Isogeny-Based Public-Key Cryptography

David Jao

Department of Combinatorics & Optimization
Centre for Applied Cryptographic Research

UNIVERSITY OF WATERLOO

June 17, 2016

Motivation: Post-Quantum Cryptography

- DH (1976)
- ECDH (1986)
- Shor's algorithm (1994)

How do we make elliptic curve cryptography into something post-quantum?

SIDH

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

- An analog of Diffie-Hellman, using supersingular isogenies.

What are supersingular isogenies?

- See next slide(s).

Why isogenies?

- Because they seem to work (discussed later in this talk).

Why supersingular isogenies?

- Because we broke non-supersingular isogenies (ANTS IX, J. Math. Cryptol. **8**(1), 2014).

Elliptic curves

Definition

An elliptic curve over a field F is a nonsingular plane curve E of the form $y^2 = x^3 + a_4x + a_6$, for fixed $a_4, a_6 \in F$.

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

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)

Constructing isogenies

Vélu (1971): Let G be any finite subgroup of an elliptic curve E. Let S be a set of representatives of $G/\sim$, where $\sim$ is the relation $P \sim Q \iff P = \pm Q$. 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:

- Computational complexity of the formula is $O(|G|)$.
- The isogeny ϕ and the codomain E' are unique up to isomorphism (a kernel determines a group homomorphism, up to isomorphism).
- Borrowing notation from group theory, we denote E' by E/G.

Basic key exchange

1. Public parameters: An elliptic curve E defined over a finite field F.
2. Alice chooses a kernel A and sends E/A to Bob.
3. Bob chooses a kernel B and sends E/B to Alice.
4. The shared secret is $(E/A)/B = (E/B)/A$.

$$\begin{tikzcd} E \arrow[r, "\phi_A"] \arrow[d, "\phi_B"'] & E/A \arrow[d] \\ E/B \arrow[r] & (E/A)/B \end{tikzcd}$$

Questions

$$
\begin{array}{ccc}
E & \xrightarrow{\phi_A} & E/A \\
{\scriptstyle \phi_B}\downarrow & & \downarrow \\
E/B & \longrightarrow & (E/A)/B
\end{array}
$$

- 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 $(E/A)/B$, 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.

Isogenies with large kernels

- In order to compute E/A for large A, we arrange it so that A is isomorphic to $\mathbb{Z}/2^e\mathbb{Z}$. Then the subgroup tower

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

  yields the chain of isogenies

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

  of length e, whose composition equals $E \to E/A$. Each isogeny in the chain is easy to compute.
- Similarly, we arrange Bob's B to be isomorphic to $\mathbb{Z}/3^f\mathbb{Z}$.

Constructing suitable elliptic curves

In order to obtain the necessary A's and B's:

- We require an elliptic curve over a finite field, containing a point of order 2^e, and a point of order 3^f.
- The field size, and the quantities 2^e and 3^f, should be of cryptographic size.
- The extension degree of the field needs to be much smaller than cryptographic size.

Strategy:

- Let E be the curve $y^2 = x^3 + x$, defined over a prime p such that $p + 1 = 2^e \cdot 3^f \cdot g$
- Then $p \equiv 3 \pmod 4$ and $\#E(\mathbb{F}_p) = p + 1$ (easy)
- Embedding degree of E is 2 (Menezes-Okamoto-Vanstone)
- Hence $E(\mathbb{F}_{p^2}) \cong (\mathbb{Z}/(2^e \cdot 3^f \cdot g)\mathbb{Z})^2$
- Let A be a one-dimensional subgroup of $(\mathbb{Z}/2^e\mathbb{Z})^2 \subset E(\mathbb{F}_{p^2})$.

Computing $(E/A)/B$

- Alice knows ϕ_A and Bob knows 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.
- Alice computes $\phi_A(P)$ and $\phi_A(Q)$ and sends them to Bob.
- Bob computes

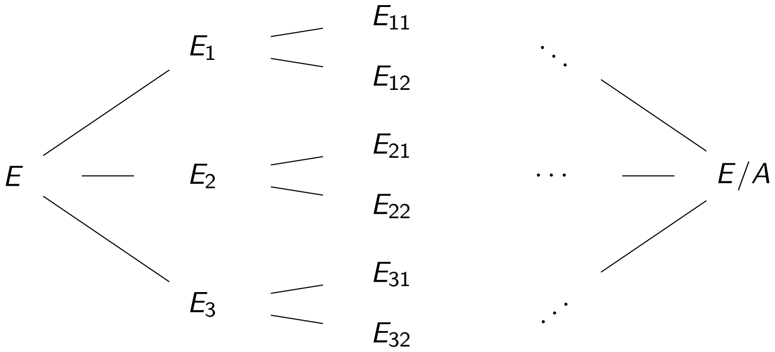
$$m\phi_A(P) + n\phi_A(Q) = \phi_A(mP + nQ)$$

to obtain $\phi_A(B)$.

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

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

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

Parameter sizes and performance

Quantum security level of SIDH is conjecturally

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

Public key size (bits):

- $8 \log_2 p$ (naive)
- $6 \log_2 p$ (Costello et al., Crypto 2016 — no performance penalty)
- $4 \log_2 p$ (Azarderakhsh et al., AsiaPKC 2016 — some performance penalty)
- Example: For 128-bit quantum security,
  - $6 \log_2 p$ bits $=$ 4608 bits $=$ 576 bytes
  - $4 \log_2 p$ bits $=$ 3072 bits $=$ 384 bytes

Performance:

- 14 ms per key-exchange round on x86-64 (Costello et al., Crypto 2016)

Open problems

- Generalizations (hyperelliptics, Jacobians)
- Cryptanalysis (classical and quantum)
- Protocols (authentication, signatures)
- Performance improvements