An Introduction to SIDH

David Jao

Department of Combinatorics & Optimization
Centre for Applied Cryptographic Research

November 18, 2018

SIDH

Supersingular **I**sogeny **D**iffie-**H**ellman (Jao and De Feo, 2011):

- A key-exchange protocol, similar to Diffie-Hellman, using isogenies between supersingular elliptic curves

Why isogenies?

- Because they seem to be quantum-resistant

Why supersingular curves?

- We found a subexponential attack against ordinary (i.e. non-supersingular) curves (Childs, Jao, and Soukharev 2014)

Supersingular elliptic curves in cryptography

Previous applications of supersingular elliptic curves in cryptography include:

- Discrete logarithm attacks (Menezes, Okamoto, Vanstone)
- Pairing-based cryptography (Joux)
- Hash functions from expander graphs (Charles, Goren, Lauter)

The CGL hash function was the first published isogeny-based cryptosystem.

- However, a hash function is "only" a one-way function, and cannot be used to achieve public-key encryption.
- Public-key encryption requires a trapdoor one-way function.

History of isogeny-based public-key cryptography

Couveignes (1997), Rostovstev & Stolbunov (2006):

- Proposed a public-key cryptosystem using ordinary curves
- Optimized by De Feo, Kieffer & Smith (2018)
- Very slow
- Has subexponential security

Jao & De Feo, SIDH (2011):

- Uses supersingular curves
- (Relatively) fast
- Exponential security

Castryck et al., CSIDH (2018):

- Based on the CRS concept, but uses supersingular curves and SIDH-like optimizations
- Almost as fast as SIDH
- Subexponential security

SIDH vs. CSIDH

SIDH:

- Faster (so far).
- Superior (exponential) asymptotic security.
- Direct key validation is not possible.
- Admits active attacks — Galbraith, Petit, Shani, Ti (2016).
- Requires extra steps, e.g. SIKE (`https://sike.org/`) for CCA2 security or non-interactive key exchange (Urbanik & Jao, MathCrypt 2018).

CSIDH:

- Slower (so far).
- Inferior (subexponential) asymptotic security.
- Supports direct key validation.
- No known active attacks.
- Naturally supports non-interactive key exchange.

Isogenies

Definition

An isogeny is a morphism ϕ of algebraic varieties between two elliptic curves, such that ϕ is a group homomorphism.

Concretely:

$$\phi \colon E \to E'$$
$$\phi(x, y) = (\phi_x(x, y), \phi_y(x, y))$$
$$\phi_x(x, y) = \frac{f_1(x, y)}{f_2(x, y)}$$
$$\phi_y(x, y) = \frac{g_1(x, y)}{g_2(x, y)}$$

(f_1, f_2, g_1, and g_2 are all polynomials)

Vélu's formulas for constructing isogenies (1971)

Let G be any finite subgroup of an elliptic curve E. Let S be a set of representatives of $(G \setminus \{\mathcal{O}_E\})/\sim$, where $P \sim Q \iff P = \pm Q$. Then there exists an isogeny $\phi\colon E \to E'$ with $\ker \phi = G$, given by

$$\phi_x(x,y) = x + \sum_{Q \in S} \left[\frac{t_Q}{x - x_Q} + \frac{u_Q}{(x - x_Q)^2} \right]$$

$$\phi_y(x,y) = y - \sum_{Q \in S} \left[u_Q \frac{2y}{(x - x_Q)^3} + t_Q \frac{y - y_Q}{(x - x_Q)^2} - \frac{g_Q^x g_Q^y}{(x - x_Q)^2} \right]$$

$$Q = (x_Q, y_Q)$$

$$g_Q^x = 3x_Q^2 + a_4$$

$$g_Q^y = -2y_Q$$

$$t_Q = \begin{cases} g_Q^x & \text{if } Q = -Q \\ 2g_Q^x & \text{if } Q \neq -Q \end{cases}$$

$$u_Q = (g_Q^y)^2$$

Vélu's formula

Remarks:

- The computational complexity of Vélu's formulas is $O(|G|)$.
  - This is exponentially slow when $|G|$ is large.
- The isogeny ϕ and the codomain E' are unique up to isomorphism (a kernel determines an isogeny, up to isomorphism).
- Borrowing notation from group theory, we denote E' by E/G.

CRS (2006), CSIDH (2018)

1. Public parameters: Ordinary elliptic curve E with $\mathrm{End}(E) = \mathcal{O}_d \subset \mathbb{Q}(\sqrt{d})$, $d < 0$.
   - CSIDH: Supersingular E with $\mathrm{End}_{\mathbb{F}_p}(E) = \mathcal{O}_d$.
2. Alice chooses an ideal $\mathfrak{a} \subset \mathcal{O}_d$ and sends $\mathfrak{a} * E$ to Bob.
3. Bob chooses an ideal $\mathfrak{b} \subset \mathcal{O}_d$ and sends $\mathfrak{b} * E$ to Alice.
4. The shared secret is $(\mathfrak{a}\mathfrak{b}) * E = \mathfrak{a} * (\mathfrak{b} * E) = \mathfrak{b} * (\mathfrak{a} * E)$.

$$\begin{array}{ccc} E & \longrightarrow & \mathfrak{a} * E \\ \downarrow & & \downarrow \\ \mathfrak{b} * E & \longrightarrow & (\mathfrak{a}\mathfrak{b}) * E \end{array}$$

The $*$ operator is the complex multiplication operator:

$$\mathfrak{a} * E = E / \ker(\mathfrak{a}) = E / \{P \in E : \phi(P) = \mathcal{O}_E \text{ for all } \phi \in \mathfrak{a}\}$$

SIDH (overview)

1. Public parameters: Supersingular elliptic curve E over $\mathbb{F}_{p^2}$.
2. Alice chooses a kernel $A \subset E(\mathbb{F}_{p^2})$ and sends E/A to Bob.
3. Bob chooses a kernel $B \subset E(\mathbb{F}_{p^2})$ and sends E/B to Alice.
4. The shared secret is

$$E/\langle A, B\rangle = (E/A)/\phi_A(B) = (E/B)/\phi_B(A).$$

$$\begin{array}{ccc} E & \xrightarrow{\phi_A} & E/A \\ {\scriptstyle \phi_B}\downarrow & & \downarrow \\ E/B & \longrightarrow & E/\langle A, B\rangle \end{array}$$

We will discuss security later: For ℓ-bit (quantum) security against current attacks, use a prime of size 6ℓ bits (i.e. $p \approx 2^{6\ell}$).

Making it work

$$\begin{array}{ccc} E & \xrightarrow{\phi_A} & E/A \\ {\scriptstyle \phi_B}\downarrow & & \downarrow \\ E/B & \longrightarrow & E/\langle A, B\rangle \end{array}$$

- In order to be secure, A and B must be of cryptographic size, but Vélu's formulas are impractical for such large kernels.
- In order to compute $(E/A)/\phi_A(B)$, Bob needs not only E/A but also the image of B in E/A, i.e. $\phi_A(B)$. But B is known only to Bob, and ϕ_A is known only to Alice.

Computing isogenies with large kernels

To control the cost of isogeny evaluation, we need to use kernels that factor as abelian groups: $A \cong \mathbb{Z}/p_1^{e_1}\mathbb{Z} \times \cdots \times \mathbb{Z}/p_g^{e_g}\mathbb{Z}$, where the primes p_i are small.

- For simplicity, we use $A \cong \mathbb{Z}/2^e\mathbb{Z}$. Then the subgroup tower

$$0 \subset \mathbb{Z}/2\mathbb{Z} \subset \mathbb{Z}/4\mathbb{Z} \subset \cdots \subset \mathbb{Z}/2^e\mathbb{Z}$$

  allows us to factor $\phi_A \colon E \to E/A$ into the composition

$$E \to E/(\mathbb{Z}/2\mathbb{Z}) \to E/(\mathbb{Z}/4\mathbb{Z}) \to \cdots \to E/(\mathbb{Z}/2^e\mathbb{Z})$$

  Each isogeny in the composition is easy to compute.
- Similarly, Bob's subgroup B is isomorphic to $\mathbb{Z}/3^f\mathbb{Z}$. (Bob cannot also use $B \cong \mathbb{Z}/2^e\mathbb{Z}$; if he did, then the shared secret $E/\langle A, B\rangle$ would be $E/E[2^e] \cong E$.)

Computing $E/\langle A, B\rangle$

Alice knows ϕ_A and Bob knows B. How can Bob find $\phi_A(B)$?

- Fix a generating set $\{P, Q\}$ of $(\mathbb{Z}/3^f\mathbb{Z})^2 \subset E(\mathbb{F}_{p^2})$.
- Let $mP + nQ$ be a generator of B. (Without loss of generality, we may take $m = 1$.)
- Alice computes $\phi_A(P)$ and $\phi_A(Q)$ and sends them to Bob.
- Bob can now compute $m\phi_A(P) + n\phi_A(Q) = \phi_A(mP + nQ)$ to obtain $\phi_A(B)$.

Therefore:

- While Alice (for example) is computing ϕ_A, she also needs to compute $\phi_A(P)$ and $\phi_A(Q)$.
- Alice needs to send $\phi_A(P)$ and $\phi_A(Q)$ to Bob as part of her public key.
- We refer to $\phi_A(P)$ and $\phi_A(Q)$ as "auxiliary points."

SIDH (detailed construction)

Public parameters:

- Prime $p = \ell_A^{e_A} \ell_B^{e_B} \cdot f \pm 1$
- Supersingular elliptic curve $E/\mathbb{F}_{p^2}$ of order $(p \mp 1)^2$
- $\mathbb{Z}$-basis $\{P_A, Q_A\}$ of $E[\ell_A^{e_A}]$ and $\{P_B, Q_B\}$ of $E[\ell_B^{e_B}]$

Alice:

- Choose $R_A = m_A P_A + n_A Q_A$ of order $\ell_A^{e_A}$
- Compute $\phi_A \colon E \to E/\langle R_A \rangle$
- Send $E/\langle R_A \rangle, \phi_A(P_B), \phi_A(Q_B)$ to Bob

Bob:

- Same as Alice, with A's and B's swapped

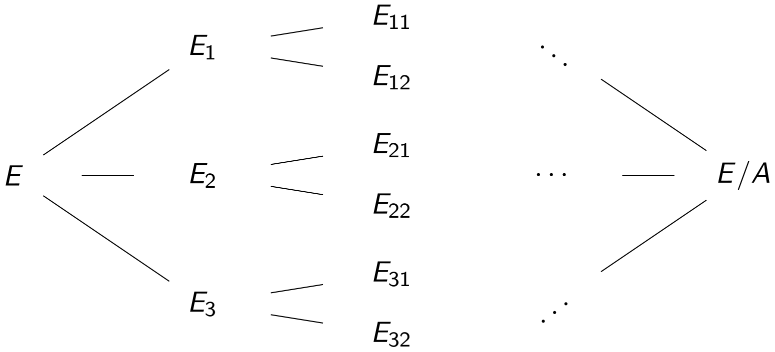
The shared secret is

$$
\begin{aligned}
E/\langle R_A, R_B \rangle &= (E/\langle R_A \rangle)/\langle m_B \phi_A(P_B) + n_B \phi_A(Q_B) \rangle \\
&= (E/\langle R_B \rangle)/\langle m_A \phi_B(P_A) + n_A \phi_B(Q_A) \rangle
\end{aligned}
$$

Security

Hardness problem: Given E and E/A, find A.

Fastest known attack is meet-in-the-middle search (Galbraith, Hess, Smart 2002):



Attack complexity

For a **generic** meet-in-the-middle attack, the values in the table are provable lower bounds.

	Alice	Bob
Classical	$\sqrt{2^e}$	$\sqrt{3^f}$
Quantum	$\sqrt[3]{2^e}$	$\sqrt[3]{3^f}$

Quantum security level of SIDH is conjecturally

$$\min(2^{e/3}, 3^{f/3}) \approx p^{1/6}$$

Key sizes

Public key size (bits):

- $8 \log_2 p$ (naive)
- $6 \log_2 p$ (Costello et al., Crypto 2016 — no key compression)
- $\frac{7}{2} \log_2 p$ (Costello et al., Eurocrypt 2017 — key compression)

Example: For 128-bit quantum security, we have $p \approx 2^{768}$ and:

- $8 \log_2 p$ bits = 6144 bits = 768 bytes
- $6 \log_2 p$ bits = 4608 bits = 576 bytes (no key compression)
- $\frac{7}{2} \log_2 p$ bits = 2688 bits = 336 bytes (key compression)

Performance (Intel x86-64 Skylake):

- Faz-Hernandez et al., `https://ia.cr/2017/1015`: 10ms
- Zanon et al., `https://ia.cr/2017/1143`: 20ms for key compression

These numbers are likely improvable, since the $p^{1/6}$ estimate seems to be too conservative (Adj et al., `https://ia.cr/2018/313`)

Summary

At first glance, the fact that [SIDH-based] SIKE is the only isogeny-based KEM submitted to the NIST post-quantum process, competing with 58 others mostly based on codes, lattices, and polynomial systems, might suggest that it is a strange outlier. However, this uniqueness is not so much an indicator of lack of support, so much as a sign of rare convergence and consensus in the elliptic-curve cryptography community—convergence that did not occur to the same extent in the communities working on other post-quantum paradigms. The fact that there was only one isogeny-based submission reflects the general agreement that this was the right way to do isogeny-based key agreement at that point in time. The more flexible CSIDH scheme was not developed until later, when the NIST process was already underway, and so it was not part of the conversation.

—Ben Smith, `https://ia.cr/2018/882`