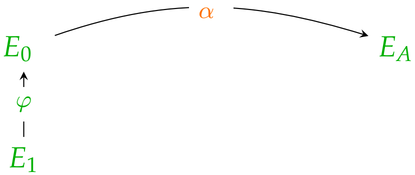
Recovering the secret

Finding the secret isogeny α of known degree.



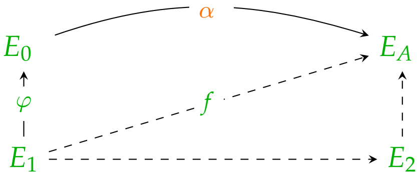
Recovering the secret

Finding the secret isogeny α of known degree.



Recovering the secret

Finding the secret isogeny α of known degree.



Kani's theorem constructs the above such that

$$\Phi = \begin{pmatrix} \varphi & -\widehat{\alpha} \\ * & * \end{pmatrix} : E_1 \times E_A \to E_0 \times E_2$$

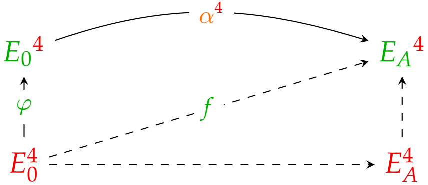
is a structure preserving isogeny of degree N^2, and

$$\ker(\Phi) = \{(\deg(\alpha)P, f(P)) : P \in E_1[N]\}$$

$\rightsquigarrow$ can compute Φ and read off secret α!

Recovering the secret with Robert's trick

Finding the secret isogeny α of known degree.



constructs the above such that

$$\Phi = \begin{pmatrix} \varphi & -\widehat{\alpha}^4 \\ * & * \end{pmatrix} : E_0^4 \times {E_A}^4 \to {E_0}^4 \times E_A^4$$

is a structure preserving isogeny of degree N^8, and

$$\ker(\Phi) \text{ is known}$$

$\rightsquigarrow$ can compute Φ and read off secret α!

What next?

- Fouotsa, Moriya, and Petit proposed mitigations
  - Masks either torsion point images or isogeny degrees
  - The mitigations make SIKE/SIDH unusably slow and big
  - For advanced protocols may still be a good option (c.f. Basso's OPRF, threshold schemes, etc.)
  - Cryptanalysis ongoing effort

What next?

- Fouotsa, Moriya, and Petit proposed mitigations
  - Masks either torsion point images or isogeny degrees
  - The mitigations make SIKE/SIDH unusably slow and big
  - For advanced protocols may still be a good option (c.f. Basso's OPRF, threshold schemes, etc.)
  - Cryptanalysis ongoing effort
- Constructive applications?
  - FESTA: New KEM. Fast and small as SIKE was?
  - SQISignHD: Small, fast signatures with clean security reduction.
  - VDF-like construction
  - Work in progress with Maino and Robert
    $\rightsquigarrow$ computing genus 2 cyclic isogenies.

What about signatures?

CSI-FiSh (S '06, D-G '18, Beullens-Kleinjung-Vercauteren '19)

Identification scheme from $H \times S \to S$:

Prover | **Public** | **Verifier**

$E \in S, \mathfrak{l}_i \in H$

$s_i \leftarrow \$\, \mathbb{Z}$

$\mathsf{sk} = \prod \mathfrak{l}_i^{s_i},$

$\mathsf{pk} = \mathsf{sk} * E \xrightarrow{\mathsf{pk}} \mathsf{pk}$

$c \leftarrow \$\, \{0, 1\}$

$\xleftarrow{c}$

$t_i \leftarrow \$\, \mathbb{Z}$

$\mathsf{esk} = \prod \mathfrak{l}_i^{t_i},$

$\mathsf{epk}_1 = \mathsf{esk} * E,$

$\mathsf{epk}_2 = \mathsf{esk} \cdot \mathsf{sk}^{-c} \xrightarrow{\mathsf{pk}, \mathsf{epk}_1, \mathsf{epk}_2}$ check:

$\mathsf{epk}_1 = \mathsf{epk}_2 * ([\mathsf{sk}^c] * E).$

After k challenges c, an imposter succeeds with prob 2^{-k}.

SQISign (De Feo-Kohel-Leroux-Petit-Wesolowski '20)

Hard Problem in CSIDH, CSI-FiSh, etc:
Given elliptic curves E and $E' \in S$, find $\mathfrak{a} \in H$ such that
$\mathfrak{a} * E = E'$.

SQISign (De Feo-Kohel-Leroux-Petit-Wesolowski '20)

Hard Problem in CSIDH, CSI-FiSh, etc:

Given elliptic curves E and $E' \in S$, find an isogeny* $E \to E'$

(*rational map + group homomorphism)

SQISign (De Feo-Kohel-Leroux-Petit-Wesolowski '20)

Hard Problem in CSIDH, CSI-FiSh, etc:
Given elliptic curves E and $E' \in S$, find an isogeny* $E \to E'$
(*rational map + group homomorphism)

SQISign(HD) is a newer signature scheme based on this idea:

SQISign (De Feo-Kohel-Leroux-Petit-Wesolowski '20)

Hard Problem in CSIDH, CSI-FiSh, etc:
Given elliptic curves E and $E' \in S$, find an isogeny* $E \to E'$
(*rational map + group homomorphism)

SQISign(HD) is a newer signature scheme based on this idea:

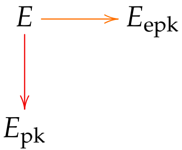
$$E \downarrow E_{\text{pk}}$$

public, secret, ephemeral secret, public challenge, public proof

SQISign (De Feo-Kohel-Leroux-Petit-Wesolowski ‘20)

Hard Problem in CSIDH, CSI-FiSh, etc:
Given elliptic curves E and $E' \in S$, find an isogeny* $E \to E'$
(*rational map + group homomorphism)

SQISign(HD) is a newer signature scheme based on this idea:

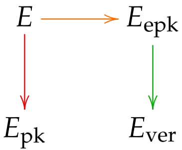


public, secret, ephemeral secret, public challenge, public proof

SQISign (De Feo-Kohel-Leroux-Petit-Wesolowski '20)

Hard Problem in CSIDH, CSI-FiSh, etc:
Given elliptic curves E and $E' \in S$, find an isogeny* $E \to E'$
(*rational map + group homomorphism)

SQISign(HD) is a newer signature scheme based on this idea:

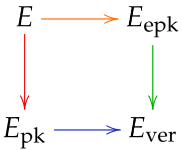


public, secret, ephemeral secret, public challenge, public proof

SQISign (De Feo-Kohel-Leroux-Petit-Wesolowski ‘20)

Hard Problem in CSIDH, CSI-FiSh, etc:
Given elliptic curves E and $E' \in S$, find an isogeny* $E \to E'$
(*rational map + group homomorphism)

SQISign(HD) is a newer signature scheme based on this idea:



public, secret, ephemeral secret, public challenge, public proof

Summary and overview

- SIKE '11 KEM. Was in NIST, recently broken in polynomial-time.

Summary and overview

- SIKE '11 KEM. Was in NIST, recently broken in polynomial-time.
- CSIDH '18 / SQALE '22 Key exchange. Small, many applications (c.f. group actions), slow, known quantum attack needs further study, other attack avenues non-obvious.

Summary and overview

- SIKE '11 KEM. Was in NIST, recently broken in polynomial-time.
- CSIDH '18 / SQALE '22 Key exchange. Small, many applications (c.f. group actions), slow, known quantum attack needs further study, other attack avenues non-obvious.
- FESTA '23 KEM. Fast-ish and small. Relies on new attack ideas. Hard to implement well.
- CSI-FiSh '19 / SCALLOP '23 Digital signature. Small-ish, slow, flexible.

Summary and overview

- SIKE '11 KEM. Was in NIST, recently broken in polynomial-time.
- CSIDH '18 / SQALE '22 Key exchange. Small, many applications (c.f. group actions), slow, known quantum attack needs further study, other attack avenues non-obvious.
- FESTA '23 KEM. Fast-ish and small. Relies on new attack ideas. Hard to implement well.
- CSI-FiSh '19 / SCALLOP '23 Digital signature. Small-ish, slow, flexible.
- SQISign '20 Digital signature. Small, slow, clean-ish security assumption, no known attack avenues. In NIST.
- SQISignHD '23 Digital signature. Small, fast-ish, security reduction to very well-studied problem in number theory, hard to implement well.

References

[B²C²LMS²21] ctidh.isogeny.org

[BD17] ia.cr/2017/334

[BDLS20] velusqrt.isogeny.org

[BEG19] ia.cr/2019/485

[BLMP19] quantum.isogeny.org

[BMP23] ia.cr/2023/660

[CCJR22] ia.cr/2020/1520

[CD19] ia.cr/2019/1404

[CDV20] ia.cr/2020/1108

[DFKLMPW23] ia.cr/2023/058

[DLRW23] ia.cr/2023/436

[FM19] ia.cr/2019/555

[GMT19] ia.cr/2019/431

[Wes21] ia.cr/2021/1583